



Target2-Securities

General Functional Specifications

R2025.NOV

05 August 2025

Table of Contents

1 Approach to T2S Functional Design 5
1.1 Presentation of the document 5
1.2 Methodological elements 6
1.3 Reference documents 7
1.4 Specifications references – cross-referencing URD 7
1.5 Conventions used 7
2 General Functional Overview 8
2.1 Introduction 8
2.1.1 Objective and scope of T2S 8
2.1.2 T2S Actors 23
2.1.3 Multi-currency in T2S 29
2.1.4 Other systems interacting with T2S 29
2.2 Overall high level diagram 30
2.3 Description of functional domains 30
2.3.1 Interface 31
2.3.2 Reference Data Management 32
2.3.3 Lifecycle Management and Matching 33
2.3.4 Settlement 35
2.3.5 Liquidity Management 37
2.3.6 Statistics, Queries, Reports and Archive 38
2.3.7 Operational Services 39
2.4 Security Management 40
3 Functional Description 43
3.1 Introduction 43
3.2 Interface 43
3.2.1 General Introduction 43
3.2.2 Dynamic data managed by the domain 46
3.2.3 Communication Module 52
3.2.4 Inbound Processing Module 60
3.2.5 Outbound Processing Module 74
3.2.6 T2S Graphical User Interface 88
3.2.7 Interface (IN) Use Cases 96
3.2.8 Processing of the Interface Use Cases 97
3.3 Reference Data Management 105
3.3.1 General Introduction 105
3.3.2 Dynamic data managed by the domain 108

All rights reserved.

3.3.3 Functions 110
3.3.4 Input / Output 122
3.3.5 Data Accessed 123
3.3.6 Common Information 123
3.3.7 Party Data Management 127
3.3.8 Securities Data Management 133
3.3.9 Securities Account Data Management 141
3.3.10 T2S Dedicated Cash Account Data Management 146
3.3.11 Rules and Parameters Data Management 153
3.3.12 Reference Data Management (SM) Use Cases 189
3.3.13 Processing of Reference Data Management Use Cases 190
3.4 Lifecycle Management and Matching 194
3.4.1 General Introduction 194
3.4.2 Dynamic data managed by the domain 197
3.4.3 Instruction Validation 214
3.4.4 Instruction Maintenance 234
3.4.5 Instruction Matching 255
3.4.6 Status Management 261
3.4.7 Penalty Mechanism 300
3.4.8 Settlement Instructions (SI) Use Cases 337
3.4.9 Processing of Settlement Instructions Use Cases 338
3.4.10 Maintenance Instruction (MI) Use Case 354
3.4.11 Processing of Maintenance Instructions Use Cases 355
3.5 Settlement 370
3.5.1 General Introduction 370
3.5.2 Dynamic data managed by the domain 373
3.5.3 Optimisation Algorithms common features 389
3.5.4 Multi-currency aspects 390
3.5.5 Standardisation and Preparation to Settlement 393
3.5.6 Daytime Validation, Provisioning and Booking 434
3.5.7 Daytime Recycling and Optimisation 468
3.5.8 Night-time Settlement 479
3.5.9 Auto-collateralisation 490
3.5.10 Processing of Settlement Instruction Use Cases 506
3.6 Liquidity Management 546
3.6.1 General Introduction 546
3.6.2 Dynamic data managed by the domain 555
3.6.3 Liquidity Operations 559
3.6.4 Outbound Information Management 568
3.6.5 CB Business Procedures 574
3.6.6 Liquidity Transfers (LT) Use Cases 576

All rights reserved.

3.6.7 Processing of Liquidity Transfers Use Cases 579
3.7 Statistics, Queries, Reports and Legal Archiving 595
3.7.1 General Introduction 595
3.7.2 Statistical Information 597
3.7.3 Query Management 606
3.7.4 Report Management 621
3.7.5 Legal Archiving 631
3.8 Operational Services 635
3.8.1 General Introduction 635
3.8.2 Scheduling 636
3.8.3 Operational Monitoring 651
3.8.4 Data Migration 661
3.8.5 Billing 663
4 Appendices 681
4.1 List of Use Cases 681
4.1.1 Presentation: devising the use cases 681
4.1.2 SI: Settlement Instruction 682
4.1.3 MI: Maintenance Instructions (Hold/Release, Amendment and Cancellation instructions) 692
4.1.4 SM: Reference Data Management 695
4.1.5 QU: Queries 701
4.1.6 RE: Reports 705
4.1.7 LT: Liquidity Transfers 708
4.1.8 EV: Events 711
4.1.9 LA: Legal Archiving 712
4.1.10 IN: Interface 713
4.2 Data Model 714
4.3 UR Index 714
4.4 Most frequently used acronyms 714

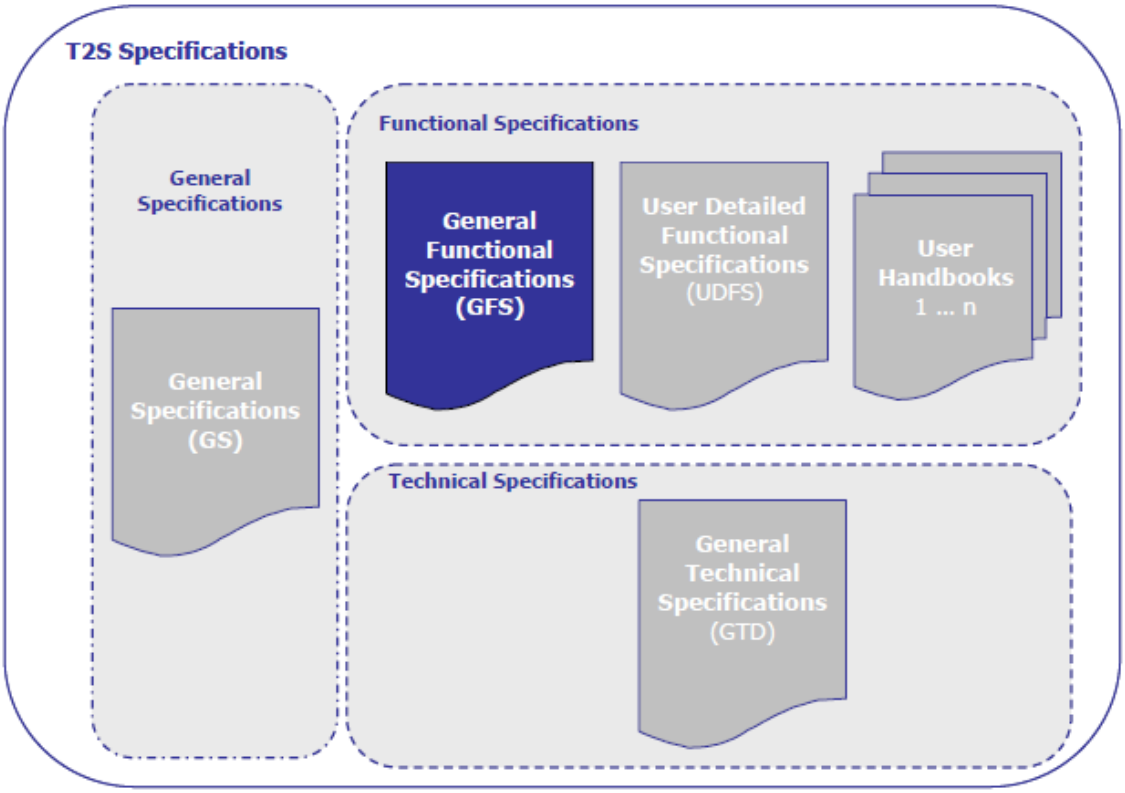
All rights reserved.

1 Approach to T2S Functional Design

1.1 Presentation of the document

The T2S deliverables for the specification phase are considered as documents aiming at allowing users to understand how services described in the T2S User Requirements Documentation will be provided by the T2S platform.

The diagram below presents an overview of all the T2S deliverables for the specification phase.



The General Functional Specifications aim at presenting from a functional perspective the solution envisaged for T2S. To that purpose, the structure of this document aims at providing an overall picture of the functional design envisaged for T2S before entering into more an in-depth description of each domain, module and function.

As regards general descriptions, after giving an overview of the approach retained for the T2S functional design in the current first chapter of the GFS, the second chapter relates to the general functional overview of T2S. This second chapter provides an overall description of the objective and the scope of T2S, as well as an overall picture of the T2S functional architecture with a high-level functional diagram representing the interactions between domains and modules, irrespective of implementation choices. This chapter provides an overview of the Data Model foreseen for T2S with both static and dynamic data as well as general description of the relationship between the different entities of the data model (detailed description of entities and attributes are provided in the relevant chapters of the functional description). This overall description is complemented by an overview of the security management issues for T2S.

All rights reserved.

On the basis of this general view of the functional solution envisaged for T2S, the third chapter provides a functional description of the domains and modules identified in the overall high level diagram. For each of these domains, this chapter describes the role and behaviour of their respective modules and functions by presenting their respective data flow processes. It also covers a presentation of the processing of some representative use cases illustrating the respective roles and interactions of modules in this processing. This graphic and text description of use cases processing aims at facilitating the understanding of the internal behaviour of the system to answer the requests originating from external parties.

1.2 Methodological elements

The Functional Design process is based on a phased approach leading to design in a first step the GFS on the basis of the T2S User Requirements and then expand the GFS into detailed functional specifications. The functional specifications aim at describing from a functional perspective the architecture of the future platform (i.e. description of the different functional domains and modules and identification of their internal interactions) with an increasing level of granularity and comprehensiveness at each stage of the process.

The methodology retained for this process relies on a waterfall approach according to which the GFS should be delivered before moving to the next phase. The functional documentation is structured to ensure full consistency, upward compatibility and reusability between the successive deliverables (i.e. parts of GFS documentation can be reused for being expanded and enriched in the following phase).

In this process, the GFS provide a description of the overall solution envisaged for the T2S functional design. More precisely, the GFS encompass (i) the complete logical data model (with both static and dynamic data and the description of entities and attributes), (ii) the exhaustive list of use cases for ensuring self consistency of the design and (iii) the descriptions of the functionalities of the future platform according to a hierarchical description in three levels: domains, modules and functions:

- The concept of "Domain" refers to the highest level of the hierarchical description of the T2S functionalities. Each of the seven domains identified for T2S covers several modules consistently grouped together according to their proximity of activity. The breakdown into domains has been done with a view to stick to the largest possible extent to the breakdown already identified in the T2S User Requirements;
- The concept of "Module" refers to the second level of the hierarchical description of the T2S functionalities. Each module covers several functions consistently grouped together according to their proximity of activity. For the provision of the T2S services identified in T2S User Requirements, each module can exchange information with T2S Actors as well as with external systems via the Interface domain and with other modules belonging to the same or different domains, as illustrated in the High-Level Diagram;
- Finally, the concept of "Function" refers to the most detailed level of the hierarchical description of the T2S functional solution in the GFS. In this context, a function is a process unit having access to permanent data with a view to process and exchange data flows with other functions or external T2S Actors and systems via the Interface domain.

The provision of services identified in the T2S User Requirements relies on the intervention of one or several functions pertaining to one or several module(s) and domain(s).

All rights reserved.

Since the GFS document is to be published, the UML (Unified Modelling Language) formalism is used for the graphical representation of the aforementioned concepts. As market-standard representation for the functional descriptions of T2S it aims at easing the reading and understanding of functional deliverables, via the provision of symbols and diagrams adequate to different representations of a system such as the functional requirements view, static structural view and dynamic behaviour view.

1.3 Reference documents

The present version 5.1 of the GFS relies on the version 5.04 of the T2S User Requirements document.

1.4 Specifications references – cross-referencing URD

In order to enable cross-referencing functional specifications and user requirements, the current version of the GFS includes the following elements:

- Within the text of the document, a clear identification of relevant User Requirement (to be recognised through the following formalism: **{T2S.xx.yyy}**) justifying the treatment described;
- Within chapter 3, a breakdown of each functional description into elementary specifications identified specifically via a unique code (to be recognised through the following formalism: [Domain].[Module].[Function].xx.yy}) allowing their cross-referencing between with User Requirements with the functions and sub-sections of the document;
- As a dedicated appendix, a matrix providing for each and every functional UR the following information: the text label of the UR, the GFS part, the GFS domain or chapter, the module or sub-chapter, the section, the function or data entity or sub-section, and the identifier of the specification.

1.5 Conventions used

UML proposes a large choice of diagrams allowing the description of systems under several perspectives. In the context of T2S GFS, the following subset of UML diagrams is used:

- Use case diagrams showing interactions between the T2S platform and external entities to T2S;
- Sequence diagrams, focusing on communications between various objects over time and the messages triggering those communications;
- Activity diagrams showing the workflow from a start point to the finish point and detailing the different decision paths existing in the progression of events contained in the activity;
- Class diagrams describing the data structure with the entities, the attributes and the relations between these entities and Object diagrams proposing some concrete examples of data entities instances in specific business conditions. For example, while a "Party class diagram" describes in general the attributes that allow describing a party, a "Party object diagram" shows the values taken by these attributes for a specific party.

The UML formalism provides a useful basis for the description of the behaviour of a module or a function, but the text accompanying the diagrams is critical for the understanding of this behaviour.

All rights reserved.

2 General Functional Overview

2.1 Introduction

2.1.1 Objective and scope of T2S

On the basis of selected extracts and abstracts of the User Requirements reflected in italics, this section of the GFS aims at providing an overview of the scope of the business of T2S, the processing schedule and calendar.

2.1.1.1 Scope

This section covers the securities categories and type of transactions, by quoting when relevant the Principles of T2S and the URD.

Securities categories

"The T2S scope shall include all securities that comply with the following eligibility criteria, i.e. that:

- *have an ISIN code, as instrument identifier;*
- *are held with a CSD in T2S;*
- *settle in book-entry form; and*
- *are fungible (from a settlement process perspective).*

These criteria should cover all securities currently settling in EU CSDs. Eurobonds, for example, have an ISIN code, settle in book-entry form and are fungible. Therefore, they are eligible for settlement in T2S if they are held with a CSD in T2S. In addition, certain securities, compliant with the first three criteria, but non-fungible from a settlement perspective, may still be entered in and processed by T2S under specific conditions. T2S would identify these securities as specific non-standardised securities pertaining to certain markets (...)" ***{T2S.02.010}***.

When prior or subsequent steps to the settlement procedure are required to register, identify or update additional codes (registration codes, reference numbers, etc.), those procedures are executed by CSDs as currently done. These criteria should cover all securities currently settling in EU CSDs. Nevertheless, T2S may process (under specific conditions) specific categories of securities which may not be fungible from a settlement process perspective.

Type of instructions

The scope of T2S:

"shall be restricted to settlement services, including the functionalities required to support settlement activities relating to asset-servicing business. Activities that extend beyond the provision of settlement services, such as the management of corporate actions, lie outside the T2S business scope. However, the system shall process the settlement instructions in relation to those CSD processes. T2S shall settle only those settlement transactions with a CeBM cash leg (or no cash leg). T2S will not provide settlement in commercial bank money (CoBM)" ***{T2S.02.020}***.

All rights reserved.

"T2S shall provide for a set of transaction types that allow transactions to be distinguished according to one or more of the following parameters:

- *priority;*
- *deadline;*
- *lifecycle type;*
- *matching mechanism; and*
- *settlement process."* ***{T2S.02.030}***

According to chapter 5.2.1 of the User Requirements:

"The instruction types covered by T2S are the following:

- *FOP (Free of Payment) consists of DFOP (Delivery Free of Payment) and RFOP (Receive Free of Payment). In both cases, securities are delivered / received without payment being made;*
- *DVP (Delivery versus Payment) and RVP (Receive versus Payment) define an exchange of securities for cash;*
- *DWP (Delivery with Payment) defines the delivery of cash and securities from one party to another. For example, trade netting by a CCP may result in such instructions;*
- *PFOD (Payment Free of Delivery) defines an exchange of cash without the delivery of securities;*
- *Settlement restriction (the action of setting or removing a settlement restriction) comprises the blocking, earmarking and reservation of positions within the overall position in a security in a securities account as well as the blocking and reservation of a cash balance in a T2S dedicated cash account.".*

2.1.1.2 Schedule and calendar

This section presents an outline of the T2S settlement day and calendar.

T2S settlement day structure

T2S provides a single harmonised timeframe for the centralised settlement procedures in central bank money.

The T2S settlement day is divided in several periods corresponding to different T2S statuses. The value of each T2S status corresponds to the ongoing period or main process of the T2S settlement day **{T2S.03.010}**.

The T2S status is defined by the value of the Status attribute in the *T2S System Status* entity. The following status attributes have been identified so far:

- The *"SOD period"*, for the start of day;
- For the night-time settlement period:
  - The "*First night-time cycle*",
  - The "*Last night-time cycle*",
- The "*Maintenance window*";

All rights reserved.

- The "Real-time settlement period";
- The "Real-time settlement closure period";
- The "*EOD period*" for the end of day.

The T2S Schedule may include two or more settlement cycles during the night time settlement. This will depend on settlement volumes by the time of T2S launch date. The current assumption is to process only two cycles: the first night-time cycle and the last night-time cycle. In case a new cycle is added, it will be created with the relevant sequence and processed between the first night-time cycle and the last night-time cycle.

The transition between the various periods is managed as an event and for each event T2S manages a planned time, a revised time and an effective time **{T2S.03.015}**.

The planned time corresponds to the standard schedule applied by default by the T2S platform for every T2S settlement day. This planned time can be updated by the T2S Operator only in case of a permanent change in the regular schedule **{T2S.03.016}**.

The revised time is the foreseen time for the current settlement day, which usually coincides with the planned time except when a delay has occurred. In contingency situations the T2S Operator updates the revised time while the planned time remains unchanged **{T2S.03.017}**.

The effective time is the time at which the event has actually occurred during the current settlement day **{T2S.03.018}**.

The T2S Operator is entitled to change the T2S Schedule deadlines of the settlement day, independently for a T2S settlement currency, in exceptional circumstances or contingency situations, based on a request by the relevant central bank. The change is valid for the current settlement day only **{T2S.03.303}**.

Overall view of the T2S settlement day

The overall T2S settlement day stands along the following timing and deadlines that are only indicative within the GFS document:

TIME	DESCRIPTION
18:45 – 20:00	SOD period
20:00 – 03:00	First night-time cycle Night-time static data update First night-time cycle reporting (22:20 set as target objective for completion time) [1] Last night-time cycle [2] Night-time static data update Last night-time cycle reporting (00:00 set as target objective for completion time) [1] Night-time reporting Last night-time static data update reporting

1 The time provided covers standard peak volumes and excludes extreme peak day volumetrics.

2 Following further detailed definition of the timing of the T2S Schedule and further consultation with T2S stakeholders, additional cycles may need to be added during the night-time settlement period.

All rights reserved.

TIME	DESCRIPTION
03:00 – 05:00	Maintenance window
05:00 (Or after NTS if NTS ends before 03:00) – 18:00	Realtime settlement (RTS)
18:00 – 18:45	EOD period

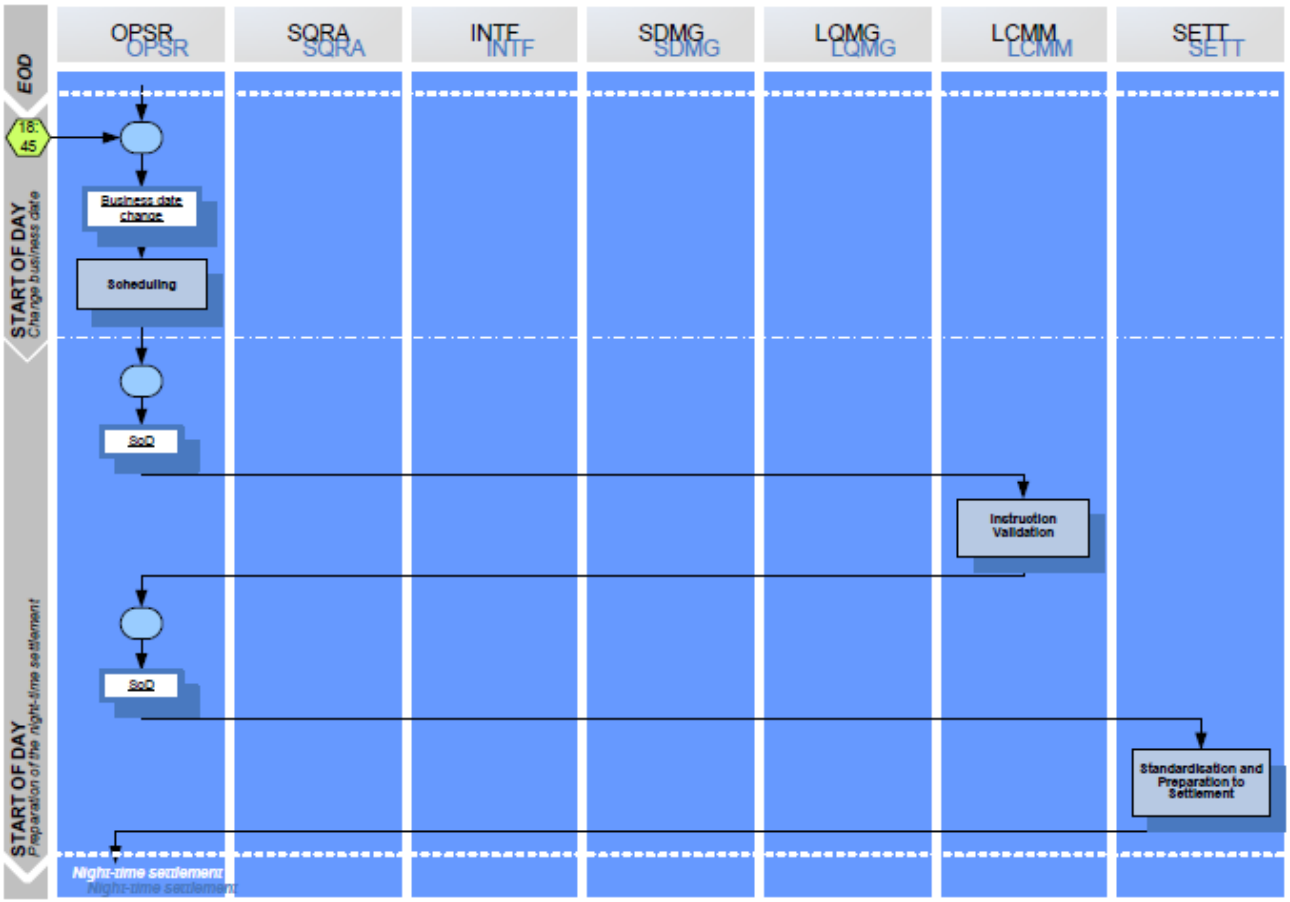
Each period of the T2S settlement day is described in the following diagrams including:

- The domains available (OPSR/Operational Services, SQRA/Statistics Queries Reports and Archiving, INTF/Interface, SDMG/Static Data Management, LQMG/Liquidity Management, LCMM/Lifecycle Management and Matching, SETT/Settlement);
- The event exchanged between the domains in order to trigger one specific module of the domain and the processing associated to a period;
- When appropriate the time which triggers the processing associated to a period.

T2S services are available throughout the T2S settlement day as per user requirements **{T2S.03.210} {T2S.03.220} {T2S.06.230}.**

SOD period

Diagram 1 - T2S Settlement day (SOD period)



All rights reserved.

The SOD period starts after the successful completion of the previous EOD period and ends before the start of the night-time settlement period.

During the SOD period, the following steps are processed successively:

- Change of business date: After successful completion of the previous EOD period and when time 18:45 is reached, the *Scheduling* module of the *Operational Services* domain **{T2S.03.020}**:
  - Changes the Business Date according to the T2S Calendar (or according to the *Closing Day* as being defined in section Calendar hereunder),
  - Creates the new T2S settlement day schedule on the basis of the default schedule of events;
- Preparation of the night-time settlement: after the change of the Business Date and prior to the start of night-time settlement period, there is a start of day procedure in order to prepare the night-time settlement **{T2S.03.030}**. During the preparation of the night-time settlement:
  - The *Instruction Validation* module of the *Lifecycle Management and Matching* domain selects the instructions that have to be revalidated against the static data valid as of the new Business Date **{T2S.03.050}**. The LCMM Instructions to be revalidated are:
    - The settlement instructions and settlement restrictions that finished the previous settlement day unsettled or partially settled and that are recycled for the current settlement day,
    - The amendment instructions, cancellation instructions and Hold/Release instructions that finished the previous settlement day not executed,
  - The *Standardisation* and *Preparation to Settlement* module of the *Settlement* domain valuates the *Securities Positions* and eligible *Settlement Transactions* for the new T2S settlement day for auto-collateralisation purposes **{T2S.03.060}** based on the prices received from collateral management systems and identifies the instructions eligible for settlement in the course of the new settlement day **{T2S.03.040}**.

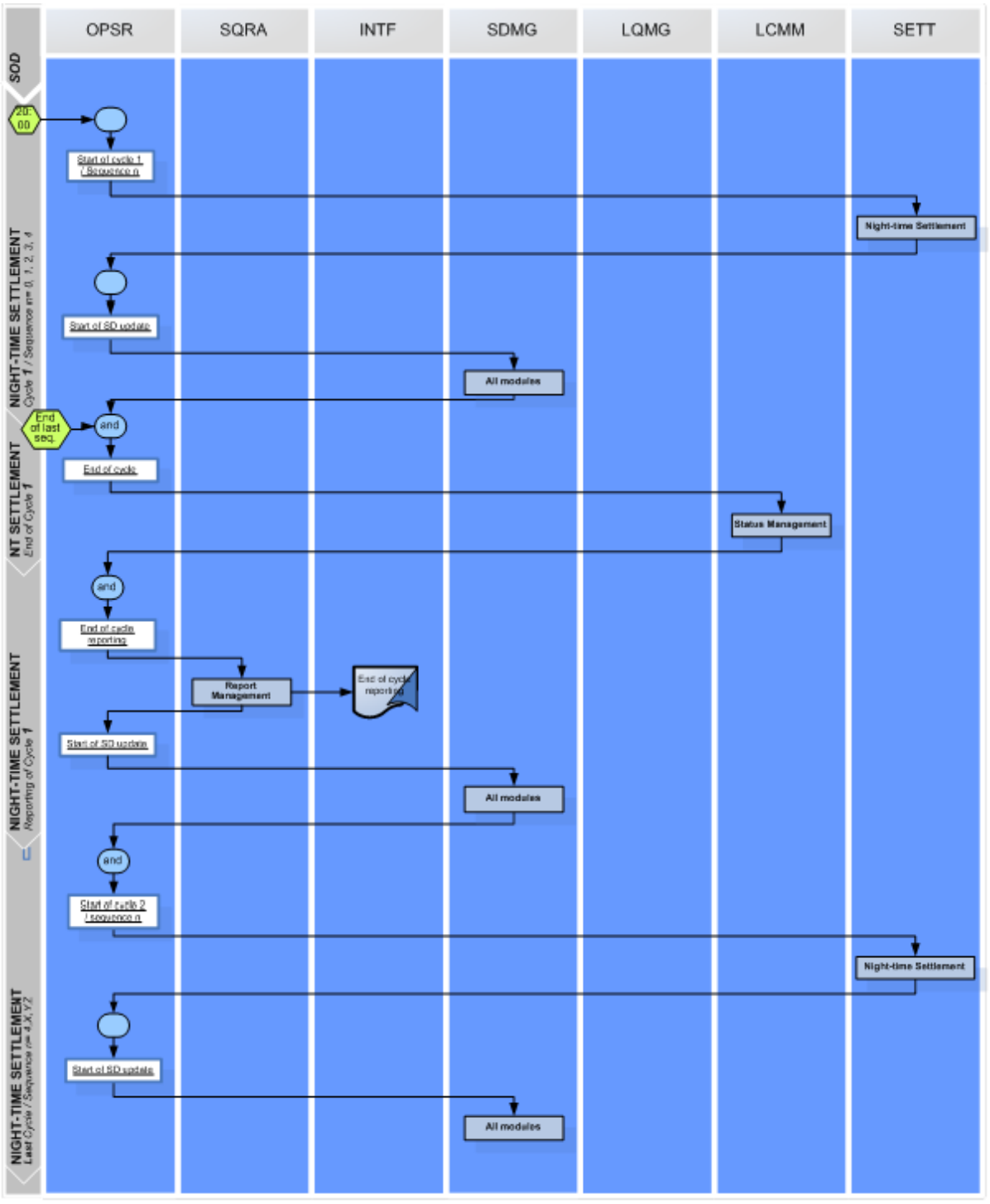
Service availability during SOD period

The LCMM, Static Data and Interface services are available during the SOD as per **{T2S.03.210}** **{T2S.03.220}** **{T2S.06.230}**.

Night-time settlement period

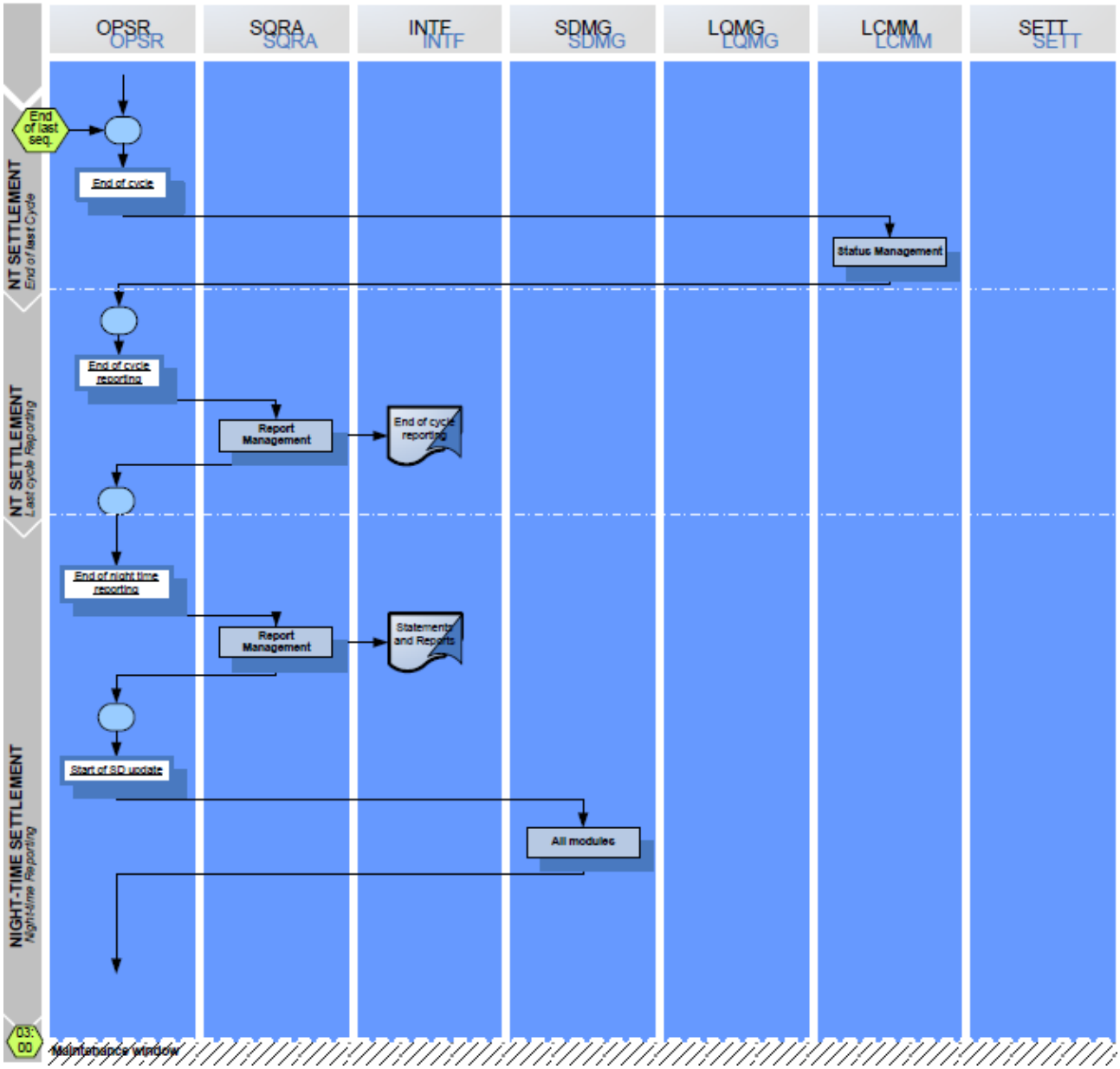
All rights reserved.

DIAGRAM 2 - T2S SETTLEMENT DAY (NIGHT-TIME SETTLEMENT PERIOD)



All rights reserved.

Diagram 3 - T2S settlement day (night-time settlement period)



The night-time settlement period starts after the successful completion of the previous SOD period and ends before the start of the maintenance window.

During the night-time settlement period, T2S provides a continuous settlement service by processing the settlement instructions within cycles and sequences with minimal time gaps between them **{T2S.03.090} {T2S.03.100}**.

The night-time period includes a number of statuses and corresponding main processes.

Sequencing

Settlement transactions are submitted to settlement according to a T2S automatic order, called sequence.

The first night-time cycle includes the following sequences:

All rights reserved.

FIRST NIGHT-TIME CYCLE	
Sequence 0	Only the following transactions sent to T2S before start of sequence 0 are proposed for settlement: ▪ Liquidity transfers from RTGS accounts to T2S Dedicated cash accounts ▪ Liquidity transfers from a T2S Dedicated cash account to another T2S Dedicated cash account of the same T2S Party **{T2S.03.070} {T2S.06.063} {T2S.06.230}**; ▪ The application of cash settlement restrictions corresponding to CoSD blocking which were regenerated by T2S at the end of the previous business day are processed in this sequence.
Sequence 1	Only the following transactions sent to T2S before start of sequence 1 are proposed for settlement: ▪ Corporate Actions [3] **{T2S.07.030} {T2S.09.270}**; ▪ All liquidity transfers **{T2S.06.230}**.
Sequence 2	Only the following transactions sent to T2S before start of sequence 2 are proposed for settlement: ▪ FOP transactions for rebalancing purposes (between securities accounts of the same T2S Party) **{T2S.07.040}**; ▪ All liquidity transfers **{T2S.06.230}**; ▪ Corporate Actions **{T2S.09.270}**. Sequence 2 includes transactions not settled in sequence 1 **{T2S.03.110}**.
Sequence 3	Only the following transactions sent to T2S before start of sequence 3 are proposed for settlement: ▪ Central Bank Operations (CBO) **{T2S.07.050}**; ▪ All liquidity transfers **{T2S.06.230}**; ▪ Corporate Actions **{T2S.09.270}**; ▪ FOP transactions as per sequence 2 **{T2S.07.040}**. Sequence 3 includes transactions not settled in sequence 2 **{T2S.03.110}**.
Sequence 4	All eligible transactions for the current T2S settlement day which are sent to T2S before start of sequence 4 are proposed for settlement. Sequence 4 includes CA on flows Sequence 4 includes transactions not settled in sequence 3 **{T2S.07.060} {T2S.03.110}**.

3 Corporate Actions refers only to Corporate Actions on stocks (events applicable on final securities positions, e.g. cash distributions). Corporate Actions on flows are events related to pending transactions (e.g. market claims and transformations) and are proposed for settlement during sequences 4.

All rights reserved.

The last night-time cycle includes the following sequences:

LAST NIGHT-TIME CYCLE	
Sequence 4	All eligible transactions for the current T2S settlement day which are sent to T2S before start of sequence 4 are proposed for settlement. Sequence 4 includes transactions not settled during the first night-time cycle **{T2S.07.070} {T2S.03.110}**.
Sequence X	All eligible transactions for the current T2S settlement day which are sent to T2S before start of sequence X are proposed for settlement. Partial settlement is applied to all unsettled transactions which are eligible for submission to partial settlement processing **{T2S.03.135} {T2S.07.080} {T2S.08.220}**. Sequence X includes transactions not settled in sequence 4 **{T2S.03.110}**.
Sequence Y	All eligible transactions for the current T2S settlement related to reimbursement of the "multiple liquidity providers"**{T2S.06.067}**.
Sequence Z	All eligible transactions for the current T2S settlement related to liquidity transfer **{T2S.06.220}**.

In case a new cycle is added between the first night-time cycle and the last night-time cycle, this complementary night-time cycle will be made of a sequence 4 composed of all the settlement transactions and a complementary night-time cycle reporting.

For each sequence and depending on the settlement transaction type involved:

- The *Night-time Settlement* module of the *Settlement* domain selects all the settlement transactions which (1) are received before the start of the sequence (2), corresponding to the criteria of the sequence and (3) are eligible for settlement at this sequence. The selection also includes settlement transactions not settled during previous sequences **{T2S.03.110}**. T2S proposes for settlement together as a collection all settlement transactions eligible for a given sequence **{T2S.03.080}**. The resulting bookings of this settlement proposal are processed on a gross basis (one by one);
- The T2S platform continuously accepts and validates static data maintenance instructions during the night-time settlement period. However, when intraday static data changes affect the settlement process, T2S platform processes these updates only during the processing periods between the different sequences **{T2S.03.220}**.

Reporting

Information on the status of the static data maintenance instructions are sent by T2S platform to T2S Actors immediately after end of processing of the static data entity update.

Settlement related messages (e.g. settlement confirmations and settlement failure notifications) are sent to T2S actors bundled into files during the night-time period. All other messages (e.g. matching notifications, reports, query responses, static data related messages) are sent individually **{T2S.13.080}**.

The *Report Management* module of the *Statistics, Queries, Reports and Legal Archiving* domain:

All rights reserved.

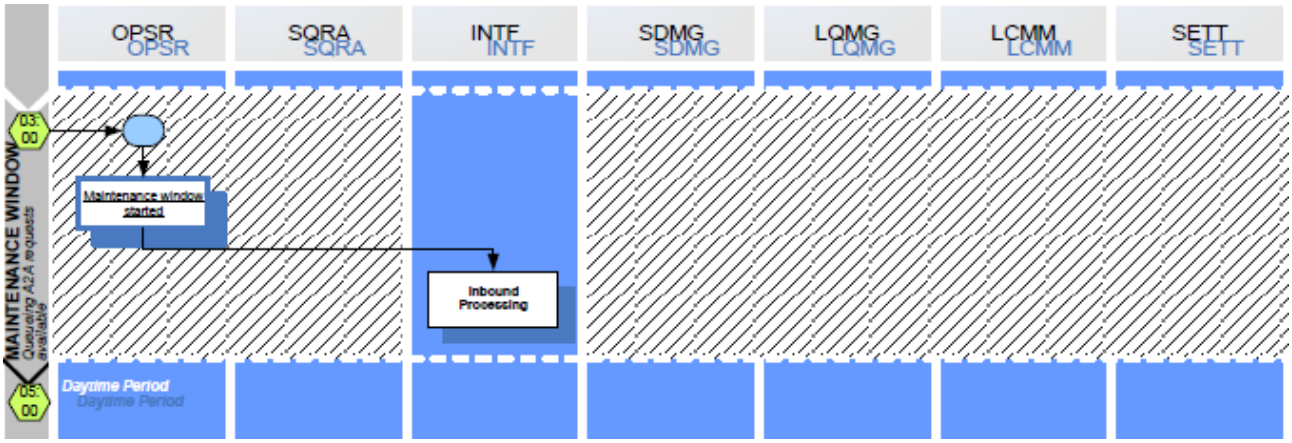
- At the end of the first night-time cycle reporting process, sends reports to T2S Actors on the results of the settlement attempts of the first night-time cycle **{T2S.03.120}**;
- At the end of the last night-time cycle reporting process, sends reports to T2S Actors on the results of the settlement attempts of the last night-time cycle **{T2S.03.120}**;
- At the end of the night-time reporting process, sends reports of the whole night-time period to T2S Actors **{T2S.03.130}**.

Service availability during night-time settlement

The LCMM, Static Data and Interface services are available during the night-time settlement periods as per **{T2S.03.210} {T2S.03.220} {T2S.06.230}.**

Maintenance window

Diagram 4 - T2S settlement day (maintenance window)



The maintenance window starts after the successful completion of the previous night-time settlement period and when time 03:00 is reached, and ends before the start of the daytime period.

During the "maintenance window":

- T2S has a technical window for system maintenance purposes **{T2S.03.140}** during the period where the lowest volumes and least critical settlement activity are expected **{T2S.03.150}**;
- The *Inbound Processing* module of the *Interface* domain starts the queuing of all requests received in A2A mode.

Service availability during the maintenance window

Except for *Interface* services, all T2S services are unavailable **{T2S.03.210} {T2S.03.220} {T2S.03.240}**.

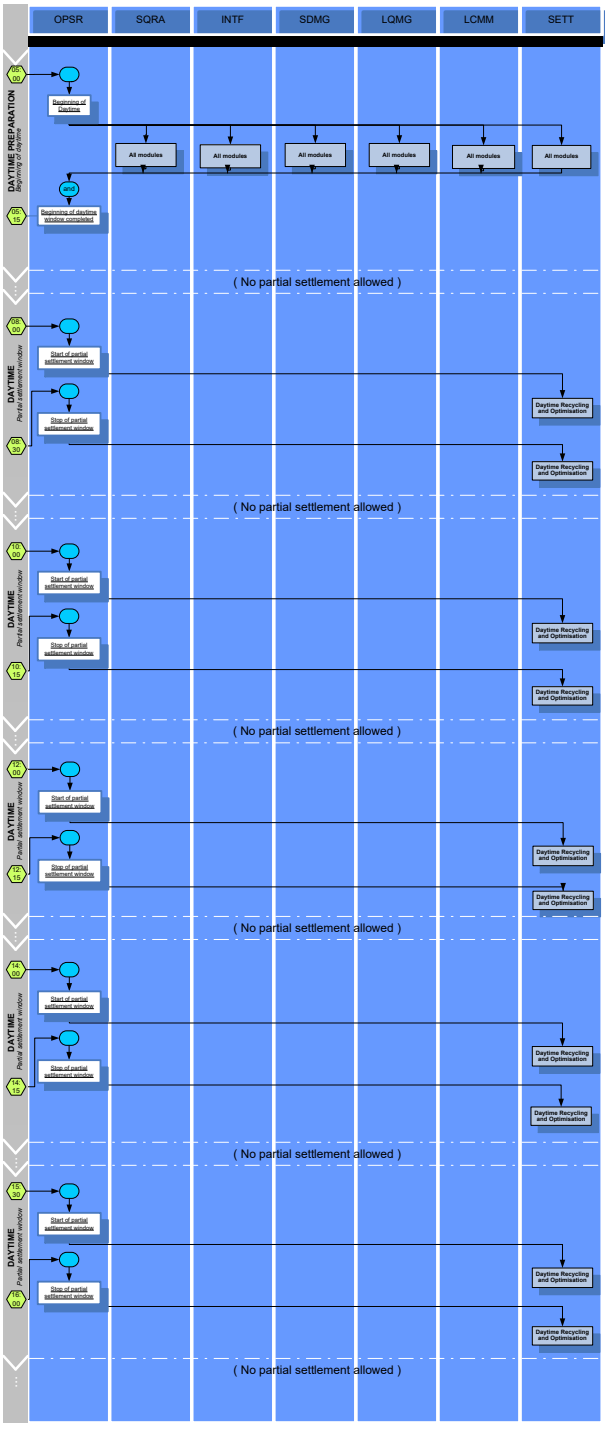
T2S *Interface* services are restricted, meaning that instructions and static data updates received in application-to-application mode during the maintenance window are queued for processing at the end of the maintenance window. The static data interfaces, in user-to-application mode, and the queries are not available during the maintenance window **{T2S.03.230}**.

Daytime period

All rights reserved.

DIAGRAM 5 - T2S SETTLEMENT DAY (DAYTIME PERIOD)

Diagram 5: T2S settlement day (daytime period)



All rights reserved.

DIAGRAM 6 - T2S SETTLEMENT DAY (DAYTIME PERIOD)

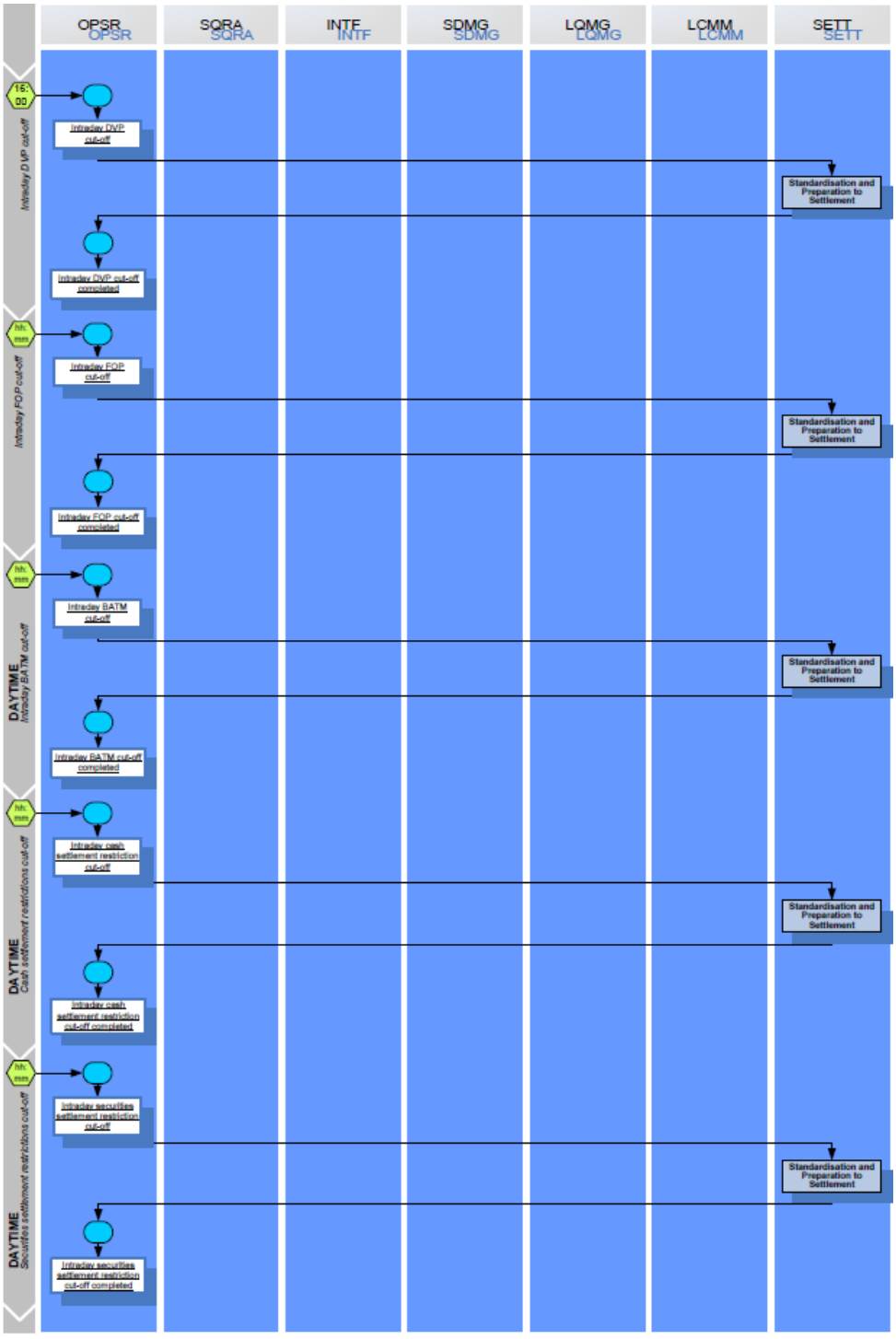
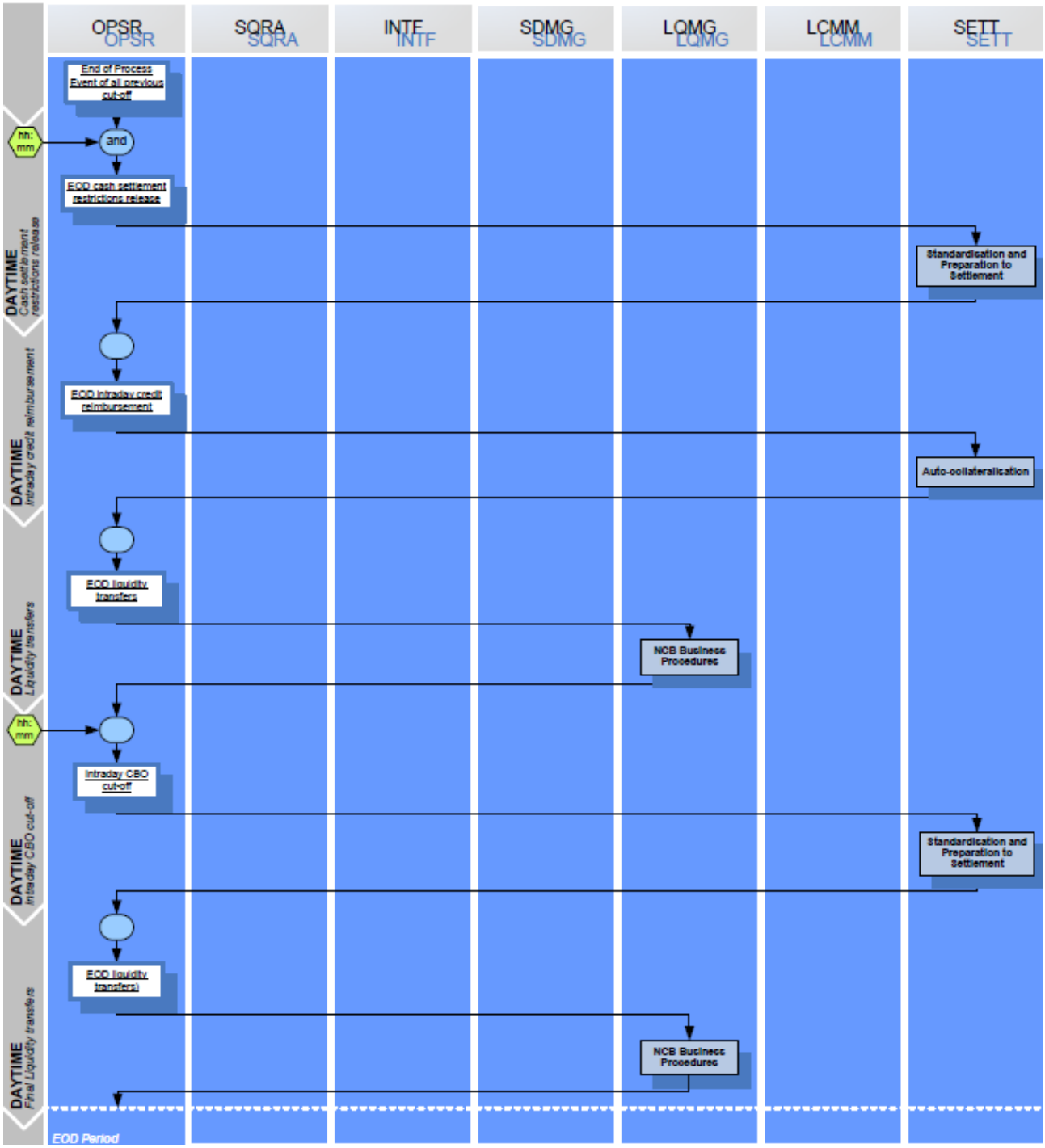


DIAGRAM 7 - T2S SETTLEMENT DAY (DAYTIME PERIOD)



The daytime starts when time 05:00 is reached with the daytime preparation allowing the preparation for the daytime settlement with the start of all domains/modules.

The day time period is mainly used for T+0 transactions and for settling transactions that failed to be settled during the night-time settlement period **{T2S.03.160}**.

During the daytime, the following processes are managed:

All rights reserved.

- Upon start and stop of partial settlement windows, the *Daytime Recycling and Optimisation* module of the *Settlement* domain allows partial settlement process in order to minimise the number and value of pending settlement instructions **{T2S.03.165}**;
- At the intraday DVP cut-off **{T2S.03.250}**, intraday FOP cut-off **{T2S.03.280}** and intraday BATM cut-off **{T2S.03.270}**:
  - Settlement instructions arriving after the deadline are moved to the night-time settlement on the next settlement day,
  - The *Standardisation and Preparation to Settlement* module of the *Settlement* domain processes, if not yet done, the Settlement Transactions corresponding to the related Settlement Instructions having entered T2S before the cut-off in order to submit them at least once to a settlement attempt during the current settlement day;
- At the intraday cash settlement restrictions cut-off and intraday securities settlement restrictions cut-off:
  - Settlement restrictions arriving after this deadline are moved to the night-time settlement on the next settlement day,
  - The *Standardisation and Preparation to Settlement* module of the *Settlement* domain processes, if not yet done, the settlement transactions corresponding to the related settlement restrictions having entered T2S before the cut-off in order to submit them at least once to a settlement attempt during the current settlement day;

The following processes, which are also known as end of day procedures throughout the T2S documentation, are managed at the end of the daytime period:

- For the end of day cash restrictions release, the *Standardisation and Preparation to Settlement* module of the *Settlement* domain:
  - Releases any blocked (including CoSD blocking) or reserved cash amount not used during the current settlement day,
  - Regenerates only cash settlement restrictions corresponding to CoSD blocking with an intended settlement date value equal to the next settlement day;
- For the end of day intraday credit reimbursement, the *Auto-collateralisation* module of the *Settlement* domain allows the reimbursement of all pending intraday credit with all available liquidity by releasing the "hold" reverse collateral settlement instructions;
- For the end of day liquidity transfer from the T2S Dedicated cash accounts to RTGS systems (i.e. cash sweep) **{T2S.03.180}**:
  - The *Inbound Processing* module of the *Interface* domain processes all liquidity transfers received prior to the beginning of the event and which have not yet been processed,
  - The *CB Business Procedures* module of the *Liquidity Management* domain creates liquidity transfers for the remaining cash of each T2S Dedicated cash account to automatically transfer it to the corresponding RTGS account;
- For the intraday CBO cut-off **{T2S.03.290}**:

All rights reserved.

- CBO settlement instructions arriving after this deadline are moved to the night-time settlement on the next settlement day,
- The *Standardisation and Preparation to Settlement* module of the *Settlement* domain processes, if not yet done, the settlement transactions corresponding to the CBO settlement instructions having entered T2S before the cut-off in order to submit them at least once to a settlement attempt during the current settlement day;

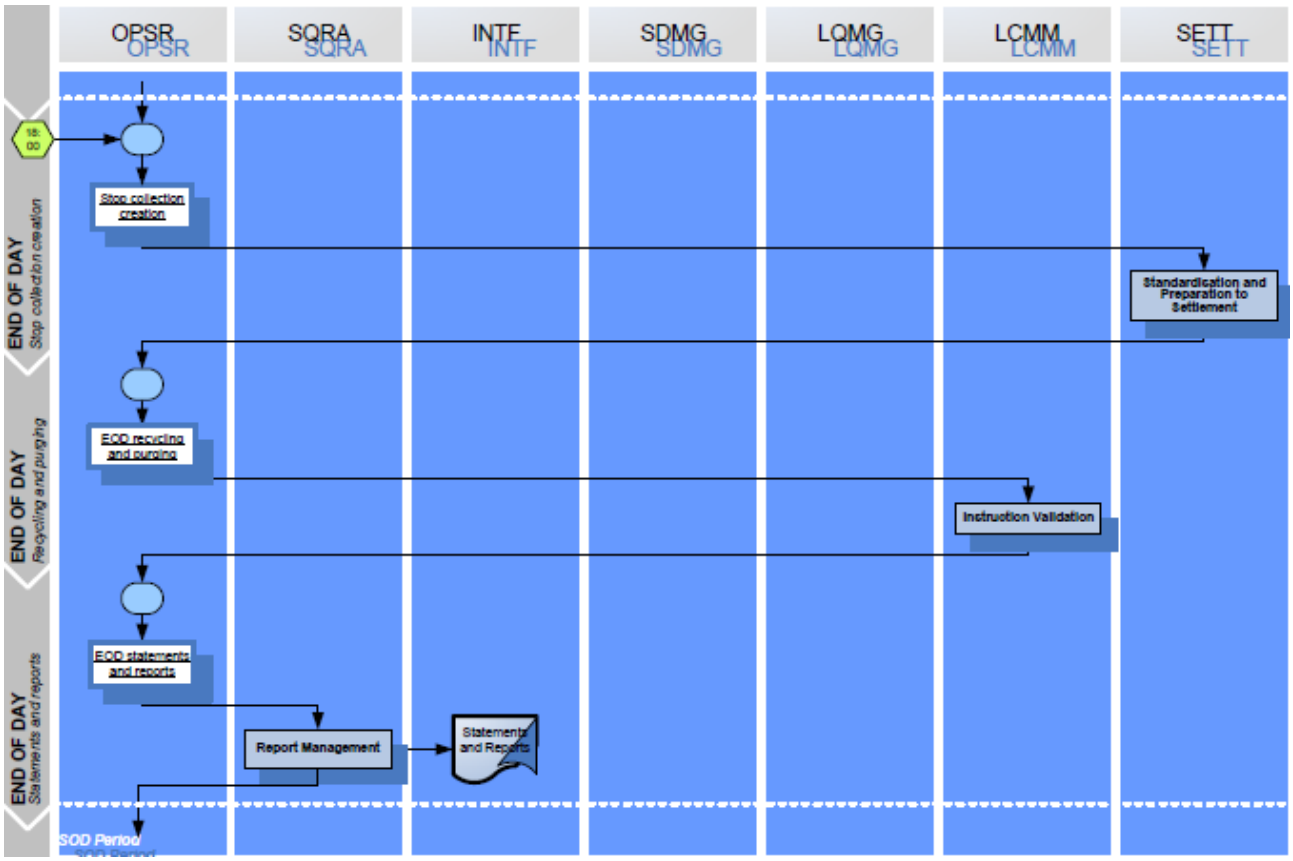
- For the final end of day liquidity transfer (i.e. final cash sweep) from the T2S Dedicated cash accounts to RTGS systems **{T2S.03.180}**, the CB Business Procedures module of the *Liquidity Management* domain creates liquidity transfers for the remaining cash of each T2S Dedicated cash account to automatically transfer it to the corresponding RTGS account.

Service availability during daytime period

The LCMM, Static Data and Interface services are available during the day-time settlement period as per **{T2S.03.210} {T2S.03.220} {T2S.06.230}**.

EOD Period

Diagram 8 - T2S Settlement Day (EOD period)



The EOD period starts after the successful completion of the daytime period following the event "Stop Collection Creation", meaning settlement is no longer possible, and finishes before the start of the next settlement day's night-time settlement **{T2S.03.170}**.

During the EOD period, the following processes are managed:

All rights reserved.

- For the stop collection creation, the *Standardisation and Preparation to Settlement* module of the *Settlement* domain stops the creation of the collection;
- For the end of day recycling and purging, the *Instruction Validation* module of the *Lifecycle Management and Matching* domain selects and cancels all settlement instructions and settlement restrictions that have past their last recycling day **{T2S.03.190}**;
- For the end of day statements and reports, the *Report Management* module of the *Statistics, Queries, Reports and Legal Archiving* domain creates and sends the end of day reports and statements to the T2S Actors having opted for that **{T2S.03.200}**.

T2S also ensures that the end-of-day securities position for the current business day for every securities account is equal to the previous end of business day's position plus the movements of the current business day **{T2S.03.195}**.

Service availability during EOD period

The LCMM, Static Data and Interface services are available during the EOD as per **{T2S.03.210}** **{T2S.03.220}** **{T2S.06.230}**.

Calendar

T2S is open for the settlement of the FOP instructions from Monday to Friday every week, independently of TARGET2 closing days **{T2S.03.305}**. For settlement in Euro central bank money (i.e. settlements against payment or free of delivery), the calendar is the same as the calendar of TARGET2 **{T2S.03.310}**.

For the settlement in non-Euro central bank money, the calendar is established according to the opening days of the relevant central bank **{T2S.03.320}**.

At the end of a Friday settlement day, T2S moves to the Monday settlement day and performs the related schedule until the end of the night-time settlement period (finishing during the night between Friday and Saturday). Realtime-settlement for Monday may already start, if the NTS ends before 3:00 am on Saturday. On Monday, T2S starts performing the schedule with the preparation of the daytime settlement as the continuation of the same settlement day or continue the RTS if started already before 3:00 am on Saturday **{T2S.03.340}**.

During the weekends, T2S interfaces and processes are not available on a regular basis **{T2S.03.350}**. Nevertheless, T2S should be technically capable to provide those services seven days a week based on specific needs (migration, issuance in direct holding countries), when required **{T2S.03.360}**.

2.1.2 T2S Actors

On the basis of selected extracts and abstracts of the User Requirements reflected in italics, this section of the GFS aims at describing the various T2S Actors. A *T2S Actor* is any legal entity or organisation interacting with T2S either directly or indirectly, i.e. through a CSD in T2S, for the purpose of securities settlement. The T2S Actors are:

- CSDs in T2S;
- T2S Parties;
- T2S Operators;

All rights reserved.

- Central Banks in T2S;
- Payment Banks.

T2S Actor is a business concept to be kept separate from the more technical concepts of a T2S System User - which is an individual or a technical process/application allowed to log into T2S with a login name and password - and role that defines the access rights a T2S System User is granted. As a consequence, the T2S services are not accessed by T2S Actors but, more precisely, by T2S System Users (belonging to these T2S Actors) linked to their relevant roles.

Although different, the aforementioned concepts are strictly linked: each T2S Actor, in fact, may have one or more T2S System Users whose role/s are different depending on the typology of the T2S Actor, meaning that there is a specific association among each T2S Actor type and the list of roles its T2S System User/s can be granted.

Below the definition of each *T2S Actor* typology is provided with a general description of the possible roles their respective T2S System User/s can be granted. Such roles are described, just as examples, from a business perspective, not being specified, for the time being, the exhaustive, detailed list of available privileges to be granted to each specific role.

2.1.2.1 CSD in T2S

A *CSD* in T2S is a CSD that (i) is recognised under Article 10 of the Settlement Finality Directive; (ii) settles in central bank money in a T2S eligible currency; and (iii) is a legal entity that has entered into a contractual relationship for the use of T2S.

"The CSD role classification shall include all T2S system users of a CSD participating in T2S. It does not include the T2S System Users of the CSD's participants. T2S makes no differentiation between the roles of Investor CSD and Issuer CSD. Most CSDs take on both aforementioned roles. With the exception of possible national specificities, T2S provides the harmonised scope of services to CSDs" ***{T2S.04.050}***.

1 – CSD System Administrator

This role is "responsible for:

- *the user administration for all of the CSD's T2S system users, including the assignment/de-assignment of roles and privileges;*
- *the configuration of roles with a set of privileges for the T2S system users of the CSD's T2S parties;*
- *the configuration of groups of secured* Reference *data objects with a set of individual secured* Reference *data objects for the T2S system users of the CDS's T2S parties;*
- *the T2S system users of the CSD's T2S parties;*
- *and the day-to-day monitoring of system applications, processes, and communication channels at the CSD.*
- *CSDs shall be responsible for defining and granting privileges to use functionality for their T2S parties. Therefore, it shall be possible for CSDs to configure roles and access rights for their T2S parties to functionality, based on their business requirements."* ***{T2S.04.060}***

2 – CSD business user

All rights reserved.

This role is "responsible for:

- *maintaining the CSD's securities account* Reference *data in T2S;*
- *the parameterisation of its securities account structure;*
- *maintaining T2S Party* Reference *data, including securities accounts, for its participants;*
- *maintaining CSD-specific instrument* Reference *data and, where applicable, the instrument* Reference *data across all CSDs;*
- *maintaining any settlement restrictions;*
- *the possibility of querying T2S dedicated cash account balances linked to the securities accounts of its participant at that CSD, when granted this privilege by the relevant CB and Payment Bank;*
- *maintaining privileges for all positions, settlement instructions and* Reference *data for the CSD and its participants that are required for business support."* ***{T2S.04.070}***

2.1.2.2 T2S Party

A *T2S Party* is a legal entity or, in some markets, an individual that has a contractual relationship with a CSD in T2S or with a CB in T2S for the processing of its settlement-related activities in T2S. It does not necessarily hold a securities account with the CSD. Examples of such parties (non-exhaustive) are:

- Indirect and direct CSD participants, (including those acting as Payment Banks for other CSD participants);
- Stock exchanges and multilateral trading platforms that route pre-match trades or settlement instructions to CSDs on behalf of trading participants;
- Central counterparts (CCPs);
- Central banks as CSD participants;
- CSDs as participants of other CSDs; and
- Securities processing outsourcer that process securities transactions on behalf of other financial institutions.

"The T2S Party role shall include all T2S System Users that a CSD maintains for the legal entities with which it has a legal relationship and which have direct connectivity to T2S. The model shall support two types of role: T2S Party system administrator and T2S Party business user" ***{T2S.04.080}***.

1 – T2S Party system administrator

This role is "responsible for user administration for all T2S system users of the T2S party of a specific CSD including the assignment/de-assignment of roles and privileges." ***{T2S.04.090}***

2 – T2S Party business user

"The scope of functions and processes that a T2S Party business user can access depends on the business services provided by the CSD. However, the data access of a T2S Party shall be limited to its own accounts, positions and transactions" ***{T2S.04.100}***.

All rights reserved.

2.1.2.3 T2S Operator

The *T2S Operator* is the legal and/or organisational entity/entities that operates/operate the T2S platform.

"The T2S operator is the top level of the hierarchical role and access rights model. The T2S Operator role classification includes all T2S System Users of the entity, which are responsible for the day-to-day operation and management of T2S. The T2S actors managed by this entity are CSDs and CBs participating in T2S. At the highest level, the T2S operator has access to all data and functionality in the subordinate level" ***{T2S.04.020}***.

A *T2S System User* of a *T2S Operator* can be granted access rights to functions and data of the platform according to the following roles.

1 – T2S System Administrator

This role is "responsible for:

- *the user administration for all T2S system users of the T2S operator;*
- *the user administration for the CSD system administrators;*
- *the user administration for the CB system administrators;*
- *the day-to-day monitoring of system operations, applications, processes, and communication channels;*
- *the configuration of privileges and default roles in T2S (...);*
- *the assignment/de-assignment of privileges to default roles and users of the T2S operator;*
- *the configuration of roles for T2S business and operations support users;*
- *the archiving of production data and the retrieval of archived data;*
- *contingency operations, e.g. starting and stopping processes outside of the normal operating schedule, in T2S;*
- *and the configuration of CSDs and CBs as system entities (...)* ***{T2S.04.030}***.

2 – T2S business and operations support user

This role is "responsible for:

- *maintaining T2S party* Reference *data, excluding securities accounts, for CSDs participating in T2S;*
- *maintaining T2S party* Reference *data, excluding T2S Dedicated Cash Accounts, for CBs participating in T2S;*
- *providing business and operations support to CSDs and CBs;*
- *maintaining T2S domains for global and Market-specific attribute lists, i.e. the valid list of values for a field (...);*
- *technical support (e.g. network and communications) for directly connected T2S parties;*
- *and query and maintenance of privileges and roles for all T2S actors for provision of business and operations support.*

All rights reserved.

Maintenance and query privileges of CSDs, the CSDs' participants, and CBs with respect to business data, such as securities and cash positions and transactions, shall be limited to contingency response situations only. The T2S system administrator shall restrict access to maintenance and query functionality to a subset of T2S business and operations support users, based on the support requirements of CSDs and CBs. For example, maintenance privileges in relation to a CSD could be limited only to the business support user for that specific CSD.

Staff on the T2S Service Desk shall have the role of T2S business and operations support." ***{T2S.04.040}.***

2.1.2.4 Central Bank in T2S

A *Central Bank* in T2S provides cash account services to banks for securities settlement in T2S in central bank money.

"The CB role classification shall include all T2S System Users of a NCB as a liquidity provider through T2S Dedicated Cash Accounts" ***{T2S.04.110}***.

1 – CB system administrator

This role is "responsible for:

- *the user administration for all T2S system users of the CB, including the assignment/de-assignment of roles and privileges;*
- *and the configuration of roles with a set of privileges for the T2S system users of the CB's participating payment banks;*
- *and the configuration of groups of secured* Reference *data objects with a set of individual secured* Reference *data objects for the T2S system users of the CB's T2S parties."* ***{T2S.04.120}****:*

2 – CB business user

This role "describes all T2S system users in CBs that require access to the static and transactional data of payment banks operating T2S Dedicated Cash Accounts. The role shall enable the T2S system user of the CB to:

- *maintain the payment banks with T2S Dedicated Cash Accounts as T2S parties;*
- *maintain the limits for payment banks on T2S Dedicated Cash Accounts;*
- *query all T2S Dedicated Cash Accounts for which the CB is responsible;*
- *query the credit line utilisation on T2S Dedicated Cash Accounts;*
- *grant/ revoke a CSD the privilege of querying T2S Dedicated Cash Account balances;*
- *identify the postings resulting in the utilisation of liquidity;*
- *identify the expected postings of cash on a T2S Dedicated Cash Account;*
- *identify the owner of every T2S Dedicated Cash Account;*
- *identify the cash leg of a settlement instruction(s), posted on the T2S Dedicated Cash Account by providing a unique transaction reference;*
- *and query the balances and postings on T2S Dedicated Cash Accounts for which the CB is responsible.*

All rights reserved.

However, it will not be possible for the CB to query the settlement instructions, securities transactions and securities positions of a T2S securities account unless the CSD participant and the CSD have granted this privilege explicitly to a CB for the securities account. This also includes the securities leg associated with a cash posting." ***{T2S.04.130}***

2.1.2.5 Payment Bank

A *Payment Bank* is either a central bank or a private bank used to settle the cash leg of securities settlements: it provides the cash account to support the settlement of the securities transactions of another financial institution in central bank money (CeBM).

"The Payment Bank role includes all T2S System Users of Payment Banks that require access to the T2S Dedicated Cash Account balances and postings of the T2S Dedicated Cash Accounts they provide for the purpose of securities settlement" ***{T2S.04.140}***.

1 – Payment Bank system administrator

This role is "responsible for the user administration of the T2S system users of the payment bank, including the assignment/de-assignment of roles and privileges" ***{T2S.04.150}.***

2 – Payment Bank business user

"The business user role for payment banks includes all T2S system users of payment banks providing a T2S Dedicated Cash Account for securities settlement. The role shall enable the T2S system user of the payment bank to:

- *maintain the limits for payment banks on T2S Dedicated Cash Accounts;*
- *grant/revoke a CSD the privilege of querying its T2S Dedicated Cash Account balances;*
- *maintain standing instructions for the transfer of liquidity between the relevant RTGS account and the T2S Dedicated Cash Account(s);*
- *query all its T2S Dedicated Cash Accounts and the balances on those accounts;*
- *query the credit line utilisation on T2S Dedicated Cash Accounts;*
- *query the postings resulting in the utilisation of liquidity;*
- *maintain limits for banks using their T2S dedicated account(s) for securities settlement;*
- *query the corresponding securities transaction of a cash posting against the T2S Dedicated Cash Account(s);*
- *and query the balances and postings on its T2S Dedicated Cash Account(s).*

It will not be possible for the payment bank to query the settlement instructions, securities transactions securities positions of a T2S securities account unless the CSD participant and the CSD have granted this privilege explicitly to the payment bank for the securities account. This also includes the securities leg associated with a cash posting." ***{T2S.04.160}***

All rights reserved.

2.1.3 Multi-currency in T2S

Although the primary focus of T2S shall be settlement services in Euro (Principle 9), according to Principle 10, T2S must be technically capable of settling currencies others than Euro. The T2S User Requirements Document specifies that T2S must be "multi-currency capable" from its first release **{T2S.02.040}**.

Furthermore, it should be possible for a granted user (*T2S system administrator*) to add a new currency **{T2S.16.340}**, to update it **{T2S.16.350}** and to delete it **{T2S.16.360}**.

Currencies for which T2S provides settlement in central bank money on T2S Dedicated Cash Accounts for securities transactions are called T2S Settlement Currencies. T2S Settlement Currencies are currencies for which an appropriate arrangement has been put in place between T2S and the central bank issuing the currency (or a central bank authorised to hold T2S Dedicated Cash Accounts denominated in this currency and to settle transactions on these accounts) to ensure cash settlements in central bank money on T2S Dedicated Cash Accounts denominated in this currency **{T2S.08.440}**. Only CeBM is accepted as a *T2S settlement currency*.

In spite of its multi-currency capability, T2S should not be considered as a foreign exchange settlement platform, since T2S does not provide any service to manage foreign exchange risks (e.g. cross-currency limits) or process cross-currencies settlements (no cross-currencies payment versus payment services):

- A *T2S Dedicated Cash Account* is denominated in only one currency;
- Each single settlement transaction has one single cash leg in one single currency;
- Cash netting is impossible between instructions having cash legs in different currencies.

However, T2S supports the settlement of T2S eligible securities issued in one currency and settled in another T2S Settlement Currency **{T2S.02.070}**, e.g. euro cash settlement are allowed for securities issued in non-euro currencies (and vice versa).

2.1.4 Other systems interacting with T2S

T2S also interacts with the following external systems:

- Euro and non-Euro RTGS systems providing liquidity, in CeBM, for settlement purposes;
- Collateral management systems providing data for collateral evaluation.

Although distinguished from the *T2S Actors* such systems exchange communication with T2S passing through the *Interface* domain.

2.1.4.1 Euro and non-Euro RTGS systems

Depending on the decision taken by the relevant *CB*, T2S interacts with Euro and non-Euro RTGS systems for the processing of liquidity transfers from such systems to T2S and vice versa, aiming at providing *T2S Dedicated Cash Accounts* with the liquidity needed for settlement.

2.1.4.2 Collateral management systems

T2S receives from collateral management systems a daily update of the Reference data for the valuations of securities positions to be used during the auto-collateralisation process.

All rights reserved.

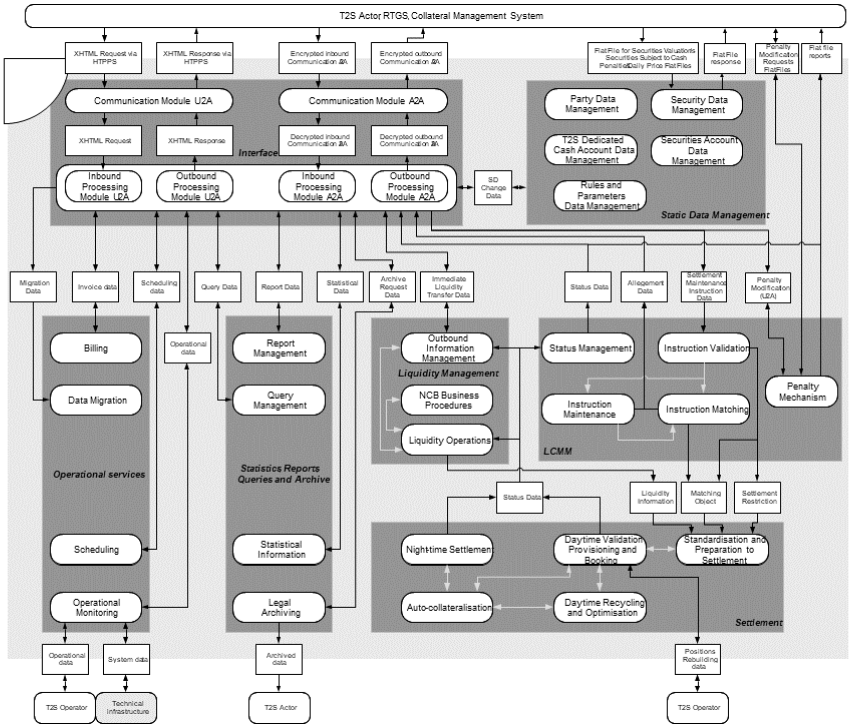
2.2 Overall high level diagram

The high-level functional diagram provides an overview of T2S at domains level. It refers to a functional conceptual representation, irrespective of the technical choices to be retained for the implementation of the aforementioned domains.

The following elements are presented on the diagram:

- Domains;
- Modules;
- Data-flows exchanged between the modules and with the outside of the platform.

Only a limited number of flows figures on the diagram in order to limit the complexity of the representation. The exhaustive list of detailed flows can be found in the domains and modules descriptions of chapter 3.



2.3 Description of functional domains

This section of the GFS provides a high level description of the role of each domain depicted above in the overall high-level diagram and recalls the different modules and functions covered by each domain and described in a more detailed way within the chapter 3.

All rights reserved.

2.3.1 Interface

The T2S *Interface* domain handles all incoming and outgoing communication with all *T2S Actors* [4] and other external systems [5] (such as RTGS systems and Collateral Management Systems), manages the use of the appropriate communication medium and conducts the relevant technical entry checks. Communication from T2S Actors connecting to T2S must have to comply with the formats and specifications defined in T2S.

T2S supports the connectivity of T2S Actors as follows:

- T2S communication is available by using messages or files containing messages in Application-to-Application mode (A2A) that allows direct communication between software applications via XML messages as well as through online-screen based activities in User-to-Application mode (U2A) for activities performed by T2S System Users.
- For the T2S communication via messages the ISO 20022 is the single standard, concerning both inbound and outbound communication. In addition, the T2S *Interface* complies with Giovannini protocol recommendations. Therefore, all messages exchanged between T2S and T2S Actors are based on XML technology and comply with the ISO 20022 standards on messages. They are sent to T2S either individually or in a file containing one or several messages.

The T2S *Interface* services are available continuously during settlement days. However, their availability is restricted during the Maintenance Window and is not guaranteed during weekends and closing days (except in exceptional cases).

The T2S *Interface* domain includes three modules:

- *Communication Module*;
- *Inbound Processing Module*;
- *Outbound Processing Module*.

2.3.1.1 Communication Module

The main purpose of the *Communication Module* is to ensure secure and reliable communication between the T2S platform and T2S Actors. T2S System Users can use Application-to-Application (A2A) and User-to-Application (U2A) communication channels to access the T2S platform. The *Outbound Processing Module* for Application-to-Application mode provides a single set of standard messages to facilitate communication with multiple external RTGS systems and with multiple external collateral management systems.

Additionally, the *Communication Module* ensures protection of the T2S platform against an intrusion and unauthorised access. It validates that only trusted party transmits the inbound communication through a secure channel. It provides an authentication functionality to verify the identity of the T2S System User.

The *Certificate Management* functionality is closely linked to the network providers and connectivity services.

---

4 For some tasks of the T2S operator (e.g. operational monitor, technical tools, and alternative access in emergency situations) a direct access to the relevant tool is envisaged.

5 See chapters 2.1.2 "T2S Actors" and 2.1.4 "Other systems interacting with T2S" for a detailed description.

All rights reserved.

2.3.1.2 Inbound Processing Module

The *Inbound Processing Module* receives

- *U2A - XHTML request;*
- *Stored inbound communication – A2A file;*
- *Stored inbound communication – A2A individual message*

from the *Communication Module* and performs a series of technical verification checks. The processing of this module includes the *Authorisation Check* which is based on privileges.

As the availability of T2S is restricted during the maintenance window and the night-time settlement, the *Inbound Processing Module* for Application-to-Application mode includes queuing and restart functionalities, which allow the deferred processing of requests.

The *Inbound Processing Module* allows the initiation of resending of messages or files. This functionality can be used in contingency situations if the messages or files have been lost during the communication or due to technical problems on the receiver side.

The *Inbound Processing Module* includes the extraction of all relevant business data from the received request in order to structure them in an internal format, determines the relevant T2S back end modules and routes the structured data to them. In case of a rejection (e.g. the privilege for the requested action is not assigned to the T2S System User) the *Interface* creates the respective *inbound processing rejection* and delivers it to the *Outbound Processing Module*.

2.3.1.3 Outbound Processing Module

The *Outbound Processing* Module receives the data in the internal format from the processing modules of other domains. It checks the message subscription preferences, generates the relevant messages in the necessary format and sends these messages (if relevant bundled into files) via the *Communication* Module to the T2S System Users in U2A mode or to the Party Technical Address in A2A mode.

It is ensured that outbound communication reaches the appropriate party technical address and is delivered on due time to the receiving T2S actors.

2.3.2 Reference Data Management

The Common Reference Data management component (CRDM) provides T2S System Users with an integrated and consistent set of common information. It is the single access point for the creation, update and deletion of Reference data "relevant" for T2S in performing its functions, such as parties, securities, securities accounts and T2S Dedicated Cash Accounts among others.

T2S System Users of this domain belong to CSDs, CBs, T2S parties, Payment Banks and the T2S Operator - each T2S system user accessing and using Reference data management facilities according to its own specific access profile. Different T2S system users have different roles assigned and, as a consequence, are allowed to see and possibly change different pieces of information.

All rights reserved.

T2S System Users of the T2S Operator belong to a specific user category devoted to system administration activities. With respect to Reference data management, they are responsible for entering and managing CSDs and CBs data and a set of global rules and parameters. They can also act on behalf of other user categories in order to perform some specific actions or within some pre-defined contingency scenarios.

All changes in Reference data can be executed both in Two-Eyes and Four-Eyes mode, the actual mode to be adopted being established by the data owner. Proper functions are envisaged in order to allow each domain of the system to access the needed set of Reference data or to feed the other domains using some specific load procedures.

Versioning facilities allow the implementation of data history and data revision features, in order to keep track of all past data changes and to enter changes meant to become effective at a future date.

The Reference *Data Management* functions are grouped in five modules, each module being in charge of a given subset of Reference data objects: *Party Data Management, Security Data Management, Securities Accounts Data Management, T2S Dedicated Cash Account Data Management and Rules* and *Parameters Data Management*.

2.3.3 Lifecycle Management and Matching

The *LCMM* domain deals with instructions received through the Interface domain. It is responsible for (i) the validation and matching of settlement instructions, before they are submitted to the *Settlement* domain, and (ii) the management and execution of maintenance instructions, and (iii) the computation and management of Penalties for failed settlement instructions. This domain is also in charge of checking the possible impact of reference data changes on pending instructions, managing the revalidation and the consequences of such impact when relevant, while keeping tracks of the changes in the lifecycle of instructions. The services provided by this domain are available continuously during the whole day T2S operating hours, with the exception of the maintenance window.

The *LCMM* domain encompasses five modules:

2.3.3.1 Instruction Validation

The *Instruction Validation* module validates all incoming instructions received during the settlement day, based on a harmonised set of validation rules. This module checks the consistency of incoming *Inbound LCMM Messages (Incoming Settlement Instructions, Settlement Restrictions* and maintenance instructions) sent by a CSD or by directly connected T2S Party and *LCMM Instructions* with T2S Reference Data. No syntax (or format) checks are performed by this module, as this kind of validation is carried out by the *Interface* domain.

If an *Inbound LCMM Message,* a *Settlement Instruction* or a *Settlement Restriction* fulfils a *Restriction Processing Type*, this module is in charge of rejecting the *Inbound LCMM Message* or putting the *LCMM Instruction* On Hold.

This module is also in charge of revalidating all the instructions at the SOD (Start of Day) of every business day or those instructions that can have been impacted by a reference data change.

All rights reserved.

2.3.3.2 Instruction Matching

The *Instruction Matching* module is responsible for matching *Settlement Instruction* that requires to be matched in T2S. It compares the settlement details provided by the buyer and the seller of securities to ensure that both parties agree on the settlement terms of the transaction. The *Instruction Matching* module matches trading instructions in a standardised way (compliant to ECSDA matching proposals).

The *Instruction Matching* module is also responsible to create the corresponding *Matching Object* for each pair of matched *Settlement Instructions*, assigning them a T2S Matching Reference.

Those instructions which do not require matching are forwarded from the *Instruction Validation* module to the *Standardisation and Preparation for Settlement* module for their processing, and therefore are not under the scope of the *Instruction Matching* module.

The Instruction Matching module routes each *Matching Object* and the related *Settlement Instructions* to the *Settlement* domain for further processing.

The Instruction Matching module is available continuously during settlement days, with the exception of the maintenance window.

2.3.3.3 Instruction Maintenance

The Instruction Maintenance module handles maintenance instructions that:

- Cancel, hold, release or partially release a *Settlement Instruction* and amend its process indicator;
- Cancel, hold or release *Settlement Restriction*;
- Release or cancel a *Settlement Instruction* for Conditional Securities Delivery purpose.

This module is also in charge of the cancellation of unmatched *Settlement Instructions* that remain as such after a standard period beyond their Intended settlement date, or the date of their last status value change, and also those instructions which have reached the end of their recycling period. The module forwards all instruction status values updates to the *Status Management* Module.

2.3.3.4 Status Management

The *Status Management* module receives status values changes information from the *LCMM* modules and *Settlement* domain, analyses it, collects the relevant data, and forwards it as *Message Data* to the T2S *Interface* domain for transmission to the directly connected T2S parties and CSDs as per the message subscription service, in a consistent way. All the modules dealing with changes in the status values of the *Settlement Instructions* (and/or the *Settlement Instruction* itself), *Settlement Restrictions* and maintenance instructions, activate this module by providing it with relevant information on the changes carried out as *Instruction Status Information* or *Maintenance Status Information*. This module also collects the data for sending *Allegement* messages in case of Unmatched *Settlement Instructions*, when needed to the relevant T2S Parties, via *Interface* domain.

All rights reserved.

2.3.3.5 Penalty Mechanism

The Penalty Mechanism module is responsible for the daily calculation and preparation for reporting of Penalties for failed settlement instructions as well as the complementary features, which are necessary for T2S Actors in this context. Every business day, the Penalty Mechanism module analyses all the failed Settlement Instructions from previous business day in order to compute the necessary Penalties. This module also handles requests from CSDs to modify an existing Penalty, as well as the recalculation of Penalties that have been modified or impacted by a reference data change. This module prepares the information, including the aggregation of amounts (daily or monthly), for the ad-hoc reports for Penalties.

2.3.4 Settlement

The modules belonging to this domain are in charge of the actual booking of *Settlement Instructions*, *Settlement Restrictions* and *Liquidity Transfers* received from the other domains. The settlement occurs on a continuous basis during the daytime, and in sequences, grouped in pre-defined cycles, during the night-time.

This domain creates settlement transactions in a standard format, taking notably into account specific use cases such as Cross-CSD Settlements and Conditional Securities Delivery. Settlement transactions are either submitted to the daytime settlement in real-time or made available for the night-time settlement (*Standardisation and Preparation to Settlement* module).

During the daytime, a first settlement is systematically attempted (module *Daytime Validation, Provisioning and Booking*). The outcome of the settlement always triggers *Daytime Recycling and Optimisation* functions. These functions identify settlement transactions to be submitted to another settlement attempt, with an expected success, due to settled transactions bringing new resources, or due to unsettled transactions resolving gridlocks (*Daytime Recycling and Optimisation* module).

During the night-time, settlement transactions are grouped into sequences, submitted to optimisation procedures in order to identify sets of settlement transactions that can settle successfully, and then booked (*Night-time Settlement* module).

Both for daytime and night-time, auto-collateralisation procedures are used in order to reduce the number of cases with lack of cash, lack of securities or insufficient external guarantee headroom (*Auto-collateralisation* module for auto-collateralisation with a central bank or with a payment/settlement bank).

Furthermore, the settlement resorts to partial settlement procedures in case of lack of securities, if requested and allowed, according to a partial window event received from the *Scheduling* module.

Lastly, T2S generated settlement instructions and liquidity transfers are created in this domain:

- Whenever the settlement requests additional settlement transactions to process realignment, conditional securities delivery or auto-collateralisation, and corresponding *Settlement Instructions* have to be generated;
- Whenever the settlement requests additional settlement transactions, to process corporate action rebalancing liquidity detected on *Settlement Instructions*, or auto-collateralisation reimbursement and corresponding *Liquidity Transfers* have to be generated.

All rights reserved.

2.3.4.1 Standardisation and Preparation to Settlement

The *Standardisation and Preparation to Settlement (SPS)* module is the entry module in the *Settlement* domain for *Settlement Instructions, Settlement Restrictions* and *Liquidity Transfers* to be settled.

It analyses the context of the settlement to be processed (Cross-CSD, In/Out T2S, CoSD, Blocking/Reservation/Earmarking, or standard settlement) and creates accordingly, on the intended settlement date, the associated settlement transactions including the amount of cash and/or securities to be settled.

The module also creates, when relevant, T2S generated settlement instructions:

- for realignment, at the validation or matching date;
- for CoSD at the intended settlement date.

It then either puts the settlement transactions into a collection immediately sent to the *Daytime Validation, Provisioning and Booking* module, or leaves them available for later selection by the *Night-time Settlement* module.

Similarly, this module handles flows received from the *Instruction Maintenance* module with the aim to update the relevant settlement transactions and the relevant T2S generated *Settlement Instructions* according to their context of settlement.

Lastly, the *SPS* module also includes complementary functions (*Cut-off Processing* and EOD/SOD Processing functions) that handle respectively cut-off and end of day/start of day events in order to trigger the relevant process in the *Settlement* domain.

2.3.4.2 Daytime Validation, Provisioning and Booking

The *Daytime Validation, Provisioning and Booking* module updates, in daytime, the securities positions, the cash balances and the limit headroom associated to credit memorandum balances (CMB). These updates take place based on the securities and cash movements of the incoming collection of settlement transactions and in accordance with the resources available.

To that purpose, the module receives collections containing a set of settlement transactions, either for their first settlement attempt or following an optimisation or recycling process.

The module settles all the received settlement transactions of a collection in an "all or none" basis and does not attempt any selection or de-selection inside a collection.

2.3.4.3 Daytime Recycling and Optimisation

This module selects and submits to another settlement attempt in daytime, unsettled settlement transactions that previously failed to settle.

It is triggered by a *Collection Status Information* event and launches, either optimisation functions when receiving unsettled collections, or recycling functions, when receiving settled collections.

2.3.4.4 Night-time Settlement

This module performs the settlement, during the night-time period only, of settlement transactions which are selected through sequences grouped into a predefined number of cycles.

All rights reserved.

These collections are made of new settlement transactions, created before the beginning of the sequence, and, settlement transactions which failed to settle in a previous settlement attempt, notably the settlement transactions recycled at the Start of Day.

Each sequence runs a succession of optimisation procedures to identify settlable collections which are booked in the module.

2.3.4.5 Auto-collateralisation

This module implements the provision of intraday credit, through auto-collateralisation with a CB or a payment/settlement bank, by creating additional collateral settlement transactions, in case of lack of cash or in case of insufficient external guarantee headroom (CMB), identified at the settlement of settlement transactions.

Such collateral settlement transactions for auto-collateralisation with CBs fit to each CB context (pledge sub, pledge or Repo) through pre-defined collateralisation procedures. In case of auto-collateralisation with a payment/settlement bank, the Repo context is systematically used.

Auto-collateralisation is offered both on stock and on flow for eligible counterparts and eligible instructions, both during night-time and real-time settlement windows.

In addition to this main feature related to lack of cash or to insufficient external guarantee headroom (CMB), the module also manages the dynamic reimbursement or automated substitution, in case of fail during settlement, due to a lack of securities already collateralised.

Lastly, this module handles the reimbursement of intraday credit, due to a decreased CB limit during the settlement day, or due to the end of day intraday credit reimbursement process.

2.3.5 Liquidity Management

The *Liquidity Management* domain is responsible for all the activities related to liquidity transfers between RTGS accounts and T2S Dedicated Cash Accounts as well as between two *T2S Dedicated Cash Accounts*. The domain encompasses three modules and performs the overall preparation of immediate, predefined and standing liquidity transfer orders on the *T2S Dedicated Cash Accounts*. Furthermore it triggers the related communication between T2S and the involved RTGS system.

The initiator of a liquidity transfer order can be both a payment bank as the account holder of the T2S Dedicated Cash Account and another party which is authorised by the account holder.

When initiating a *Liquidity Transfer* between T2S Dedicated Cash Accounts and the RTGS system, the liquidity transfer order is checked and treated according to the specific features of the different liquidity transfer types by the *Liquidity Operations* module.

Liquidity transfers between an RTGS system and T2S are based on the exchange of inbound and outbound liquidity transfers and on the use of dedicated transit accounts (both in the RTGS system and in T2S) managed by the *Outbound Information Management* module in order to enable the other system to align its accounting.

All rights reserved.

At the end of the settlement day, liquidity available on T2S Dedicated Cash Accounts is automatically transferred to the relevant RTGS accounts and the specific end-of-day procedures are implemented by the *CB Business Procedures* module.

2.3.5.1 Liquidity Operations

This module ensures the checking and management of liquidity transfer orders between two T2S Dedicated Cash Accounts or towards RTGS system (and vice versa), taking notably into account the specific features of the different transfer types.

2.3.5.2 Outbound Information Management

Liquidity transfers between an RTGS system and T2S require the use of technical transit accounts. This module manages inbound and outbound liquidity transfers in order to enable the other system to align its accounting. Moreover, it provides information on special booking events (e.g. ceiling, floor, and partial/no execution) to T2S Actors.

2.3.5.3 CB Business Procedures

This module is responsible for the end-of-day liquidity transfers between T2S and an RTGS system. It is in charge of the automated transfer of surplus liquidity at the end of the day.

2.3.6 Statistics, Queries, Reports and Archive

This domain includes modules for data exploration services available to the T2S System Users according to their needs in terms of time scope, nature of data (detailed or aggregated), flexibility of the tools and response time.

2.3.6.1 Statistical Information

The *Statistical Information* module provides T2S System Users (i.e. the T2S operator and, on an optional basis, CSDs, CBs and authorised directly connected parties) with business intelligence tools for statistical analysis and as a decision support system.

The scope of this module is twofold:

- To provide historical data and statistics to the T2S operator and the CSDs on the level of use of the different components of T2S over time, to support the operational management of the system;
- To offer to CSDs, CBs, T2S operator and possibly authorised directly connected parties, historical data and statistics for regulatory reporting purposes and, on an optional basis, wider scope statistical analysis.

Both repositories are separated from the live data repositories, in order to provide an easy access to high quality and business oriented data without the risk of impacting the performance of the operational settlement environment.

All rights reserved.

In the present document, one scope(short term) is covered by the single *Statistical Information* module. Nevertheless, this general description will be followed, during the next phase of the project, by detailed specifications aiming at serving as a basis of the design of the technical (development and infrastructure) solution(s): (i) one covering the access to historical data and statistics needed for the monitoring and management of T2S by the T2S operator and the CSDs and (ii) the second covering access to historical data and statistics for regulatory reporting purposes by CSDs and CBs, T2S operator and possibly authorised directly connected parties and, on an optional basis, wider scope statistical analysis.

The *Statistical Information* module provides functions to manage historical and statistical data repositories, in order to perform historical and statistical query and reporting and multi-dimensional analysis.

The specific set of available historical and statistical data and functions depends on the specific privileges of each T2S System User.

2.3.6.2 Query Management

The *Query Management* module allows different categories of real-time queries and historical queries on the production data (e.g. settlement instructions, securities positions, cash balances, reference data, and audit trails). It extracts the requested data from their respective data stores and sends it back to the *Interface* domain.

2.3.6.3 Report Management

The *Report Management* module provides T2S Actors with a set of defined reports periodically (settlement instructions, balance and reference data reports) which, however, do not have to cover the regulatory reporting. T2S generates reports as XML messages and the reports comply with ISO 20022 where an ISO report standard is available.

All reports are available for all T2S Actors. T2S sends reports to T2S Actors, based on the data of the party and user for which the report is configured. T2S triggers the creation of a report based on a business event or pre-defined, scheduled time.

2.3.6.4 Legal Archiving

The *Legal Archiving* module stores for a given period processed inbound and outbound messages [6] . Events, sent by the *Scheduling* module or on request of the T2S Operator, trigger the processing within the module.

2.3.7 Operational Services

This domain includes modules providing functions specific to the T2S operational teams.

2.3.7.1 Operational Monitoring

The aim of this module is to support the T2S Operator and, for some specific tasks, other T2S System Users (e.g. online access to the Trouble Management System for CSDs and authorised T2S parties) in the monitoring of T2S, from two different angles:

6 The module does not address the needs for Data Revision (replication of reference data structures as revision tables) and Data History (data stored with "valid from" and "valid to" date attributes) described in UR section 16.4.

All rights reserved.

- The operational monitoring for the detection of functional or operational problems in real-time, the monitoring related to the SLA indicators, and the information provisioning for crisis management scenarios;
- The technical monitoring for the detection of hardware and software problems via real-time monitoring of all the technical components involved in the processing, including the network connections.

Furthermore, this module provides an overview of the message flows in the whole system to the T2S operator.

2.3.7.2 Scheduling

This module contains a set of functions for the management of operating day events and related business processes. The current business date event schedule can be changed at run-time, in order to insert, update, time-shift and complete an event instance or a set of events. The broad categories of events are EOD/SOD events, End of cycle events, Maintenance events and Dynamic events.

2.3.7.3 Billing

This module automatically or on request produces monthly bills containing all billable events (e.g. events related to the lifecycle of an instruction), fixed and variable fees. At the end of a billing period, an invoice is sent to the CSDs and sums up all the relevant billing information per T2S Party. All invoices are stored electronically and are available for later inquiries.

2.3.7.4 Data Migration

The *Data Migration* module provides the migration functionality which allows the CSDs and the CBs to automatically transfer the major part of relevant data (e.g. securities account data, party reference data, securities reference data, T2S Dedicated Cash account data…) to T2S. The CSDs and CBs can send their data via the standard channels of communication: via A2A using the supported standard XML messages or files, via U2A using the standard T2S GUI functionality to enter data required for its migration. Alternatively, the CSDs can provide the T2S Operator outside T2S (e.g. via secured e-mail) with structured files (flat files, Excel files). The T2S Operator uses the data migration module to convert those structured files into standard T2S XML format and sends the data to T2S.

2.4 Security Management

This section of the GFS aims at providing a generic functional description of the way the GFS integrate the security requirements expressed in the UR document, notably in the chapter 18 "Information Security Requirements", but also in chapter 5 "Instruction Lifecycle Management and Matching Requirement", in chapter 10 "Securities Positions and Cash Balances" and in chapter 16 "Static Data Requirements". The solutions (i.e. tools or specific developments) to be implemented to meet these requirements will be defined and described in detail in the detailed functional specifications.

In this perspective, this transversal subject will be analysed in each domain/module/function, particularly the functions which (i) update the data, (ii) send or/and receive flows and (iii) ensure the interface with T2S Actors and external systems.

All rights reserved.

At this level, the analysis remains general and defines the methodology that will be implemented in the next phase of the functional design. Indeed, applying this methodology, the relevant security measures will be identified at the function level.

The objective of the methodology is ensuring that the security issues are exhaustively and consistently addressed, in each function. It consists of raising and answering the following questions at functional level with respect to the kind of treatments performed by the considered function:

- Should the function enforce any confidentiality requirement, i.e. ensuring that information is accessible only to authenticated and authorised parties?
  - What is the business sensitivity (level of confidentiality) of the processed information?
  - Who / what has a good reason to access the function (need to know principle)?
  - Is the function expected to receive the information from the sender or send the information to the recipient?

In the current functional design, the access control to T2S is performed by the Interface domain and by the Business Interface layers of the specific Common Components.

- Should the function enforce any integrity requirement, i.e. safeguarding the accuracy and completeness of information through the various processing, transmitting and storage methods?
  - To what extent must the input to application be validated to ensure that this data is correct and appropriate?
  - To what extent must validation checks be incorporated into applications to detect any corruption of information through processing errors or deliberate acts?
  - To what extent must the output of an application be validated to ensure that the processing of stored information is correct and appropriate?

In the current functional design, the integrity and the segregation of information are ensured by the CRDM component .

- Should the function enforce any authentication (origin and recipient sides) requirements, i.e. determining whether someone or something (function, component...) is, in fact, who or what it is declared to be?
  - What is the level of trust required in the identity of the counterparty before giving authorisation for accessing/transmitting the information?
  - What are the legal requirements enforcing the application to:
    - Ensure that a party cannot deny having received or sent a message (non repudiation requirement)?
    - Log the needed legal proofs for being able to establish this non repudiation in a long-term period?
    - Appeal to a notarisation service for trusted third party witnesses legally recognised in case of dispute?
    - Make use of timestamps proving the existence of a document at a given time?

All rights reserved.

In the current functional design, the authentication requirements are ensured by the Interface domain and by the Business Interface layers of the specific Common Components.

- Should the function enforce any monitoring requirements, i.e. detecting unauthorised information processing activities and recording appropriate information for future investigations?
  - What are the legal requirements in terms of monitoring and logging activities?
  - What event must be logged such as access, change of system configuration, use of privileges, and de-activation of protection systems, faults, system administrator and operator activities...?
  - What information details (user ids, date time...) must be logged and how long must the archived data be retrievable for future investigations?
  - How is the logged and archived information protected against tampering and unauthorised access?

In the current functional design, the monitoring requirements are ensured by the Operational monitoring module.

- Should the function enforce any availability requirement, i.e. ensuring that authorised users have access to information and associated assets when required (need to have principle)?
  - What is the business criticality of the information (need for accessibility/ availability) i.e. what the business risk is if this information would not be available (Business continuity)?
  - How fast should the function return information?
  - How to react if incoming information is not received when needed? How to react if information cannot be sent when expected?

The availability of the T2S services is ensured by the Infrastructure design.

- Should the function enforce any "auditability" [7] requirement (audit trail), e.g. the possibility of establishing whether a system is functioning properly and that it has worked properly?
  - What are the requirements in terms of "auditability", particularly coming from the service auditors examining the system in accordance with recognised auditing standards such as "Statement on Auditing Standards (SAS) N°70"?
  - What appropriate documentation will provide sufficient knowledge about the system and its structure, functions, controls, etc.?
  - What kind of integrity-related modifications to the system and its data must be made visible for proving the proper functioning?
  - What are the requirements in terms of Audit reports?

In the current functional design, the "auditability" of the T2S services is ensured by the design of the Data Model and the legal archiving module.

For obvious efficiency reasons, it is planned for the next phases to draw up an inventory of the global functional assets mentioning their sensitivity, criticality and identifying their owners. This inventory will be used as a common validated reference for helping the functional designer answering the above listed questions.

7 Also called « Controllability »

All rights reserved.

3 Functional Description

3.1 Introduction

On the basis of this general view of the functional solution envisaged for T2S, this chapter provides a functional description of the domains and modules identified in the overall high level diagram. For each of these domains, this chapter describes the role and behaviour of their respective modules and functions by presenting their respective data flow processes. It also includes a presentation of the processing of some representative [8] use cases illustrating the respective roles and interactions of modules in this processing.

This graphic and text description of representative use cases processing aims at facilitating the understanding of the internal behaviour of the system to answer the requests originating from external parties. For each domain, these representative use cases are described through:

- A sequence diagram showing the interactions between the T2S modules in the processing of the use case;
- A description of the processing of the use case and of the role of each module into this processing.

The approach for the choice of the representative use cases is explained for each category

3.2 Interface

3.2.1 General Introduction

The T2S *Interface* domain provides access to the T2S platform for all T2S System Users.

T2S supports access via two communication modes:

- The application-to-application mode (A2A), allowing direct communication between software applications via XML messages ;
- The user-to-application mode (U2A), supporting activities performed manually by the T2S System Users via the Graphical User Interface (GUI).

The *Interface* domain is also used for communication between:

- T2S and external RTGS systems;
- T2S and external collateral management systems.

The T2S Interface services are available continuously during the settlement day. However, the availability of the Interface services is restricted during the maintenance window (during the week), weekends and closing days.

The T2S Interface domain is composed of the following three modules for each communication mode:

- *Communication Module;*
- *Inbound Processing Module;*

8 Considering the huge number of defined Use Cases in some categories and the very similar processing of some "close" Use Cases, the description is limited in each category to a set of representative uses cases chosen in each category. For the other uses cases, the reference of the relevant representative Use Case is given into the lists of Use Cases provided in appendix.

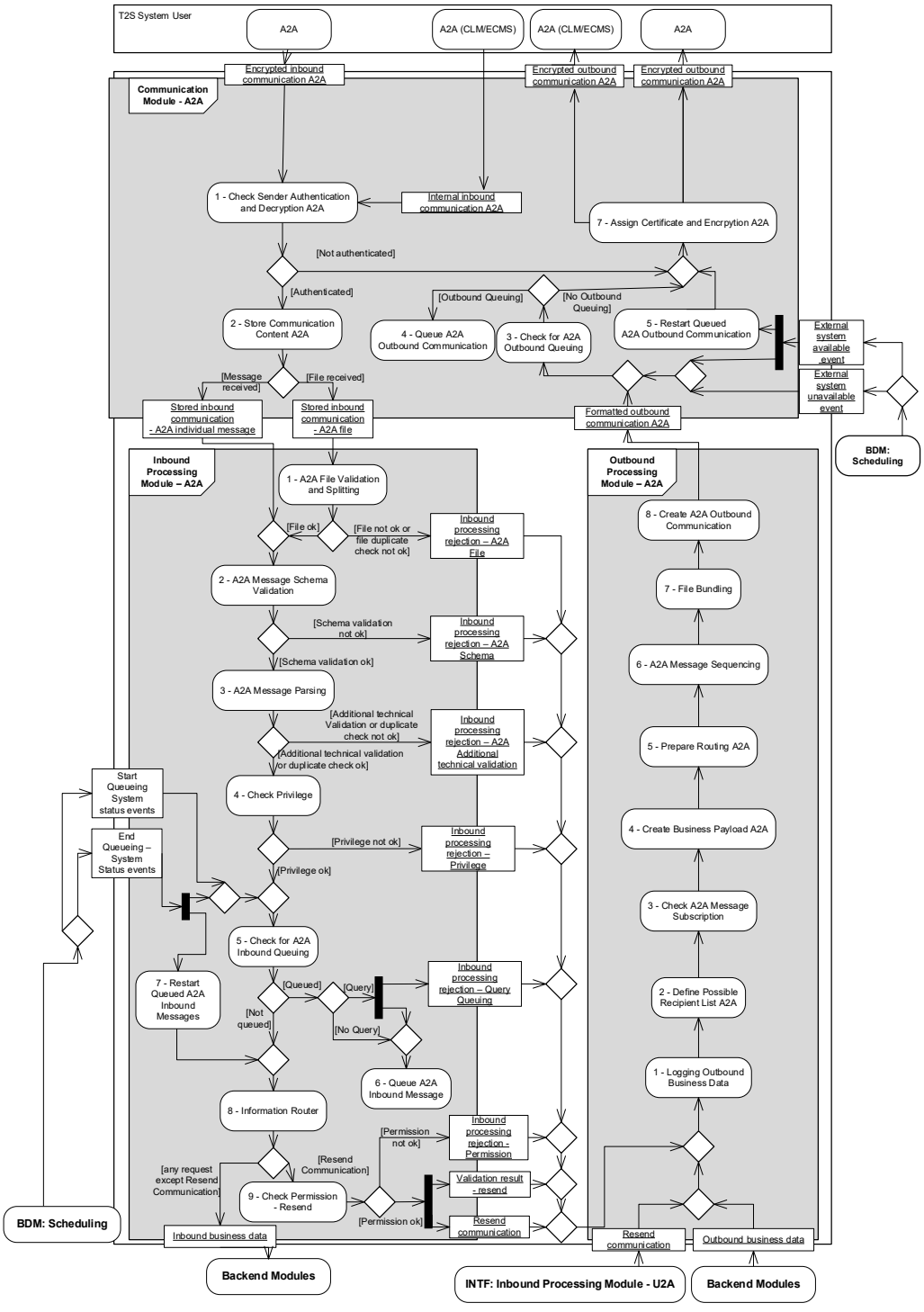
All rights reserved.

- *Outbound Processing Module.*

The *Interface* domain complies, to the possible extent, with the Giovannini protocol recommendations **{T2S.12.050}**. All messages support the formats and specifications defined by the Eurosystem Single Interface **{T2S.12.240}**.

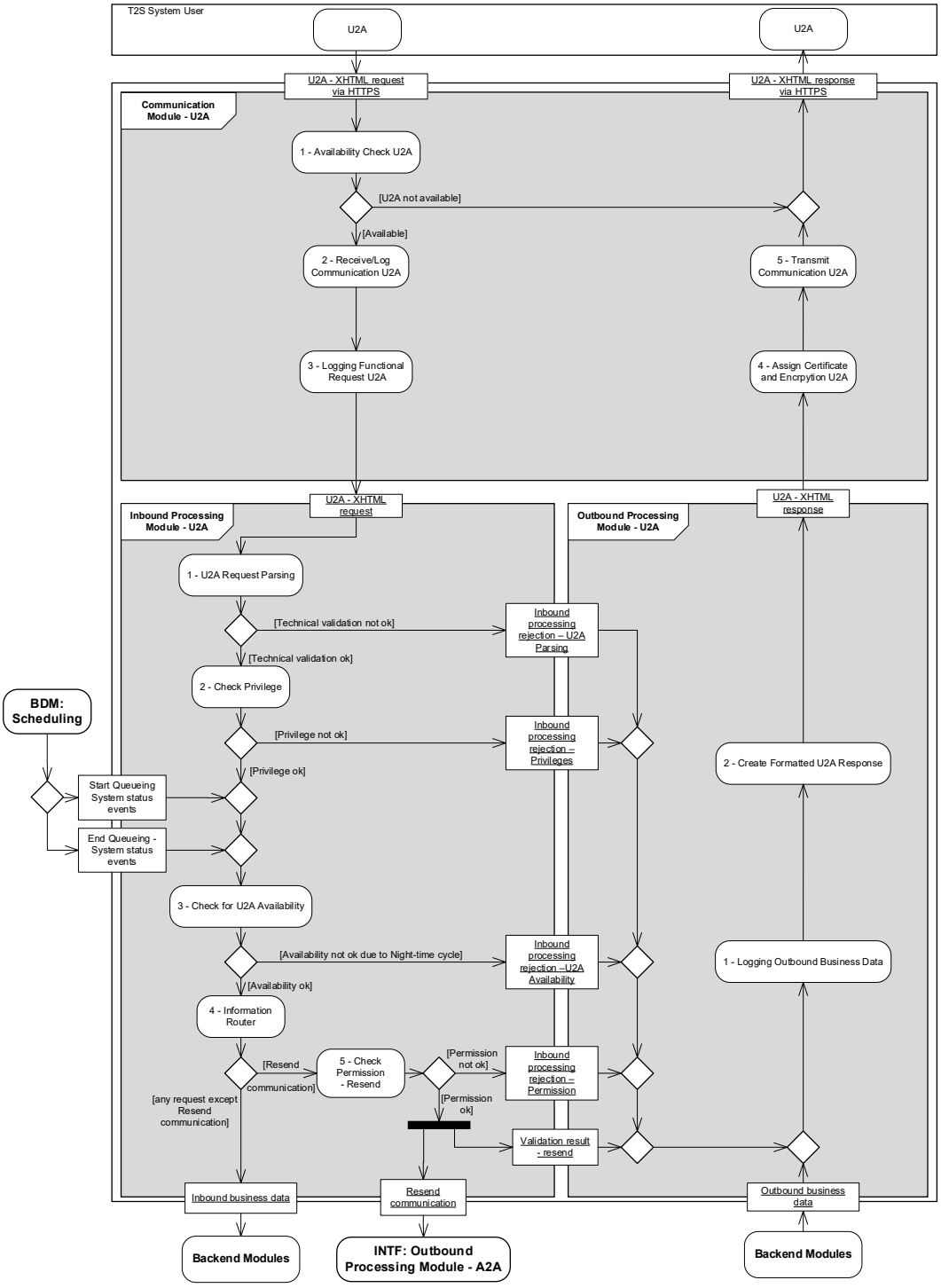
All rights reserved.

The following activity diagrams depict the high-level interactions between the modules and functions of the *Interface* domain. [9]



9 Please note that the described functionality won´t be available before CSLD go-live.

All rights reserved.



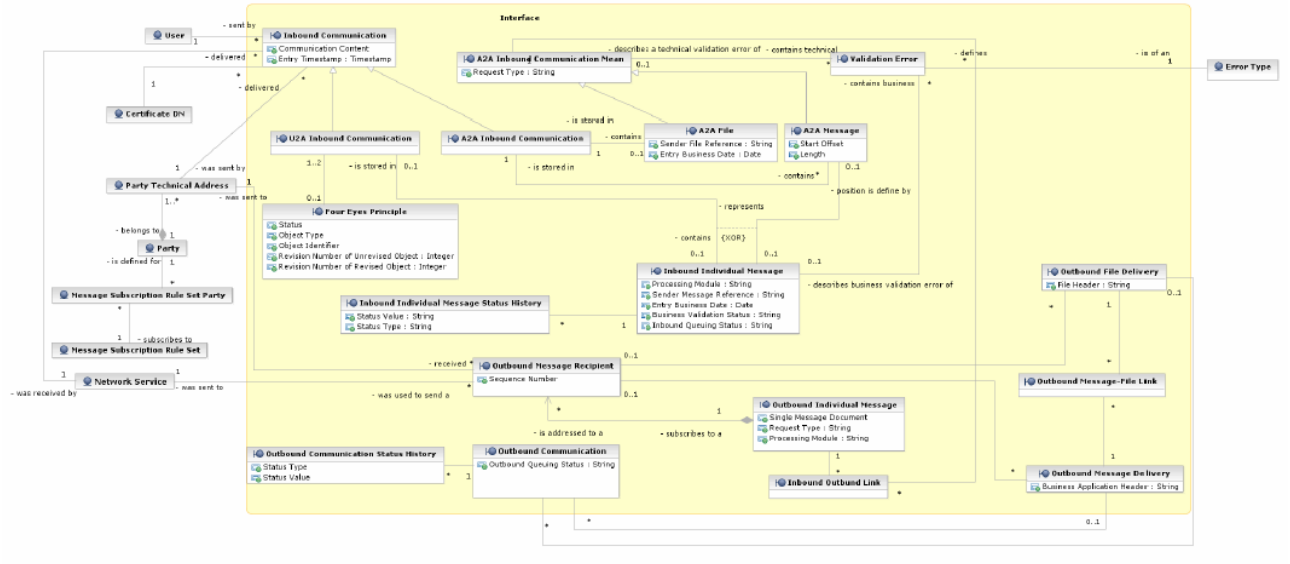
3.2.2 Dynamic data managed by the domain

The T2S System User sends or receives communication to/from T2S. A communication is a collective term for single messages and files. A file is a communication with specific header, which may contain zero, one or more single requests . A single message is a communication containing exactly one request , containing the necessary business data to be processed by T2S.

Remark:

All rights reserved.

All inbound/outbound communication except queries and the related results are stored for the sake of audit trail and traceability. Additionally, it allows a retrieval of information received from T2S System Users, including as well non-settlement related information. This retrieval of information is only possible for T2S System Users with the appropriate access rights **{T2S.12.020}**.



Inbound Communication:

This entity stores incoming U2A and A2A communications in their original (unchanged) decrypted form. It is also used to link the *User, Certificate DN, Party Technical Address* and *Network Service* to the received communication. The entity *A2A Inbound Communication* and the entity U2A Inbound Communication are specialisation of the abstract entity of Inbound Communication.

ATTRIBUTE	DESCRIPTION
Communication Content	This attribute stores the decrypted incoming communication in its original form received by T2S.
Entry Timestamp	The attribute stores the Entry Timestamp of the communication within T2S.

A2A Inbound Communication:

The entity contains for message-based communication only the incoming messages related to requests to change business data in T2S. The entity also contains all files received by T2S for file-based communication. It is stored by *Store Communication Content* function.

The *A2A Inbound Communication* entity is a specialisation of the abstract entity of *Inbound Communication* and, as such, exclusively for the communication, received through the A2A communication mode only. It inherits the Communication Content and Entry Timestamp attributes as well as all associations of the super entity.

Furthermore, this entity is linked to the *A2A Message* entity for the purpose of the identification of all single messages in an incoming communication. Only a single message is present for the message-based communication. A file-based communication may have zero, one or many messages.

All rights reserved.

A link exists between this entity and the *A2A File* entity. Interface processing creates this link only when the incoming communication is file-based to ensure that every file received by T2S is identified clearly and can be easily referenced in case of a failure of the technical validation of its syntax.

A2A Inbound Communication Mean:

This entity stores incoming A2A messages and A2A files. The entity *A2A Message* and the entity *A2A File* are specialisation of the abstract entity of *A2A Inbound Communication Mean*.

ATTRIBUTE	DESCRIPTION
Request Type	This attribute stores the request type for the message according to ISO 20022 standard.

A2A Inbound Communication Mean – A2A File:

For each authenticated A2A communication of the type *"File"* the function *A2A File Validation and Splitting* stores the entity including the sender's file reference. It documents that the received communication is a file and it enables the duplicate check on file level.

ATTRIBUTE	DESCRIPTION
Sender File Reference	This attribute stores the identification of the file which was delivered by the sender.
Entry Business Date	The attribute stores the business day, when T2S received the file.

A2A Inbound Communication Mean – A2A Message:

For each individual A2A message the Start Offset and Length are stored in this entity by the function *A2A Message Schema Validation*. It allows a documentation of messages, received as single message and as file without redundancy.

This entity only includes messages received in a file, if the file is positively validated.

ATTRIBUTE	DESCRIPTION
Start Offset	This attribute stores the starting position of the message within the communication
Length	This attribute stores the length of the message

U2A Inbound Communication:

The entity is used to link the *User, Party Technical Address* and *Network Service* to the occurred communication received in U2A. The complete decrypted U2A communication is stored in this entity by the function *U2A Request Parsing*.

The *U2A Inbound Communication* entity is a specialisation of the abstract entity of *Inbound Communication* and, as such, exclusively used for the communication, received through the U2A communication mode only. It inherits the Communication Content attribute as well as all associations of the super entity.

Four-Eyes principle:

In case of T2S operations where the Four-Eyes principle is required (due to the T2S System User's privileges or the activity itself) it is ensured that a second authorised T2S System User has confirmed the T2S operation before its execution. Details related to the processing of operations are provided in *Inbound Processing Module* (Check Privilege function). Linked secured objects are reference data.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Object Type	The Object Type describes the kind of identification according to responsible domain naming.
Object ID	The Object ID stores the technical identifier of the added / changed / deleted object. The Object ID is returned by the subsequent module after a successful business validation.
Revision Number of Unrevised Object	The Revision Number of Unrevised Object stores the revision number of the changed / deleted object, which was valid before the revision.
Revision Number of Revised Object	The Revision Number of Revised Object stores the revision number of the added / changed / deleted object, which is valid after the revision.
Status	This attribute identifies which requests under the Four-Eyes principle require attention of the approver. Its main purpose is to provide efficient way of filtering requests where the Four-Eyes principle is required. The exhaustive list of values is: ▮ Awaiting approval; ▮ Queued; ▮ Approved; ▮ Revoked; ▮ Rejected.

Inbound Individual Message:

This entity is used to store the relevant information about a single inbound message after a positive technical message validation by the *U2A Request Parsing* function or by *Message Parsing* function. Also, the target Processing Module attribute as a result of the *Information Router* function is stored.

ATTRIBUTE	DESCRIPTION
Sender Message Reference	This attribute stores the identification of the message, which is delivered by the sender.
Entry Business Date	The attribute stores the business day, when T2S received the message.
Processing Module	This attribute stores the module (component) to which the message is forwarded. Possible values are (exhaustive list): ▮ Inbound Processing Module (INTF) ▮ Outbound Processing Module (INTF) [9] ▮ Outbound Information Management (LQMG) ▮ Liquidity Operations (LQMG) ▮ Query Management (SRQA)

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▮ Instruction Validation (LCMM) ▮ Operational Monitoring (OPSR) ▮ BDM
Business Validation Status	This attribute stores the actual business validation status of the inbound individual message. Possible values are (exhaustive list): ▮ False ▮ True
Inbound Queuing Status	This attribute stores the actual inbound queuing status of the inbound individual message. Possible values are (exhaustive list): ▮ False ▮ True

Inbound Individual Message Status History:

This entity is used to store the status history for the Business Validation Status or Inbound Queuing Status for each inbound message.

ATTRIBUTE	DESCRIPTION
Status Value	This attribute stores the status value from Inbound Individual Message entity. Possible values are (exhaustive list): ▮ False ▮ True
Status Type	This attribute stores the status type from Inbound Individual Message entity. Possible values are (exhaustive list): ▮ Business Validation Status ▮ Inbound Queuing Status

Validation Error:

This entity is used to link the A2A communication mean or the Inbound individual message to the occurred validation errors. The link to the Error Type (T2S Internal Reason Code / ISO 20022 Error Code / ISO 20022 Reason Code) is stored in this entity by the functions *File Validation and Splitting (A2A), Message Scheme Validation (A2A), Message Parsing (A2A) or Check Privilege (A2A/U2A)*.

Inbound – Outbound Link:

The entity links the inbound communication to the outbound messages. It is only present for an erroneous A2A inbound communication.

All rights reserved.

Outbound Individual Message:

The entity is created by *Create Business Payload* function to store the request type, processing module and payload of the outbound communication.

ATTRIBUTE	DESCRIPTION
Request Type	This attribute stores the request type for the message according to ISO 20022 standard.
Processing Module	This attribute stores the module (component) from which the message has been received. Possible values are (exhaustive list) : ▪ Inbound Processing Module (INTF) ▪ Outbound Information Management (LQMG) ▪ Liquidity Operations (LQMG) ▪ Query Management (SQRA) ▪ Report Management (SQRA) ▪ Status Management (LCMM) ▪ Instructions Matching (LCMM) ▪ Instruction Maintenance (LCMM) ▪ Operational Monitoring (OPSR)
Common Communication Content	This attribute includes the container storing the message payload generated out of the information given by processing module.

Outbound Message Recipient:

After receiving sequenced outbound business data the function *Create Outbound Communication* stores the sequence number of the message in this entity. The *Outbound Individual Message* and the *Outbound Message Recipient* is stored at the same time.

ATTRIBUTE	DESCRIPTION
Sequence Number	This attribute includes an independent counter for message sequence, which is available once per party technical address.

Outbound Communication:

In this entity the complete information about the formatted outbound message is stored by *Check for Outbound Queuing* function.

ATTRIBUTE	DESCRIPTION
Outbound Queuing Status	This attribute stores the actual outbound queuing status of the outbound communication. Possible values are (exhaustive list): ▪ True

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▮ False

Outbound Communication Status History:

This entity documents the status changes to the entity *Outbound Communication* and is therefore linked to it.

ATTRIBUTE	DESCRIPTION
Status Type	This attribute stores the status type from the Outbound Communication entity. Possible value (exhaustive list): ▮ Outbound Queuing Status
Status Value	This attribute stores the status value. Possible values for Outbound Queuing Status are (exhaustive list): ▮ True ▮ False

Outbound Message Delivery:

This entity stores the Business Application Header information.

ATTRIBUTE	DESCRIPTION
Business Application Header	Container storing the Business Application Header (BAH) content, which is different for each receiver

Outbound Message-File Link:

This entity links the Outbound Message Delivery to the Outbound File Delivery.

Outbound File Delivery:

This entity includes the Business File Header information.

ATTRIBUTE	DESCRIPTION
File Header	Container storing the File Header content, which is different for each receiver.

3.2.3 Communication Module

3.2.3.1 Description of the module

The main purpose of the *Communication Module* is to ensure secure and reliable communication between the T2S platform and its users. T2S System Users can use application-to-application (A2A) and user-to-application (U2A) communication channels to access the T2S platform. The respective communication channels support the push or pull mode **{T2S.12.260}**.

The T2S *Communication Module* supports several connectivity options **{T2S.12.010}**. The T2S connectivity services support store-and-forward and real-time file and message transfers **{T2S.12.260}**.

All rights reserved.

The module also supports communication between T2S and external RTGS systems, and T2S and external collateral management systems. The domain deploys a single set of standard messages to provide the interfacing with multiple external RTGS systems **{T2S.12.340}**. T2S supports interfacing with external collateral management system to receive securities prices for collateral valuation and to exchange transactional data for auto-collateralisation operations **{T2S.12.350}**, using a single set of standard communication **{T2S.12.360}** and for the specific case of securities valuation via flat files.

Additionally, the *Communication Module* provides functionality to protect the T2S platform against intrusion and unauthorised access. It provides an authentication functionality to verify identity of the T2S System User to other functions or modules.

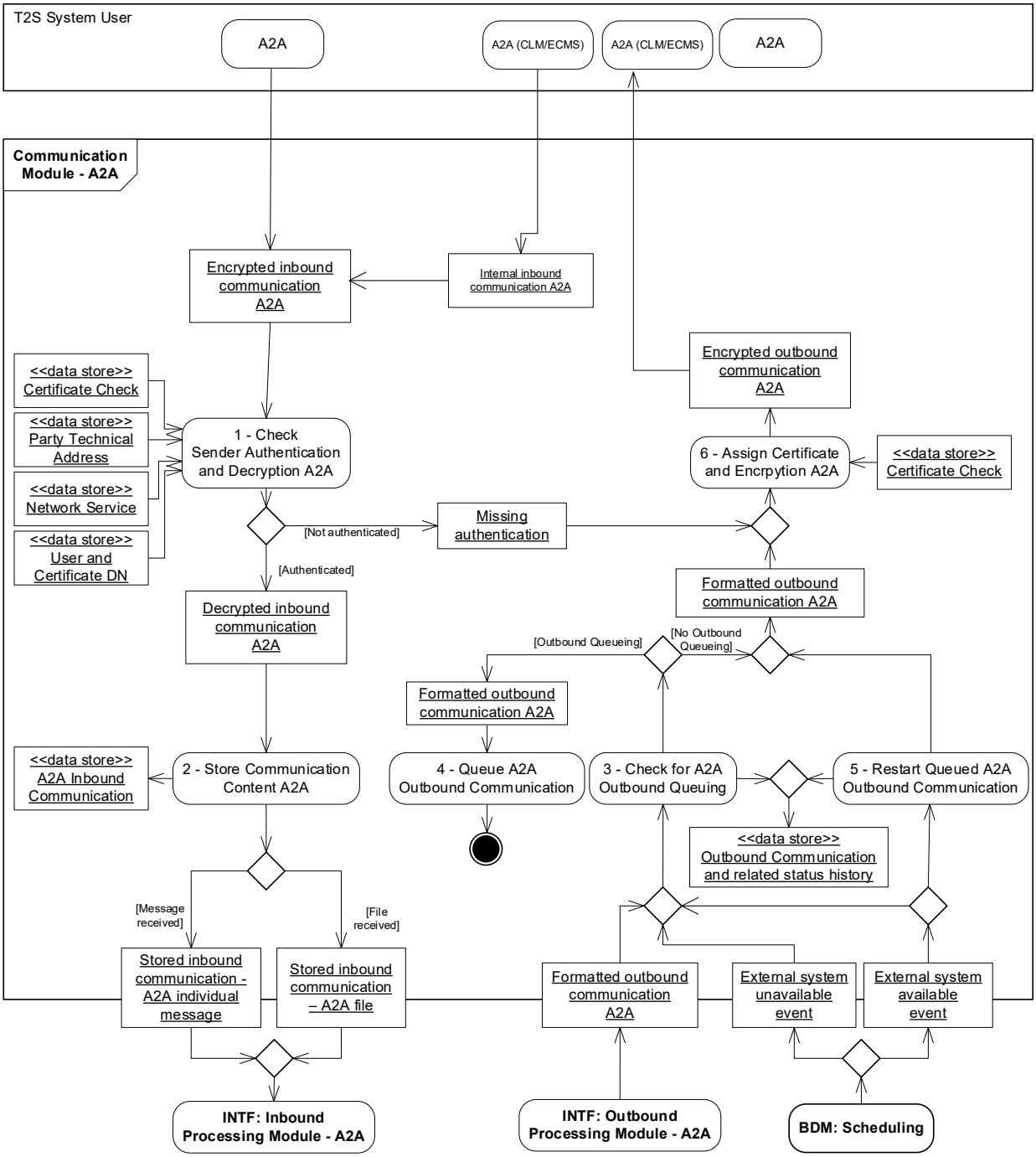
The *Communication Module* is composed out of three sub-modules:

- *Communication Module* for Application-to-Application mode;
- *Communication Module* for User-to-Application mode.

All rights reserved.

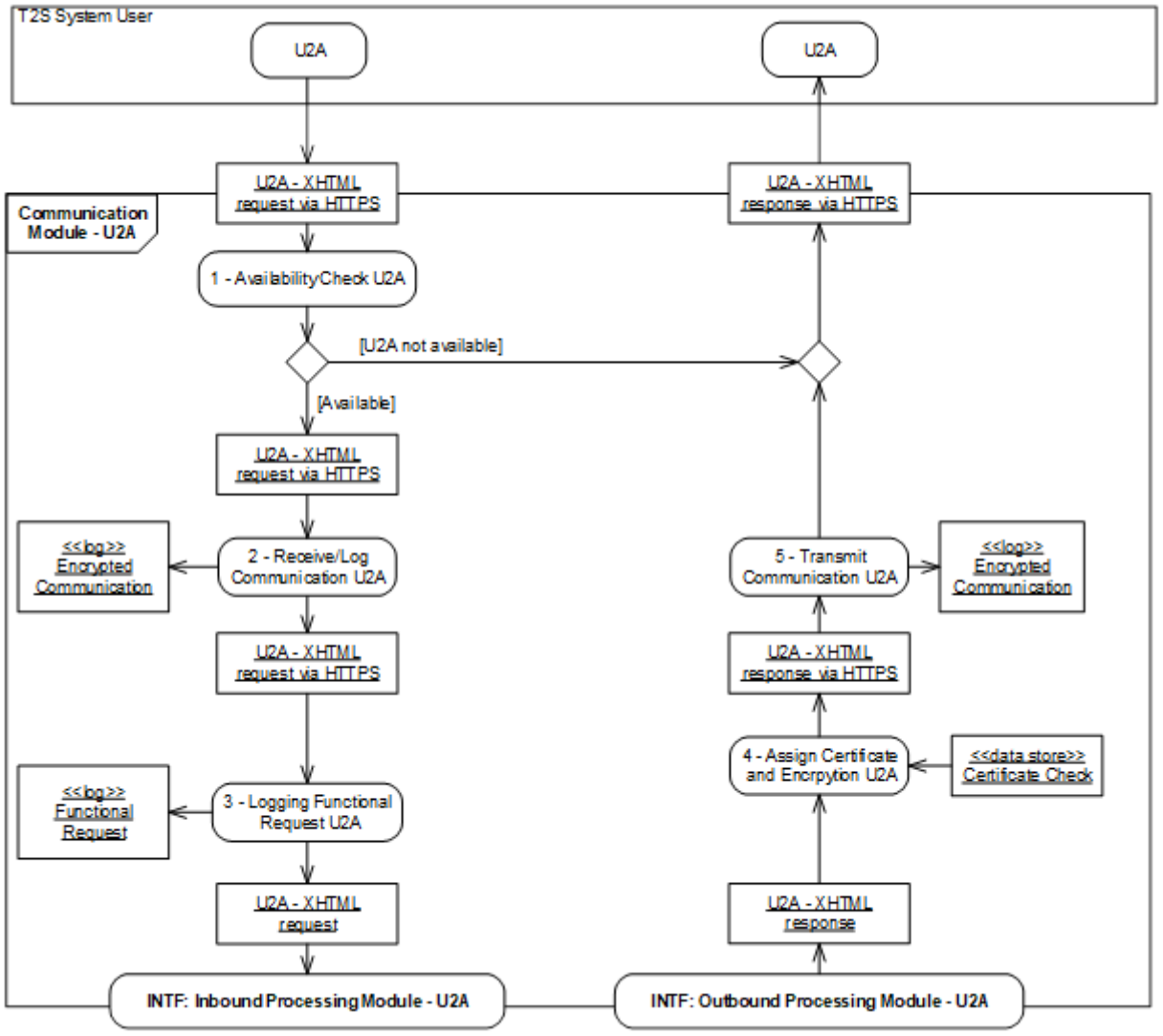
3.2.3.2 Diagram of the module

Communication Module for Application-to-Application mode (A2A)



All rights reserved.

Communication Module for User-to-Application mode (U2A)



3.2.3.3 Description of the functions of the module

Communication Module – Application-to-Application

1 – Check Sender Authentication and Decryption A2A

Reference Id	***INTF.COM-A2A.CSA.1.1***

This function ensures that users without authentication are not able to compromise the communication processing in T2S.

For every *Encrypted inbound communication A2A* flow the function identifies the sender and checks if the digital signature of the inbound communication corresponds to the certificate of the sender. Furthermore this function also receives the Internal inbound communication A2A flow that handles the inbound communication sent by CLM or ECMS to T2S.This function also identifies the T2S System User, the used Certificate DN

All rights reserved.

and the communicating T2S Party **{T2S.12.120}** and it checks the *party technical address* **{T2S.12.110}** and the *network service* used for the communication by associating the sender information of the inbound communication with the relating reference data information.

The function rejects the *Encrypted inbound communication A2A* or the Internal inbound communication A2Aif one of these checks fails. The function forwards the Missing authentication flow (including error information) to the Assign Certificate *and Encryption A2A* function and initiate the process to deliver it to the sender if the mandatory routing information (that means the data of the original sender) is available. Otherwise, the message *Missing authentication* cannot be delivered.

If the function has positively checked the sender authentication of the communication, it converts the *Encrypted inbound communication A2A* flow and the Internal inbound communication A2A flow into the *Decrypted inbound communication A2A* flow (including decompression if necessary)and routes it to the function *Store Communication Content A2A*.

2 – Store Communication Content A2A

Reference Id	***INTF.COM-A2A.SCC.1.1***

The function *Store Communication Content A2A* receives the *Decrypted inbound communication A2A* flow and stores it into the *A2A Inbound Communication* data store.

The function checks whether the communication is a file or a message. The function starts the processing for a single message, when the communication is not a file **{T2S.12.130}**.

The function identifies a file via the file header. It may include zero, one or many single individual messages.

If the communication is a message (including acknowledgements), this function sends the *Stored inbound communication – A2A individual message* flow to the *Inbound Processing Module - A2A*, otherwise it sends the *Stored inbound communication – A2A file* flow to the same module.

3 – Check for A2A Outbound Queuing

Reference Id	***INTF.COM-A2A.COQ.1.1***

This function receives the following flows:

- *Formatted outbound communication A2A* flow;
- *External systems available* event;
- *External systems unavailable* event.

When receiving the *Formatted outbound communication A2A* flow, the function stores the *A2A Outbound Communication. Outbound* queuing is not applied for queries responses. Unless the communication is a query response, this function checks the availability of the relevant external system (i.e. RTGS system or Collateral management system). If the external system is available, this function stores the *Outbound Communication* and related status history with Status Type "Outbound Queuing Status" and Status Value *"False"* and forwards the *Formatted outbound communication A2A* flow to the function *Assign Certificate and Encryption A2A*. Otherwise it stores the *Outbound Communication* and the related *Status History* with Status Type "Outbound Queuing Status" and Status Value "*True"* and forwards the *Formatted outbound communication A2A* flow to the function *Queue A2A Outbound Communication*.

All rights reserved.

4 – Queue A2A Outbound Communication

Reference Id	***INTF.COM-A2A.QOM.1.1***

This function receives the flow *Formatted outbound communication A2A* for the case that the regarding external system is not available and it queues the message for the deferred sending.

5 – Restart Queued A2A Outbound Communication

Reference Id	***INTF.COM-A2A.ROM.1.1***

This function receives an event *(External systems available* event), which is delivered by BDM. Afterwards this function retrieves all the queued communication for the activated external system, stores the *Outbound Communication* and the related status history with Status Type *"Outbound Queuing Status"* and Status Value *"False"* and submits them as the *Formatted outbound communication A2A* flow to the function *Assign Certificate and Encryption A2A*.

6 - Assign Certificate and Encryption A2A

Reference Id	***INTF.COM-A2A.ACE.1.1***

This function receives:

- *Formatted outbound communication A2A* flows;
- *Missing authentication* flows.

In case of the *Missing Authentication* flow this function creates the related XML error message.

For both cases the function checks if the receiver of the communication will be CLM/ECMS or another T2S System User. In case the receiver of the communication is CLM or ECMS the function delivers the Encrypted outbound communication A2A directly.

After that the function delivers the *Encrypted outbound communication A2A* to ESMIG.

Communication Module – User-to-Application

1 – Availability Check U2A

Reference Id	***INTF.COM-U2A.ACU.1.1***

This function is the entry point for the U2A access of T2S System Users to the T2S *Interface* domain **{T2S.12.060}**. When a T2S System User sends an *U2A-XHTML request via HTTPS* (including queries **{T2S.14.040}**), the system availability is checked. If the system is unavailable due to maintenance window or any other reason, the function delivers an unavailability information using the *U2A – XHTML response* via HTTPS to the T2S System User.

When the system is available, the function forwards the *U2A-XHTML request* via HTTPS to the *Receive/Log Communication U2A* function.

2 - Receive/Log Communication U2A

Reference Id	***INTF.COM-U2A.RLC.1.1***

All rights reserved.

This function is responsible for the logging of the received *U2A-XHTML request via HTTPS*. It stores the communication in the log *Encrypted Communication* for short term traceability and then forwards it to the function Logging Functional Request U2A .

3 – Logging Functional Request U2A

Reference Id	***INTF.COM-U2A.LFR.1.1***

This function stores the log *Functional Request* for short term traceability and sends the *U2A-XHTML request* flow to the *Inbound Processing Module-U2A*.

4 – Assign Certificate and Encryption U2A

Reference Id	***INTF.COM-U2A.ACE.1.1***

This function receives the *U2A-XHTML response* flow , creates the enciphered XHTML response and delivers it as the *U2A-XHTML response* via HTTPS flow to the *Transmit Communication U2A* function.

5 – Transmit Communication U2A

Reference Id	***INTF.COM-U2A.TRC.1.1***

This function receives an *U2A-XHTML response* via HTTPS, stores it in the log *Encrypted Communication* for short term traceability and sends it to the original sender.

3.2.3.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
U2A - XHTML request via HTTPS	IN	Request received by T2S via U2A	T2S System User	
U2A - XHTML response	IN		INTF: Outbound Processing Module	
Encrypted inbound communication A2A	IN	Messages and files received by T2S via A2A	T2S System User	
Internal inbound communication A2A	IN	Messages and files received by T2S from CLM/ECMS	T2S System User	
Formatted outbound communication A2A	IN		INTF: Outbound Processing Module	
External system available events	IN		BDM	
External system unavailable events	IN		BDM	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
U2A - XHTML request	OUT			INTF: Inbound Processing Module
U2A - XHTML response via HTTPS	OUT			T2S System User
Stored inbound communication – A2A individual message	OUT	Messages (A2A) validated by Communication Module		INTF: Inbound Processing Module
Stored inbound communication – A2A file	OUT	Files (A2A) validated by Communication Module		INTF: Inbound Processing Module
Encrypted outbound communication A2A	OUT	Messages sent by T2S via A2A		T2S System User

3.2.3.5 Data accessed by the module

Data stores

All inbound/outbound communication except queries and the related results are stored.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
Certificate Check		Read	Read access for check on authentication and for assignment of certificate for outbound processing
Network Service		Read	Read information about the used network service
Party Technical Address		Read	Allows the check on the secured and recognised technical address based on reference data
User and Certificate DN		Read	Allows the check of the existence and correct usage of user data and certificate DN based on reference data
A2A Inbound Communication		Write	Allows the storage of the A2A Inbound Communication
Outbound Communication and related status history		Write	Allows the storage for each status update in Outbound Communication.

All rights reserved.

Logs

The logs are stored for short term traceability. Therefore they are not archived.

DATA	ACCESS MODE	COMMENTS
Encrypted Communication	Write	U2A/A2A communication is logged for inbound and outbound processing
Functional Request	Write	U2A functional requests are logged for inbound processing

3.2.4 Inbound Processing Module

3.2.4.1 Description of the module

The *Inbound Processing Module* receives:

- *U2A - XHTML request*;
- *Stored inbound communication – A2A file*;
- *Stored inbound communication – A2A individual message*;

and performs a series of technical verification checks. The processing of this module includes the part of the authorisation check which is based on privileges. Roles Based Access Control (RBAC) is done in T2S by using the defined static data of T2S. The management of privileges (creation, update or deletion) is done in CRDM.

As the availability of T2S is restricted during the maintenance window and the night-time settlement, the *Inbound Processing Module* includes queuing and restart functionalities, which allow the deferred processing of requests.

The *Inbound Processing Module* allows the initiation of resending of messages or files. This functionality can be used in contingency situations if the messages or files have been lost during the communication or due to technical problems on the receiver side.

The *Inbound Processing Module* includes the extraction of all relevant business data from the received request in order to structure them in an internal format, determines the relevant T2S modules and routes the structured data to them. In case of a rejection (e.g. the privilege for the requested action is not assigned to the T2S System User) the *Interface* creates the respective *Inbound processing rejection* and delivers it to the *Outbound Processing Module*.

The term "internal format" is used for all T2S internal flows between the *Interface* domain and the subsequent modules of the other domains. This term indicates that the *Interface* only forwards the relevant business data included in the message instead of the original message to the subsequent modules. Therefore possible structure changes of XML messages only impact the *Interface* domain and not the whole system.

The *Inbound Processing Module* is composed of two sub-modules:

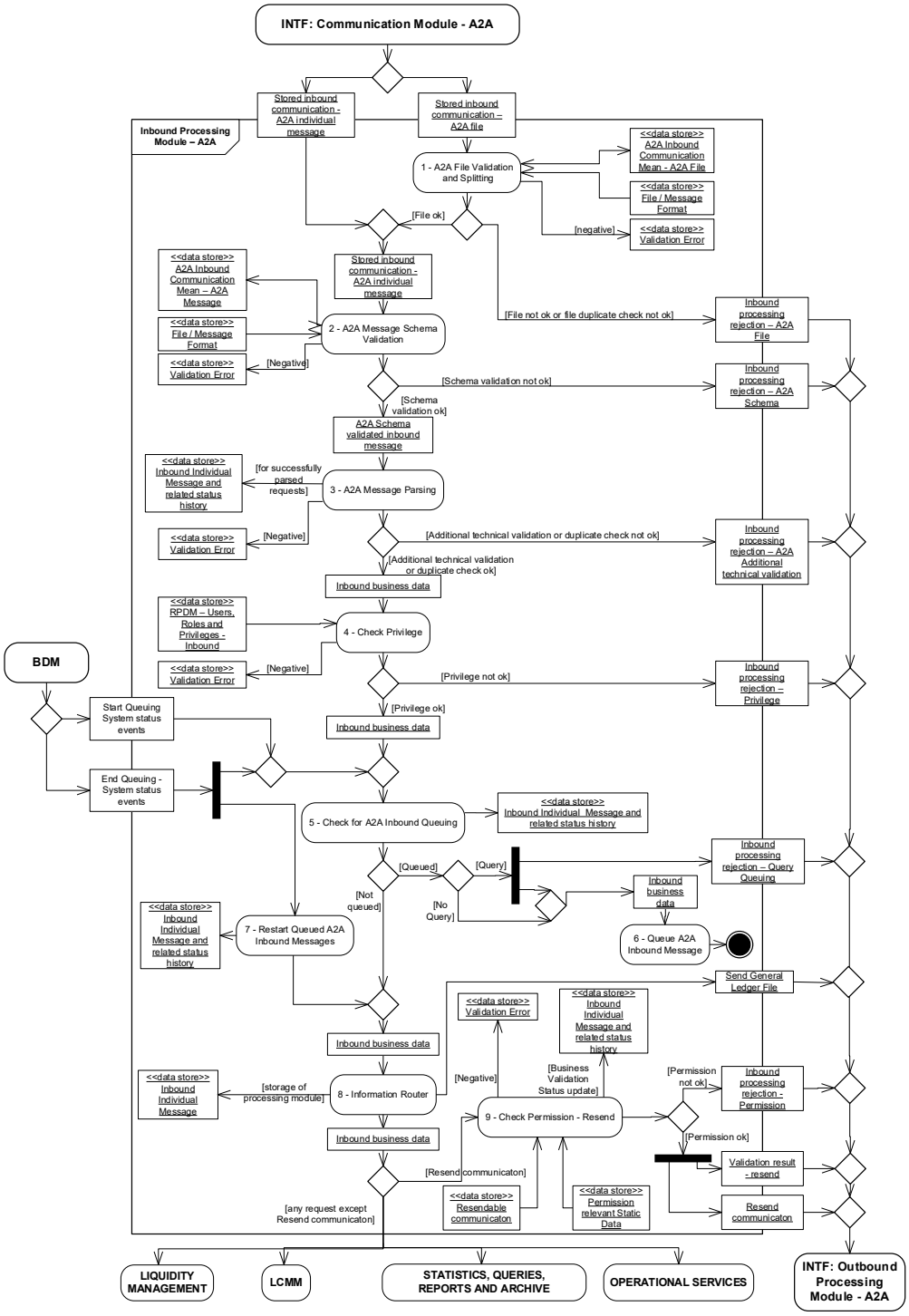
- *Inbound Processing Module* for Application-to-Application mode;

All rights reserved.

- *Inbound Processing Module* for User-to-Application mode.

3.2.4.2 Diagram of the module

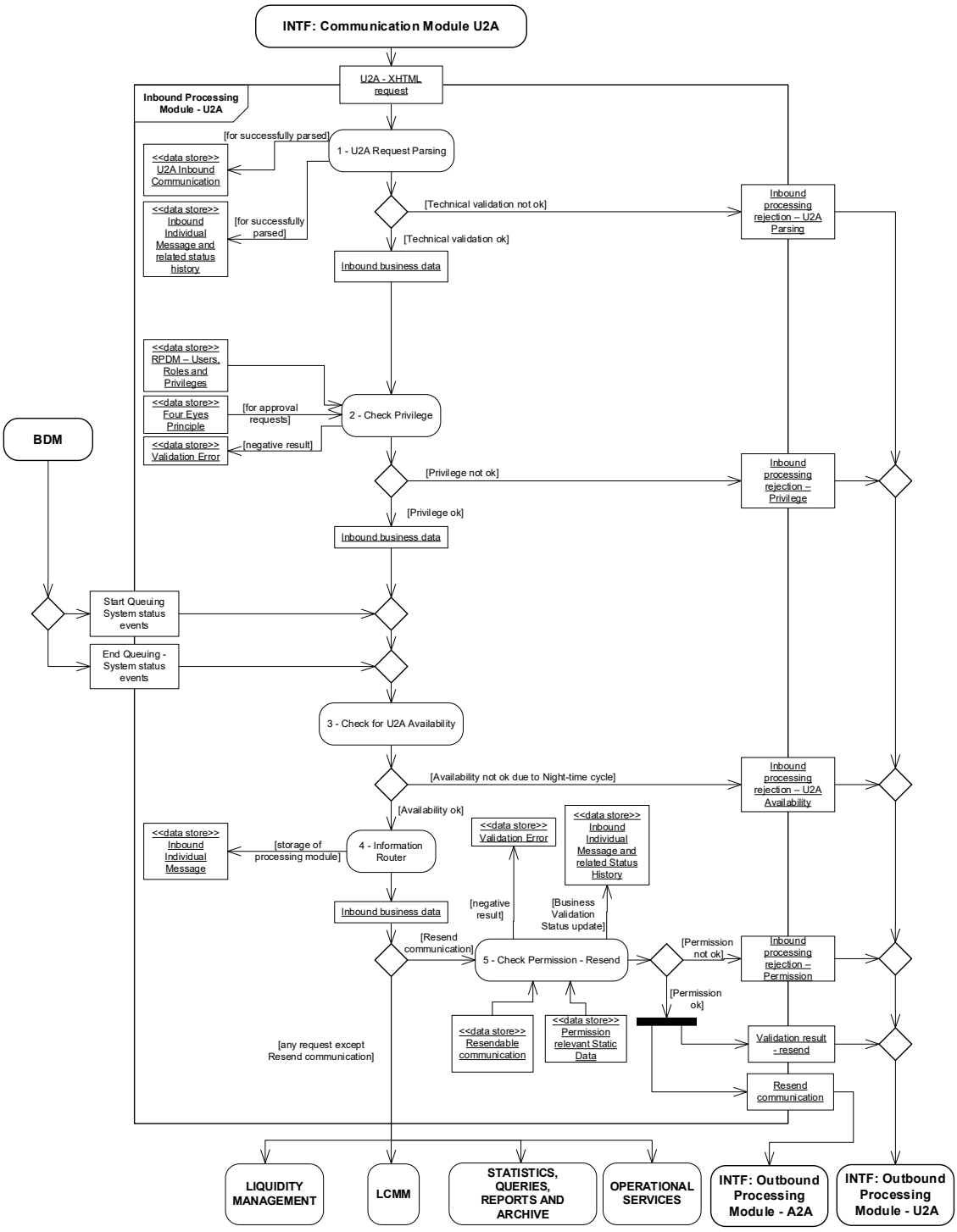
Inbound Processing Module for Application-to-Application mode (A2A) [11]



11 Please note that the described functionality won´t be available before CSLD go-live.

All rights reserved.

Inbound Processing Module for User-to-Application mode (U2A)



3.2.4.3 Description of the functions of the module

Inbound Processing Module – Application-to-Application

1 – A2A File Validation and Splitting

All rights reserved.

This function receives the *Stored inbound communication – A2A file* flow and validates inbound files. The file structure is in an XML format. Therefore the *Interface* domain validates the inbound files with the same standards as single messages **{T2S.12.080}**. The file processing complies with Giovannini file transfer rule-book for file operations **{T2S.12.090}**.

The first processing step validates that the file has a proper XML structure. In case of file structure problems (i.e. file header as first record in a file and file trailer as last record in a file are not identifiable or individual messages are not recognisable), this function rejects the file entirely **{T2S.12.070}**.

In the second step this function does a duplicate emission check on the file level **{T2S.12.090}** by comparing the sender's file reference with file references, which were delivered before by the same sender.

In case of a negative result of the validation, it creates an *Inbound processing rejection - A2A File* flow and sends it immediately to the *Logging Outbound Business Data* function in *Outbound Processing Module* A2A. This function uses the data store *Validation Error* to store the reason for the rejection.

If the file structure is valid then the file is processed (split in individual messages) **{T2S.16.167}**. All technical validations done by *Message Schema Validation* and *Message Parsing* functions are then applied at the level of a single message (which were included in a file) in a subsequent processing step. If one of these checks is not successful, this results in the creation of a rejection for the single invalid message only **{T2S.12.100}**.

The function stores a file reference in the *A2A Inbound Communication Mean – A2A File* data store **{T2S.12.090}**.

The function forwards each single message as *Stored inbound communication – A2A individual message* flow after successful splitting to the function *A2A Message Schema Validation*.

2 – A2A Message Schema Validation

Reference Id	***INTF.INP-A2A.MSV.1.1***

The function *A2A Message Schema Validation* receives the *Stored inbound communication – A2A individual message* flow and stores the *A2A Inbound Communication Mean – A2A Message* data store.

This function checks the type of the delivered message (nature of the communication) **{T2S.12.130}**. Unless the message is supported by T2S, the communication is rejected.

The syntax and structure of messages supported by T2S in A2A mode are based on XML and compliant with the ISO 20022 standard **{T2S.12.040}** (including queries **{T2S.14.010}**). ISO20022 complaint T2S specific messages are used when necessary. Therefore, T2S verifies the compliance of the syntax, format and structure in the inbound single message according to T2S requirements **{T2S.12.070}**.

Each request type is connected to only one XML schema file [12], therefore the validation is done against these schemas. The function ensures that the XML message is:

- Well-formed (e.g. each opening tag has a corresponding closing tag);

12 A XML schema is a description of a type of XML document, typically expressed in terms of constraints on the structure and content of documents of that type, above and beyond the basic syntax constraints imposed by XML itself.

All rights reserved.

- Valid against the rules of the schema, i.e. the structure of the message corresponds to the schema file. This includes (exhaustive list):
  - correct order (hierarchy) of blocks and elements,
  - naming of blocks and elements are correct,
  - the occurrence of the elements is correct: elements can appear once or repetitively in an XML message,
  - all mandatory blocks and elements of the XML message are filled (e.g. for Inbound liquidity transfers **{T2S.06.520} {T2S.06.521}**),
  - the tags of the message are filled in the correct format: including the data type (e.g. text, alphanumeric, amount) and the format (e.g. maximum and minimum length).

The processing of this function does not include checks on dependencies between elements.

In case of a negative result of the validation, the *Message Schema Validation* function creates the *Inbound processing rejection A2A - Schema* flow, assigns the error code information and sends it to the *Logging Outbound Business Data* function in the *Outbound Processing Module* A2A. This function uses the data store *Validation Error* to store the reason for the rejection.

After completing the schema validation with a positive result, the function forwards the *A2A Schema validated inbound message* flow to the *A2A Message Parsing* function.

3 – A2A Message Parsing

Reference Id	***INTF.INP-A2A.MSP.1.1***

This function receives the *A2A Schema validated inbound message* flows.

For all messages this function performs the following processing:

- Process additional technical validations, which are not included in the schema validation (e.g. conditional availability of tags, dependencies between elements);
- Extract the business data from the communication which is needed by the subsequent modules (e.g. ISIN code, counterparty codes for settlement instructions, T2S System User initiating the communication) **{T2S.12.180} {T2S.12.210}**;
- Convert the business data into the internal format to be processed by the subsequent modules;
- Perform a duplicate check on individual message level by comparing the sender's message reference with messages references, which were delivered before by the same sender.

In case of any negative result (i.e. duplicate check or another technical validation), the function creates an *Inbound processing rejection - A2A Additional technical validation* flow including the error code information and sends it to the *Logging Outbound Business Data* function in the *Outbound Processing Module* A2A. This function uses the data store *Validation Error* to store the reason for the rejection.

After completion of the parsing with a positive result, the function stores the *Inbound Individual Message* and the related status history. Afterwards it forwards the *Inbound business data* flow to the function *Check Privilege*.

All rights reserved.

4 – Check Privilege

Reference Id	***INTF.INP.CHP.1.1***

The *Check Privilege* function receives the *Inbound business data* flow for the U2A and A2A access. It checks the privileges **{T2S.11.360}**, which are assigned to the T2S System User (Based on the System User Reference) **{T2S.12.250}** and validates if the sender of an incoming communication is authorised to execute the intended T2S function (administrative or business operation) **{T2S.12.060}**. There are two different types of privileges available:

- System Privileges;
- Object Privileges **{T2S.11.361}**.

Both are described in the CRDM_domain.

The checks include system privileges and object privileges. In case of object privileges the checks, which are done by this function, only include the T2S functional capability and do not include the check on the SD secured elements (single or via SD Secured Group) linked to the T2S functional capability. The respective subsequent module provides the validation of access to the SD secured elements.

During the processing of U2A writing requests, a possible usage of the Four-Eyes principle has to be taken into account. This function identifies the requests, which require the Four-Eyes principle (i.e. the requestor is only allowed to initiate the writing request according to the Four-Eyes principle) and forwards this information together with the request for further processing. While receiving an approval request including a confirmation or a revocation, this function checks the current status and existence of the respective initial request. In case of approval request, it is additionally checked, if the approval user is different from the initiating user.

In case of a negative result of the privilege check (for example the T2S System User does not have the required privilege), the function creates an *Inbound processing rejection - Privilege* including the error code information, it uses the data store *Validation Error* to store the reason for the rejection and it sends it for:

- A2A access, to the *Logging Outbound Business Data* function in the *Outbound Processing Module* A2A;
- U2A access, to the *Logging Outbound Business Data* function in the *Outbound Processing Module* U2A.

When the privilege check is completed with a positive result, the *Inbound business data* flow is forwarded to the

- *Check for A2A Inbound Queuing* function for A2A access;
- *Check for U2A Availability* function for U2A access.

5 - Check for A2A Inbound Queuing

Reference Id	***INTF.INP-A2A.CIQ.1.1***

This function receives the following data flows:

- *Inbound business data*;
- *Start Queuing System Status events*;
- *End Queuing System Status events*;

All rights reserved.

- *Inbound processing rejection – Query Queuing*.

It checks if the A2A request can be forwarded or if it has to be queued. Queuing is relevant for:

- All requests during maintenance window **{T2S.03.230}**;
- For Liquidity Transfers
  - Between Business date change and Change of business date completed and
  - During cycle 2 / preparation of sequence y
- Securities position related Queries during sequences with impact on positions
- Balance Queries during night time sequences **{T2S.14.080}**.

If the queued message is a query, the function closes the real-time channel by sending the flow Inbound processing rejection – Query Queuing to the *Logging Outbound Business Data* function in the *Outbound Processing Module* A2A.

The processing of the inbound communication during the maintenance window is independent from the message type **{T2S.14.040}**. It only depends on the communication mode.

Files received during the maintenance window are queued as single messages, because the files are already split **{T2S.20.030}**.

The T2S *Interface* domain receives the respective events from BDM. The function *Check for A2A Inbound Queuing* processes single messages accordingly. The following alternatives may occur:

- Delivery: the function forwards the requests to the *Information Router* function and set the Inbound Queuing Status in *Inbound Individual Message* and related status history to "*False*";
- Queuing: the function starts the queuing mechanism to allow a restart after the end of the maintenance window or the end of the night-time settlement in case of Balance Queries and set the Inbound Queuing Status in *Inbound Individual Message* and related status history to *"True"*.

The above mentioned rules do not apply, if T2S Interface is not available. T2S *Interface* domain is not available on a regular basis during weekends **{T2S.03.350}** and on closing days. During weekends and closing days the *Interface* domain is only available when required, based on specific needs (e.g. migration, issuance in direct holding countries) **{T2S.03.360}**.

6 – Queue A2A Inbound Message

Reference Id	***INTF.INP-A2A.EQM.1.1***

This function receives the *Inbound business data* flow and queues the message for deferred execution. Sending an extra notification in case the message is queued is therefore not needed **{T2S.14.040}**.

7 – Restart Queued A2A Inbound Messages

Reference Id	***INTF.INP-A2A.RQM.1.1***

This function is triggered by an event (*End Queuing – System status events*) received from the BDM. When activated, this function retrieves all the queued single messages, stores the Inbound Queuing Status in *Inbound Individual Message* and related status history to *"False"* and submits them as the *Inbound business data* flow to the *Information Router* function.

All rights reserved.

8 – Information Router

Reference Id	***INTF.INP.INR.1.1***

The *Information Router* function receives the *Inbound business data* flows.

It is responsible for identifying the respective processing module and for the routing of the business data related to the messages listed in the following table [13]:

MODULE	MESSAGE	FLOW (INTERNAL FORMAT)	REMARKS	URD REFERENCES
LQMG: Outbound Information Management	Receipt	RTGS answer		
		CLM Request		
LQMG: Liquidity Operations	Liquidity Credit Transfer	Immediate / inbound liquidity transfer	Immediate liquidity transfer originally coming from T2S actors or Inbound Liquidity Transfer coming from RTGS actors	**{T2S.06.195}** **{T2S.13.083}** **{T2S.16.170}**
		Liquidity transfer approval request	Only U2A	
SQRA: Query Management	User Query	User query		**{T2S.14.010}** **{T2S.14.020}**
LCMM: Instructions Validation	Settlement Instruction, Cancellation Instruction, Condition Modification, Settlement Restriction	Inbound LCMM Message	Condition Modification includes Hold and Release and Amendment Instruction,	**{T2S.06.120}** **{T2S.09.180}** **{T2S.12.200}**
		Incoming Instruction approval request	Only U2A	
OPSR: Operational Monitoring		Business Information request	Only U2A	
		Trouble case access request	Only U2A	
INTF: Outbound Processing Module	Resend communication	Resend communication		
INTF: Outbound Processing Module	Resend communication	Resend communication		

13 Please note that the described functionality won´t be available before CSLD go-live.

All rights reserved.

MODULE	MESSAGE	FLOW (INTERNAL FORMAT)	REMARKS	URD REFERENCES
		Send General Ledger File		

9 – Check Permission – Resend

Reference Id	***INTF.INP.CPR.1.1***

This function receives the *Inbound business data* flow related to a Resend Communication request, which can be initiated via U2A or A2A. It validates if the T2S system user is allowed to resend the required message of file. The function also processes validations, which are specific for the approval according to the Four-Eyes Principle.

Through the U2A mode a privileged T2S system user may only request a resend of messages or files T2S sent out via the A2A mode. Responses sent by T2S via the U2A mode cannot be resent in any form.

After finalisation of the business validation (e.g. requested message exist) the function sets the Business Validation Status in the *Inbound Individual Message* and related status history data store to document the successful finish of the validation. In case of a negative result, the *Check Permission – Resend* function creates an *Inbound processing rejection - Permission* flow including the error code information and sends it to the *Logging Outbound Business Data* function in the *Outbound Processing Module* (U2A or A2A). Additionally this function uses the data store Validation Error to store the reason for the rejection.

When the permission check is completed with a positive result, the function creates the *Validation Result – resend* flow, sends it to the *Logging Outbound Business Data* function in the *Outbound Processing Module* (U2A or A2A) and the *Resend Communication* flow is forwarded to the function *Logging Outbound Business Data* function in the *Outbound Processing Module* A2A **{T2S.12.160}**.

Inbound Processing Module – User-to-Application

1 – U2A Request Parsing

Reference Id	***INTF.INP-U2A.MSP.1.1***

This function:

- Receives inbound *U2A - XHTML request* flows;
- Performs the formal set of technical validations to validate the syntax and structure of messages required by T2S in the U2A mode (ISO 20022 standard is not adopted);
- Extracts the business data from the communication which are needed by the subsequent modules (e.g. ISIN code, counterparty codes for settlement instructions, T2S System User initiating the communication) **{T2S.12.180} {T2S.12.210}**;
- Structures the business data into the internal format to be processed by the subsequent modules;
- Stores the *U2A Inbound Communication* and the *Inbound Individual Message* and related status history.

All rights reserved.

In case the validation fails, this function creates an *Inbound processing rejection - U2A Parsing* flow and sends it to the *Logging Outbound Business Data* function in the *Outbound Processing Module* - U2A. Otherwise, if the result of the parsing is positive, the function sends the *Inbound business data* flow to the function *Check Privilege*.

2 – Check Privilege

Reference Id	***INTF.INP.CHP.1.1***

The same *Check Privilege* function is used for the U2A and A2A mode. It is described in the *Inbound Processing* Module Application-to-Application mode.

3 – Check for U2A Availability

Reference Id	***INTF.INP-U2A.CFA.1.1***

This function receives the following data flows:

- *Inbound business data*;
- *Start Queuing System status events*;
- *End Queuing System status events*.

The events concerning the System Status inform the function about the current system status therefore they enable the function to check on the availability. The function uses this information for processing the *Inbound business data* flow.

After receiving the *Inbound business data* flow the function checks, if the U2A request has to be rejected due to restricted availability for the dedicated functionality. This is relevant for:

- Liquidity Transfers
  - Between Business date change and Change of business date completed and
  - During cycle 2 / preparation of sequence y
- Securities position related Queries during sequences with impact on positions
- Balance Queries **{T2S.14.080}** during night-time sequence.
- Penalty queries and modifications during the processing of Penalties.

If both conditions are fulfilled, the function rejects the request due to running sequence. It therefore creates an *Inbound processing rejection – U2A Availability* flow.

The function forwards all other *Inbound business data* flows to the *Information Router* function.

4 – Information Router

Reference Id	***INTF.INP.INR.1.1***

The same *Information Router* function is used for the U2A and A2A mode. It is described in the *Inbound Processing* Module Application-to-Application mode.

5 – Check Permission – Resend

Reference Id	***INTF.INP.CPR.1.1***

All rights reserved.

The same *Check Permission - Resend* function is used for the U2A and A2A mode. It is described in the *Inbound Processing* Module Application-to-Application mode.

3.2.4.4 Description of the Input/Output of the module

The *Inbound Business Data* flow described before is used as place holder for the various different business flows, which have to be delivered to the back end modules (special case is the *Resend Message* flow, which is sent to the *Outbound Processing Module*. [14]

FLOW	IN/OUT	DESCRIPTION	FROM	TO
U2A - XHTML request	IN		INTF: Communication Module	
Stored inbound communication – individual message	IN	Inbound individual message received in A2A	INTF: Communication Module	
Stored inbound communication – file	IN	File received in A2A	INTF: Communication Module	
Start Queuing System status events	IN		BDM	
End Queuing System status events	IN		BDM	
Inbound processing rejection U2A- Parsing	OUT	Rejection of a single inbound message delivered in U2A due to parsing error		INTF: Outbound Processing Module
Inbound processing rejection – Privilege	OUT	Rejection of a inbound individual message delivered in U2A or in A2A due to missing privilege		INTF: Outbound Processing Module
Inbound processing rejection – U2A Availability	OUT	Rejection of a inbound individual message delivered in U2A due to unavailability		INTF: Outbound Processing Module
Inbound processing rejection – Permission	OUT	Rejection of a inbound individual message delivered in U2A or in A2A due to missing permission		INTF: Outbound Processing Module

14 Please note that the described functionality won´t be available before CSLD go-live.

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Inbound processing rejection – Query Queuing	OUT	Information about query queuing in order to close real-time channel		INTF: Outbound Processing Module
Validation result – resend	OUT	Result of business validation for Resend Communication to be sent in U2A or in A2A		INTF: Outbound Processing Module
Inbound processing rejection - A2A File	OUT	Rejection of a file delivered in A2A		INTF: Outbound Processing Module
Inbound processing rejection - A2A Schema	OUT	Rejection of a single inbound message delivered in A2A due to schema validation		INTF: Outbound Processing Module
Inbound processing rejection - A2A Additional technical validation	OUT	Rejection of a single inbound message delivered in A2A due to duplicate check or parsing error		INTF: Outbound Processing Module
Resend communication	OUT	U2A or A2A initiation of the resending of an already sent A2A message or file		INTF: Outbound Processing Module
Send General Ledger File	OUT	Initiation of the sending of General Ledger File to CLM		INTF: Outbound Processing Module
RTGS answer	OUT			LQMG: Outbound Information Management
CLM Request	OUT			LQMG: Outbound Information Management
Immediate / inbound liquidity transfer	OUT	Immediate liquidity transfer originally coming from T2S actors or Inbound Liquidity Transfer coming from RTGS actors		LQMG: Liquidity Operations

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Liquidity transfer approval request	OUT			LQMG: Liquidity Operations
User query	OUT			SQRA: Query Management
Inbound LCMM message	OUT			LCMM: Instruction Validation
Inbound Penalty Modification message	OUT			LCMM: Penalty Mechanism
Incoming Instruction approval request	OUT			LCMM: Instruction Validation
Incoming Instruction approval request	OUT			LCMM: Penalty Mechanism
Business Information request	OUT	Only available in U2A		OPSR: Operational Monitoring
Trouble case access request	OUT	Only available in U2A		OPSR: Operational Monitoring

3.2.4.5 Data accessed by the module

Data stores

All inbound/outbound communication except queries and the related results are stored.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
File / Message Format		Read	Basis of this data store are the schema files of all XML files and messages used in T2S.
U2A Inbound Communication		Write	Stored for each authenticated U2A communication
A2A Inbound Communication Mean – A2A File		Read/Write	Stored for each authenticated A2A communication with file structure
A2A Inbound Communication Mean – A2A Message		Read/Write	Stored for each individual A2A message. This includes messages received in a file, if the file was positively validated.

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
Inbound Individual Message		Read/Write	Update of the Inbound Individual Message due to Processing Module.
Inbound Individual Message and related status history	Inbound Individual Message Inbound Individual Message Status History	Read/Write	It allows the storage of the Inbound Individual Message including the status.
RPDM – Users, Roles and Privileges - Inbound	Privilege System Privilege Role Privilege Role Role User User Object Privilege Role Privilege Object	Read	It allows the check on the definition of privileges which are assigned to a user.
Four-Eyes Principle		Read	Documentation of processing of requests in Four-Eyes principle
Validation Error		Write	For the link between A2A Communication Mean and Error Type
Resendable Communication	Outbound Individual Message Outbound Message Recipient Outbound Communication Outbound Message Delivery Outbound Message-File Link Outbound File Delivery	Read	It allows the check on existing outbound communication and their recipients
Permission relevant Reference Data	Role Privilege Object SD Secured Group SD Secured Object SD Secured Object with profile	Read	It allows the check on the definition of secured objects.

All rights reserved.

3.2.5 Outbound Processing Module

3.2.5.1 Description of the module

The *Outbound Processing Module* receives the data in the internal format from the processing modules of other domains. It checks the message subscription preferences, generates the relevant messages in the necessary format and sends these messages (if relevant bundled into files) via *Communication Module* to the T2S System Users in U2A or to the Party Technical Address in A2A.

It is ensured that outbound communication reaches the appropriate party technical address **{T2S.12.140}** and is delivered on due time to the receiving T2S actors **{T2S.12.160}**.

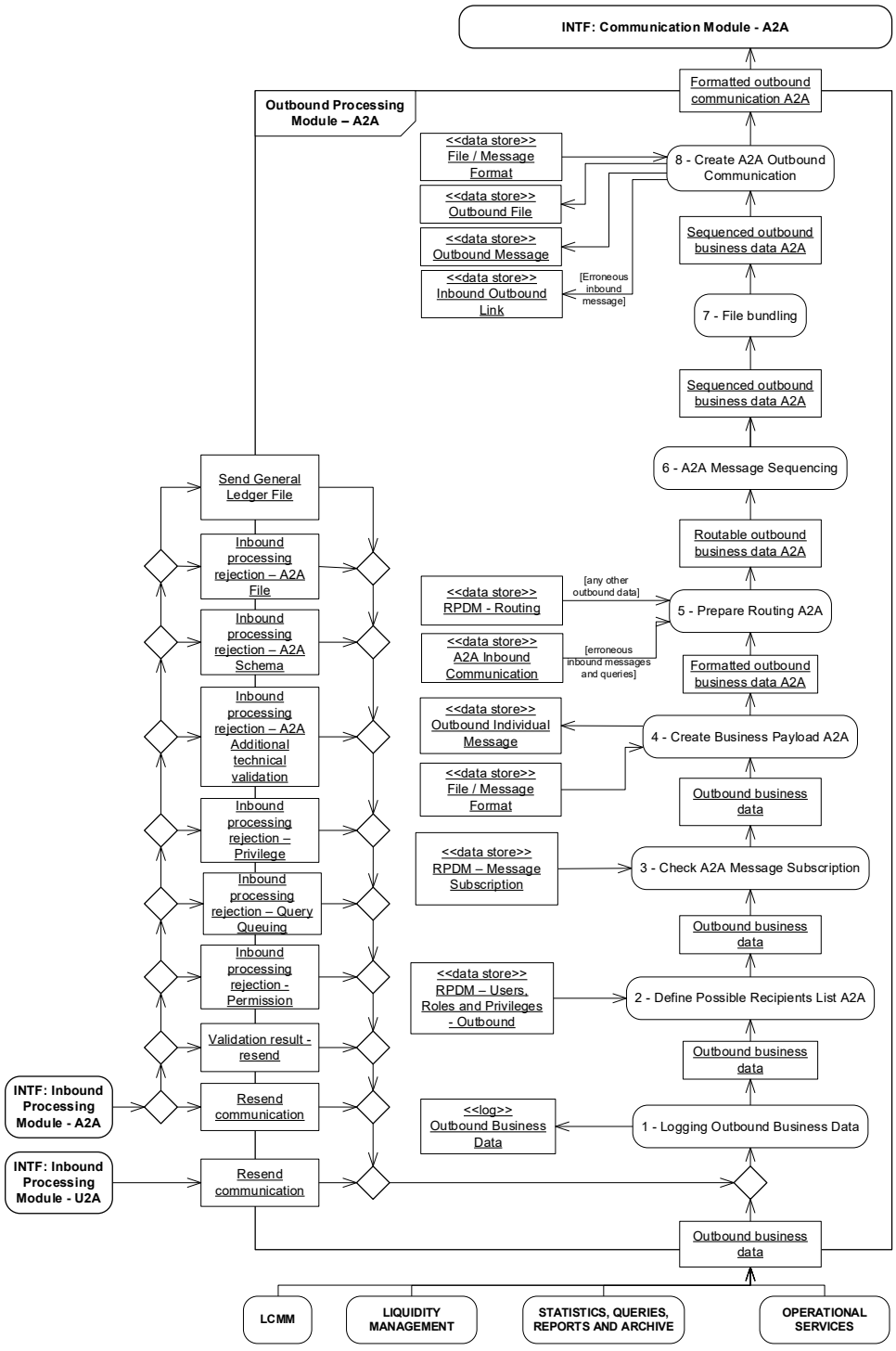
The *Outbound Processing Module* is composed out of two sub-modules:

- *Outbound Processing Module* for Application-to-Application mode;
- *Outbound Processing Module* for User-to-Application mode.

All rights reserved.

3.2.5.2 Diagram of the module

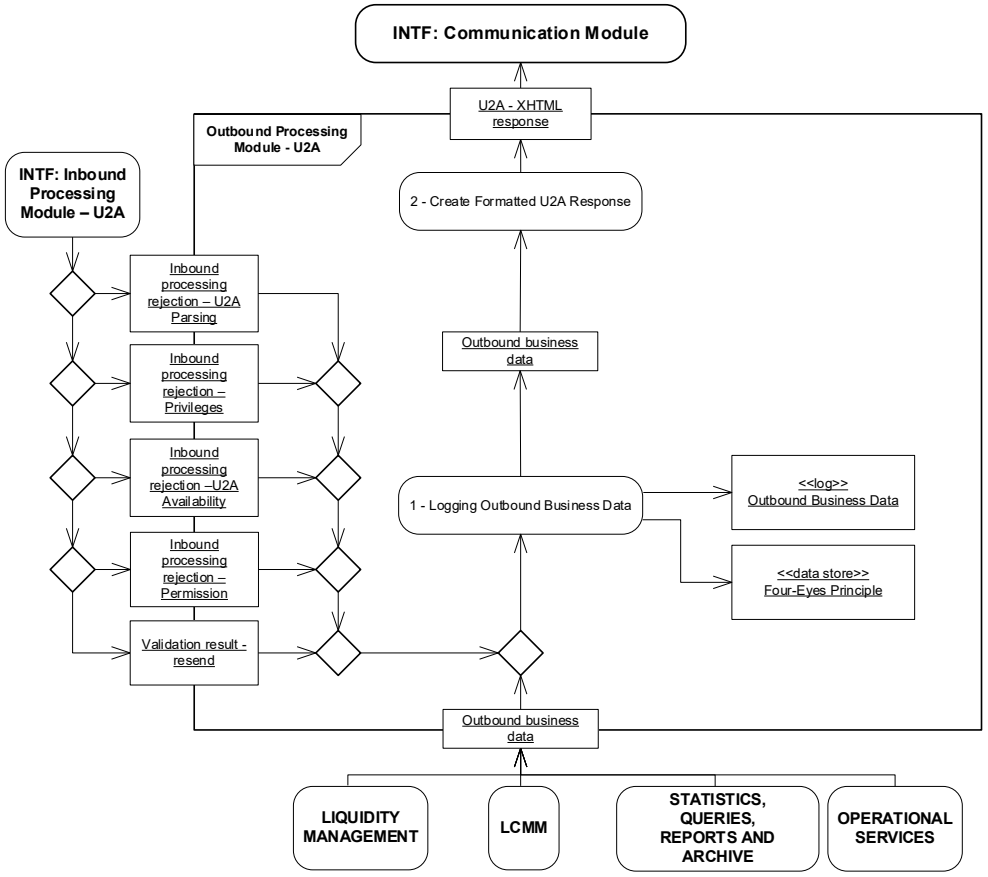
Outbound Processing Module for Application-to-Application mode (A2A) [15]



15 Please note that the described functionality won´t be available before CSLD go-live.

All rights reserved.

Outbound Processing Module for User-to-Application mode (U2A)



3.2.5.3 Description of the functions of the module

Outbound Processing Module – Application-to-Application mode

1 – Logging Outbound Business Data

Reference Id	***INTF.OUP.LOB.1.1***

This function receives:

- The *Outbound business data* flows for the U2A and A2A mode from the processing modules (e.g. results on queries);
- The *Resend communication* flow*;*
- The error condition flows (e.g. *Inbound processing rejections)*.
- The Send General Ledger flow [16]

It logs the *Outbound business data* flow in the log *Outbound Business Data* for short term traceability purposes.

16 Please note that the described functionality won´t be available before CSLD go-live.

All rights reserved.

For the U2A mode the function additionally checks, whether the *Outbound business data* flow contains positive result of the business validation of an initial request or a second user action related to the Four-Eyes principle. If it contains an initial request in Four-Eyes principle the function creates a new instance in the *Four-Eyes principle* data store, stores the reference to the business object and sets the status value to *"Awaiting Approval"*. If it contains a second user action related to the Four-Eyes principle the function updates the status value of the respective Four-Eyes principle depending on the result to *"Queued", "Approved"*, *"Revoked"*or *"Rejected"*.

After successful business validation the function start the inbound copy workflow for A2A and U2A requests subject to inbound copies.

As the function is used for the U2A and the A2A mode, the *Outbound business data* flow is delivered to the function *Create Formatted U2A Response* or to the function *Define Possible Recipients List A2A*.

T2S sends messages to the T2S Actors according to the information included in the event-driven data flows delivered to the *Interface* from the back end modules **{T2S.13.060}**. Such messages are only send out via A2A.

2 – Define Possible Recipients List A2A

Reference Id	***INTF.OUP-A2A.PRL.1.1***

The function receives the *Outbound business data* flow including additional business information (e.g. relevant accounts/parties).

The function uses the business data received from the backend modules and the respective privilege to enlarge the list of parties which are allowed to receive the business data in order to cover for messages subject to copies also third party receiver business cases in which a party not included in the list of interested parties shall receive copies of messages.

Afterwards the *Outbound business data* flow including the possible recipients is forwarded to the *Check A2A Message Subscription* function **{T2S.12.140}**.

3 – Check A2A Message Subscription

Reference Id	***INTF.OUP-A2A.CMS.1.1***

This function receives the *Outbound business data* flow.

In case of:

- Reactions on erroneous inbound messages;
- Reports (option for subscription is defined via Report Configuration);
- Query results,
- Floor/Ceiling notifications
- Outbound liquidity transfers

All rights reserved.

the message is sent independent from the message subscription. For any other outbound communication the function checks for all the listed possible recipients, whether they opted for receiving the message **{T2S.12.150}**. For each party that wants to receive the message, the *Party* and a copy of the *Outbound business data* flow is delivered to the *Create Business Payload A2A* function. If no party wants to receive the message, the Outbound Processing is finished.

The message subscription service gives flexibility to all CSDs and directly connected T2S parties to choose the messages they wish to receive or not, in order to handle their business activities **{T2S.13.010}**. The message subscription functionality is available independently of the case why the T2S Actor is recipient (Interested Party or Privileged Third Party).

All messages, which are used by T2S, are available for message subscription **{T2S.13.020}** (except the above mentioned, which are always delivered. The message subscription definition is stored and maintained by T2S System Users in CRDM **{T2S.13.030}**. The respective *Message Subscription* definitions for the message subscription services are based on the following parameters:

- Message type;
- Instruction type;
- Message status;
- Instruction status;
- Party;
- Securities Account;
- T2S Dedicated Cash Account;
- ISIN;
- Transaction Code;
- Currency;
- Already matched flag;
- Settlement Transaction Condition Code;
- Business Sending Party;
- System User Reference;
- Instructing Party;
- Instruction Status Reason Code.

The usage of the above mentioned criteria is optional. They can be combined or the criteria can be ignored. In case the above mentioned criteria are ignored, it is still possible to subscribe to all messages in all cases or no messages at all **{T2S.13.050}**.

All rights reserved.

The positive/negative parameter allows subscribing Parties to explicitly refuse to receive messages that match certain criteria. If the parameter is set to negative it means that the Party does not receive messages which match the rule set. [17] By setting it to positive the Party receive messages matching the rule set (i.e. the normal subscription behaviour). In the event of an overlap, "negative" rule sets have priority over "positive" rule sets.

These criteria are used according to the validity period **{T2S.13.040}** of each Message Subscription definition within a system entity. The Message Subscription definition is composed out of several rules. Each single rule has a Rule Sequence number which defines the processing order **{T2S.11.640}**.

The Message Subscription Matrix shown below gives an example rule set of three rules for Interested Party B, which in this case is a T2S Party of CSD X. The configuration is valid from January 1st, 2013.

Message Subscription Matrix – Example:

System Entity: CSD X

Recipient: Interested Party B

Rule Set Valid From: January 1st, 2013

RULE SEQUENCE	MESSAGE TYPE	MESSAGE STATUS	INSTRUCTION TYPE	INSTRUCTION STATUS	PARTY	SECURITIES ACCOUNT	ISIN	T2S DEDICATED CASH ACCOUNT	TRANSACTION CODE	CURRENCY	ALREADY MATCHED FLAG	SETTLEMENT TRANSACTION CONDITION CODE	BUSINESS SENDING PARTY	SYSTEM USER REFERENCE	INSTRUCTING PARTY	Instruction Status Reason Code
1								Cash A/C 9876								
2	Settlement Instruction					Account ABC										
3					Bank A											

17 Exception exists for parameter "Instruction Status Reason Code". For this parameter, also the paths of Message status, Instruction status and all Instruction Status Reason Codes reported in the respective message are relevant. As long as one or more statuses or codes are provided in the respective message that do not match the one set up in the negative message subscription rule, the negative rule is not applied and the messages are sent provided a corresponding positive subscription rule exists.

All rights reserved.

RULE SEQUENCE	MESSAGE TYPE	MESSAGE STATUS	INSTRUCTION TYPE	INSTRUCTION STATUS	PARTY	SECURITIES ACCOUNT	ISIN	T2S DEDICATED CASH ACCOUNT	TRANSACTION CODE	CURRENCY	ALREADY MATCHED FLAG	SETTLEMENT TRANSACTION CONDITION CODE	BUSINESS SENDING PARTY	SYSTEM USER REFERENCE	INSTRUCTING PARTY	Instruction Status Reason Code
					Bank B											

- According to Rule 1, the Interested Party B receives all status information associated to any posting performed on the T2S Dedicated Cash Account 9876.
- According to the second rule, Interested Party B receives all status information linked to the posting on the securities account ABC of the security leg of Settlement Instruction of any type.
- According to the final rule, Interested Party B receives copies of all messages between T2S and the specified parties, Bank A and Bank B.

4 – Create Business Payload A2A

Reference Id	***INTF.OUP-A2A.CBP.1.1***

This function receives the *Outbound business data* flow and it creates the business payload of the message, which is common for all possible recipients by using the defined structure, syntax and format defined in data store File / Message Format. Afterwards the *Formatted outbound business data* flow is forwarded to the *Prepare Routing A2A* function.

The business data include all necessary data (e.g from the *Life Cycle Management and Matching* **{T2S.12.220}**) to generate the appropriate messages in correct structure, syntax and format **{T2S.12.030}**. For the T2S communication via XML messages the ISO 20022/UNIFI is the single standard **{T2S.12.040}** (including queries **{T2S.14.010}**).

In detail, the function creates the following messages out of flows from other modules [18]:

MODULE	MESSAGE	FLOW (INTERNAL FORMAT)	REMARKS	URD REFERENCES
LCMM: Status Management	Validation Status, Matching Status, Maintenance Instruction Status, Blocking Status, Reservation	Message data	Settlement-related information (settlement confirmation, settlement status message)	**{T2S.03.120}** **{T2S.05.480}** **{T2S.05.530}** **{T2S.09.220}** **{T2S.09.230}**

18 Please note that the described functionality won´t be available before CSLD go-live.

All rights reserved.

MODULE	MESSAGE	FLOW (INTERNAL FORMAT)	REMARKS	URD REFERENCES
	Status, Settlement Status, CoSD Status, Eligibility Status		Auto-collateralisation related information (transferring/blocking and retransferring/un-blocking status message) In the status message the source of input is reported (e.g. amendment instruction of a process indicator made by a CSD following a corporate event on a pending instruction sent by a directly connected T2S party).	**{T2S.09.250}** **{T2S.12.220}** **{T2S.13.070}** **{T2S.13.080}** **{T2S.13.090}** **{T2S.13.100}** **{T2S.13.110}** **{T2S.13.120}** **{T2S.13.130}**
	Bank to Customer Debit Credit Notification	Message data	Credit Debit Notification stemming from instructions	
SQRA: Query Management	Query Result	Queried data		**{T2S.14.010}** **{T2S.14.020}**
	Error	Error data queries		
LQMG: Outbound Information Management	Liquidity Credit Transfer	Outbound Liquidity Transfer		**{T2S.06.060}**
	Liquidity Credit Transfer	**Liquidity Transfer - Pull Request** **{CR-720}**		
	Liquidity Debit Transfer	Outbound Liquidity Transfer	Liquidity Debit Transfer for Direct Debits	
	Receipt	Information to RTGS System		**{T2S.06.680}**
	Bank to Customer Debit Credit Notification	Information to T2S Actor (here: Credit or Debit Notification)		**{T2S.13.086}** **{T2S.06.690}**

All rights reserved.

MODULE	MESSAGE	FLOW (INTERNAL FORMAT)	REMARKS	URD REFERENCES
	Receipt	Information to T2S Actor (here: Liquidity Transfer Status, CB Info RTGS-Rejection, Invalid RTGS answer)		**{T2S.06.196}** **{T2S.06.710}** **{T2S.13.086}**
	Return Account	Information to T2S Actor (here: Floor/Ceiling)		**{T2S.06.233}** **{T2S.06.234}** **{T2S.13.086}**
	Receipt	Information to CLM		
LQMG: Liquidity Operations	Receipt	Rejection	Includes rejection of Inbound Liquidity transfer, Immediate liquidity transfer order	**{T2S.06.580}**
	Receipt	Accepted Liquidity Transfer		**{T2S.06.580}**
SQRA: Report Management	Report	Report		**{T2S.13.160}**
OPSR: Operational Monitoring	Alert / Report	Alert / display / report	Only available in U2A	
		Trouble case access response	Only available in U2A	

5 – Prepare Routing A2A

Reference Id	***INTF.OUP-A2A.PRR.1.1***

This function receives the *Formatted outbound business data A2A* flow.

In case of:

- Reactions on erroneous inbound messages;
- Query results,

the message is directly sent to the *Party Technical Address* and *Network Service*, which were used for sending the related A2A inbound communication.

Any other outbound communication is routed according to the defined conditional routing or according to the default routing. The default routing definition is used if no condition matches for this kind of outbound communication {**T2S.12.150}**.

The possible criteria for conditional routing are:

- Size boundaries,

All rights reserved.

- Currency,
- Message type [19].

When defining a routing configuration, a T2S actor can specify through the compression setting whether T2S must compress the relevant data before sending them to the recipient. This function forwards the respective compression information for respective handling in *Communication Module*.

The definition of the conditional routing by the Rule Sequence number is stored in CRDM

The output of this function is the *Routable outbound business data A2A* flow which is sent to the *A2A Message Sequencing* function.

6 – A2A Message Sequencing

Reference Id	***INTF.OUP-A2A.MSE.1.1***

This function receives the *Routable outbound business data A2A* flow. It ensures that all outbound messages, where the correct "business" order cannot be derived from the message content itself (i.e. LCMM status notification), include a sequence number which allows directly connected T2S parties to process messages delivered by T2S in the correct business order. All other messages do not contain a sequence number. For each party technical address, the *Message Sequencing* function uses an independent counter. Every time a message, which is subject to sequencing, is sent to the given technical address the function increments the value of the counter by one [20]. The function delivers the *Sequenced outbound business data A2A* flow to the File Bundling function.

7 – File Bundling

This function receives the Sequenced outbound business data A2A flow and is responsible for the file bundling process.

This process is divided into two parts: the mandatory bundling during the night-time settlement and the optional file bundling during the whole day. Exceptions during the business day are the maintenance window and during the period close to the DVP cut off.

Afterwards the function delivers the Sequenced outbound business data A2A flow to the Create Outbound Communication A2A function.

8– Create A2A Outbound Communication

Reference Id	***INTF.OUP-A2A.COC.1.1***

The function *Create A2A Outbound Communication* receives the *Sequenced outbound business data A2A* flow.

The *Create A2A Outbound Communication* function:

- creates the (Business Application Header and the Business File Header) supported by the defined structure, syntax and format defined in data store File / Message Format;
- delivers the message as the *Formatted outbound communication A2A* to the *Communication Module*;

19 The list of message types includes "NightTimeSettlementRelatedMessages" as possible routing criterion.

20 Additional details concerning the counter usage are provided in the User Detailed Functional Specification (UDFS).

All rights reserved.

- sends all the settlement-related messages (see UDFS for exhaustive list of messages) grouped into files on single sequence level during night-time settlement,
- stores the *Outbound File*, the *Outbound Message* and, in case of inbound processing, rejections it also stores the *Inbound – Outbound Link* [21].

In the A2A mode all messages are sent in due time to T2S System Users. To ensure that the information reaches the requesting T2S System User, positive and negative acknowledgement features are used (T2S expects the receiving party to acknowledge the message).

Outbound Processing Module – User-to-Application mode

1 – Logging Outbound Business Data

Reference Id	***INTF.OUP.LOB.1.1***

The same *Logging Outbound Business Data* function is used for the U2A and A2A mode. It is described in the *Outbound Processing* Module Application-to-Application mode.

2 – Create Formatted U2A Response

Reference Id	***INTF.OUP-U2A.CFR.1.1***

The function receives the *Outbound business data* flow which is sent to the user in the U2A mode. This function creates the *U2A - XHTML response* flow, which is forwarded to the *Assign Certificate and Encryption U2A* function. The *U2A - XHTML response* does not adopt the ISO 20022 message standard.

3.2.5.4 Description of the Input/Output of the module

The *Outbound Business Data* flow described before is used as place holder for the various different business flows, which have to be received from the back end modules (special case is the *Resend Communication* flow, which is received from the *Inbound Processing Module*.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Message data	IN		LCMM: Status Management	
Outbound Penalty Modification message	IN		LCMM: Penalty Mechanism	
Approval request response	IN		LCMM: Penalty Mechanism	
Outbound Liquidity Transfer	IN		LQMG: Outbound Information Management	
Information to T2S Actor	IN		LQMG: Outbound Information Management	

21 The resent message is not stored a second time.

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Information to RTGS System	IN		LQMG: Outbound Information Management	
Liquidity Transfer - Pull Request	**IN**		**LQMG: Outbound Information management** **{CR-720}**	
Accepted Liquidity Transfer	IN		LQMG: Liquidity Operations	
Rejection	IN		LQMG: Liquidity Operations	
Queried data	IN		SQRA: Query Management	
Error data queries	IN		SQRA: Query Management	
Report	IN		SQRA: Report Management	
Alert / Display / Report	IN		OPSR: Operational Monitoring	
Trouble case access response	IN		OPSR: Operational Monitoring	
Event Maintenance response	IN		BDM	
Inbound processing rejection – U2A Parsing	IN	Rejection of a single inbound message delivered in U2A due to parsing error	INTF: Inbound Processing Module	
Inbound processing rejection – Privilege	IN	Rejection of a inbound individual message delivered in U2A or in A2A due to missing privilege	INTF: Inbound Processing Module	
Inbound processing rejection – U2A Availability	IN	Rejection of a inbound individual message delivered in U2A due to unavailability	INTF: Inbound Processing Module	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Inbound processing rejection – Permission	IN	Rejection of a inbound individual message delivered in U2A or in A2A due to missing permission	INTF: Inbound Processing Module	
Inbound processing rejection – Query Queuing	IN	Information about query queuing in order to close real-time channel	INTF: Inbound Processing Module	Inbound processing rejection – Query Queuing
Validation Result – resend	IN	Result of business validation for Resend Message to be sent in U2A or in A2A	INTF: Inbound Processing Module	
Inbound processing rejection – A2A File	IN	Rejection of a file delivered in A2A	INTF: Inbound Processing Module	
Inbound processing rejection – A2A Schema	IN	Rejection of a single inbound message delivered in A2A due to schema validation	INTF: Inbound Processing Module	
Inbound processing rejection – A2A Additional technical validation	IN	Rejection of a single inbound message delivered in A2A due to duplicate check or parsing error	INTF: Inbound Processing Module	
Resend communication	IN		INTF: Inbound Processing Module	
U2A - XHTML response	OUT			INTF: Communication Module
Formatted outbound communication A2A	OUT			INTF: Communication Module

3.2.5.5 Data accessed by the module

Data stores

All inbound/outbound communication except queries and the related results are stored.

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
File / Message Format		Read	Basis of this data store are the network specific file or message formats and the schema files of all XML messages used in T2S.
Outbound Individual Message		Write	It allows the storage of the business payload.
Outbound Message	Outbound Message Recipient Outbound Message Delivery	Write	It allows the storage of the possible recipients and the related sequence number plus Business Application Header related information.
Outbound File	Outbound File Delivery Outbound Message-File Link	Write	It allows the storage of outbound file related information.
RPDM – Message subscription	Message Subscription Rule Parameter Type Message Subscription Rule Parameter Message Subscription Rule Message Subscription Rule Set Party	Read	It allows the check of message subscription rules for A2A mode.
RPDM – Routing	Party Party Technical Address Default Routing Conditional Routing Network Service Network Provider	Read	It allows the routing of a message in A2A mode.
Inbound Outbound Link		Read / Write	The link between A2A Inbound Communication Means and Outbound Individual Message is only stored for erroneous A2A inbound communication

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
RPDM – Users, Roles and Privileges - Outbound	Privilege System Privilege Role Privilege Role Role User User Object Privilege Role Privilege Object	Read	It allows the creation of a possible recipients list in A2A mode.
A2A Inbound Communication		Read	It allows the routing to the sender of the inbound message
Four-Eyes Principle		Read / Write	The data store allows the update of the existing Four-Eyes principle entry.

Logs

The log is stored for short term traceability. Therefore they are not archived.

DATA	ACCESS MODE	COMMENTS
Outbound Business Data	Write	It allows the storage of the outbound business data received from the subsequent modules.

3.2.6 T2S Graphical User Interface

3.2.6.1 General aspects

T2S offers access to T2S System Users through:

- An interface for back-office applications (A2A);
- A graphical user interface for User-to-Application mode (U2A).

The T2S Graphical User Interface (GUI) provides T2S System Users with the access to a comprehensive browser application to display from and update in T2S online information.

In comparison to the exchange of information between T2S and the internal application of the party in A2A mode, using the GUI for User-to-Application mode means that information is displayed in a browser running on a Personal Computer.

For example, the T2S System User requests information about a specific instruction sent on the current settlement day:

- A2A: T2S System User sends the respective query message to T2S and receives the related response;

All rights reserved.

- U2A: GUI lists the instructions sent by the T2S System User on the current settlement day in a first screen as an overview within a table. Detailed information can be requested by selecting the specific instruction using buttons and selection fields.

A detailed description of messages used for the A2A communication within T2S will be available in the User Detailed Functional Specifications (UDFS) whereas further details of T2S GUI (screens, buttons, system requirements, etc.) will be explained in the User Handbook (UHB). Some functionality (e.g. specific queries **{T2S.14.020}**) is only available via U2A.

T2S GUI is based on English language only **{T2S.19.280}**.

The T2S GUI supports the following non-exhaustive list of maintenance and querying functions **{T2S.12.250}**. It enables users to:

- Issue online query requests to T2S (e.g. such as balance requests, status requests **{T2S.11.330}**);
- Display results in a readable and standardised way;
- Export a query result by using common industry-wide standard formats (The extract includes exactly the same information, which is provided by the query without update, including the query parameters and the query timestamp.);
- Input and maintain settlement instructions and liquidity transfer orders;

3.2.6.2 Common Principles and Design of the Screens

The T2S GUI is based and developed on the style and navigation guidelines agreed in the context of the European Single Interface (ESI) **{T2S.12.230}**.

Therefore, it is ensured that it follows the main principles mentioned in the following non-exhaustive list:

- Neutrality: The design reflects that T2S is an European system. It does not point out a specific country or financial centre.
- Clarity and easy navigation: The T2S GUI is visually, conceptually and linguistically clear by having a clear and simple navigation which easily allows the identification of different areas.
- User-Task-orientation (Usability): The structure of the menu groups the information available from the perspective of a user. In addition, colours and graphics are used to support the orientation of the user. It permits the user not to focus on the mechanism of the GUI but on the information and tasks to be done.
- User control: The T2S GUI enables the user's control by providing a basic behaviour and placement of controls (e.g. Menu bar, pull-down menu, check box, radio button) which are consistent throughout the application. Furthermore it provides immediate results of user actions and feedback in case an action is not possible.
- Comprehensibility: The T2S GUI is designed in a way that it can be easily understood and learned. A user is able to know:
  - What to do,
  - What to look at,

All rights reserved.

- When, where and how to do it.
- The flow of actions, responses, visual preparations and information are designed in a way that is easy to recollect and place in context.

- Consistency: Based on an agreed styling and layout the identical implementation of common functions is ensured.
- Security Aspects: Authentication functions enable authorised users to access the T2S GUI. The user authentication is required only once while establishing a connection to T2S and will be valid through the running session. Therefore user identification is not requested for each action/message like in A2A mode. That means that a user does not need to log in more than once during a single session.
  The access rights assigned to a user determine which functions the user can execute and the data the user can see and maintain. This includes the Four-Eyes principle process.
- Reaction times: The T2S GUI presents online information which is provided by the back end modules for 95% of the basic queries within three seconds. If the complexity of the query to be processed requires more time, the screen includes a signal that the query execution is still ongoing. A progress information (e.g. remaining execution time) will not be available. During the ongoing query execution the T2S System User is able to stop loading the response **{T2S.17.140}**.
  Any data to be created, modified or deleted via the user-to-application interface is updated directly. The time limit is five seconds for 95% of standard changes **{T2S.17.160}**.
- Support: The T2S GUI provides help at least within a dedicated help page (e.g. including FAQs) for each screen.

3.2.6.3 Four-Eyes Principle

T2S supports a dual authorisation concept (Four-Eyes Principle) when entering and updating data online through the GUI (U2A communication mode). A privileged T2S System User can specify sets of T2S operations that require a second independent verification and confirmation of an operation before that operation becomes active in T2S.

Action

Three actions Create, Update and Delete are indicated in the following table comparing the Two-Eyes principle with the Four-Eyes principle and their respective results (exhaustive list).

Please note, that not every action is available for every possible object, e.g. there is no update or delete of an immediate liquidity transfer.

ACTION	PRINCIPLE	RESULT
Create	Two-Eyes	A new object is created by one user and immediately available for processing **{T2S.16.230}**.

All rights reserved.

ACTION	PRINCIPLE	RESULT
	Four-Eyes	A new object is created but not yet available because an action from a second user is required (status *"Awaiting Approval"*). This can be twofold: - In case the second user confirms the creation, the new object becomes available for processing, - In case the second user does not confirm the creation the new object is *"Revoked"*.
Update	Two-Eyes	Requested changes are applied by creating a new revision of the relevant object on the back end module side and that new revision becomes immediately available for processing.
	Four-Eyes	Requested changes are applied by creating a new revision of the relevant object on the back end module side, but this new revision is not yet available because an action from a second user is required (status *"Awaiting Approval"*). This can be twofold: - In case the update is confirmed the new revision becomes available for processing, - In case the update is not confirmed the new revision is *"Revoked"*.
Delete	Two-Eyes	A new revision of the relevant object is created on the back end module side containing a flag marking it as logically deleted and making it immediately no longer available for processing **{T2S.16.270}.**
	Four-Eyes	A new revision of the relevant object containing the flag for logical deletion is created on the back end module side, but this new revision is not yet

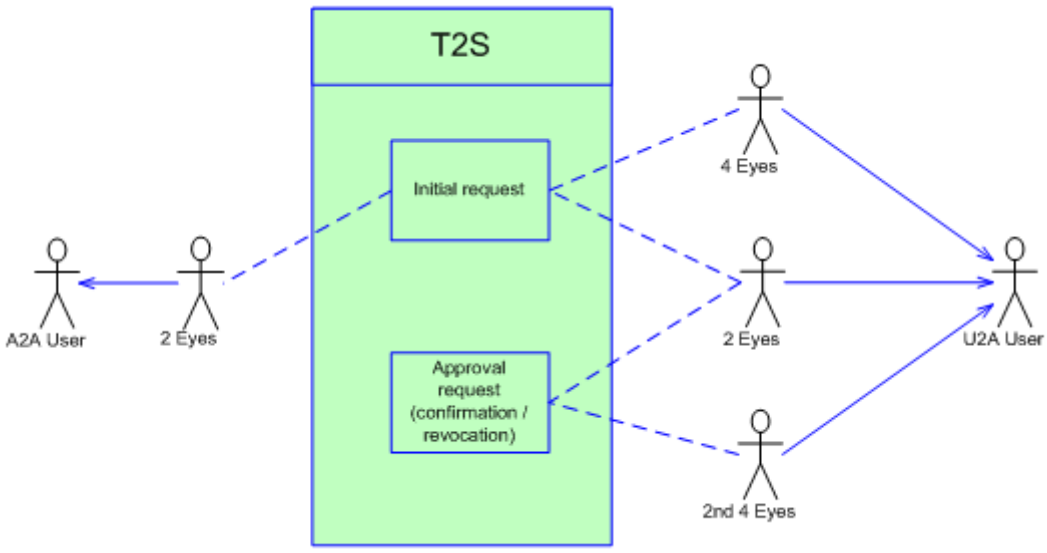
All rights reserved.

ACTION	PRINCIPLE	RESULT
		available for processing because an action from a second user is required (status "*Awaiting Approval*"). This can be twofold: ▮ In case the deletion is confirmed the new revision becomes active (thus marking the object as deleted), ▮ In case the deletion is not confirmed the new revision is "*Revoked*" (and the object remains available).

The following attribute is considered by the entities of the respective back end modules

ATTRIBUTE	DESCRIPTION
Approval Status	It defines whether a request has been approved or revoked by an authorised T2S System User or if it is in the middle of an approval process or rejected by T2S. The exhaustive list of possible values is as follows: ▮ Approved ▮ Awaiting Approval ▮ Rejected ▮ Revoked ▮ Queued

The next diagram depicts the interaction between different types of users with T2S:



All rights reserved.

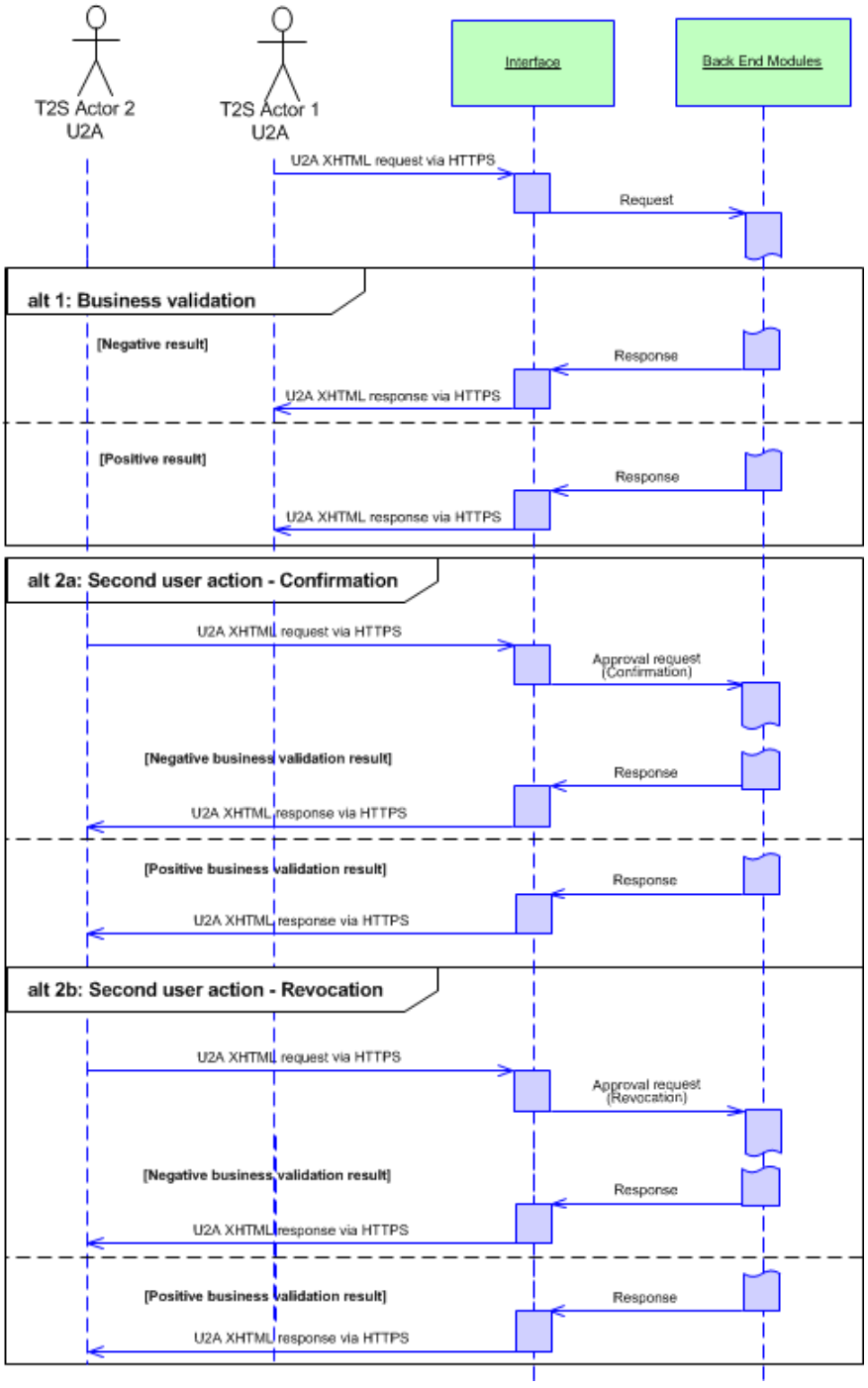
When updating data in T2S that is subject to an independent user verification (Four-Eyes principle), the initial request is sent by a user with a Four-Eyes privilege. As a result, the back end module creates a new revision with a status *"Awaiting Approval"* in the Approval Status attribute of the applicable reference data class. To approve a task, a subsequent confirmation by either a different user with a Four-Eyes privilege or a user with a Two-Eyes privilege is required. The revocation of the task can be performed either by the initial user, or by another user with a Four-Eyes or Two-Eyes privilege.

Processing of UC: Four-Eyes Principle

The general approach for processing Four-Eyes Principle requests is described blow.

All rights reserved.

DIAGRAM 9 - INTERFACE FOUR EYES PRINCIPLE



The *Interface* domain receives an incoming initial request in the U2A mode from a T2S System . In case this request fails the technical validation (see UC-INTF-4: Interface via U2A for details), the *Interface* domain rejects the request. After a successful technical validation, the *Interface* domain forwards the request to the relevant back end module. The forwarded request contains both the relevant business data and the information that the request requires a second user action. The relevant back end module performs the business validation.

All rights reserved.

Alternative 1a - Business Validation: Negative Result

If business validation fails, the back end module rejects the request and sends reference information back to the *Interface* domain. The *Interface* domain delivers the information to the T2S System User.

Alternative 1b - Business Validation: Positive Result

If the business validation is successful, the back end module processes the initial request by creating a pending change, as it requires subsequent action by a second, independent user. The back end module sets the status of the relevant business object to *"Awaiting Approval"* and sends the reference information about the business object that is waiting for approval back to the *Interface* domain. The *Interface* domain creates a new instance of the *Four-Eyes Principle* entity, stores the reference of the business object that is waiting for approval and sets its status value to "*Awaiting Approval"*. In the next step, the *Interface* domain delivers this information to the T2S System User.

A second authorised T2S System User has to confirm or to revoke the request that is waiting for approval to complete the processing.

Alternative 2 - Second user action

The *Interface* domain receives an incoming *Four-Eyes approval request* (confirmation or revocation) in the U2A mode from a T2S System User. If the request fails technical validation the *Interface* domain rejects the approval request. If technical validation is successful, the *Interface* domain forwards the approval request to the relevant back end module. The relevant back end module performs the business validation of the approval request.

If the business validation is successful, the back end module processes the approval request following the described below alternative 2a and 2b. It sets the status of the relevant business object to *"Approved"* or "*Revoked*" and sends the reference information about the business object back to the *Interface* domain. The *Interface* domain links the related approval request with the initial maintenance request and updates the status of the corresponding instance of the *Four-Eyes Principle* entity depending on the result to *"Queued", "Approved", "Revoked" or "Rejected". In* the next step, the *Interface* domain delivers relevant information to the T2S System user.

If the approval request fails business validation the back end module rejects the approval request and sends the reference information back to the *Interface* domain. In the next step, the *Interface* domain delivers this information to the T2S System User.

Alternative 2a - Second user action: Confirmation

The back end module performs a revalidation of the initial request to ensure that the previous update is still consistent:

- If the business revalidation fails, the back end module updates the status of the respective business object to *"Rejected"* and sends the reference information back to the *Interface* domain. The *Interface* domain updates relevant *Four-Eyes Principle* entity instance, as described above, and delivers this information to the sender of the approval request.

All rights reserved.

- After a successful business revalidation, the back end module makes the initial request available for processing in T2S. It updates the status of the respective business object to *"Approved"* and sends the reference information to the *Interface* domain. The *Interface* domain updates relevant *Four-Eyes Principle* entity instance, as described above, and delivers this information to the sender of the approval request.

Alternative 2b - Second user action: Revocation

The back end module revokes initial request.

The back end module does not perform business revalidation of the initial request but sets the status of the respective business object to *"Revoked"*. The respective business object is not available for processing in T2S. The back end module sends back the reference information to the *Interface* domain. The *Interface* domain updates relevant *Four-Eyes Principle* entity instance, as described above, and delivers it to the sender of the approval request.

3.2.7 Interface (IN) Use Cases

Scope

This category of use cases covers the situations where either inbound or outbound communication in A2A or requests and responses in U2A take place between the T2S System User and the *Interface.*

Criteria

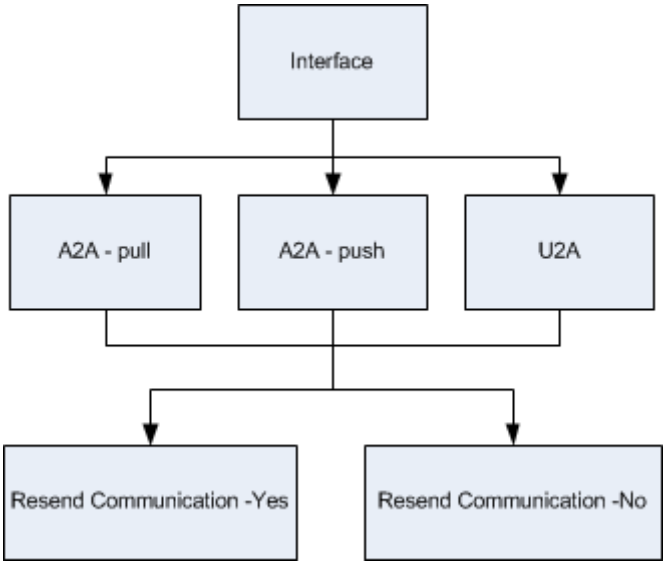
The criteria that characterise an *Interface* use case are: the communication mode, the instructing party and the distinction whether it is a resend communication (i.e. messages or files) or not. These criteria are summarised in the table below (exhaustive list):

CRITERIA	POSSIBLE VALUES	DESCRIPTION
Communication mode	A2A – pull, A2A – push, U2A	See glossary
Resend Communication	Yes, No	Instructing party sends a request to T2S that an already sent communication should be sent again.

List of Use Cases

The criteria described above are reported in the following tree:

All rights reserved.



3.2.8 Processing of the Interface Use Cases

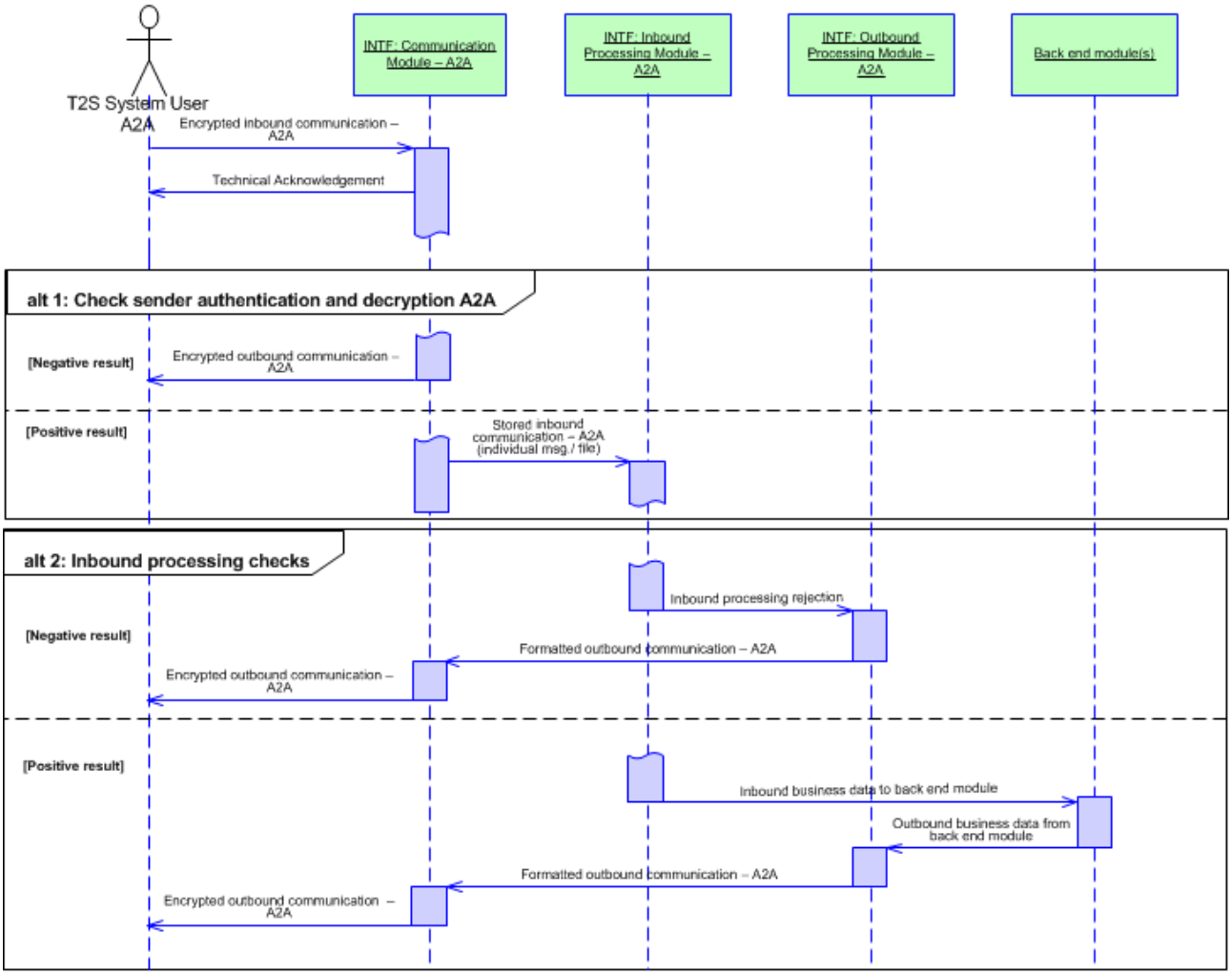
The processing of the following representative use cases is described:

- UC-INTF-1: Interface via A2A - pull;
- UC-INTF-2: Interface via A2A – pull - Resend;
- UC-INTF-3: Interface via A2A – push;
- UC-INTF-4: Interface via U2A;
- UC-INTF-5: Interface via U2A – Resend.

All rights reserved.

3.2.8.1 Processing of UC-INTF-1: Interface via A2A – pull

DIAGRAM 10 - INTERFACE VIA A2A



Business assumption

A T2S System User sends a query request via A2A to the *Interface*.

Processing

The query request in the form of an *Encrypted inbound communication – A2A* enters the system via the *Interface* domain. In a first step the *Communication Module – A2A* sends a *Technical acknowledgment* and checks whether it was sent by a secured and recognised address (authentication):

Alternative 1: Check sender authentication and decryption A2A: Negative Result

In case the T2S System User cannot be authenticated or one of the checks mentioned in *Communication Module – A2A* is not positively validated a rejection including error information is sent to the technical address from which the communication was originally received.

Alternative 1: Check sender authentication and decryption A2A: Positive Result

The communication (message or file) is forwarded to the *Inbound Processing Module – A2A*.

All rights reserved.

In the *Inbound Processing Module – A2A* several checks are performed.

If the communication is a file, the *Inbound Processing Module – A2A* checks that the file has a proper XML structure and performs a duplicate emission check on file level based on the sender and the sender's file reference. Only after passing these checks the file is split into single messages.

The *Inbound Processing Module – A2A* ensures that the individual message is well-formed and the message type is supported by T2S. It performs additional technical validations which were not included in the schema validation. In the next step it guarantees that the sender of an incoming communication is authorised to execute the intended T2S function (based on Certificate DN).

Alternative 2: Inbound processing checks: Negative Result

If the request fails one of these checks, the *Inbound Processing Module – A2A* sends an *Inbound processing rejection* to the *Outbound Processing Module – A2A*. After the formatting and encryption the *Communication Module – A2A* sends the *Outbound Communication* to the T2S System User.

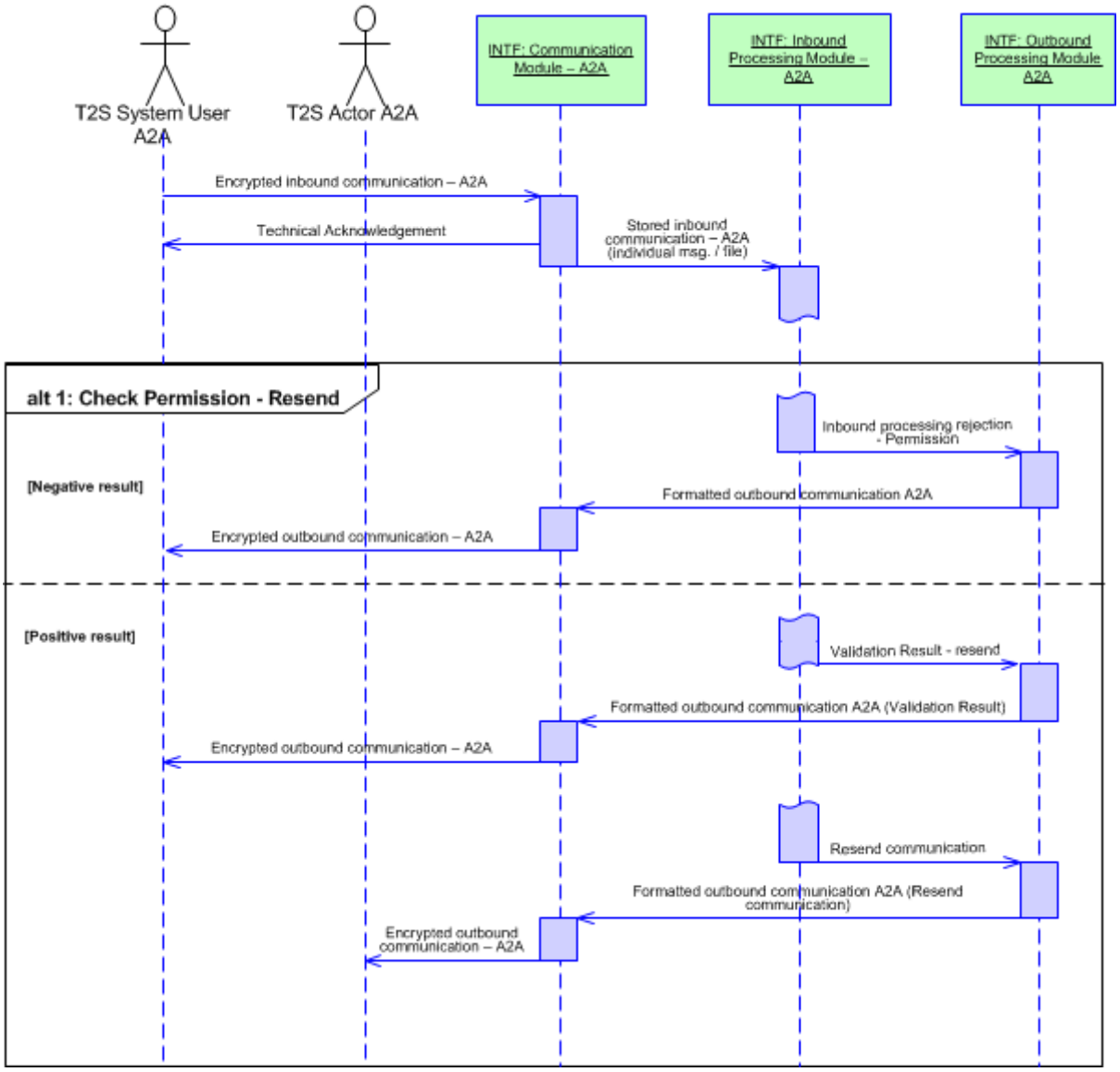
Alternative 2: Inbound processing checks: Positive Result

If the request passes all these checks, the *Inbound business data* is sent to the respective back-end module. After processing the request the module sends the data back to the *Outbound Processing Module* – A2A. After the formatting and encryption the *Communication Module – A2A* sends the *Outbound Communication* – A2A to the T2S System User.

All rights reserved.

3.2.8.2 Processing of UC-INTF-2: Interface via A2A – pull – Resend

DIAGRAM 11 - INTERFACE VIA A2A - RESEND



Business assumption

A T2S System User sends a Resend communication (i.e. messages or files) request via A2A to the *Interface*.

Processing

The *Encrypted inbound communication – A2A* enters the system via the *Interface* domain. In a first step the *Communication Module – A2A* sends a *Technical acknowledgment* and checks whether it was sent by a secured and recognised address (authentication).

Alternative 1: Check Permission - Resend: Negative Result

All rights reserved.

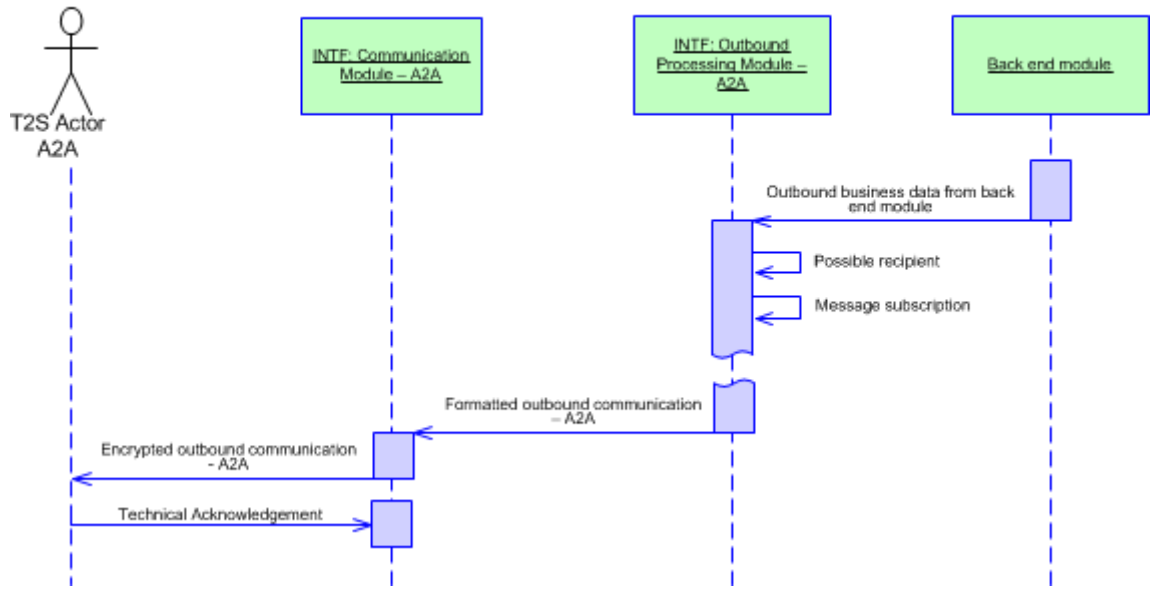
When the communication (i.e. messages or files), which should be resent, is not found or the permission check failed, the *Inbound Processing Module – A2A* sends an *Inbound processing rejection - permission* to the *Outbound Processing Module – A2A*. After formatting and encryption, the *Communication Module – A2A* delivers the outbound communication to the T2S System User.

Alternative 1: Check Permission - Resend: Positive Result

In case of a positive result the *Inbound Processing Module – A2A* sends the *Validation result - resend* as well as the *Resend communication* (i.e. messages or files) to *Outbound Processing Module – A2A.* After formatting and encryption, the *Communication Module – A2A* delivers the outbound communication for *Validation result - resend* to the T2S System User and the outbound communication for *Resend communication* (i.e. messages or files) to the T2S Actor A2A.

3.2.8.3 Processing of UC-INTF-3: Interface via A2A – push

DIAGRAM 12 - INTERFACE VIA A2A - PUSH



Business assumption

A T2S Actor A2A is a potential recipient of a push message.

Processing

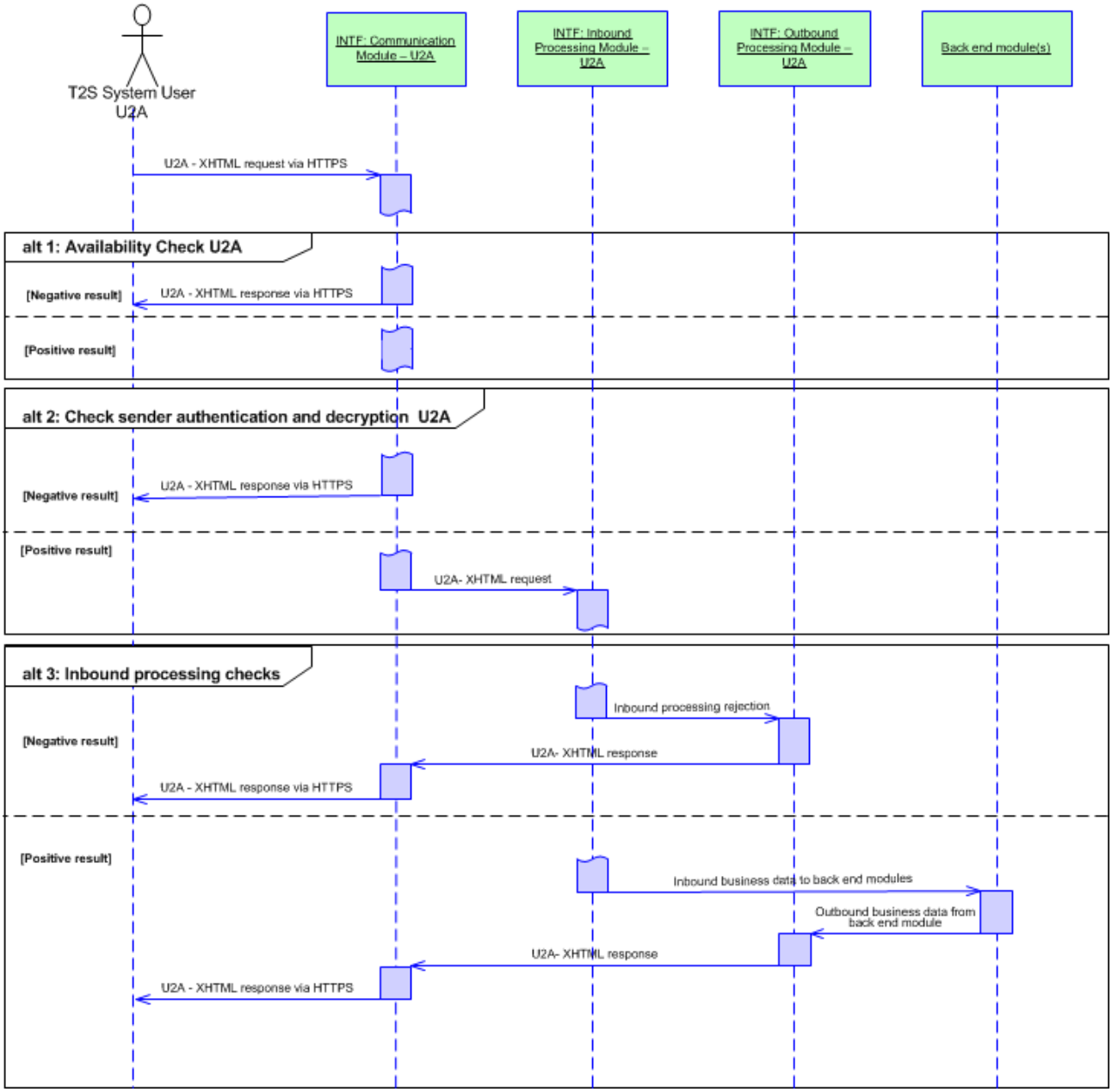
The back end module sends the *Outbound business data* flow to the *Outbound Processing Module – A2A*. The *Outbound Processing Module – A2A* creates the payload and a list of possible recipients. Afterwards it checks for all possible recipients, whether the party opted for receiving the message.

The *Communication Module – A2A* forwards the formatted and encrypted *Outbound Communication* to the T2S Actor A2A. The T2S Actor sends a Technical Acknowledgement back to the *Communication Module – A2A*

All rights reserved.

3.2.8.4 Processing of UC-INTF-4: Interface via U2A

DIAGRAM 13 - INTERFACE VIA U2A



Business assumption

A T2S System User sends a communication via U2A to the *Interface*.

Processing

The *U2A - XHTML request via HTTPS* enters the system via the *Interface* domain. In a first step *Communication Module – U2A* checks the availability of the T2S System.

Alternative 1: Availability Check U2A: Negative Result

If the system is unavailable, the T2S System User receives an *"Unavailable"* response.

All rights reserved.

Alternative 1: Availability Check U2A: Positive Result

If the system is available the request is subject to the authentication check performed by the *Communication Module – U2A*.

Alternative 2: Check sender authentication and decryption U2A: Negative Result

If one of the checks fails, the *Communication Module – U2A* creates and sends a rejection including reason information to the T2S System User.

Alternative 2: Check sender authentication and decryption U2A: Positive Result

If the communication passes the checks successfully, the *Communication Module – U2A* forwards, the decrypted *U2A- XHTML request* to the *Inbound Processing Module - U2A*.

The *Inbound Processing Module - U2A* performs a series of technical verification and authorisation checks.

Alternative 3: Inbound processing checks: Negative Result

If the *U2A- XHTML request* fails one of these checks, the *Inbound Processing Module* - U2A creates an *Inbound processing rejection*. The *Outbound Processing Module - U2A* creates the corresponding *U2A XHTML response* and the *Communication Module – U2A* sends it out to the T2S System User.

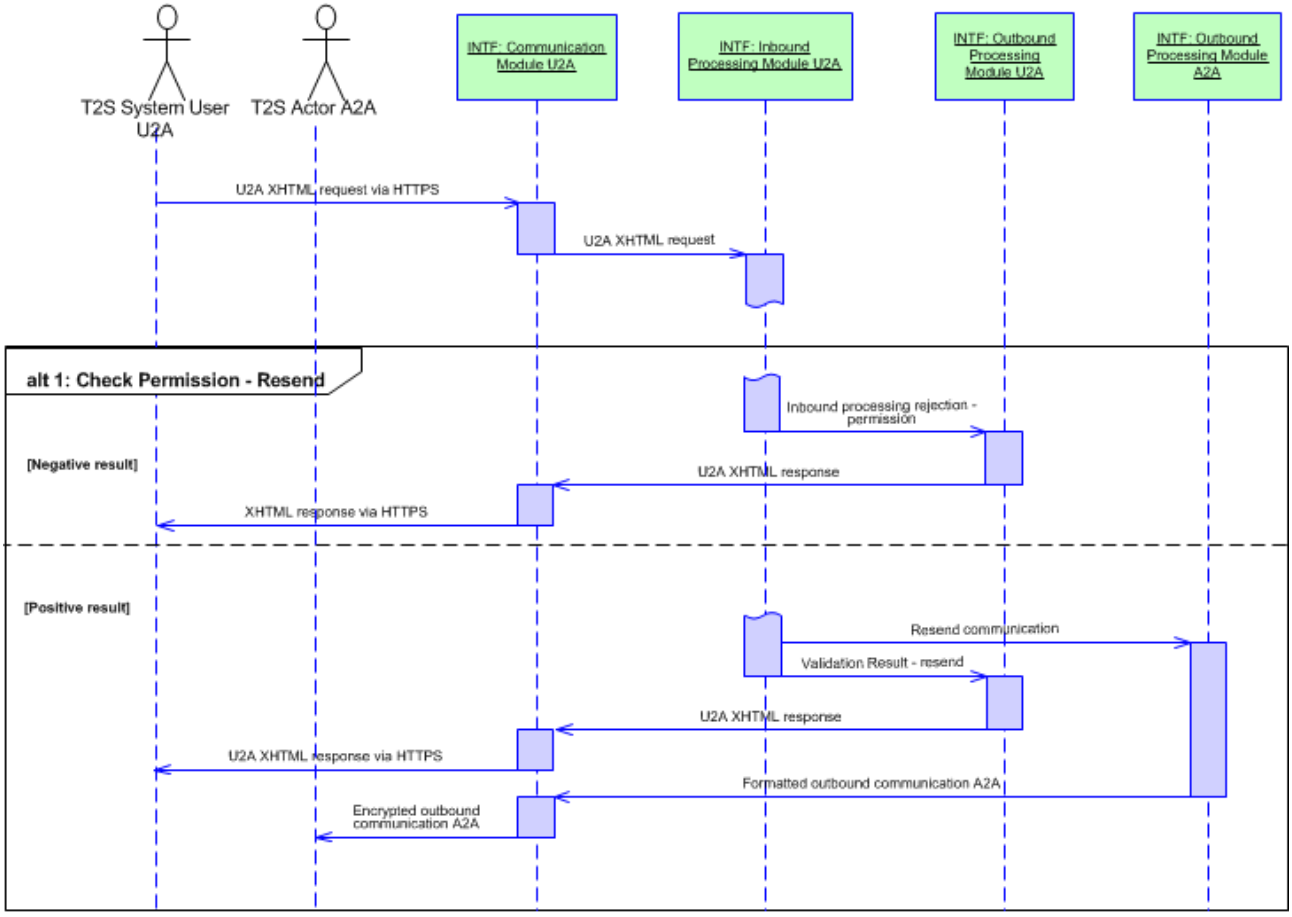
Alternative 3: Inbound processing checks: Positive Result

If the *U2A XHTML request* passes all checks successfully the *Inbound Processing Module - U2A* forwards it as *Inbound business data* to the relevant back end module. After processing the request the back end module(s) send(s) the *Outbound business data* to the *Outbound Processing Module - U2A*. The *Outbound Processing Module - U2A* creates the corresponding *U2A XHTML response* and the *Communication Module – U2A* sends it out to the T2S System User.

All rights reserved.

3.2.8.5 Processing of UC-INTF-5: Interface via U2A – Resend

DIAGRAM 14 - INTERFACE VIA U2A - RESEND



Business assumption

A T2S System User sends a resend communication (i.e. messages or files) request via U2A to the *Interface*.

Processing

The *U2A XHTML request via HTTPS* enters the system via the *Interface* domain. If the checks performed in the *Communication Module - U2A* as well as *Inbound Processing Module - U2A* succeed, *U2A XHTML request* is subject to a business validation and permission check based on the privileges of the T2S System User.

Alternative 1: Check Permission - Resend: Negative Result

When the communication (i.e. messages or files), which should be resent, is not found or the permission failed, an *Inbound processing rejection - permission* is sent to the *Outbound Processing Module - U2A*. After formatting it, the *Communication Module* - U2A delivers a *U2A XHTML response* to the T2S System User via U2A.

Alternative 1: Check Permission - Resend: Positive Result

The *Inbound Processing Module – U2A* sends the *Validation result - resend* to the *Outbound Processing Module - U2A* which creates a *U2A XHTML response* before the *Communication Module* - U2A sends it to the T2S System User.

All rights reserved.

At the same time the *Inbound Processing Module* - U2A forwards the *Resend communication* (i.e. messages or files) to the *Outbound Processing Module* - A2A. After formatting and encryption the *Communication Module – A2A* delivers the outbound communication A2A to the T2S Actor A2A.

3.3 Reference Data Management

3.3.1 General Introduction

Reference Data Management for T2S is carried out through the Common Reference Data Management component (CRDM). CRDM includes the entire set of functions for managing reference data for parties (e.g. CSDs, CBs, CSD participants, payment banks), securities, securities accounts, T2S dedicated cash accounts, configuration rules and system parameters. CRDM allows authorised T2S system users to maintain and access Reference data objects.

A Reference data object is a set of logically related, self-consistent information. Parties, securities, securities accounts and message subscription rules are examples of Reference data objects. A Reference data object may consist of one or more classes of information with a defined set of attributes. For example, a party is a Reference data object, consisting of classes of information for the party, party code, party name, party address and party technical address. The party address has attributes such as the street, city or town, country and postal code. When a user creates a party, then the user must provide all mandatory attributes of the mandatory classes for the party Reference data object.

Reference *Data Management* processes:

- Reference *Data Access Requests*, which retrieve Reference data objects;
- Reference *Data Maintenance Requests*, which update the information of Reference data objects.

Reference *Data Management* supports the update of Reference data objects according to the:

- Two-Eyes principle, where Reference *Data Management* applies an update of a Reference data object in one step;
- or the Four-Eyes principle, where a second independent user must approve or revoke the processed *Maintenance Request* through a subsequent *Maintenance Approval Request*.

The Reference *Data Management* domain consists of five modules:

- *Party Data Management*;
- *Securities Data Management*;
- *Securities Account Data Management*;
- *T2S Dedicated Cash Account Data Management*;
- *Rules and Parameters Data Management*.

The Reference *Data Management* domain provides the functions to maintain all the objects included in these modules. The following table shows the complete list of these Reference data objects and their mapping versus the responsible Reference Data module [22].

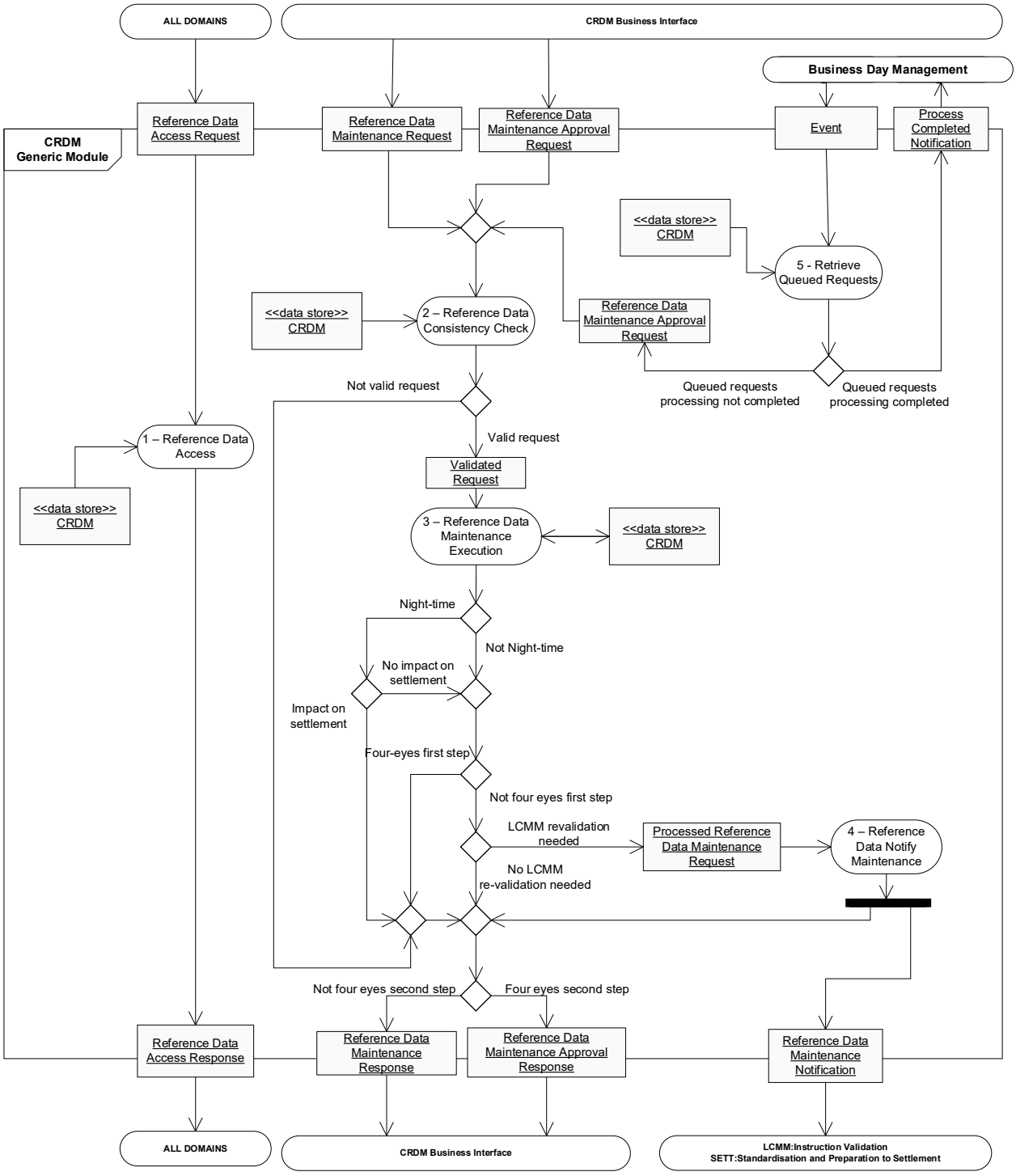
22 Please refer to the section of the relevant module for a detailed description of each Reference Data object.

All rights reserved.

REFERENCE DATA MODULE	REFERENCE DATA OBJECT
Party Data Management	Party, Eligible Counterpart CSD Link, Technical Address Network Service Link.
Securities Data Management	Securities, Securities Valuation, Close Link, Security CSD Link, Security Auto-collateralisation Eligibility.
Securities Account Data Management	Securities Account, CSD Account Link, CMB Securities Account Link.
T2S Dedicated Cash Account Data Management	T2S Dedicated Cash Account, Credit Memorandum Balance, Liquidity Transfer Order, Liquidity Transfer Order Link Set, External RTGS Account.
Rules and Parameters Data Management	User, Role, Privilege, Secured Object, Secured Group, Message Subscription Rule, Message Subscription Rule Set, Routing, Network Service, Report Type, Report Configuration, Attribute Domain, Attribute Reference, Event Type, Operating Day Type, Closing Day, Market-Specific Attribute, Restriction Type, Restriction Type Rule, Conditional Securities Delivery Rule, Conditional Securities Delivery Rule Set, Service Item, Country, Currency, Limit, Partial Settlement Threshold, System Entity, Tolerance Amount, T2S BIC Directory, Certificate Distinguished Name, User-Certificate Distinguished Name Link.

All rights reserved.

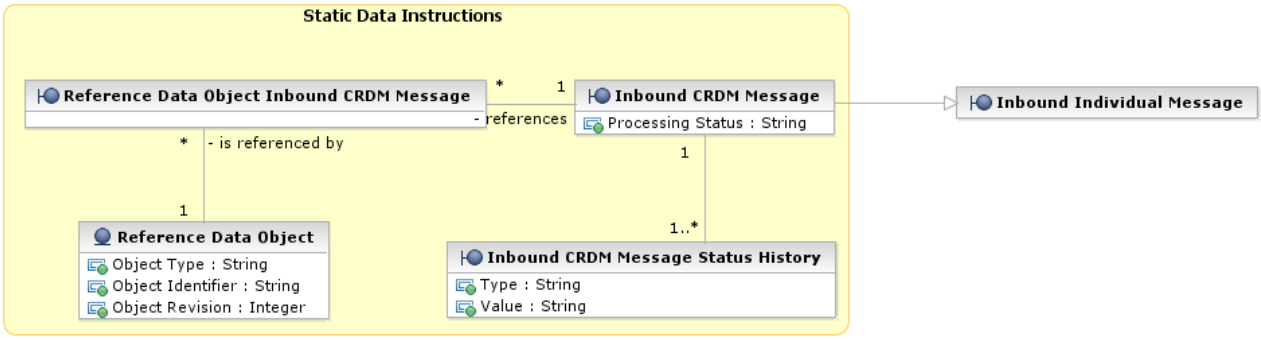
The following generic activity diagram describes at module level the common set of functions for the different types of Reference data objects managed by CRDM .



All rights reserved.

3.3.2 Dynamic data managed by the domain

The following diagram depicts the conceptual dynamic data model for reference data maintenance instructions.



1 – Inbound CRDM Message

This entity stores information concerning all Reference data maintenance instructions, i.e. Reference *Data Maintenance Requests* and Reference *Data Maintenance Approval Requests,* received by the business interface of the CRDM component **{T2S.16.167}**. It is a specialisation of the *Inbound Individual Message* entity (refer to the dynamic data description of the *Interface* domain for more information), with an additional attribute for storing the current processing status of the request.

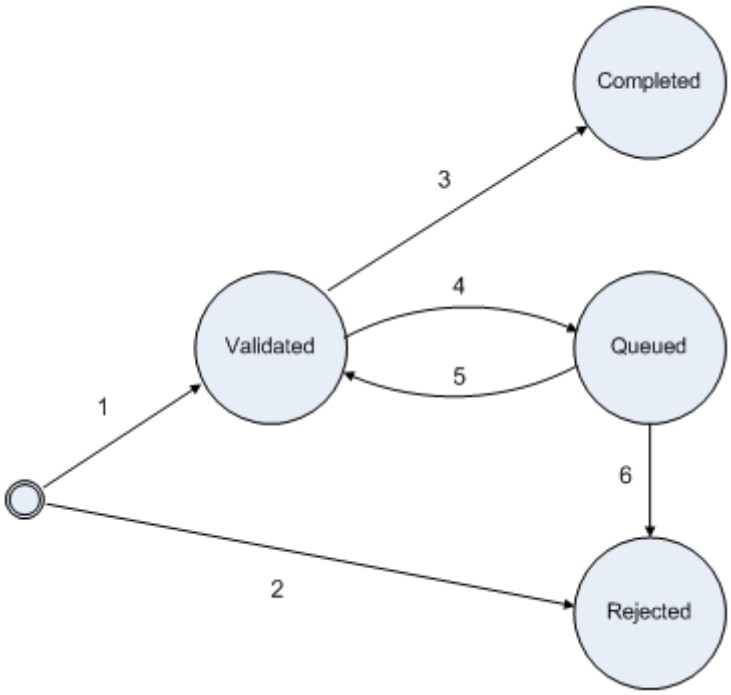
ATTRIBUTE	DESCRIPTION
Processing Status	It specifies the current processing status of the request. The exhaustive list of possible values is as follows: ▪ Validated ▪ Queued ▪ Rejected ▪ Completed

Each *Inbound* CRDM *Message* is linked to one or many occurrences of *Inbound* CRDM *Message Status History*, storing the full history of its status values. In addition, a one-to-zero-to-many association exists between the *Inbound* CRDM *Message* entity and the Reference *Data Object* entity, owing to the fact that:

- a Reference *Data Maintenance Request* may impact multiple Reference data objects, when the Reference *Data Maintenance Execution* function successfully processes a validated request (one-to-one-or-many association);
- or a Reference *Data Maintenance Request* may not execute when the Reference *Data Maintenance Execution* function cannot process a request as it failed validation (one-to-zero association).

The processing status transition diagram for *Inbound* CRDM *Messages* is provided below:

All rights reserved.



The following table provides a description of the possible status transitions:

TRANSITION	DESCRIPTION
1	The request is checked and no validation errors are found.
2	The request is checked and at least one validation error is found.
3	The request can be processed immediately after the validation.
4	The request can be processed but it cannot be completed immediately after the validation (e.g. a night-time sequence is running).
5	The request is checked again (e.g. after the end of a night-time sequence) and no validation errors are found.
6	The request is checked again (e.g. after the end of a night-time sequence) and at least on validation error is found.

2 – Inbound CRDM Message Status History

This entity stores the full history of the status values of all *Inbound* CRDM *Messages.* Every time the Reference *Data Management* domain creates a new *Inbound* CRDM *Message*, it also creates a new occurrence of *Inbound* CRDM *Message Status History* and it stores the value assigned to the initial processing status in both entities. Every time the Reference *Data Management* domain updates the value assigned to the processing status of an *Inbound* CRDM *Message*, it creates a new revision of the relevant occurrence of *Inbound* CRDM *Message Status History* to store the new value of the processing status, propagating it also into the Processing Status attribute of the *Inbound* CRDM *Message* entity.

ATTRIBUTE	DESCRIPTION
Type	It specifies the status type of the request. The only possible value for this attribute is "Processing Status".

All rights reserved.

ATTRIBUTE	DESCRIPTION
Value	It specifies the status value assigned to the status type. The exhaustive list of possible values for the "Processing Status" status type is as follows: ▪ Validated ▪ Queued ▪ Rejected ▪ Completed

Each occurrence of *Inbound* CRDM *Message Status History* is linked to the relevant *Inbound* CRDM *Message*.

3 – Reference Data Object

This entity is a generic representation of a Reference data object in the Reference *Data Management* domain.

ATTRIBUTE	DESCRIPTION
Object Type	It specifies the type of Reference data object that the Reference data maintenance instruction updated (e.g. party, security, T2S dedicated cash account, etc.).
Object Identifier	It specifies the technical identifier of the Reference data object (refer to section 3.3.6 [▸ 123]).
Object Revision	It specifies the revision number of the Reference data object (refer to section 3.3.6 [▸ 123]).

A Reference data object has a link to one or many *Inbound* CRDM *Messages*, i.e. to those messages, being either a Reference *Data Maintenance Request* or a Reference *Data Maintenance Approval Request* that updated the related Reference data object. This association serves as an audit trail and provides an efficient query mechanism in that it documents which Reference data maintenance instructions led to which changes in which Reference data object(s).

3.3.3 Functions

Reference Id	***SDMG.ALL.FUN.1.1***

This section provides a description of the functions available within each of the modules of the *Reference Data Management* domain **{T2S.16.160}**.

Reference Data Management domain is always available during the settlement day, except during the technical maintenance window. Users of T2S can access the functions of this domain in user-to-application and application-to-application mode interactively **{T2S.03.220}**. The non-exhaustive list of Reference data objects that can be maintained in user-to-application mode only is as follows **{T2S.16.163}**:

- Attribute domain;
- Closing day;
- Conditional securities delivery rule set;
- Message subscription rule set;

All rights reserved.

- Partial settlement threshold;
- Restriction type;
- System entity;
- Tolerance amount.

CRDM data supports the processing of Reference Data Maintenance Requests **{T2S.13.140}** using either the Two-Eyes principle or the Four-Eyes principle, whereas the latter is available in user-to-application mode only **{T2S.16.170} {T2S.16.310}**:

- Two-Eyes principle: the domain processes the *Reference Data Maintenance Request* immediately and immediately applies the relevant change to make it available for processing in T2S, without requiring an approval of the change by a second independent T2S system user;
- Four-Eyes principle: the domain processes the *Reference Data Maintenance Request* immediately and it applies the change to the Reference data object used for processing only after approval by a different T2S system user.

1 – Reference Data Access

Reference Id	***SDMG.ALL.SDA.1.1***

This function enables authorised T2S system users to access Reference data objects based on their access rights.

The *Reference Data Access* function receives as input from all other domains a *Reference Data Access Request.* In the case of queries coming from T2S users, this *Reference Data Access Request* is sent by the *Query Management* module, after it has been checked at syntax level by the CRDM Business Interface . Each access request refers to the access to one or more objects of the same type and it is processed immediately as follows by the *Reference Data Access* function: the data of the referenced objects that are accessible (according to the segregation of data based on system entities [23] and of the relevant granted object privileges) to the relevant T2S system user are retrieved and collected in an output access response. For each retrieved object, the response includes all the relevant classes of information (e.g. for the party object, classes of information for the party, party code, party name, party address and party technical address are retrieved) or part of them (e.g. only securities code for the securities object), depending on the specific request. In case the T2S system user is not allowed to access any of the referenced objects (i.e. though being authorised to use this function by the CRDM Business Interface , the T2S system user has not been previously granted the object privilege to access the referenced objects), no data are retrieved and a negative access response, containing the applicable error code, is issued.

In any case, as last step of the processing, the access response is provided in order to inform the requestor about the result of the corresponding request.

Other domains can access this function to retrieve the information they need from the Reference data repository. In this case, the *Reference Data Access Request* is received directly from the relevant domain and it does not require a specific authorisation check based on roles and object privileges. The access response is

23 See section 3.3.11.11 [▸ 180] for more information.

All rights reserved.

sent back to the same domain. Please note that, by convention, any access to Reference Data performed by other domains is represented in the respective activity diagrams with a direct data flow between the relevant function and the Reference data store, in order to make the diagrams as simple as possible. For this reason, the diagrams do not mention explicitly the data flows between other domains and the *Reference Data Access* function.

2 – Reference Data Consistency Check

Reference Id	***SDMG.ALL.SDC.1.1***

This function is the initial point of entry for the processing of any *Reference Data Maintenance Request*, regardless of whether it is a *Reference Data Maintenance Request* sent according to the Two-Eyes principle or the Four-Eyes principle. Consequently, it can receive as input both *Reference Data Maintenance Requests* and *Reference Data Maintenance Approval Requests* that the CRDM Business Interface has checked at syntax level. This function also performs a check of the relevant access rights at object level, so to enable authorised T2S system users to maintain Reference data objects according to their own specific access rights, i.e. to modify information about the objects based on the granted privileges of their role(s).

The *Reference Data Consistency Check* function performs a validation of a *Reference Data Maintenance Request* against the relevant Reference data. The validation that this function executes depends on the actual request that Reference data is to process (i.e. on the object type and the type of maintenance to be performed on it), as a different set of business validation rules applies to different objects [24]. This function also impedes any concurrent updates of Reference data objects, i.e. it does not allow a T2S system user to perform an update of a Reference data object if a former *Reference Data Maintenance Request* sent according to the Four-Eyes principle changed the same object and a second independent user has not yet approved or rejected this change **{T2S.16.310}**.

When the *Reference Data Consistency Check* function receives a positive Reference *Data Maintenance Approval Request* (i.e. the second step of an object maintenance performed according to the Four-Eyes principle), it revalidates the Reference data revision object that is pending approval to ensure it still fulfils the validation requirements as a change may have occurred between the initial update and the approval. This revalidation is not needed in the case of a negative *Reference Data Maintenance Approval Request*, i.e. when the second independent user revokes the Reference data change requested by the first user.

If validation is not successful (i.e. the request violates at least one of the concerned business rules), the maintenance process stops, the function sets the status of the request to "Rejected" and the function forwards the relevant information (i.e. status and error codes) to the CRDM Business Interface , in order to inform the requestor about the object maintenance rejection. If validation is successful, the function sets the status of the request to *"Validated"* immediately before the *Reference Data Maintenance Execution* function starts processing it.

3 – Reference Data Maintenance Execution

Introduction

24 The exhaustive list of business validation rules applicable for each object type (e.g. party, securities account, limit and so forth) and maintenance type (i.e. create, update, delete) is provided in the User Detailed Functional Specifications (for A2A functions) and in the User Handbooks (for U2A functions).

All rights reserved.

Reference Id	***SDMG.ALL.SDM.1.1***

The *Reference Data Maintenance Execution* function receives a validated request that is either a *Reference Data Maintenance Request* or a *Reference Data Maintenance Approval Request* for a previous *Reference Data Maintenance Request* **{T2S.16.180}**. Each *Reference Data Maintenance Request* or *Reference Data Maintenance Approval Request* refers to the maintenance of one or more entities of the same Reference data object type. The actual processing of a *Reference Data Maintenance Request* depends on the type of maintenance **{T2S.16.200}** and the applicable approval configuration **{T2S.16.210}**, i.e. if the function has to process the request according the Two-Eyes or the Four-Eyes principle **{T2S.16.190}** **{T2S.16.220}**.

Maintenance requests processing

Reference Id	***SDMG.ALL.SDM.2.1***

Depending on the specific request (i.e. on the type of Reference data object involved) **{T2S.16.190}**, and on the phase of the settlement day, each *Reference Data Maintenance Request* submitted under the Two-Eyes principle is processed as follows.

During the daytime processing, this function processes the request and applies the requested update immediately. If the request updates a Reference data object or an attribute of a Reference data that *LCMM* uses in its validation process, the function forwards a *Reference Data Maintenance Notification* to the *LCMM* domain to enable *LCMM* to revalidate the *LCMM instructions*, affected by the Reference data update (see below the description of the *Notify Maintenance* function for more information). The function notifies the following exhaustive list of Reference data changes to the *Instruction Validation* module of the *LCMM* domain:

- Security Reference Data:
  - update of issue date, final maturity date, minimum settlement unit, settlement unit multiple,
  - creation or logical deletion of deviating settlement unit,
  - deletion of a not yet issued security or an already matured/expired security,
- Securities Account Reference Data:
  - update of opening date or closing date,
- T2S Dedicated Cash Account Reference Data:
  - update of opening date or closing date,
- CMB Securities Account Links:
  - update of validity period.
- Updates which can be one done by the T2S Operator which trigger also immediate SI revalidation:
  - attribute domain: update of the predefined time frame,
  - creation of a closing day.

The function uses the same mechanism also for an update of a limit. In this case, the function forwards a *Reference Data Maintenance Notification* to the *Settlement* domain to enable it to check if it must trigger a forced auto-collateralisation reimbursement.

All rights reserved.

During the night-time processing, the function processes and applies each request immediately. If the requested change may affect the ongoing settlement process, the function creates a revision occurrence for the Reference data object change and sets the approval status of this revision to *"Queued"*. As the function does not apply the change to the current instance of the object, the update does not affect the settlement transactions in the ongoing settlement process. The *Reference Data* Domain revalidates and applies all objects with an approval status "*Queued*" (i.e. their approval status is set to *"Approved"*) at the end of the current settlement sequence and before the beginning of the next settlement sequence (or cycle). If the *Reference Data* Domain receives a request outside of a settlement sequence, i.e.:

- between two sequences within the same night-time settlement cycle;
- between the end of the last sequence of a night-time settlement cycle and the beginning of the first sequence of the following settlement cycle;
- after the end of the last night-time settlement cycle and the beginning of the day-time processing;

then the function processes and applies the requested change immediately.

As a general principle and regardless of the settlement day phase, the *Reference Data Maintenance Execution* processes do not synchronise with any of the concurrent lifecycle management and settlement processes in order to avoid any performance issues for the processing of settlement instructions. This lack of synchronisation may cause, in some specific cases, technical or business data inconsistencies (e.g. deletion of a *Securities Account* linked to pending transactions or open positions). Such inconsistencies shall be avoided a priori by defining specific business constraints. Typically, these business constraints restrict the possibility to perform some specific data changes intra-day, making them possible only as of a future date.

The following list provides examples of these Reference data changes [25]:

- update of a party – securities account relationship;
- update of a CSD link;
- update of a tolerance amount;
- update of a CoSD rule;
- update of configuration data for auto-collateralisation;
- update of an eligible counterpart CSD;
- update of an attribute domain;
- update of the calendar.

By definition, these Reference data changes may have relevance for the validation of *LCMM instructions* with an intended settlement date in the future [26]. Therefore, such updates do not result in inconsistencies for the ongoing lifecycle management and settlement processes as they do not access changed data as of a future date.

25 The exhaustive list of reference data changes not allowed intra-day is identified during the detailed specification phase.

26 The revalidation of instructions due to these reference data updates is not done immediately, but it is triggered in certain periods of time during the business day.

All rights reserved.

The function processes a *Reference Data Maintenance Request* using the Four-Eyes principle immediately, regardless of the specific request and the type of Reference data object involved, but it does not apply the change to the active instance of the Reference data object until receiving and successfully processing a Reference *Data Maintenance Approval Request*.

The following tables summarise to the previous descriptions regarding the processing of *Reference Data Maintenance Requests* during the daytime and night-time processing, respectively:

TABLE 1 - DAYTIME PROCESSING

ACTION	PRINCIPLE	RESULT
CREATE	**TWO-EYES**	A new object is created and made immediately available for processing **{T2S.16.230}**. The processing status of the request is set to *"Completed"*. The creation of a new object entails the creation of all the classes of information related to the object (e.g. the creation of a party implies the creation of a party as a class of information, a party code, a party name, a party address, only for specific party types, and one or many party technical addresses).
	FOUR-EYES	A new object is created, but it is not made available until a positive maintenance approval request for this object maintenance is received and executed. The processing status of the request is set to *"Completed"*.
UPDATE	**TWO-EYES**	All the requested changes are applied to the relevant object and its new version is made immediately available **{T2S.16.240}**. The processing status of the request is set to *"Completed"*. The update of an object entails not only the possible update of its classes of information, but also the possibility to create or delete classes of information (e.g. the update of a party may imply the creation of a new a party name and the deletion of a party technical address).

All rights reserved.

ACTION	PRINCIPLE	RESULT
		Each change is applied either as a revision or as an occurrence in a data history, depending on the content of the request [27]. In the case of a revision, the old version of the object is made no longer available for processing. In an entity that supports a data history, both the new and old versions are available for processing, but with different, non-overlapping validity periods.
	FOUR-EYES	All the requested changes are applied to the relevant object as revisions or one or more occurrences in a data history, but its new version is not made available until a positive maintenance approval request for this object maintenance is received and executed. The processing status of the request is set to *"Completed"*.
DELETE	**TWO-EYES**	The relevant object is deleted and it is no longer available for processing. The processing status of the request is set to "*Completed*". The deletion of an object entails the deletion of all the classes of information related to the object. Deletion is logical and not physical in order to keep track of entities that are no longer active or valid **{T2S.16.270} {T2S.16.300}**. This also makes the reactivation of a logically deleted object possible (e.g. in case of a wrong deletion) and ensure the integrity of the data **{T2S.16.290}**.

27 Some data do not require a data history. For such data only revisions are allowed. Otherwise, either revisions or data histories are possible.

All rights reserved.

ACTION	PRINCIPLE	RESULT
		Physical deletion of data revisions and data history is applied **{T2S.16.150}** in the context of archiving procedures in such a way as to ensure the referential integrity of static and dynamic data and only after archiving processes have removed and archived the related dynamic data.
	FOUR-EYES	The relevant object is provisionally deleted, but it is still available for processing until a positive maintenance approval request for this object maintenance is received and executed. The processing status of the request is set to "*Completed*".

TABLE 2 - NIGHT-TIME PROCESSING

ACTION	IMPACT ON SETTLEMENT PROCESS	RESULT
CREATE	**YES**	A new object is created, but it is not made available when it may affect the ongoing settlement process. The processing status of the request is set to "Queued".
	NO	A new object is created and it is made immediately available for processing **{T2S.16.230}**. The processing status of the request is set to *"Completed"*.
UPDATE	**YES**	All the requested changes are applied to the relevant object as revisions or one or more occurrences of a data history, but its new version is not made available when it may affect the ongoing settlement process. The processing status of the request is set to "Queued".

All rights reserved.

ACTION	IMPACT ON SETTLEMENT PROCESS	RESULT
	NO	All the requested changes are applied to the relevant object and its new version is made immediately available **{T2S.16.240}**. The processing status of the request is set to *"Completed"*. Each change is applied either as a revision or one or more occurrences of a data history, depending on the content of the request [28]. In the case of a revision, the old version of the object is made no longer available for processing. In the case of a data history, both the new and old versions are available for processing, but with different, non-overlapping validity periods.
DELETE	**YES**	The relevant object is provisionally deleted, but it is still available for processing when it may affect the ongoing settlement process. The processing status of the request is set to *"Queued"*.
	NO	The relevant object is deleted and, with immediate effect, is no longer available for processing. The processing status of the request is set to *"Completed"*. Deletion is logical and not physical in order to keep track of entities that are no longer active or valid **{T2S.16.270}** **{T2S.16.300}**. This also makes the reactivation of a logically deleted object possible (e.g. in case of a wrong deletion) and ensures the referential integrity of data **{T2S.16.290}**.

28 Some data do not require a data history. For such data only revisions are allowed. Otherwise, either revisions or data histories are possible.

All rights reserved.

ACTION	IMPACT ON SETTLEMENT PROCESS	RESULT
		Physical deletion of data revision and data history is applied **{T2S.16.150}** in the context of archiving procedures in such a way as to ensure the referential integrity of static and dynamic data and only after archiving processes have removed and archived the related dynamic data.

The following tables show the possible status transitions related to the technical statuses, i.e. Deletion Status and Approval Status (see section 3.3.6 [▸ 123]) of any Reference data object during the processing of a *Reference Data Maintenance Request* according to the Two-Eyes principle, or the first step of a *Reference Data Maintenance Request* processed following the Four-Eyes principle, during the daytime and night-time processing, respectively **{T2S.16.230}**.

TABLE 3 - DAYTIME PROCESSING

ACTION	PRINCIPLE	OLD REVISION		NEW REVISION	
		DELETION STATUS	APPROVAL STATUS	DELETION STATUS	APPROVAL STATUS
CREATE	**TWO-EYES**	N/A	N/A	Active	Approved
	FOUR-EYES	N/A	N/A	Active	Awaiting Approval
UPDATE	**TWO-EYES**	Active	Approved	Active	Approved
	FOUR-EYES	Active	Approved	Active	Awaiting Approval
DELETE [29]	**TWO-EYES**	Active	Approved	Deleted	Approved
	FOUR-EYES	Active	Approved	Deleted	Awaiting Approval

TABLE 4 - NIGHT-TIME PROCESSING

ACTION	IMPACT ON SETTLEMENT PROCESS	OLD REVISION		NEW REVISION	
		DELETION STATUS	APPROVAL STATUS	DELETION STATUS	APPROVAL STATUS
CREATE	**YES**	N/A	N/A	Active	Queued
	NO	N/A	N/A	Active	Approved
UPDATE	**YES**	Active	Approved	Active	Queued
	NO	Active	Approved	Active	Approved

29 In case of reactivation of a logically deleted object, the values of deletion status for the old and the new revision are switched (i.e. the deletion status is "Deleted" for the old revision and "Active" for the new one).

All rights reserved.

ACTION	IMPACT ON SETTLEMENT PROCESS	OLD REVISION		NEW REVISION	
		DELETION STATUS	APPROVAL STATUS	DELETION STATUS	APPROVAL STATUS
DELETE [30]	**YES**	Active	Approved	Deleted	Queued
	NO	Active	Approved	Deleted	Approved

Maintenance approval requests processing

Reference Id	***SDMG.ALL.SDM.3.1***

If the approval is positive, the function applies the change, waiting for approval, thereby creating a new version of the object and setting its approval status to "*Approved*", making it available for processing **{T2S.05.280} {T2S.16.610} {T2S.16.620} {T2S.13.150} {T2S.16.250} {T2S.16.260}**.

The processing status of the *Reference Data Maintenance Approval Request* is set to "*Completed*". The following table summarises the possible results concerning the processing of a positive *Reference Data Maintenance Approval Request* **{T2S.17.100}**:

ACTION	RESULT
CREATE	The object created during the processing of the relevant maintenance request, is made available for processing. The processing status of the maintenance approval request is set to *"Completed"*.
UPDATE	The new version of the object, built during the processing of the relevant maintenance request, is made available for processing. The processing status of the maintenance approval request is set to *"Completed"*. The old version of the object is made no longer available for processing in the case of a revision. If the approval is for an entity with a data history, it is made available for processing with different, non-overlapping validity period with the new version.
DELETE	The object deleted during the processing of the relevant maintenance request is made no longer available for processing. The processing status of the maintenance approval request is set to *"Completed"*.

Conversely, if a change waiting for approval is revoked, the approval status of the relevant Reference data object is set to "*Revoked*", while the processing status of the *Reference Data Maintenance Approval Request* is set to "*Completed*".

As last step of the processing, in all cases, the function forwards the relevant information (i.e. a Reference data maintenance response or a Reference data maintenance approval response) to the CRDM Business Interface , in order to inform the requestor about the result of the corresponding request.

The following table shows the possible status transitions related to the technical statuses, i.e. Deletion Status and Approval Status (see section 3.3.6 [▸ 123]) of any Reference data object during the processing of a *Reference Data Maintenance Approval Request*.

30 In case of reactivation of a logically deleted object, the values of deletion status for the old and the new revision are switched (i.e. the deletion status is "Deleted" for the old revision and "Active" for the new one).

All rights reserved.

ACTION	DECISION	OLD REVISION		NEW REVISION	
		DELETION STATUS	APPROVAL STATUS	DELETION STATUS	APPROVAL STATUS
CREATE	**APPROVAL**	Active	Awaiting Approval	Active	Approved
	REVOCATION	Active	Awaiting Approval	Active	Revoked
UPDATE	**APPROVAL**	Active	Awaiting Approval	Active	Approved
	REVOCATION	Active	Awaiting Approval	Active	Revoked
DELETE [31]	**APPROVAL**	Deleted	Awaiting Approval	Deleted	Approved
	REVOCATION	Deleted	Awaiting Approval	Deleted	Revoked

4 – Reference Data Notify Maintenance

Reference Id	***SDMG.ALL.SDN.3.1***

This function receives a *Processed Reference Data Maintenance Request* and notifies the *Instruction Validation* module of the *LCMM* domain about any changes made active on a Reference data object as a result of the execution of the underlying *Reference Data Maintenance Request* or Reference *Data Maintenance Approval Request*, when such changes may affect pending *LCMM instructions* and their settlement, i.e. they may make one or more *LCMM instructions* no longer valid. The *Reference Data Notify Maintenance* function forwards the notification to the *Instruction Validation* module of the *LCMM* domain to allow it to identify all relevant pending *LCMM instructions* possibly affected by the Reference data update **{T2S.08.800}** (refer to the *LCMM* chapter for more information on the revalidation process).

This function also notifies the *Standardisation and Preparation to Settlement* module of the *Settlement* domain about any changes made active on limits, as they may trigger a forced auto-collateralisation reimbursement or the recycling of pending instructions.

5 – Retrieve Queued Requests

Reference Id	***SDMG.ALL.RQR.1.1***

An event received from the Business Day Management at the end of each night-time settlement sequence triggers this function. When activated, this function retrieves all the pending *Reference Data Maintenance Requests* (i.e. all the requests received while the last night-time settlement sequence was running and related to a change that may affect the ongoing settlement process. For each *Reference Data Maintenance Request*, it identifies the relevant Reference data object and forwards for that object a *Reference Data Maintenance Approval Request* to the *Reference Data Consistency Check* function to complete its processing by changing its approval status from "*Queued*" to "*Approved*" (making it available for processing) in the case of successful validation, or to "*Rejected*" if the validation fails.

After having completed the processing of all the pending requests, the *Retrieve Queued Requests* function sends to the Business Day Management a *Process Completed Notification*.

31 In case of reactivation of a logically deleted object, the values of deletion status for the old and the new revision is "Active" instead of "Deleted".

All rights reserved.

3.3.4 Input / Output

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Reference Data Access Request	In	Request of access specific information of Reference Data	All domains	
Reference Data Access Response	Out	Information retrieved from Reference Data as specified with Access Request		All domains
Reference Data Maintenance Request	In	Request of insert/ modify/delete information present in Reference Data	CRDM Business Interface	
Reference Data Maintenance Approval Request	In	Request of confirmation/rejection of a pending Reference Data maintenance request	CRDM Business Interface	
Reference Data Maintenance Response	Out	Response with result of Reference Data Maintenance Request		CRDM Business Interface
Reference Data Maintenance Approval Response	Out	Response with result of Reference Data Maintenance Approval Request		CRDM Business Interface
Event	In	Event received by the end of each night-time settlement sequence.	Business Day Management	
Process Completed Notification	Out	Notification of the end of processing of all the pending Reference data maintenance requests.		Business Day Management
Reference Data Maintenance Notification	Out	Notification of maintenance with possible impacts on pending settlement instructions.		LCMM: Instruction Validation SETT: Standardisation and Preparation to Settlement

All rights reserved.

3.3.5 Data Accessed

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
REFERENCE DATA			
All	All	Read/Write	None
DYNAMIC DATA			
Dynamic Data of the *CRDM Data* domain.	Inbound CRDM Message, Reference Data Object	Read/Write	None

3.3.6 Common Information

Reference Id	***SDMG.ALL.CIN.1.1***

All Reference data items have the following set of attributes in common for audit trail and Reference data change management purposes **{T2S.16.030} {T2S.16.040} {T2S.16.050} {T2S.16.060} {T2S.16.070} {T2S.16.080} {T2S.16.090} {T2S.16.100} {T2S.16.110} {T2S.16.120} {T2S.16.260}**:

ATTRIBUTE	DESCRIPTION
Technical Identifier	This attribute is the automatically assigned primary identifier for a new item of static or transaction data **{T2S.16.010}**. The technical identifier in combination with a sequential revision number **{T2S.16.130}** is used to ensure uniqueness within multiple occurrences of a single Reference data item, which has undergone multiple updates.
Revision Number	Within the technical identifier, this attribute marks every update of the item's attributes so as to ensure the uniqueness of a given item which has undergone several revisions **{T2S.16.020}**.
Deletion Status	It defines whether the Reference data may be available for processing in T2S. The exhaustive list of possible values is as follows: ▮ Active ▮ Deleted The Reference data item is available for processing only if its deletion status is "Active" and it approval status (see below) is "Approved".
Approval Status	The attribute defines whether the Reference data object is approved or revoked by an authorised T2S system user, is awaiting approval by the T2S system user, or was rejected owing to business validation errors. The exhaustive list of possible values is as follows: ▮ Approved ▮ Awaiting Approval

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▮ Queued ▮ Rejected ▮ Revoked. In case of updates of a Reference data item submitted according to the Four-Eyes principle, the modified version of the data is created with status "Awaiting Approval" and it becomes either "Approved" or "Revoked" only after the decision of the second, independent, authorised T2S system user **{T2S.16.240}**.

Furthermore, a System Entity Identifier attribute links each new static or dynamic data item to a CSD, a CB or to the T2S Operator for data segregation purposes **{T2S.11.110}**.

Finally, some Reference data items may have one or two additional attributes specifying a validity period **{T2S.16.150}**:

ATTRIBUTE	DESCRIPTION
Valid From	It specifies the date from which the Reference data item is valid.
Valid To	It specifies the date until when the Reference data item is valid.

These two attributes are indicated explicitly for the relevant entities in the data model descriptions.

To ensure the audit trail documenting events and status changes*, Reference Data Management* keeps the date and time of every change and the unique identifier of the T2S system user requesting the change **{T2S.16.140}**.

ATTRIBUTE	DESCRIPTION
Timestamp	Timestamp of the change.

The audit trail record has an association with the T2S system user (or the application) responsible for the change and to the before and after (static or dynamic) images of the records, resulting from the change**.**

Some examples below illustrate the concepts of revision and history in combination with the status transitions related to the attribute Deletion Status and Approval Status of Reference Data objects.

Example 1: T2S allows the maintenance of a Reference data object (not requiring a data history), i.e. some of its attributes are updated according to the Four-Eyes principle. In this scenario, the latest revision of the object with Deletion Status = "Active" and Approval Status = "Approved" is used as a baseline for the maintenance request processing.

TABLE 5 - BEFORE THE PROCESSING

TECHNICAL IDENTIFIER	REVISION	ATTRIBUTES	DELETION STATUS	APPROVAL STATUS
20101968	5	ABCD	Active	Approved

All rights reserved.

When processed according to the Four-Eyes principle, the processing immediately creates a new revision of the object with an Approval Status set to "Awaiting Approval". The status allows authorised users (e.g. the ones authorised either to approve or revoke it), to access the object for approval or revocation, but excludes the object for any other types of processing in T2S. After the processing (and until the approval of the new revision by a second authorised user), the old revision of the object is still available for processing in T2S.

TABLE 6 - AFTER THE FIRST STEP OF THE PROCESSING

TECHNICAL IDENTIFIER	REVISION	ATTRIBUTES	DELETION STATUS	APPROVAL STATUS
20101968	5	ABCD	Active	Approved
20101968	6	XYZ	Active	Awaiting Approval

When the second user approves the maintenance, a new revision of the object is created in order to update its Approval Status and set it to "Approved". This makes the new version of the object (i.e. with the new values for the updated attributes) available for processing in T2S.

TABLE 7 - AFTER THE PROCESSING

TECHNICAL IDENTIFIER	REVISION	ATTRIBUTES	DELETION STATUS	APPROVAL STATUS
20101968	5	ABCD	Active	Approved
20101968	6	XYZ	Active	Awaiting Approval
20101968	7	XYZ	Active	Approved

Example 2: A T2S system user maintains an item of a Reference data object subject to a data history and based on the Two-Eyes principle) to create a new version of that item valid as of a future date

TABLE 8 - BEFORE THE PROCESSING

TECHNICAL IDENTIFIER [32]	REVISION	VALID FROM	ATTRIBUTES	DELETION STATUS	APPROVAL STATUS	REF. TECH. IDENTIFIER [33]
20101968	3	2014-01-01	ABC	Active	Approved	19581027

In this scenario, a new version of the item is created with the specified validity period and it is linked to the same object. As a result, two different items are now linked to the same object, but with different validity periods.

TABLE 9 - AFTER THE PROCESSING

TECHNICAL IDENTIFIER	REVISION	VALID FROM	ATTRIBUTES	DELETION STATUS	APPROVAL STATUS	REF. TECH. IDENTIFIER
20101968	3	2014-01-01	ABC	Active	Approved	19581027
13021972	0	2014-03-15	XYZ	Active	Approved	19581027

32 It is the technical identifier of the reference data item subject to data history.

33 It is the technical identifier of the reference data object including the static data item subject to data history.

All rights reserved.

Example 3: For a static object with a data history, a T2S system user maintains an existing item of a Reference data object for an existing validity date and based on the Two-Eyes principle.

TABLE 10 - BEFORE THE PROCESSING

TECHNICAL IDENTIFIER	REVISION	VALID FROM	ATTRIBUTES	DELETION STATUS	APPROVAL STATUS	REF. TECH. IDENTIFIER
20101968	3	2014-01-01	ABC	Active	Approved	19581027

In this scenario, a new revision of the item is created with the new attributes and the same validity period and it is linked to the same object. As before the processing, one single item is linked to the relevant object, but with different values of the attributes when compared to the previous revision.

TABLE 11 - AFTER THE PROCESSING

TECHNICAL IDENTIFIER	REVISION	VALID FROM	ATTRIBUTES	DELETION STATUS	APPROVAL STATUS	REF. TECH. IDENTIFIER
20101968	3	2014-01-01	ABC	Active	Approved	19581027
20101968	4	2014-01-01	DEF	Active	Approved	19581027

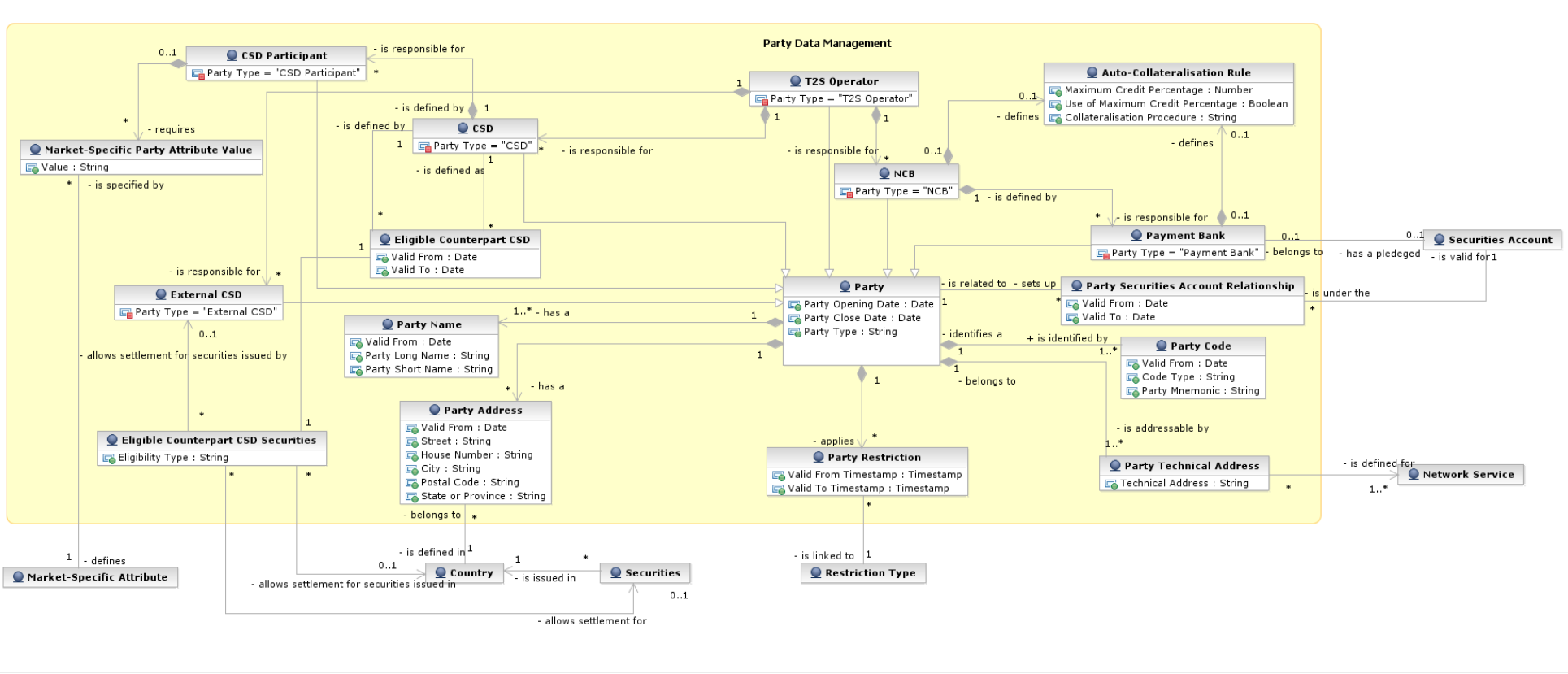
All rights reserved.

3.3.7 Party Data Management

3.3.7.1 Data model of the module

Reference Id	***SDMG.PAR.DMD.1.1***

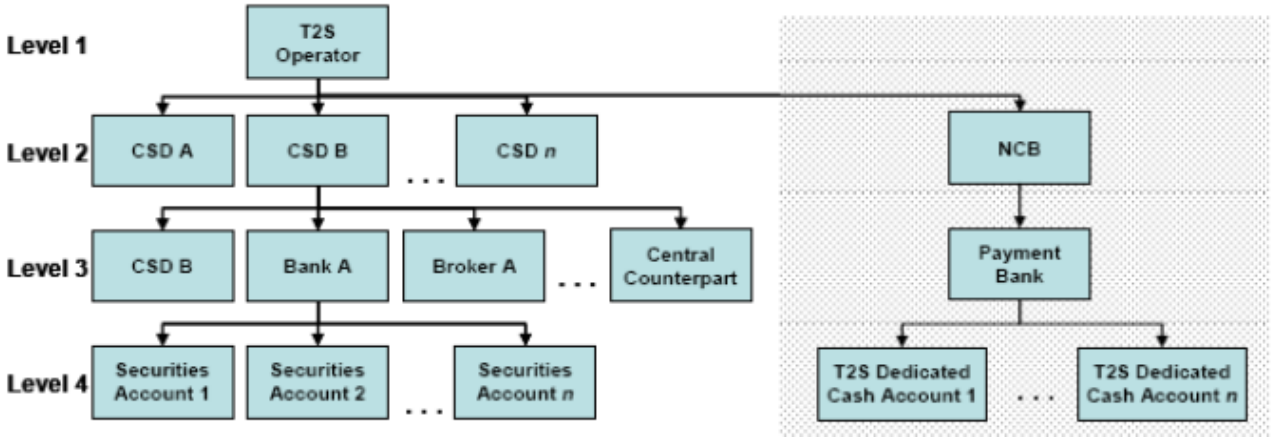
The following diagram shows the conceptual data model for Party Data Management **{T2S.16.530}**.



3.3.7.2 Description of the module

Reference Id	***SDMG.PAR.DEM.1.1***

This module allows the management of reference data related to parties, their links to the relevant restrictions and the business relationships defined in T2S between different parties, according to the hierarchical structure described in the following diagram **{T2S.16.540}**.



This hierarchical structure defines the legal relationships between the parties identified within the first three levels of the hierarchy itself. More in detail, a legal relationship exists between the T2S Operator (the only entity on the top level of the hierarchy) and each entity of the second level, i.e. each CSD and each CB in T2S. Similarly, legal relationships exist between each entity belonging to the second level of the hierarchy (i.e. a CSD or a CB) and all its participants (i.e. CSD participants for the CSDs and payment banks for the CBs). In this context, CSD participants include central counterparts, trading platforms, stock exchanges and financial institutions with a contractual relationship to a CSD. Finally, the lowest level of the hierarchy describes the links between each CSD participant and its securities account(s) and between each payment bank and its T2S dedicated cash account(s).

The *Party Data Management* module allows the managements of all the relationships between all the parties belonging to the first three levels of the hierarchy, but not the links between a party and its securities accounts or T2S dedicated cash accounts. The management of these links is performed within the *Securities Account Data Management* module and the *T2S Dedicated Cash Account* module (see sections 3.3.9 [▸ 141] and 3.3.10 [▸ 146], respectively).

The functions described below allow the authorised T2S system users to input their own parties and to access and maintain them, i.e. to create new parties or to update or delete already existing parties.

3.3.7.3 Description of the entities

1 – Party

Reference Id	***SDMG.PAR.ENT.1.1***

This entity includes all party reference data that do not require a data history, i.e. all the attributes having only one valid value for a given party, regardless the point in time taken into account **{T2S.16.550}**.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Party Opening Date	Opening date of the party.
Party Close Date	Closing date of the party.
Party Type	It specifies a classification for the party. The exhaustive list of possible values is as follows: ▮ T2S Operator ▮ Payment Bank ▮ Central Securities Depository (CSD) ▮ CSD Participant ▮ External CSD ▮ Central Bank (CB)

The party reference data that require a data history are the entities *Party Code*, *Party Name* and *Party Address*, described below. Each party is linked at least to one *Party Code*, *Party Name* and it may be linked to a *Party Address*. In addition, each party is linked to one or many *Party Technical Addresses*.

Each *Party* may be linked to one or many *Party Restrictions* [34] **{T2S.16.680}** In addition, each *Party* defined by a CSD (i.e. a CSD participant) may be linked to one or many *Market-Specific Attributes* [35] (see section 3.3.11.9 [▸ 174]). Finally, each *Party* defined as a CB and all *Parties* defined as a payment bank and offering auto-collateralisation are linked to an *Auto-collateralisation Rule* (see below).

2 – Party Code

Reference Id	***SDMG.PAR.ENT.2.1***

This entity includes the information used to identify a *Party* from a business perspective **{T2S.16.570}**. Each legal entity is identified in the financial market by its primary BIC, based on ISO 9362 standard. A legal entity may establish multiple legal relationships with several CSDs and CBs in T2S. As a consequence, a legal entity is defined multiple times in T2S, possibly multiple times for each legal relationship with a CSD or a CB. The combination of <Party BIC, System Entity Identifier [36]> ensures the uniqueness of the *Party* in T2S, i.e. any BIC is unique within a given *System Entity*.

Party codes may change in time, but only one *Party code* for a *Party* must be valid at any given point in time. For this reason, it is also necessary to specify its validity period for each *Party Code*.

ATTRIBUTE [37]	DESCRIPTION
Valid From	Starting validity date for the party code.

34 For each party restriction, a period of validity and a restriction type must be specified.

35 For each market-specific attribute, a valid value must be specified.

36 See section 3.3.11.11 [▸ 180] for more information on system entities.

37 The System Entity Identifier is not explicitly mentioned as an attribute here, because it is part of the common information for all reference data.

All rights reserved.

ATTRIBUTE [37]	DESCRIPTION
Code Type	Code type for the party. Currently, only BIC (as defined by ISO 9362 standard) is foreseen.
Party Mnemonic	Actual value for the party code, i.e. a BIC11 for the party.

Each *Party Code* is linked to its relevant *Party*.

3 – Party Name

Reference Id	***SDMG.PAR.ENT.3.1***

This entity includes Party Long Name and Party Short Name in a time line basis **{T2S.16.560}**. This is due to the fact that party names may change in time, but only one long name and one short name for a *Party* are valid at any given point in time.

ATTRIBUTE	DESCRIPTION
Valid From	Starting validity date for the party name.
Party Long Name	Full name of the party.
Party Short Name	Short name of the party.

Each *Party Name* is linked to its relevant *Party*.

4 – Party Address

Reference Id	***SDMG.PAR.ENT.4.1***

This entity includes legal addresses information for specific Party Types (i.e. *"T2S Operator"*, *"CSD"*, *"CB"* and *"Payment Banks"*) in a time line basis **{T2S.16.580}**. This is due to the fact that party legal addresses may change in time, but only one legal address for a *Party* is valid at any given point in time.

ATTRIBUTE	DESCRIPTION
Valid From	Starting validity date for the party address.
Street	Name of the street for the address.
House Number	House number for the address.
City	Name of the city for the address.
Postal Code	Postal code for the address.
State or Province	State or province for the address.

Each *Party Address* is linked to its relevant *Party* and *Country*.

5 – Party Technical Address

Reference Id	***SDMG.PAR.ENT.5.1***

This entity includes information related to all technical addresses defined for a *Party* **{T2S.16.700}**. Each Party Technical Address uniquely identifies a possible recipient technical address the *Party* can use in the message subscription service.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Technical Address	Unique technical address of a party in T2S (e.g. a BIC, a distinguished name, an IP address) [38]

Each *Party Technical Address* is linked to its relevant *Party* and to one or many *Network Services* [39].

6 –Auto-Collateralisation Rule

Reference Id	***SDMG.PAR.ENT.6.1***

This entity includes all configuration data concerning auto-collateralisation rules required in T2S for each CB and payment banks offering auto-collateralisation **{T2S.16.581}**.

ATTRIBUTE	DESCRIPTION
Collateralisation Procedure	It specifies the type of collateralisation procedure application for the CB. The exhaustive list of possible values is as follows: ▮ Pledge ▮ Pledge Sub-account ▮ Repo ▮ For payment banks this attribute always equals "Repo".
Minimum amount for auto-collateralisation	This attribute shall specify the minimum amount to be sourced in an autocollateralisation operation. This attribute will only be available for Payment Banks.
Minimum amount for client collateralisation	This attribute shall specify the minimum amount to be sourced in a client collateralisation operation. This attribute will only be available for Payment Banks.

Each *Auto-collateralisation Rule* is linked to its relevant *Party*, i.e. to the CB or payment bank to which the rule applies. Conversely, each CB and each payment bank offering auto-collateralisation is linked to an *Auto-collateralisation Rule*.

7 – Eligible Counterpart CSD

Reference Id	***SDMG.PAR.ENT.7.1***

This entity includes all configuration data allowing each CSD to define the CSDs it accepts as counterpart for settlement in T2S. The set of relevant securities is specified in the *Eligible Counterpart CSD Securities* entity (see below) **{T2S.16.910} {T2S.16.920}**.

ATTRIBUTE	DESCRIPTION
Valid From	Date from which eligible counterpart CSD is valid.

38 The actual data to be stored for the definition of a technical address is clarified during the detailed specification phase.

39 The linked *Network Service* identifies the type of *Party Technical Address*. Consequently, if a *Party Technical Address* is linked to multiple *Network Services*, then all the Technical Address Types shall be the same (see section 3.3.11.5 [▸ 164] for more information on network services).

All rights reserved.

ATTRIBUTE	DESCRIPTION
Valid To	Date until which the eligible counterpart CSD is valid.

Each occurrence of this entity is linked to the relevant *Investor CSD* and *Eligible Counterpart CSD*.

8 – Eligible Counterpart CSD Securities

Reference Id	***SDMG.PAR.ENT.8.1***

This entity includes all configuration data concerning the definition of securities, linked to an eligible counterpart CSD, for which a given investor CSD allows settlement.

ATTRIBUTE	DESCRIPTION
Eligibility Type	It specifies whether the eligibility for settlement in T2S is defined with respect to a single security or to a set of securities (either issued by the same CSD or issued in the same country). The exhaustive list of possible values is as follows: ▮ Securities ▮ Issuer CSD ▮ Country

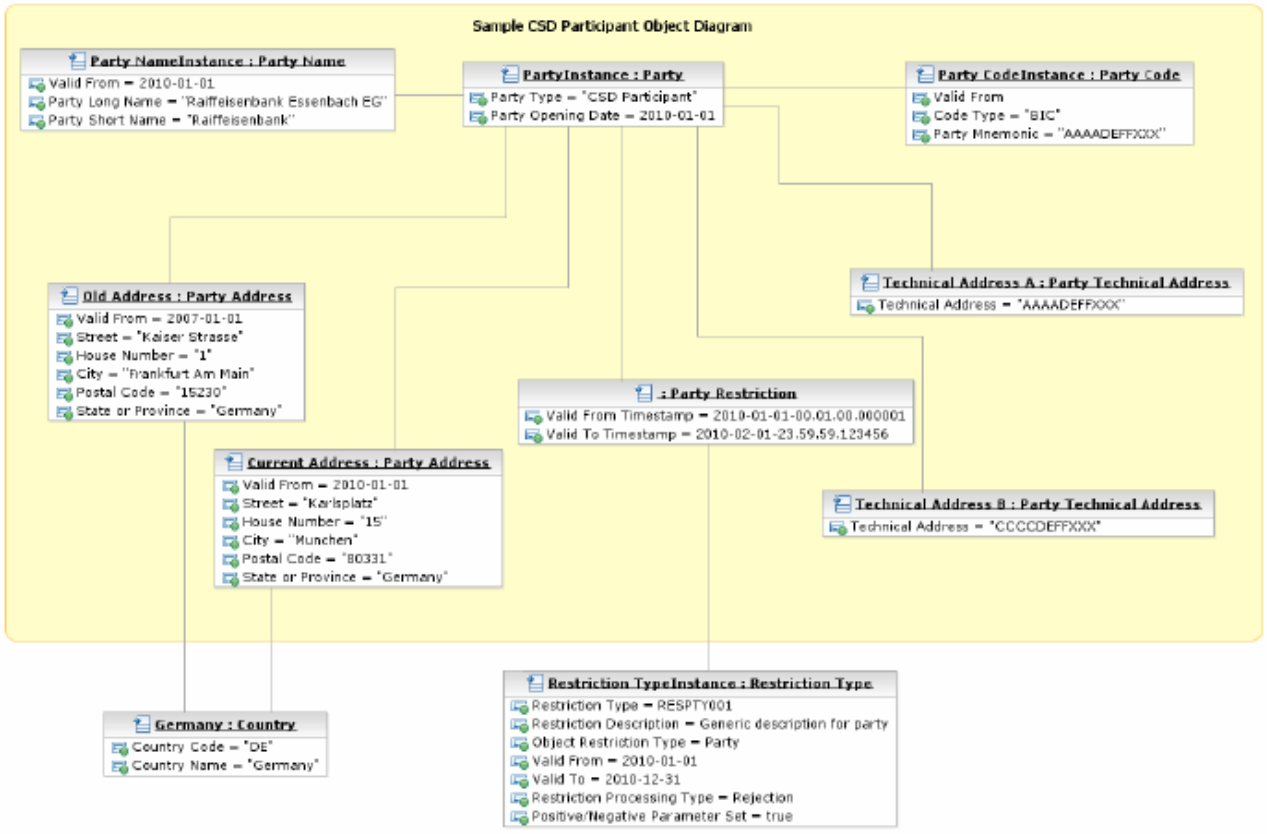
Each occurrence of this entity is linked to the relevant occurrence of *Eligible Counterpart CSD*. In addition, depending on the eligibility type, it is linked to the relevant security, or issuer CSD (to allow settlement in T2S for all the securities issued by the CSD) or country of issuance (to allow settlement in T2S for all the securities issued in the country).

3.3.7.4 Object diagram

Reference Id	***SDMG.PAR.ODI.1.1***

All rights reserved.

The next diagram represents a CSD Participant using two different *Party Technical Addresses* and changing *Party Address* within the same jurisdiction.



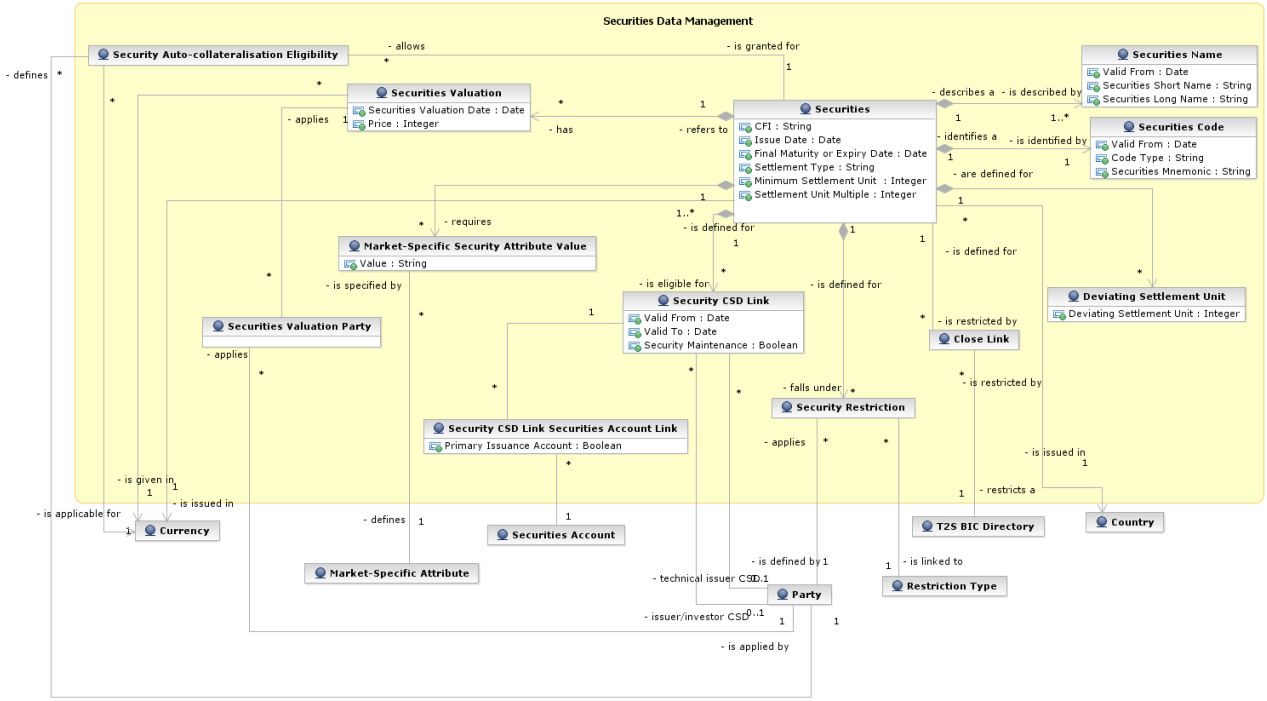
3.3.8 Securities Data Management

3.3.8.1 Data model of the module

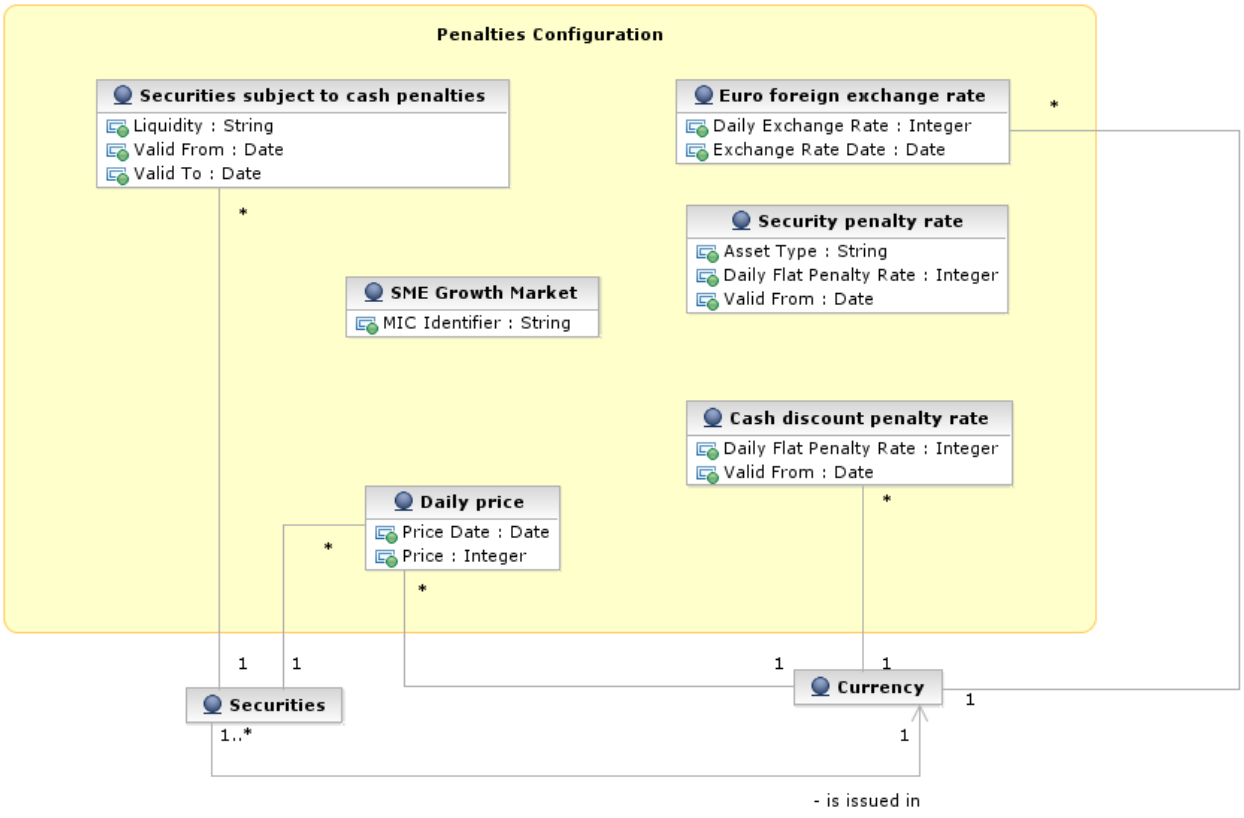
Reference Id	***SDMG.SEC.DMD.1.1***

All rights reserved.

The following diagram depicts the conceptual data model for *Securities Data Management* **{T2S.16.370}**. The scope of securities reference data is limited to the information that T2S requires for settlement.



The following diagram depicts the conceptual data model for Penalties configuration related to Securities.



All rights reserved.

3.3.8.2 Description of the module

Reference Id	***SDMG.SEC.DEM.1.1***

This module provides the services to manage securities reference data, their eligibility for settlement in issuer and investor CSDs and settlement restrictions. It supports the maintenance of close links, valuation prices and the eligibilities of securities in specific currencies for auto-collateralisation offered by NCBs or payment banks in T2S **{T2S.16.370}**. The functions described below allow the authorised T2S system users (i.e. CSD system administrators) to input their own securities and to access and maintain them, i.e. to create new securities or to update or delete already existing securities as well as managing the cash penalties configuration. When updating a security, each change is applied either as a revision or as an historic item, according to what is specified in the relevant request.

3.3.8.3 Description of the entities

1 – Securities

Reference Id	***SDMG.SEC.ENT.1.1***

This entity specifies all securities reference data that do not require a data history, i.e. all the attributes having only one valid value for a given security, regardless of the point in time taken into account **{T2S.16.380}**.

ATTRIBUTE	DESCRIPTION
CFI	Classification of the security according to ISO/DIS 10962:2015(E) standards.
Issue Date	Date from which settlement is allowed for the security.
Final Maturity or Expiry Date	Final maturity or expiry date of the security.
Settlement Type	Type of settlement foreseen for the security. The exhaustive list of possible values is as follows: ▮ Units ▮ Nominal
Minimum Settlement Unit	Minimum unit of the security in accordance with the value specified in the Settlement Type attribute.
Settlement Unit Multiple	Settlement Unit multiple of the security in accordance with the value specified in the Settlement Type attribute.

The entity *Securities Name* requires a data history. Each security has at least one corresponding item in the *Securities Name* entity. In addition, each security has one corresponding item in the Securities Code entity.

All rights reserved.

Each security inherits a country identifier for the country of issuance through the relationship with the *Country* entity and inherits a currency identifier for its currency of issuance [40] from the *Currency* entity. Furthermore, it may also be linked to one or many *Deviating Settlement Units* (see below), *Security Restrictions* [41] **{T2S.16.510}**, *Market-Specific Attributes* [42] (see section 3.3.11.9 [▸ 174]) and to one or many combinations of *Parties* (CBs or payment banks) offering auto-collateralisation and *Currencies* eligible for auto-collateralisation **{T2S.06.720} {T2S.16.899} {T2S.16.900}**. Furthermore, each instance of *Securities* is linked to one or several issuers or technical issuer CSDs and can be linked to one or many investor CSDs by using *Security CSD link* [43] **{T2S.16.460} {T2S.16.470} {T2S.16.480} {T2S.16.710} {T2S.16.935**and one or many parties for the definition of the relevant *Close Links* [44] **{T2S.06.720} {T2S.16.690}**. Finally, each security is linked to all its valuation data (see below).

2 – Securities Code

Reference Id	***SDMG.SEC.ENT.2.1***

This entity defines the identifiers of a security from a business perspective. The ISIN, based on ISO 6166 standard **{T2S.16.420}**, uniquely identifies a security. The ISIN does not change during the lifetime of a security, however the model supports the association of more than one occurrence to a single security, which allows managing exceptional cases of input errors.

The model allows only one ISIN to be valid at a given point in time.

ATTRIBUTE	DESCRIPTION
Valid From	Starting date of validity for the securities code.
Code Type	Code type assigned to the security. Currently, only ISIN (as defined by ISO 6166 standard) is foreseen.
Securities Mnemonic	Actual value for the securities code, i.e. the ISIN for the security.

The *Securities Code* entity inherits the Securities Identifier attribute from the *Securities* entity to link the security with its associated ISIN.

3 – Securities Name

Reference Id	***SDMG.SEC.ENT.3.1***

40 See section 3.3.11.13 [▸ 184] for a description of the Country and Currency entities.

41 For each security restriction, a period of validity, a restricting party (i.e. the T2S operator or a CSD) and a restriction type (see section 3.3.11.10 [▸ 176]) must be specified.

42 For each market-specific attribute, a valid value must be specified. The possibility to manage market-specific attributes shall be given not only to issuer CSD but also to investor CSDs, so to make them able to define restriction types based on their securities market-specific attributes.

43 For each Security CSD link, a period of validity, the type of link (i.e. if the CSD is issuer, investor or technical issuer for the relevant security) and a Boolean value indicating if the CSD is responsible for maintaining the security must be specified (for each security one and only one CSD can be responsible for its reference data maintenance including the cash penalties configuration Moreover, one or many issuance accounts can be defined within a given Security CSD Link.

44 Each Close Link defines a link between the relevant security and a party, i.e. specifies that the security is not eligible as collateral for that party.

All rights reserved.

This entity specifies the Securities Long Name and Securities Short Name of a security on a time line basis **{T2S.16.390}** as a security name may change in time owing, for example, to the renaming of the issuer. Therefore, several names may exist for a security, although only one name can exist for a security at any given point in time, requiring T2S to store a *Securities Name* on a timeline basis **{T2S.16.410}**.

ATTRIBUTE	DESCRIPTION
Valid From	Starting date of validity for the security name.
Securities Short Name	Short description of the security.
Securities Long Name	Long description of the security.

The *Securities Name* entity inherits a securities identifier attribute from the *Securities* entity to link the security name to the underlying security.

4 – Deviating Settlement Unit

Reference Id	***SDMG.SEC.ENT.4.1***

This entity specifies the odd lots, in which securities are eligible for settlement. Every security has a multiple settlement unit. A multiple of that defines the standard lot sizes eligible for settlement on condition of being equal or greater than the minimum settlement unit. However, securities exist that have several odd lot sizes outside of the standard settlement multiple. Therefore, T2S stores such deviations from the standard multiple in this entity to perform validations on whether T2S can settle the quantity or nominal amount for the security specified in the *Settlement Instruction* **{T2S.16.500}**.

ATTRIBUTE	DESCRIPTION
Deviating Settlement Unit	Deviating settlement unit for a security.

The *Deviating Settlement Unit* entity inherits the Security Identifier attribute from the *Securities* entity to link a deviating settlement unit to the underlying security.

5 – Securities Valuation

Reference Id	***SDMG.SEC.ENT.5.1***

This entity stores dirty prices for securities, with the haircut already deducted, for the valuation of securities positions for auto-collateralisation in T2S. Both CBs and payment banks offering auto-collateralisation, provide prices for the securities each has identified as eligible for auto-collateralisation **{T2S.16.520}**.

ATTRIBUTE	DESCRIPTION
Securities Valuation Date	Date for which the valuation applies.
Price	Price of the security.

The *Securities Valuation* entity inherits a currency identifier attribute for the currency of the valuation price from the *Currency* entity. The entity also inherits a securities identifier attribute from the *Securities* entity to link a valuation price to the underlying security. Finally, each *Securities Valuation* is linked to all the parties applying the price, i.e. either a CB (euro or non-euro) or a payment bank.

6 – Securities subject to cash penalties

All rights reserved.

Reference Id	**SDMG.SEC.ENT.6.1**

This entity defines the scope of securities subject to cash penalties **{T2S.16.930}**.

ATTRIBUTE	DESCRIPTION
Liquidity	It specifies whether a security is liquid or not. The exhaustive list of possible values is as follows: ▮ Liquid ▮ Illiquid
Valid From	Starting date of validity for the configuration of the penalties.
Valid To	Ending date of validity for the configuration of the penalties.

The Securities subject to cash penalties entity inherits the Security Identifier attribute from the linked Securities entity.

7 – SME Growth Market

Reference Id	**SDMG.SEC.ENT.7.1**

This entity is a catalogue of Market Identifier Codes (MIC Identifiers) corresponding to trading venues (Place of Trade) **{T2S.16.970}***. The catalogue is maintained by the T2S Operator only* **{T2S.16.980}.**

ATTRIBUTE	DESCRIPTION
MIC Identifier	It specifies the Market Identifier Code according to ISO 10383.

8 – Security penalty rate

Reference Id	**SDMG.SEC.ENT.8.1**

This entity stores the Security penalty rates according to the asset type **{T2S.16.990}***. The maintenance of security penalty rates is performed by T2S Operator only* **{T2S.16.1000}.**

ATTRIBUTE	DESCRIPTION
Asset Type	*It specifies the type of the asset related to the security penalty rate.*
Daily flat penalty rate	*It specifies the value for the security penalty rate.*
Valid From	*Starting date of validity for the security penalty rate.*

9 – Cash discount penalty rate

Reference Id	**SDMG.SEC.ENT.9.1**

This entity stores the Cash Discount Penalty Rates for each currency **{T2S.16.1010}***. The maintenance of cash discount penalty rates related to Euro currency is performed by T2S Operator only based on values provided by the European Central Bank* **{T2S.16.1020}.**

For other currencies, maintenance is performed by the Central Bank responsible for the currency.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Daily flat penalty rate	*It specifies the value for the cash discount penalty rate (i.e. discount rate of the currency).*
Valid From	*Starting date of validity for the cash discount penalty rate.*

The Cash Discount Penalty Rate entity inherits a currency identifier attribute for the currency of the rate from the Currency entity.

10 – Euro Foreign Exchange Rate

Reference Id	**SDMG.SEC.ENT.10.1**

This entity stores the Euro Foreign Exchange Rates against the Euro for each currency **{T2S.16.1030}.** *The maintenance of Euro foreign exchange rates is performed by the European Central Bank according to the official exchange rates* **{T2S.16.1040}.**

ATTRIBUTE	DESCRIPTION
Daily Exchange Rate	*It specifies the exchange rate for the linked currency against the Euro.*
Valid From	*Starting date of validity for the Euro foreign exchange rate.*

The Euro Foreign Exchange Rate entity inherits a currency identifier attribute for the currency of the rate from the Currency entity.

11 – Daily price

Reference Id	**SDMG.SEC.ENT.11.1**

This entity stores the Daily Prices to be used for the daily calculation of cash penalties **{T2S.16.1050***}. The maintenance of daily prices is performed by the Securities Maintaining Entity responsible for the security the price applies to* **{T2S.16.1060}**.

ATTRIBUTE	DESCRIPTION
Price Date	*Date for which the daily price applies.*
Price	*Price of the security.*

The Daily Price entity inherits a currency identifier attribute for the currency of the rate from the Currency entity. The entity also inherits a securities identifier attribute from the Securities entity to link a daily price to the underlying security.

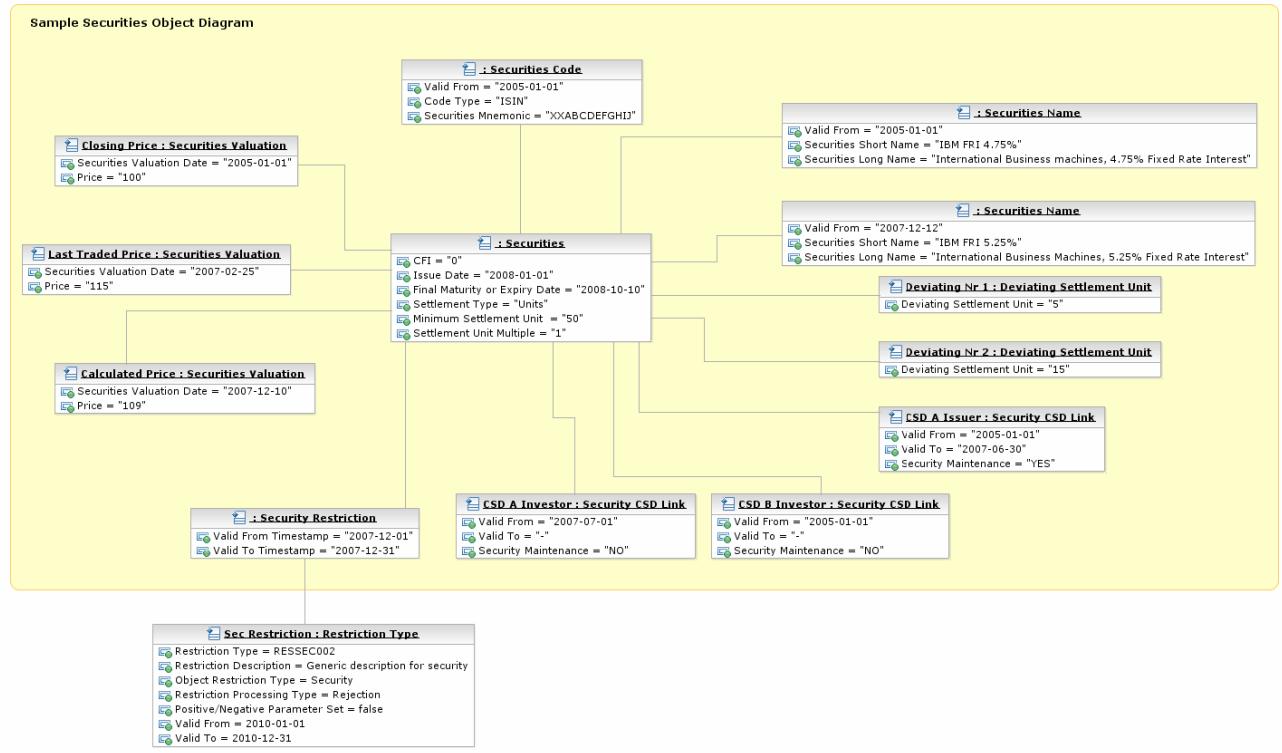
3.3.8.4 Object diagram

Reference Id	***SDMG.SEC.ODI.1.1***

The next diagram is an example of an instance of a security, according to the conceptual data model. The model links the instance of the security in the *Securities* entity to two different instances in the *Securities Name* entity owing to a history for the securities name. It specifies two occurrences for *Deviating Settlement Unit*. Furthermore, the example represents a switch of CSD A from issuer CSD to investor CSD of the secur-

All rights reserved.

ity, while retaining the responsibility to maintain the securities reference data. Finally, the example represents three occurrences of valuation prices for the security in *Securities Valuation* as well as an occurrence of a settlement restriction on the security in the *Security Restriction* entity.



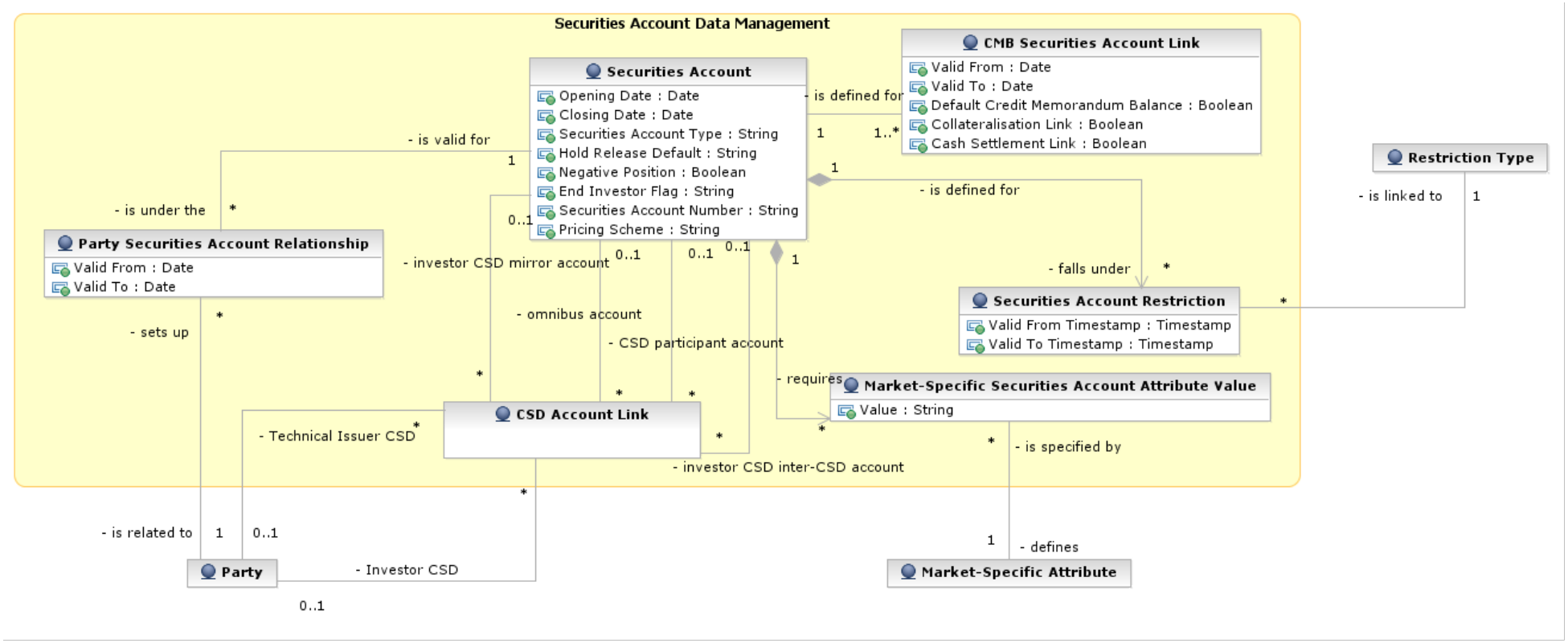
All rights reserved.

3.3.9 Securities Account Data Management

3.3.9.1 Data model of the module

Reference Id	***SDMG.SAC.DMD.1.1***

The following diagram shows the conceptual data model for Securities Account Data Management.



3.3.9.2 Description of the module

Reference Id	***SDMG.SAC.DEM.1.1***

This module allows the management of reference data related to *Securities Accounts* and their links to the relevant parties, CSDs restrictions and eligible currencies, plus the configuration of all the links defined between securities accounts for cross-CSD settlement. The functions described below allow the authorised T2S system users (i.e. CSD system administrators) to input their own *Securities Accounts* and to access and maintain them, i.e. to create new *Securities Accounts* or to update or delete already existing *Securities Accounts*. When updating a *Securities Account*, each change is applied either as a revision or as a data history, according to what is specified in the relevant request.

3.3.9.3 Description of the entities

1 – Securities Account

Reference Id	***SDMG.SAC.ENT.1.1***

This entity includes all *Securities Account* reference data required for settlement in T2S **{T2S.16.590}**.

ATTRIBUTE	DESCRIPTION
Securities Account Number	It specifies the unique securities account number.
Opening Date	Legal opening date for the securities account.
Closing Date	Legal closing date for the securities account.
Securities Account Type	It specifies a classification for the securities account for the maintenance of CSD-account links. The exhaustive list of possible values is as follows: ▮ CSD participant account ▮ CSD mirror account ▮ Inter-CSD account ▮ T2S technical offset account [45] ▮ CSD omnibus account ▮ Issuance account
Hold Release Default	Default setting for specific settlement instructions related to the securities account. The exhaustive list of possible values is as follows: ▮ Hold ▮ Release
Negative Position	Boolean attribute specifying whether the securities account can hold a negative position in a security.
End Investor Flag	It specifies information to identify securities accounts where allocation instructions are posted. The exhaustive list of possible values is as follows:

45 This category is foreseen for direct holding markets.

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▪ No account allocations ▪ FOP account allocations ▪ DVP/FOP account allocations
Pricing Scheme	It is the option to configure the pricing scheme of the customer. The exhaustive list of possible values is as follows: ▪ Account ▪ ISIN

Each *Securities Account* is linked to the *Party* (i.e. CSD or CSD participant) that operates the account. This link is time-dependent, as a CSD may transfer the relationship of a securities account from one account operator / sub-custodian to another account operator / sub-custodian within the same CSD [46] **{T2S.16.595}**. When opening a new *Securities Account*, the first link between the *Securities Account* and the relevant *Party* is automatically created in T2S and the start date of its validity period equals the opening date of the *Securities Account* **{T2S.16.596}**. When transferring the relationship of a *Securities Account* from one account operator / sub-custodian to another, the following mandatory attributes are specified by the CSD: the party identifiers of the two parties between which the CSD wishes to transfer the relationship, the date as of which the new link between the *Securities Account* and the new party is valid, the new *CMB Securities Account Links* for the *Securities Account* **{T2S.16.597}**. A relationship transfer results in the simultaneous:

- Closing of the old relationship (by setting the end date of its validity period);
- Creation of the new relationship (by creating a new link and setting the start date of its validity period);
- Replacement of the *CMB Securities Account Links* (see section 3.3.10 [▸ 146]) of the old account operator / sub-custodian with the links of the new one **{T2S.16.598}**;
- Finally, the new account operator / sub-custodian is able to view those transactions and positions of the end-investor account that existed in T2S prior to the transfer (e.g. positions, restrictions and settlement instructions) **{T2S.16.599}**;

Each *Securities Account* may then be linked to one or many *Credit Memorandum Balances* to offer auto-collateralisation for specific *T2S Dedicated Cash Accounts* **{T2S.16.900}** and it is linked to one or many CMB Securities Account Links (see section 3.3.10 [▸ 146]) **{T2S.16.591} {T2S.16.898} {T2S.16.899}**. In addition, each *Securities Account* may be linked to one or many *Securities Account Restrictions* [47] **{T2S.08.630} {T2S.16.680}** and *Market-Specific Attributes* [48](see section 3.3.11.9 [▸ 174]).

2 – CSD Account Link

Reference Id	***SDMG.SAC.ENT.2.1***

46 As a consequence, for each link a period of validity must be specified.

47 For each securities account restriction, a period of validity and a restriction type must be specified.

48 For each market-specific attribute, a valid value must be specified.

All rights reserved.

This entity includes all reference data concerning the configuration of CSD account links for cross-CSD settlement in T2S **{T2S.16.720} {T2S.16.730} {T2S.16.740}**.

ATTRIBUTE	DESCRIPTION
Valid From	This attribute specifies the date from which the CSD account link is valid from.
Valid To	This attribute specifies the date from which the CSD account link is valid to. An occurrence of the CSD account link requires a value in this attribute when the relationship is removed/closed.

Each *CSD Account Link* includes the following elements:

- A link to the party playing the role of the investor CSD;
- A link to the party playing the role of the technical issuer CSD;
- A link to a CSD participant account (within the relevant investor CSD) [49];
- A link to an omnibus account (within the relevant technical issuer CSD);
- A link to a mirror account (within the relevant investor CSD);
- A link to an inter-CSD account (within the relevant investor CSD).

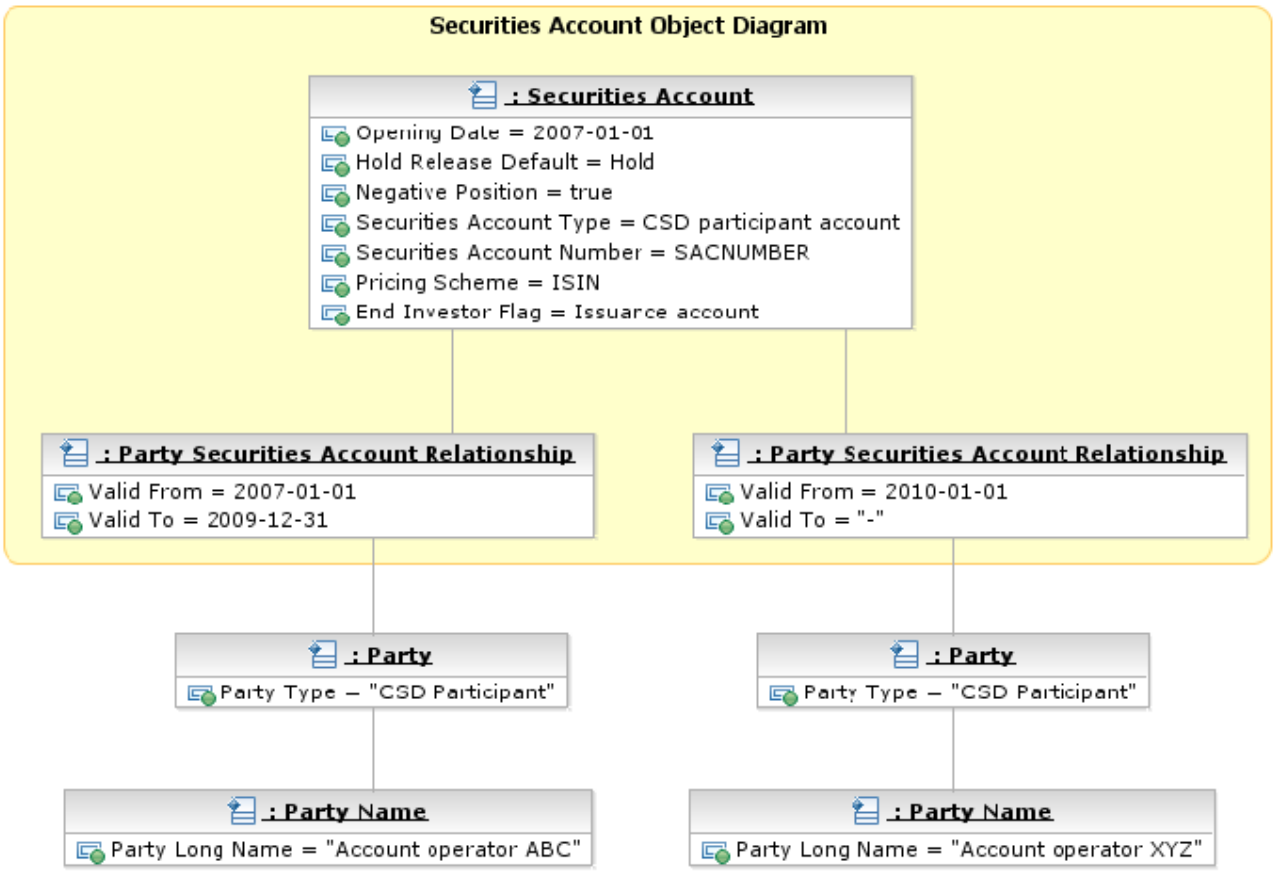
3.3.9.4 Object diagram

Reference Id	***SDMG.SAC.ODI.1.1***

49 This link is not specified for any CSD accounts link related to the default configuration of an omnibus account.

All rights reserved.

The next diagram shows a *Securities Account* and its relationship with two different parties, i.e. its account operators in different periods of time.



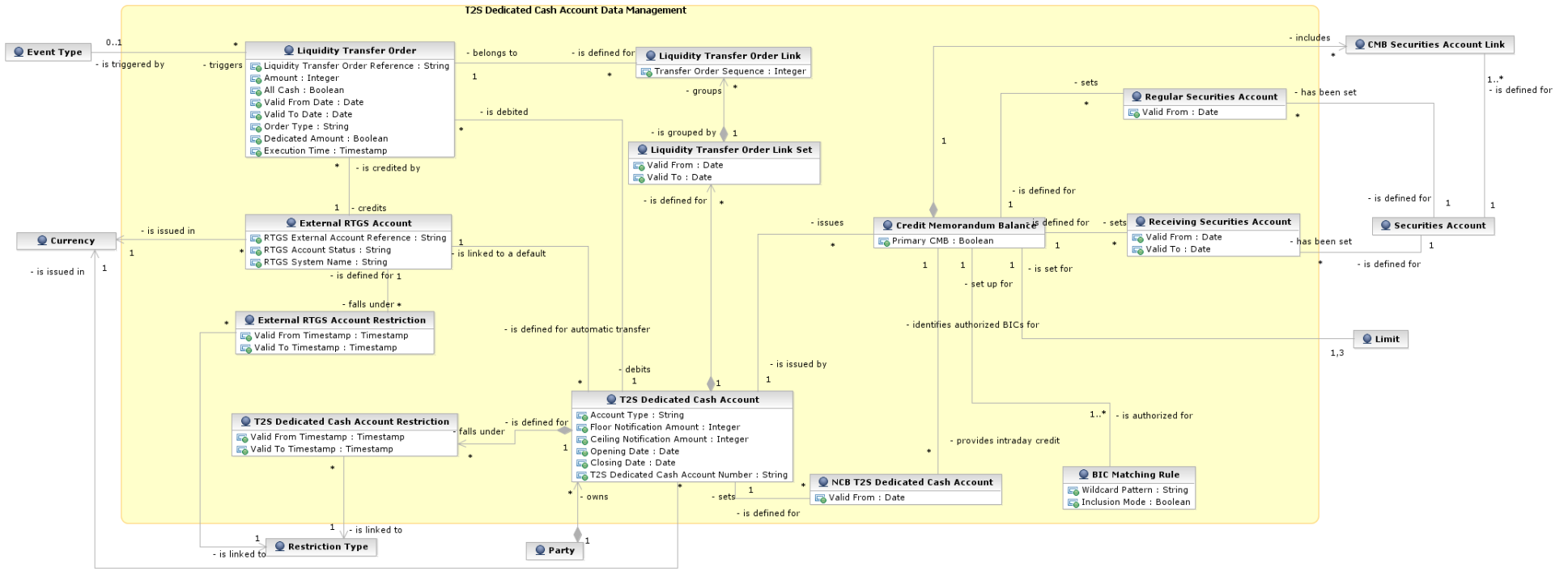
All rights reserved.

3.3.10 T2S Dedicated Cash Account Data Management

3.3.10.1 Data model of the module

Reference Id	***SDMG.DCA.DMD.1.1***

The following diagram shows the conceptual data model for *T2S Dedicated Cash Account Data Management*.



3.3.10.2 Description of the module

Reference Id	***SDMG.DCA.DEM.1.1***

This module allows the management of reference data related to *T2S Dedicated Cash Accounts* and their links to the relevant *Currencies* **{T2S.02.040} {T2S.02.050} {T2S.02.060}**, *Securities Accounts*, *External RTGS Accounts*, *T2S Dedicated Cash Accounts Restrictions* **{T2S.06.100}**, *Credit Memorandum Balances* and *Liquidity Transfer Order* configurations **{T2S.06.120} {T2S.06.350}**. The functions described below allow the authorised T2S System Users (i.e. CB system administrators) **{T2S.06.020}** to input their own *T2S Dedicated Cash Accounts* and to access and maintain them **{T2S.06.030}**, i.e. to create new *T2S Dedicated Cash Accounts* or to update or delete already existing *T2S Dedicated Cash Accounts* **{T2S.06.010} {T2S.06.110} {T2S.06.130}**, for adding or closing *External RTGS Accounts* **{T2S.16.656} {T2S.16.657}** and to configure link sets of *Liquidity Transfer Orders* (see below). When updating a *T2S Dedicated Cash Account* or an *External RTGS Account*, each change is applied either as a revision or as a data history, according to what is specified in the relevant request.

3.3.10.3 Description of the entities

1 – T2S Dedicated Cash Account

Reference Id	***SDMG.DCA.ENT.1.1***

This entity includes all *T2S Dedicated Cash Account* reference data **{T2S.16.600}**. An authorised CB business user can add **{T2S.16.610}** and close **{T2S.16.620}** *T2S Dedicated Cash Accounts* for the relevant payment banks in T2S.

ATTRIBUTE	DESCRIPTION
T2S Dedicated Cash Account Number	It specifies the unique T2S dedicated cash account number.
Floor Notification Amount	It specifies the lower threshold for notifying the cash manager.
Ceiling Notification Amount	It specifies the upper threshold for notifying the cash manager.
Account Type	It specifies a classification for the T2S dedicated cash account. The exhaustive list of possible values is as follows: ▮ RTGS Dedicated Transit Account ▮ T2S Central Bank Account ▮ T2S Dedicated Cash Account T2S Central Bank Accounts can have a negative balance. A RTGS Dedicated Transit Account per currency exists in T2S and it belongs to a central bank.
Opening Date	Opening date of the T2S dedicated cash account.
Closing Date	Closing date of the T2S dedicated cash account.

All rights reserved.

Each *T2S Dedicated Cash Account* is linked to its relevant *Party* **{T2S.05.060}** **{T2S.06.040}**, *Currency* **{T2S.02.080}** and **{T2S.06.050}** *External RTGS Cash Account* on which the liquidity available on the *T2S Dedicated Cash Account* shall automatically be repatriated at the end of the settlement day. In addition, it is linked to one or many *Credit Memorandum Balances* **{T2S.06.070}** **{T2S.06.080}** and it may be linked to one or many *T2S Dedicated Cash Account Restrictions* [50] **{T2S.16.680}** and *Liquidity Transfer Order Link Sets* (see below).

2 – Credit Memorandum Balance

Reference Id	*SDMG.DCA.ENT.2.1*

This entity includes all reference data for *Credit Memorandum Balances* defined on T2S Dedicated Cash Accounts **{T2S.16.616}**. The Credit Memorandum Balance is the mechanism to track the credit provision of a payment bank to one of its clients for a *T2S Dedicated Cash Account* on which the payment bank can set and monitor the external guarantee limit, the unsecured credit limit and the auto-collateralisation limit. The T2S Actor can create a Credit Memorandum Balance for the payment bank when a CB creates a new T2S Dedicated Cash Account for the payment bank **{T2S.16.611}** and it specifies the relevant BICs of this payment bank (the authorised system user can define a set of BICs selecting them one by one or using wildcards to specify multiple BICs at the same time. Single BICs and wildcards can be specified either in positive or negative mode, in order to include or exclude them from the set of BICs that will be linked to the Credit Memorandum Balance). T2S requires the CB the setup of the auto-collateralisation limit for the same *T2S Dedicated Cash Account* at the same time [51]. The T2S Actor can also create a *Credit Memorandum Balance* for a given *T2S Dedicated Cash Account* when the payment bank holding the account authorises one of its clients to use this account for settlement (by specifying the relevant BICs of this client) in the same way described above for the payment bank). The authorised system user can define a set of BICs selecting them one by one or using wildcards to specify multiple BICs at the same time. Single BICs and wildcards can be specified either in positive or negative mode, in order to include or exclude them from the set of BICs that will be linked to the Credit Memorandum Balance. In this case, T2S requires the set up of three client limits (for external guarantee, unsecured credit and auto-collateralisation) for the combination of BICs of the client and *T2S Dedicated Cash Account* [52] **{T2S.16.614}** **{T2S.16.592}**.

ATTRIBUTE	DESCRIPTION
Primary CMB	Boolean attribute specifying whether the credit memorandum balance is the primary one for the specified T2S Dedicated Cash Account.

Each *Credit Memorandum Balance* is linked to the relevant *T2S Dedicated Cash Account* and occurrences of the T2S BIC Directory (i.e. the BICs of the relevant payment bank or CB). In addition, it is also linked to one or three *Limits* **{T2S.16.612}** and it may be linked to one or many *CMB Securities Account Links* {T**2S.16.650 }** and to one *Securities Account* of the credit provider for the collateral. Finally, in the case of

50 For each T2S Dedicated cash account restriction, a period of validity and a restriction type (see section 3.3.11.10 [▸ 176]) must be specified.

51 If not specified, the limit is set up to zero by default.

52 If not specified, these limits are set up to zero by default.

All rights reserved.

CB collateralisation, it is linked to the *Securities Account h*olding the collateral securities positions related to the reverse (collateral settlement transactions) not reimbursed at the end of the settlement day and it may be linked to a *T2S Dedicated Cash Account* of the CB providing the intraday credit (this account is also used during the EOD period for collateral relocation).

3 – CMB Securities Account Link

Reference Id	*SDMG.DCA.ENT.3.1*

This entity specifies the links between each *Securities Account* and all the relevant *Credit Memorandum Balances as of a given date* **{T2S.16.650}**. For each new securities account, T2S requires the assignment of the securities account to one or more *T2S Dedicated Cash Accounts* of at least one payment bank that acts as liquidity provider. On this basis, T2S creates a new *CMB Securities Account Link* for each selected *T2S Dedicated Cash Account*, each of which references the *Credit Memorandum Balance* (of one of the selected T2S dedicated cash accounts) that was previously created for the party (i.e. the CSD participant) holding the *Securities Account*. Each *Securities Account* must have a default *T2S Dedicated Cash Account* **{T2S.16.591}**.

ATTRIBUTE	DESCRIPTION
Valid From	It specifies the date from which the CMB securities account link is valid.
Valid To	It specifies the date to which the CMB securities account link is valid.
Default Credit Memorandum Balance	Boolean attribute specifying whether the credit memorandum balance is the balance of the default T2S dedicated cash account for the securities account.
Collateralisation Link	Boolean attribute specifying whether T2S can use the securities, earmarked as collateral and held on the securities account, for auto-collateralisation operations on the T2S dedicated cash account it is linked with through the credit memorandum balance.
Cash Settlement Link	Boolean attribute specifying whether T2S can use the link between the securities account and the T2S dedicated cash account for the settlement of the cash leg of a settlement instruction.

Each *CMB Securities Account Link* is linked to the relevant *Securities Account* and *Credit Memorandum Balance*.

4 – Liquidity Transfer Order Link Set

Reference Id	*SDMG.DCA.ENT.4.1*

This entity includes all reference data for *Liquidity Transfer Order Link Sets*, i.e. groups of *Liquidity Transfer Orders* linked to an individual *T2S Dedicated Cash Account*, which can be used for sequencing of transfers from an individual *T2S Dedicated Cash Account* to the corresponding RTGS accounts **{T2S.16.661}**.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Valid From	It specifies the date from which the liquidity transfer order link set is valid.
Valid To	It specifies the date to which the liquidity transfer order link set is valid.

Each *Liquidity Transfer Order Link Set* is linked to one or many *Liquidity Transfer Order Links* and to the relevant *T2S Dedicated Cash Account*.

5 – Liquidity Transfer Order Link

Reference Id	*SDMG.DCA.ENT.5.1*

This entity includes all reference data for *Liquidity Transfer Order Links* **{T2S.16.662}**.

ATTRIBUTE	DESCRIPTION
Transfer Order Sequence	It specifies the sequence in which the relevant liquidity transfer order is executed within the link set. The value of this attribute is unique within all individual instances of each Liquidity Transfer Order Link Set.

Each *Liquidity Transfer Order Link* is linked to the relevant *Liquidity Transfer Order* and *Liquidity Transfer Order Link Set*.

6 – Liquidity Transfer Order

Reference Id	*SDMG.DCA.ENT.6.1*

This entity includes all reference data for *Liquidity Transfer Orders* **{T2S.16.660}**.

ATTRIBUTE	DESCRIPTION
Liquidity Transfer Order Reference	It specifies the unique reference assigned to the predefined and standing liquidity transfer order, by the instructing party.
Amount	Amount to be credited or debited through the liquidity transfer order.
All Cash	Boolean attribute specifying whether the liquidity transfer order transfers any remaining liquidity on the debit cash account. When this attribute is set to "true", then the Amount attribute must be set to zero.
Dedicated Amount	Boolean attribute specifying whether the liquidity transfer order transfers a dedicated amount stemming from a specific business event triggering the liquidity transfer order. For example, in the case of corporate actions (CARL) the liquidity transfer order transfers the amount of the corporate action. When this attribute is set to "true", then the Amount attribute must be set to zero.
Valid From Date	Date from which the liquidity transfer order is valid.
Valid To Date	Date to which the liquidity transfer order is valid.
Order Type	It specifies a classification for the liquidity transfer order. The exhaustive list of possible values is as follows:

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▪ Predefined ▪ Standing Order
Execution Time	It specifies the execution time of the liquidity transfer order, if it is not based on a specific event.

Each *Liquidity Transfer Order* is linked to the relevant *External RTGS Cash Account*, *Liquidity Transfer Order Link* and to the relevant *Triggering Event*.

It is possible to modify a *Liquidity Transfer order* via a Reference *Data Maintenance Instruction* containing its unique reference (i.e. Liquidity Transfer Order Reference). It is possible to modify any of the following pieces of information **{T2S.06.332}**:

- Link to the debit *T2S Dedicated Cash Account*;
- Link to the relevant *Currency*;
- Amount;
- All cash setting;
- Start validity date;
- End validity date;
- Execution type;
- Execution setting.

7 – External RTGS Account

Reference Id	***SDMG.DCA.ENT.7.1***

This entity includes all reference data for *External RTGS Cash Accounts* which CB business users can define for a payment bank or a settlement bank in T2S **{T2S.16.655}** **{T2S.16.656}**. It includes all the RTGS Accounts used for the repatriation of liquidity to the relevant RTGS system at the end of settlement day and all the RTGS Accounts referenced in any pre-defined and standing Liquidity *Transfer Order* or in any immediate *Liquidity Transfer*.

ATTRIBUTE	DESCRIPTION
RTGS External Account Reference	Account number of the external RTGS cash account within the relevant RTGS system.
RTGS Account Status	Business status for the external RTGS cash account. The exhaustive list of possible values is as follows: ▪ Open ▪ Closed
RTGS System Name	▪ It specifies the name of the RTGS system in which the external RTGS account is held.

All rights reserved.

Each *External RTGS Account* is linked to the relevant *RTGS System* and *Currency* **{T2S.16.655}**. Furthermore, it may be linked to one or many *T2S Dedicated Cash Accounts*, *Liquidity Transfer Orders* and *External RTGS Account Restrictions* [53]. Applying a specific restriction to an *External RTGS Account* results in applying the same restriction to all the *T2S dedicated cash accounts* linked to the same *External RTGS Account* **{T2S.16.658}**. Similarly, removing a specific restriction from an *external RTGS account* results in removing the same restriction from all the *T2S Dedicated Cash Accounts* linked to the same *External RTGS Account* **{T2S.16.659}**.

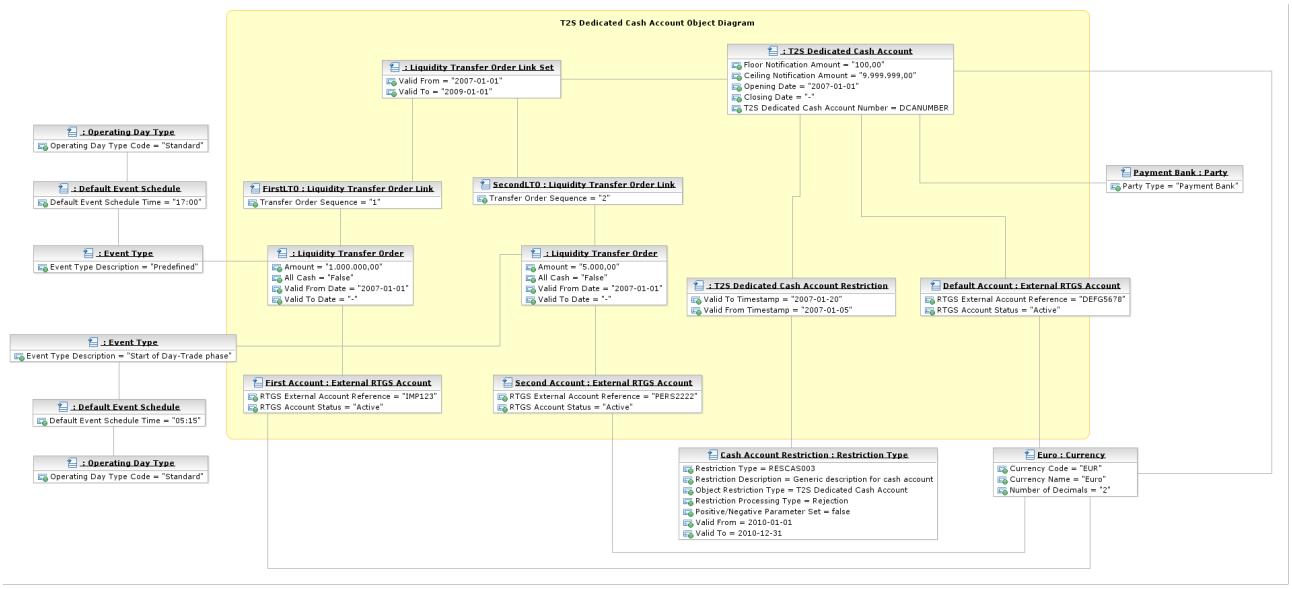
Any *External RTGS Account* can be closed by the authorised CB business user only if none of the following conditions is met **{T2S.16.657}**:

- There is a pending payment instruction specifying the *External RTGS Account*;
- The *External RTGS Account* has an active link to a *T2S Dedicated Cash Account*;
- The *External RTGS Account* is referenced in a valid *Liquidity Transfer Order*.

3.3.10.4 Object diagrams

Reference Id	***SDMG.DCA.ODI.1.1***

The next diagram is a representation of a *T2S Dedicated Cash Account* instance. It belongs to a Payment Bank and is linked to a CSD Participant *Securities Account* and to a default *External RTGS Account*. Two liquidity transfer orders are defined for the *T2S Dedicated Cash Account*: the former is the first to be executed and is based on a time constraint, the latter is linked to the start-of-day event.

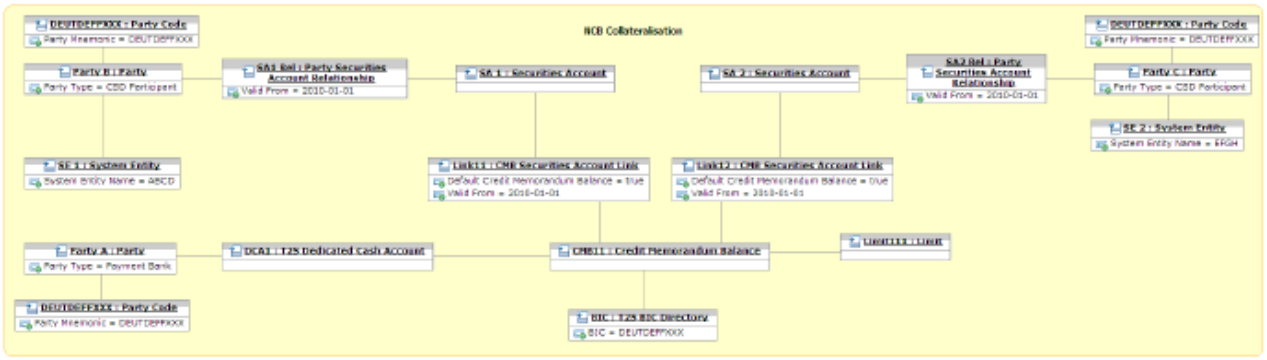


Reference Id	***SDMG.DCA.ODI.2.1***

53 For each restriction, a period of validity and a restriction type must be specified.

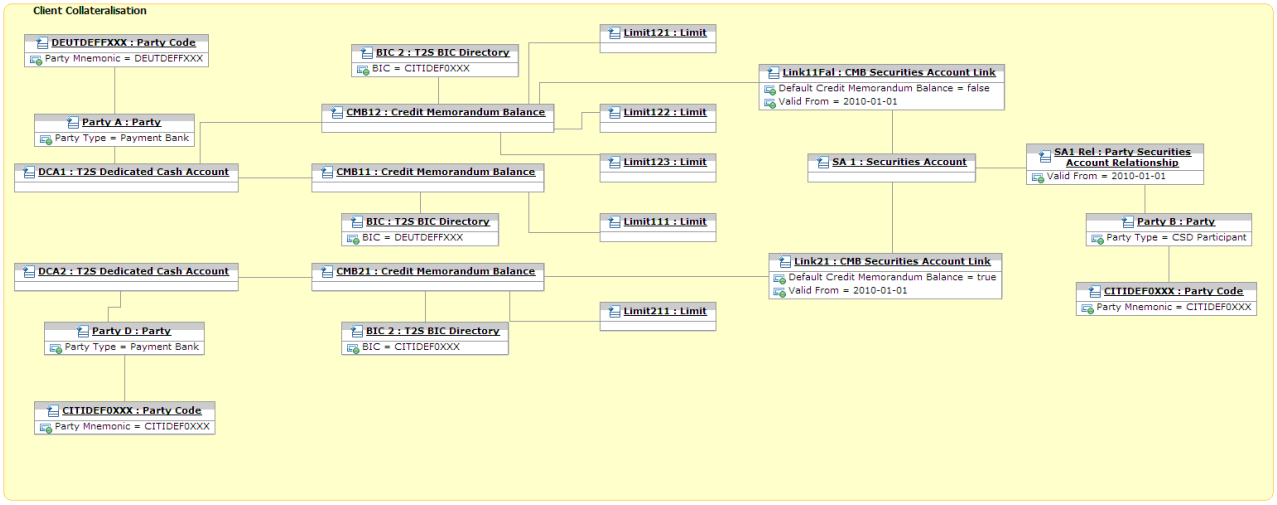
All rights reserved.

The next diagram shows the configuration of the Credit Memorandum Balance and CMB Securities Account Links set up by a legal entity (defined in T2S as a payment bank and, at the same time, as two CSD participants within two different CSDs) in order to use a single T2S Dedicated Cash Account for the settlement of the cash leg of settlement instructions related to its two securities accounts in T2S.



Reference Id	***SDMG.DCA.ODI.3.1***

The next diagram shows the configuration of the Credit Memorandum Balances and CMB Securities Account Links set up in order to allow a legal entity (defined in T2S both as a CSD participant and as a payment bank) to use its T2S Dedicated Cash Account and the T2S Dedicated Cash Account of another legal entity (defined as a payment bank in T2S) for the settlement of the cash leg of settlement instructions related to its securities account.



3.3.11 Rules and Parameters Data Management

3.3.11.1 Description of the module

Reference Id	***SDMG.RPS.DEM.1.1***

This module allows the management of reference data related to rules and parameters defined in T2S. The functions described below allow the authorised T2S system users to input their own rules and parameters and to access and maintain them, i.e. to create new rules and parameters or to update or delete already existing rules and parameters.

All rights reserved.

3.3.11.2 Data model and description of the entities

The data model for *Rules and Parameters Data Management* module is actually made of several components, each of which is related to a specific set of rules and parameters. A list of such components is given hereafter:

- users, roles and privileges;
- message subscription;
- conditional routing;
- report configuration;
- attribute domains;
- Business Day Management ;
- market-specific attributes;
- restriction types;
- conditional securities delivery;
- billing configuration;
- other rules and parameters.

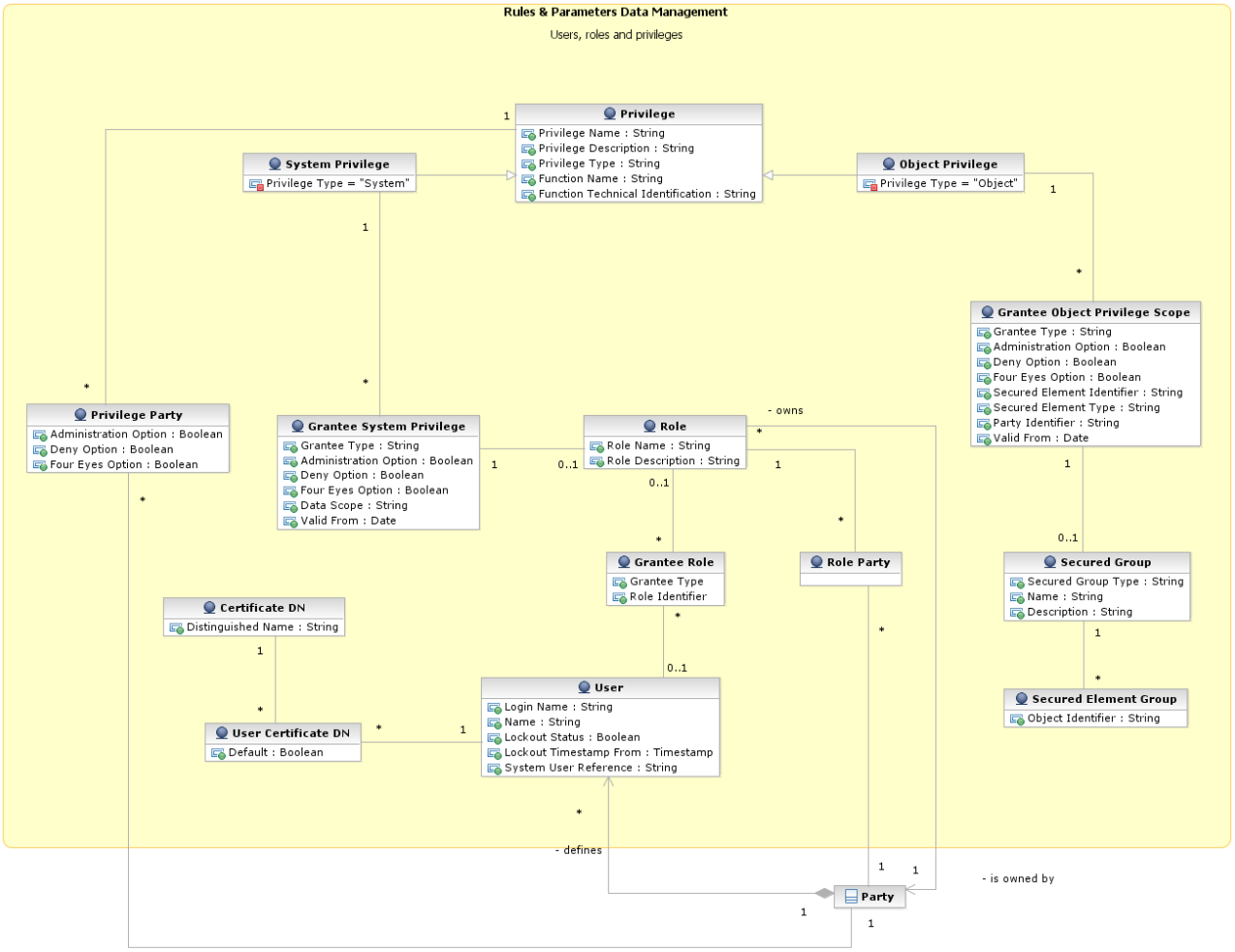
The following part of this section provides a description of the data model and the entities for all the above listed components.

3.3.11.3 Users, roles and privileges

Reference Id	***SDMG.RPS.DMD.1.1***

All rights reserved.

The following diagram shows the conceptual data model for *Users*, *Roles* and *Privileges* management.



Each T2S function is linked to a *Privilege* (i.e. the privilege to use this function), which is the means used for granting (or denying) access to functions (and possibly data) to selected *Parties*, *Users* and *Roles* **{T2S.11.355} {T2S.11.360}**. *Privileges* are created and maintained by T2S system administrators. *Privileges* can be granted or revoked by a T2S system administrator or by a privileged T2S system user, i.e. a user to whom the same *Privileges* have been previously granted with Administration Option (see below). It is not possible to grant contradicting *Privileges* to the same *Party*, *User* or *Role* (e.g. granting the same privilege twice with different configuration options) **{T2S.11.370}**

A set of *Privileges* can be grouped into a *Role*. Each *Party*, *User* and *Role* can be assigned one or more *Privileges*. Each Party and User can be assigned one or more Roles. *Roles* are created and managed by T2S system administrators or by privileged T2S system users, i.e. users to whom the privileges of creating and managing *Roles* have been previously granted. The management of *Roles* includes both their maintenance (i.e. update and logical deletion) and the possibility to grant or revoke other *Privileges* **{T2S.11.400} {T2S.11.430}**. CSDs and CBs can configure specific roles to be granted to their own CSD participants and payment banks, in order to grant them with proper access to T2S functions **{T2S.04.010} {T2S.11.410}**. In turn, system administrators of CSD participants or payment banks can use *Roles* granted by the relevant CSD or CB, in order to assign proper access rights to their own T2S system users **{T2S.11.420}**.

All rights reserved.

Based on the granted set of roles and privileges, all T2S system user are authorised to input their own Reference data objects and to access and maintain them, i.e. to create new objects or to update or delete already existing objects. For each T2S system user, the specific set of available functions and data are determined by the relevant access rights **{T2S.11.460} {T2S.11.470} {T2S.11.480} {T2S.11.490} {T2S.11.520}**.

1 – User

Reference Id	***SDMG.RPS.ENT.1.1***

This entity includes all reference data for *Users* {**T2S.11.440}**. This concept includes not only users interacting with T2S in U2A mode and triggering functions via ad hoc screens, but also applications connecting to T2S in A2A mode and using functions via XML messages. As a consequence, also the processes of authentication {**T2S.18.770}** and authorisation takes place both in case of a user interacting via the T2S graphical user interface and of an application exchanging messages with T2S [54].

ATTRIBUTE	DESCRIPTION
Login Name	Username to be provided for authentication.
Name	Full name of the user.
Lockout Status	Boolean attribute specifying whether the user is blocked from logging **{T2S.11.500} {T2S.11.510}.**
Lockout Timestamp From	Timestamp specifying the date and the time from which the user is locked out from T2S.
System User Reference	The unique system user reference associated to the user.

Users are linked to the *Party* they belong to and to one or many *Roles* and *Privileges* {**T2S.11.450}**.

Each User can be linked to one or many *Certificate DNs*.

2 – Certificate DN

Reference Id	***SDMG.RPS.ENT.2.1***

This entity includes all reference data for *Certificate DN* {**T2S.11.440}**.

ATTRIBUTE	DESCRIPTION
Distinguished Name	It specifies the distinguished name.

Each *Certificate DN* can be linked to one or many *Users*.

3 – Privilege

Reference Id	***SDMG.RPS.ENT.3.1***

This entity includes all reference data for *Privileges* {**T2S.11.360}**.

ATTRIBUTE	DESCRIPTION
Privilege Name	Name of the privilege.

54 Possible additional or different attributes for this entity needed in case of applications is provided in the detailed specifications phase.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Privilege Description	Description of the privilege.
Privilege Type	It specifies a classification for the privilege. The exhaustive list of possible values is as follows **{T2S.11.361}**: ▪ System, i.e. the associated function does not apply to a specific Reference data object type (e.g. a function to query the current phase of the settlement day). ▪ Object, i.e. the associated function applies to a specific Reference data object type (e.g. a function to display securities account).
Function Name	Name of the function associated to the privilege.
Function Technical Identification	It specifies all the data needed in order to identify and to trigger the function, e.g. the type of function (query, report, etc.), the type of interaction (push, pull, interactive), the set of required input parameters for the function and so forth [55].

Each *Privilege* can be linked to one or many *Parties*, *Users* and *Roles*. When linking a *Privilege* to a *Party*, a *User* or a *Role*, the following Boolean attributes are set:

- Deny Option, to specify whether the associated function is allowed or explicitly denied to the grantee;
- Administration Option, to specify whether the grantee of the privilege is allowed to grant the same privilege to another *Party*, *User* or *Role;* when revoking for a user a privilege that was granted previously with the administration option, the same privilege is also revoked, according to a cascade principle, for all the parties (including their users and roles) that were previously granted the same privilege by the same user;
- Four-Eyes Option, to specify whether the grantee of the privilege is allowed to use the associated function according to the Two-Eyes or Four-Eyes principle (this attribute is relevant only for privileges related to functions that can be used both according to the Two-Eyes and to the Four-Eyes principle) **{T2S.11.362}**.

System administrators can grant the available privileges to their *Parties*, *Users* and *Roles* in order to set up their change approval configuration **{T2S.16.180}**, i.e. the applicable combination of change type (e.g. create, update, delete) **{T2S.16.200}** and update type (i.e. Two-Eyes mode or Four-Eyes mode) **{T2S.16.190}** for all the relevant functions and Reference data objects **{T2S.16.210}**.

Object privileges, i.e. privileges related to functions that apply to specific Reference data objects can be granted:

- In relation to a single Reference data "secured" object only (e.g. a single securities account);
- In relation to a "secured" group of Reference data objects (e.g. a set of T2S dedicated cash accounts).

For each *Object Privilege*, the set of object types associated to it is already given by definition, i.e. it is determined by the specific function associated to the *Privilege* (e.g. the privilege to display a security can only be granted for securities and not for other Reference data object types).

4 – Role

55 The actual data to be stored for the technical identification of a function is clarified during the detailed specification phase.

All rights reserved.

Description

Reference Id	***SDMG.RPS.ENT.4.1***

This entity includes all reference data for *Roles* **{T2S.11.400}**.

ATTRIBUTE	DESCRIPTION
Role Name	Name of the role.
Role Description	Description of the role.

Each *Role* can be linked to one or many *Privileges*. Moreover, each *Role* can be linked to many *Parties* and *Users* **{T2S.11.530}**.

5 – Secured Object

Description

Reference Id	***SDMG.RPS.ENT.5.1***

This entity includes all reference data for *Roles* **{T2S.11.375}**.

ATTRIBUTE	DESCRIPTION
Secured Object Type	It specifies a classification for the secured object. The current exhaustive list [56] of possible values is as follows: ▮ Security ▮ Party ▮ Securities Account ▮ T2S Dedicated Cash Account

Each *Secured Object* is linked to the relevant Reference data object (which is of the same type expressed by the Secured Object Type attribute), it may be linked to one or many *Restriction Profiles* and may belong to one or many *Secured Groups*.

6 – Secured Group

Description

Reference Id	***SDMG.RPS.ENT.6.1***

This entity includes all reference data for *Secured Groups*, i.e. set of Reference data *Secured Objects* **{T2S.11.375}**.

ATTRIBUTE	DESCRIPTION
Secured Group Type	It specifies a classification for the secured group. The current exhaustive list of possible values is as follows: ▮ Security

56 The current list covers the actual business requirements expressed by the market. It can be enlarged with other object types at a later stage, in order to cover possible additional business needs.

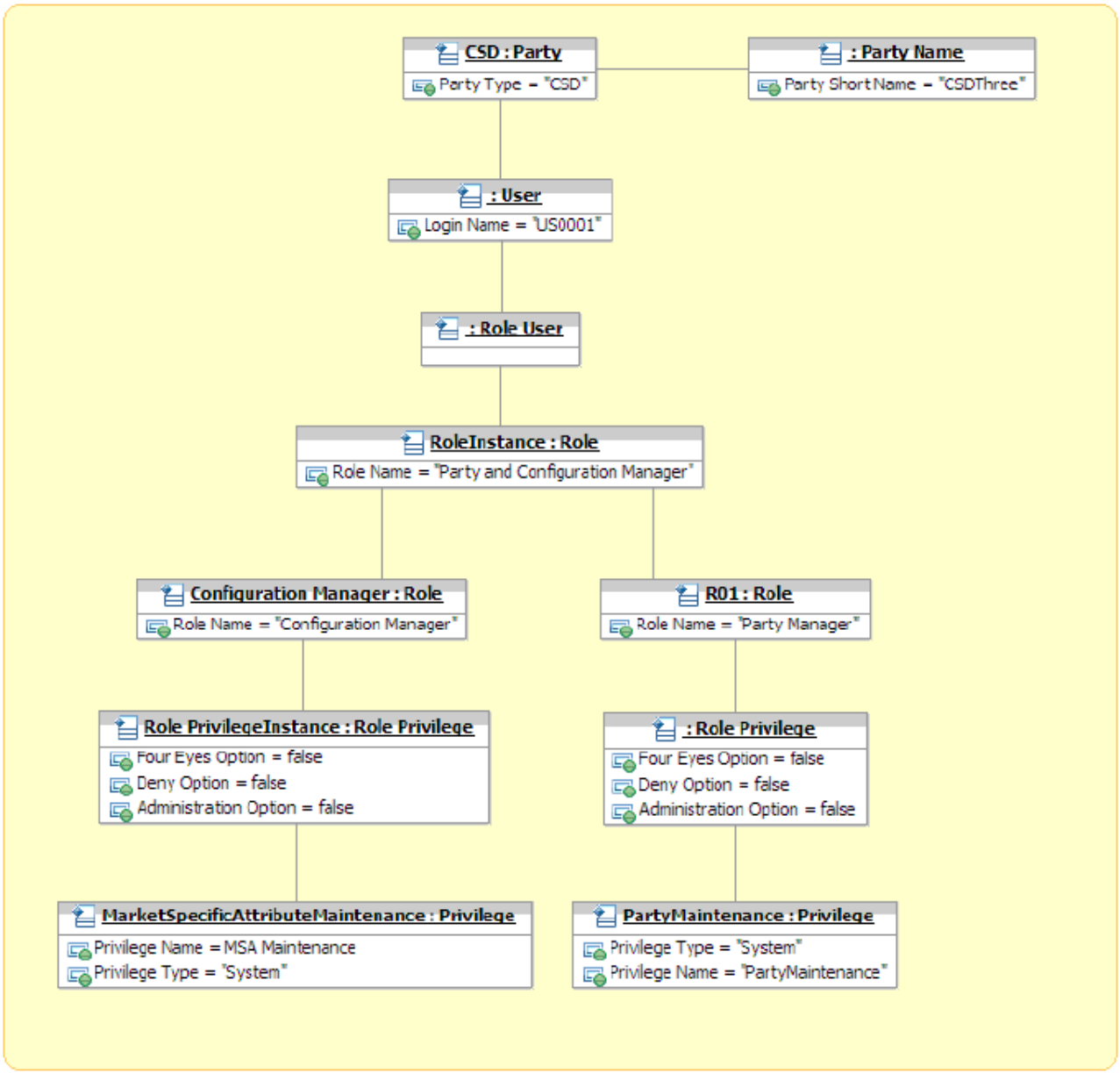
All rights reserved.

ATTRIBUTE	DESCRIPTION
	▪ Party ▪ Securities Account ▪ T2S Dedicated Cash Account
Name	The name assigned to the Secured Group.
Description	It represents the description assigned to the Secured Group.

Each *Secured Group* is linked to one or many *Secured Objects* (which is of the same type expressed by the Secured Group Type attribute).

Examples

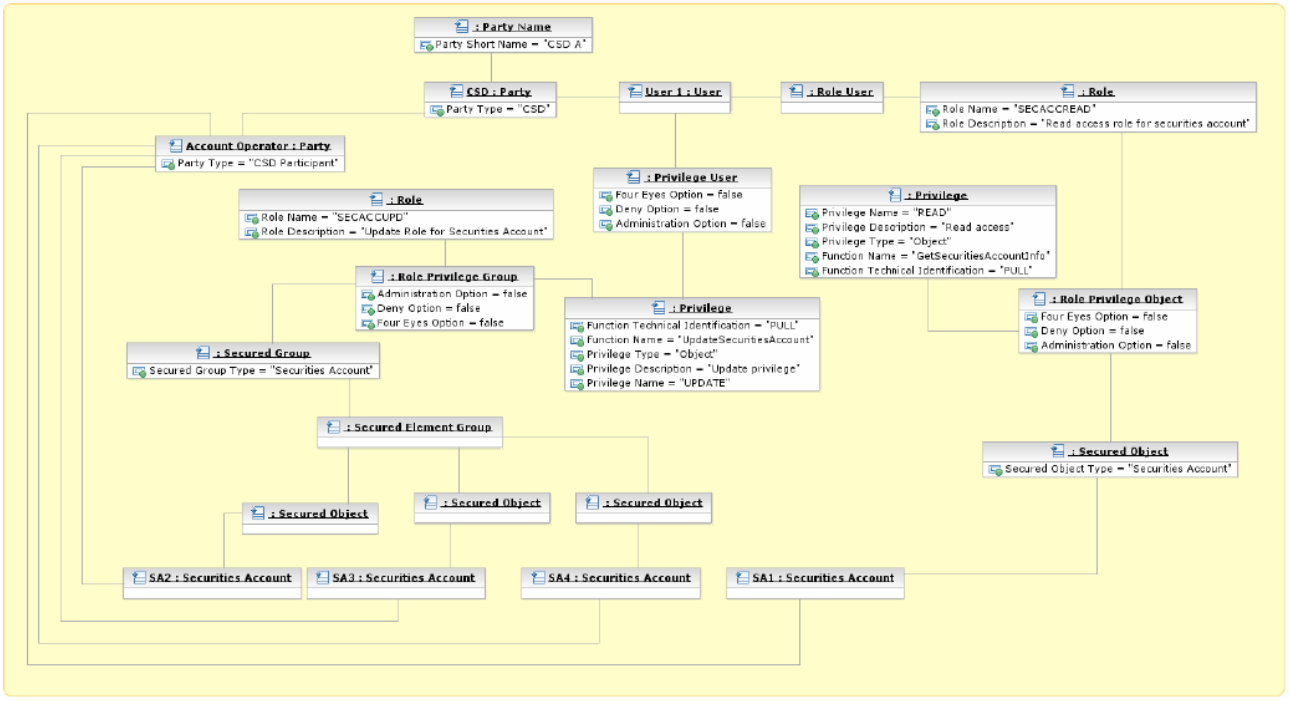
The following object diagram shows a simple example of *Roles* and *Privileges* grouping and their assignment to a T2S system user.



All rights reserved.

Reference Id	***SDMG.RPS.ODI.2.1***

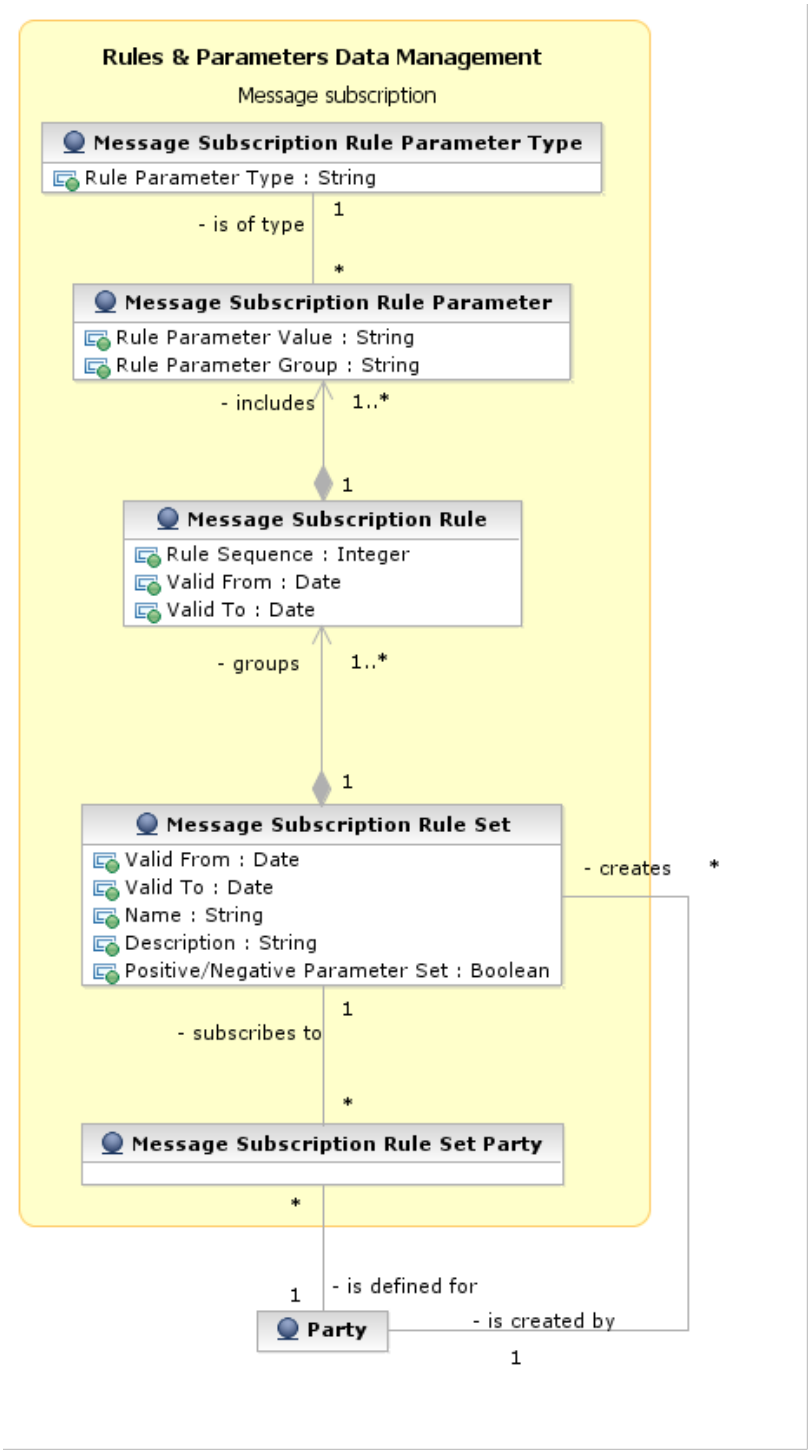
The following object diagram shows an example of *Roles and Privileges* assignment granting selective access to specific *Secured Objects* (i.e. *Securities Accounts*) or *Secured Groups* (i.e. groups of *Securities Accounts*).



3.3.11.4 Message subscription

Reference Id	***SDMG.RPS.DMD.2.1***

The following diagram shows the conceptual data model for *Message Subscription* management.



Message Subscription allows CSDs and CBs to configure, for themselves and for their directly connected parties, the specific set of messages they want to receive from T2S **{T2S.11.640} {T2S.13.030}**.

Each CSD and CB can set up several *Message Subscription Rule Sets*. Each *Message Subscription Rule Set* defines the messages one or many interested parties receives via a sequence of *Message Subscription Rules*. Each *Message Subscription Rule* specifies the parameters (e.g. message type, party, and securities account number) that have to be taken into account to identify the messages to be sent to the interested parties.

All rights reserved.

7 – Message Subscription Rule Set

Reference Id	***SDMG.RPS.ENT.6.1***

This entity defines the set of message subscription rules defined by each CSD or CB **{T2S.11.640}**.

ATTRIBUTE	DESCRIPTION
Valid From	It specifies the date from which the rule set is valid.
Valid To	It specifies the date to which the rule set is valid.
Name	The name assigned to the message subscription rule set.
Description	It represents the description assigned to the message subscription rule set.
Positive/Negative Parameter Set	It specifies whether the message subscription rule set must be used in positive or negative way.

Each *Message Subscription Rule Set* is linked to the relevant CSD or CB, to one or many interested parties (i.e. the parties that receive all the messages identified by the message subscription rule set), and to a set of *Message Subscription Rules*.

8 – Message Subscription Rule

Reference Id	***SDMG.RPS.ENT.8.1***

This entity defines the *Message Subscription Rules* defined by each CSD or CB **{T2S.11.640}**.

ATTRIBUTE	DESCRIPTION
Rule Sequence	It specifies the order in which the rules are processed within the relevant rule set.
Valid From	It specifies the date from which the rule is valid.
Valid To	It specifies the date to which the rule is valid.

Each *Message Subscription Rule* belongs to a single *Message Subscription Rule Set* and it is linked to a set of *Message Subscription Rule Parameters*.

9 – Message Subscription Rule Parameter

Reference Id	***SDMG.RPS.ENT.9.1***

This entity includes the message subscription rule parameters defined within each message subscription rule **{T2S.11.640}**.

ATTRIBUTE	DESCRIPTION
Rule Parameter Group	It specifies the group of the rule parameter. All the groups within a message subscription rule include the same number of rule parameters. A rule is matched when all the rule parameters of at least one of its groups are matched.
Rule Parameter Value	It specifies a valid value for the rule parameter.

Each *Message Subscription Rule Parameters* belongs to a single *Message Subscription Rule* and it is linked to a specific *Message Subscription Rule Parameter Type*.

10 – Message Subscription Rule Parameter Type

All rights reserved.

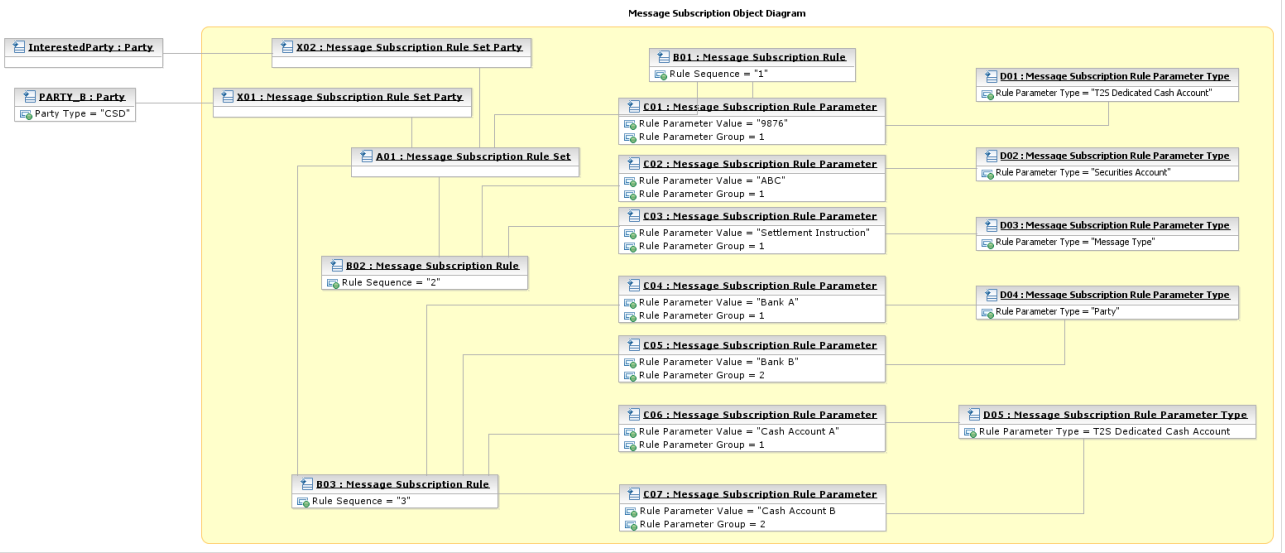
Reference Id	***SDMG.RPS.ENT.10.1***

This entity defines all message subscription rule parameters types **{T2S.11.640} {T2S.13.040}**.

ATTRIBUTE	DESCRIPTION
Rule Parameter Type	It specifies a classification for the message subscription rule parameters. The exhaustive list of possible values is as follows: ▪ Message type ▪ Instruction type ▪ Message status ▪ Party ▪ Securities account number ▪ ISIN ▪ T2S Dedicated cash account number ▪ Instruction status ▪ ISO Transaction code ▪ Currency ▪ Already matched flag ▪ Settlement Transaction Condition Code ▪ Business sending party ▪ System User Reference ▪ Instructing party

Reference Id	***SDMG.RPS.ODI.2.1***

The following object diagram provides an example of the data model for message subscription:

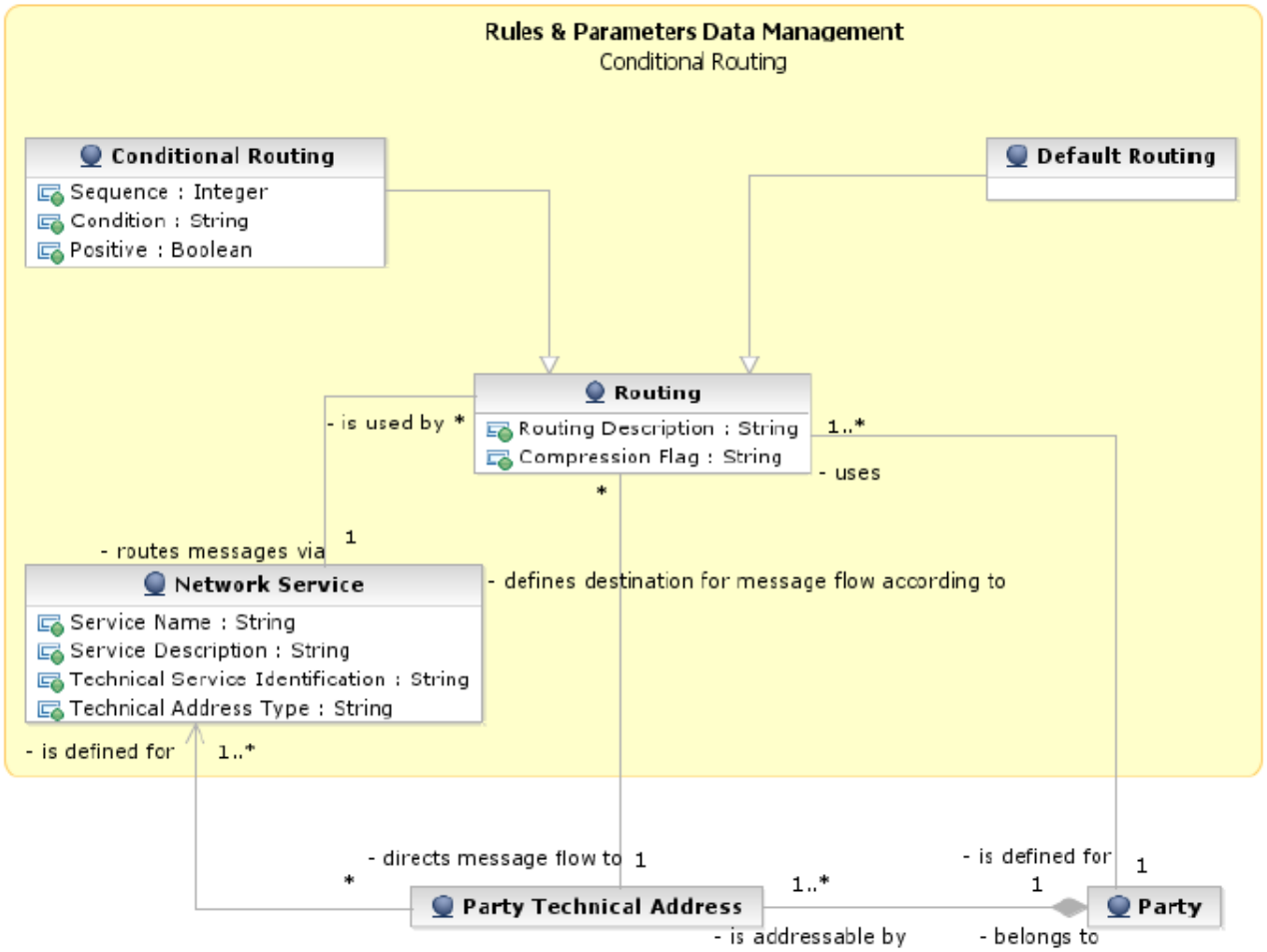


All rights reserved.

3.3.11.5 Conditional routing

Reference Id	***SDMG.RPS.DMD.3.1***

The following diagram shows the conceptual data model for *Conditional Routing*.



Conditional Routing allows parties to configure routing information that T2S uses to deliver outgoing messages to them.

Each party can set up several routing conditions. Each routing condition defines the network service and the technical address that T2S uses to route the relevant message, on the basis of the following criteria: size boundaries, currency and message type.

11 – Routing

Reference Id	***SDMG.RPS.ENT.11.1***

This entity stores all configuration information for *Conditional Routing*. For each party, a default routing configuration is defined, that is used when no specific routing conditions are defined for the same party and for a specific outgoing message.

ATTRIBUTE	DESCRIPTION
Routing Description	Description of the routing configuration.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Compression Flag	It specifies the type of compression selected. The exhaustive list of possible compression values is as follows: ▌ No Compression ▌ ZIP Algorithm

Each *Routing* configuration is linked to the relevant *Party*, *Party Technical Address* and *Network Service*. For the configuration of routing conditions, the following additional information must be provided.

12 – Conditional Routing

Reference Id	***SDMG.RPS.ENT.12.1***

This entity is a specialisation of the *Routing* entity and defines configuration of routing conditions (i.e. non-default routing configurations).

ATTRIBUTE	DESCRIPTION
Sequence	It specifies the order in which the routing conditions for the same party are processed.
Condition	It specifies the set of conditions applicable for a specific routing configuration. The exhaustive list of possible criteria for the specification of these criteria is as follows: ▌ Size boundaries ▌ Currency ▌ Message type [57]
Positive	This Boolean attribute specifies whether the set of conditions represent a positive or negative list. A positive setting would specify the combinations of size boundaries, currencies and message types that routes to a specific technical address using a specific network service. A negative setting would specify the combinations of size boundaries, currencies and message types that is not routed to the same technical address, via said network service.

Each routing condition, being a specialisation of the *Routing* entity, is linked as well to the relevant *Party*, *Party Technical Address* and *Network Service*.

13 – Network Service

Reference Id	***SDMG.RPS.ENT.13.1***

This entity stores reference data of all network services available in T2S.

ATTRIBUTE	DESCRIPTION
Service Name	Name of the network service.
Service Description	Description of the network service.

57 The actual data to be stored for the definition of a routing condition is clarified during the detailed specification phase.

All rights reserved.

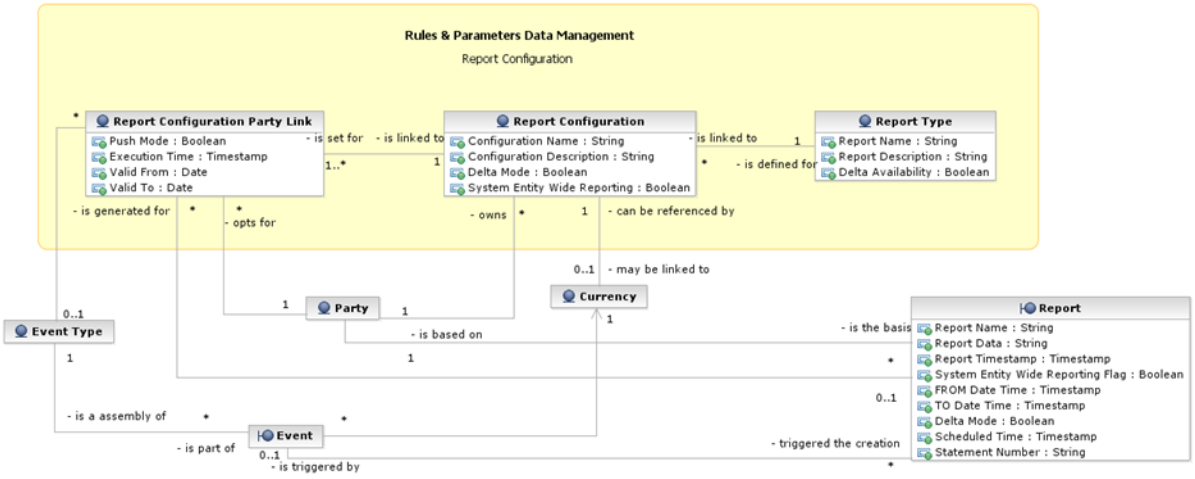
ATTRIBUTE	DESCRIPTION
Technical Service Identification	It specifies all the data needed in order to identify and to use a network service [58].
Technical Address Type	It specifies the type of technical address for the network service (e.g. BIC, Distinguished Name, IP address).

Each *Network Service* is linked to all the *Party Technical Addresses* it provides and to all the *Routing* configuration using it for the delivery of outgoing messages.

3.3.11.6 Report configuration

Reference Id	***SDMG.RPS.DMD.4.1***

The following diagram shows the conceptual data model for report configuration.



Report configuration allows T2S parties to configure the specific set of reports they want either to receive (push mode) or to download (pull mode) from T2S.

13 – Report Type

Reference Id	***SDMG.RPS.ENT.13.1***

This entity defines all types of reports available in T2S.

ATTRIBUTE	DESCRIPTION
Report Name	Name of the report type. The various reports are described in more detail in the latest versions of the User Detailed Functional Specification (UDFS) and User Handbook (UHB) for the Graphical User Interface
Report Description	Description of the report type.
Delta Availability	Boolean attribute specifying whether the report is also available in delta mode, i.e. with the possibility for the recipient to get only the changes since the last time the recipient got the same report from T2S.

58 The actual data to be stored for the technical identification of a network service is clarified during the detailed specification phase.

All rights reserved.

Each *Report Type* and can be referenced by many *Report Configurations*.

15 – Report Configuration

Reference Id	***SDMG.RPS.ENT.15.1***

This entity stores all reference data for report configurations. Each *report configuration* specifies a type of report, its data scope (i.e. full or delta report), the set of parties entitled to get said type of report and the mode they get it (i.e. push or pull).

ATTRIBUTE	DESCRIPTION
Configuration Name	Name of the report configuration.
Configuration Description	Description of the report configuration.
Delta Mode	Boolean attribute specifying whether the recipient gets the report linked to the report configuration in delta mode or in full mode.
System Entity Wide Reporting	Boolean attribute specifying whether the recipient gets the report for data belonging to the entire system entity.

Each *Report Configuration* is linked to the *Party* that owns the report configuration, to the relevant *Report Type* and *Event Types* (see section 3.3.11.8 [▶ 172]), to the currency for which the information are required, and to one or many parties entitled to get the same *Report Type* [59].

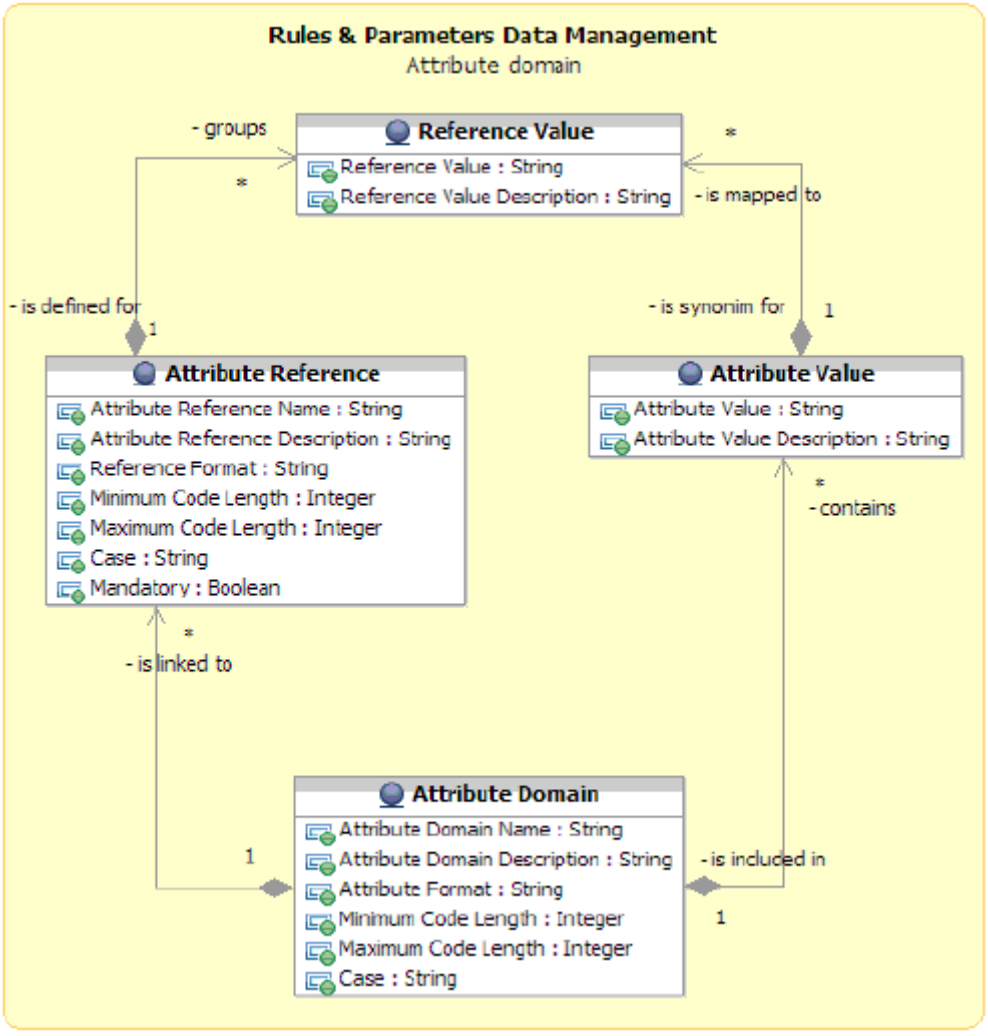
3.3.11.7 Attribute domain

Reference Id	***SDMG.RPS.DMD.5.1***

59 For each of these links a Boolean value specifies whether the party receives its report in push mode or if it downloads it in pull mode. A timestamp specifies the execution time for the relevant report. A validity period can be defined by giving a valid from and valid to date.

All rights reserved.

The following diagram shows the conceptual data model for attribute domain management **{T2S.11.210} {T2S.11.300} {T2S.11.310} {T2S.11.320} {T2S.11.270} {T2S.11.290}**.



Attribute domains provide valid lists of values allowed for an attribute **{T2S.11.200}**. *Attribute domains* are used for field validations and for documenting the business definition of a value in an attribute. Furthermore, it is possible to define additional reference values, mapped to an attribute specified by an *Attribute domain* definition.

Attribute Domains can also be used to store definitions and values of T2S configuration parameters. A list of these configuration parameters is given hereafter:

- Allegement for first unsuccessful matching attempt: it is defined as the standard delay period from the first unsuccessful matching attempt of a settlement instruction. It is calculated in hours and minutes **{T2S.11.545}**;
- Allegement before intended settlement date: it is defined as the standard delay period measured backwards from the FOP cut-off time on the intended settlement date. It is calculated in hours and minutes. T2S sends out an allegement at the earliest point in time between this period and the period defined by the allegement for first unsuccessful matching **{T2S.11.545}**;

All rights reserved.

- Recycling period for unmatched pending settlement instruction: it is defined as the number of settlement days an unmatched pending settlement instruction can be recycled in T2S before being purged **{T2S.11.900} {T2S.11.910}**;
- Recycling period for matched pending settlement instruction: it is defined as the number of settlement days an matched pending settlement instruction can be recycled in T2S before being purged {T2S.11.900} {T2S.11.910};
- Acceptable time deviation period: it is defined as the maximum number of settlement days between the reception date of a settlement instruction and its intended settlement date (in the past or in the future) that T2S can accept **{T2S.11.920}**. The current value for the acceptable deviation period in unlimited **{T2S.11.930}**;
- Settlement priority defaults: they are defined as the default settlement priority levels that T2S automatically assigns for the processing of settlement instructions, on the basis of the relevant instruction type, transaction code and instructing party type **{T2S.11.340}**. The exhaustive list of possible settlement priority default values is as follows: reserved, top, high and normal **{T2S.07.150}**;
- Sequencing rules: they specify the configuration of sequencing that T2S uses for the processing of settlement instructions, on the basis of the relevant instruction type and transaction code **{T2S.11.350}**. The exhaustive list of possible sequencing rule values is as follows: sequence 1 (corporate actions related settlements), sequence 2 (free-of-payment rebalancing of securities), sequence 3 (CB-specific operations) and sequence 4 (trading-related and other instructions) **{T2S.07.030}**;
- Duplicate check period: it specifies the predefined number of days that T2S uses within the duplicate check process related to settlement instructions, settlement restrictions, maintenance instructions and immediate liquidity transfer orders. More precisely, in order to check for possible duplicates between incoming instructions and instructions already settled or cancelled, T2S performs the duplicate check by comparing an incoming instruction in addition to all pending instructions also with all the instructions already settled or cancelled in a number of days in the past specified by this parameter;
- Billing information: it is a set of reference data concerning the billing legal entity. It includes the following parameters: name and address of the legal entity issuing the T2S invoices, contact telephone number and contact e-mail address of the person in charge of the invoicing process, remittance details, legal registration (i.e. tax identification number of the billing legal entity), payment date (i.e. number of days between the invoice issuance and the payment), VAT number, signature statement;
- Retention period: it specifies the predefined number of days (from the logical deletion of a Reference data object or since a transactional data reached its final status) that T2S waits before archiving these data and performing their physical deletion from the operational data base **{T2S.16.300}**.

16 – Attribute Domain

Reference Id	***SDMG.RPS.ENT.16.1***

This entity includes all reference data for *Attribute Domains* **{T2S.11.240}**.

ATTRIBUTE	DESCRIPTION
Attribute Domain Name	Name of the attribute domain.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Attribute Domain Description	Description of the attribute domain.
Attribute Format	It specifies a classification for the attribute format. The exhaustive list of possible values is as follows: ▪ Alphabetic ▪ Alphanumeric ▪ Numeric
Minimum Code Length	It specifies the minimum length of the code for a value in the attribute domain.
Maximum Code Length	It specifies the maximum length of the code for a value in the attribute domain.
Case	It specifies a classification for the case type of the attribute format. The exhaustive list of possible values is as follows: ▪ Upper case ▪ Lower case ▪ Both

Each *Attribute Domain* is linked to many attribute values (i.e. to all the values belonging to the domain) and may be linked to one or many *Attribute References*.

17 – Attribute Reference

Reference Id	***SDMG.RPS.ENT.17.1***

This entity includes all reference data for *Attribute References* **{T2S.11.230} {T2S.11.240} {T2S.11.250}**

ATTRIBUTE	DESCRIPTION
Attribute Reference Name	Name of the attribute reference.
Attribute Reference Description	Description of the attribute reference.
Reference Format	It specifies a classification for the attribute reference format. The exhaustive list of possible values is as follows: ▪ Alphabetic ▪ Alphanumeric ▪ Numeric
Minimum Code Length	It specifies the minimum length of the code for a reference value.
Maximum Code Length	It specifies the maximum length of the code for a reference value.
Case	It specifies a classification for the case type of the attribute reference format. The exhaustive list of possible values is as follows: ▪ Upper case ▪ Lower case

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▌ Both
Mandatory	Boolean attribute specifying whether the input of a reference value is mandatory.

Each *Attribute Reference* is linked to the relevant *Attribute Domain* and to many reference values (i.e. to all the values belonging to the *Attribute Reference*).

18 – Attribute Value

Reference Id	***SDMG.RPS.ENT.18.1***

This entity includes all reference data for attribute values **{T2S.11.220}**.

ATTRIBUTE	DESCRIPTION
Attribute Value	Value of the attribute.
Attribute Value Description	Description of the attribute value.

Each *Attribute Value* is linked to the relevant *Attribute Domain* and may be linked to one or many *References Value*.

19 – Reference Value

Reference Id	***SDMG.RPS.ENT.19.1***

This entity includes all reference data for *Reference Values*.

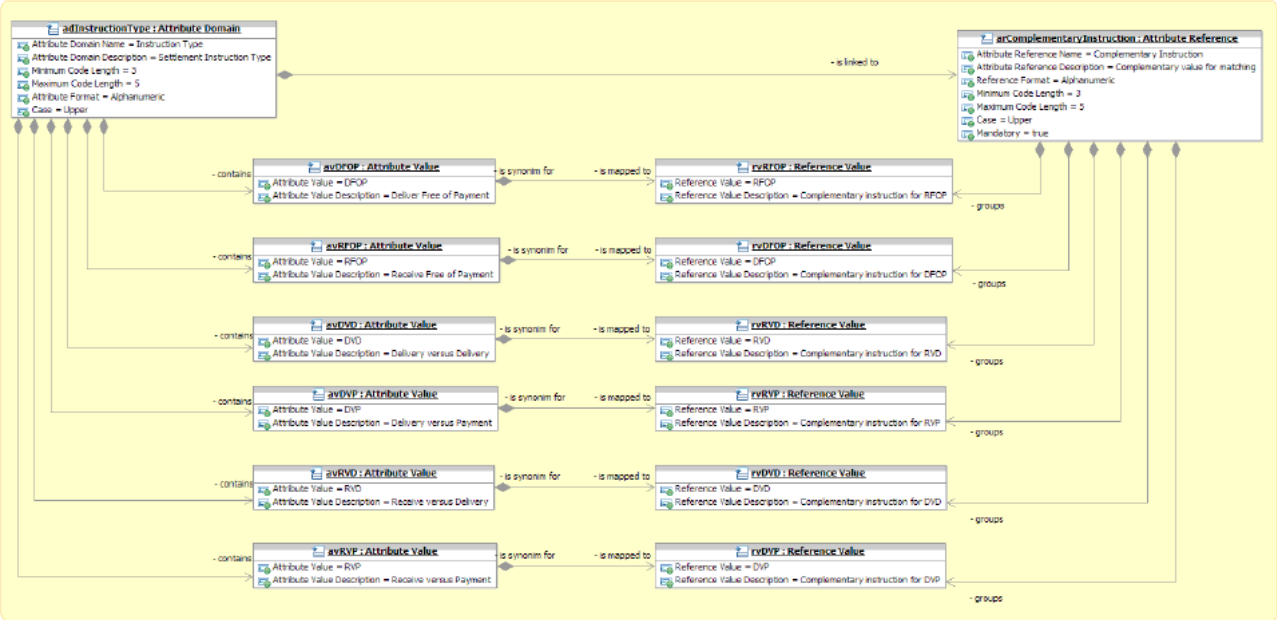
ATTRIBUTE	DESCRIPTION
Reference Value	Value of the attribute reference.
Reference Value Description	Description of the reference value.

Each *References Value* is linked to the relevant *Attribute Reference* and *Attribute Value*.

Reference Id	***SDMG.RPS.ODI.5.1***

All rights reserved.

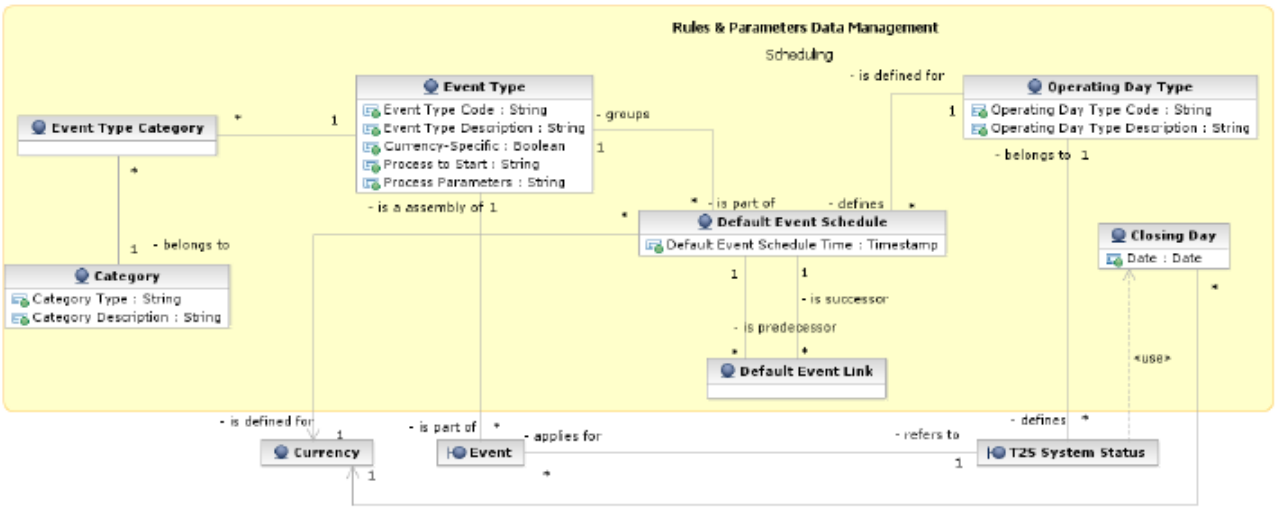
The following object diagram provides an example of the data model for attribute domain management:



3.3.11.8 Scheduling

Reference Id	***SDMG.RPS.DMD.6.1***

The following diagram shows the conceptual data model for scheduling management.



The scheduling of any actual operating day (see the section on *Scheduling* module for more information) is based on a set of pre-defined *Event Types* that can be combined, according to different default schedules, into different *Operating Day Types* **{T2S.11.040}**.

20 – Event Type

Reference Id	***SDMG.RPS.ENT.20.1***

All rights reserved.

This entity includes all the information concerning the *Event Types* defined in T2S **{T2S.03.250} {T2S.03.270} {T2S.03.280} {T2S.03.290} {T2S.03.300}**. An *Event Type* is any possible kind of event in T2S (e.g. EOD, SOD, Cut-off, Beginning of day time and so forth), regardless of its planned or actual triggering time.

ATTRIBUTE	DESCRIPTION
Event Type Description	Description of the event type.
Event Type Code	The unique name of the event type.
Currency-Specific	Boolean attribute specifying whether the event type can be triggered at different times for different currencies.
Process to Start	It specifies the process(es) that must be started when an event of this type is triggered.
Process Parameters	It specifies all the parameters of the process(es) that must be started when an event of this type is triggered, e.g. the identification of the software component(s) to be started, the set of required input parameters, an indication whether a process completed notification is expected, etc. [60]

Currencies, in case the *Event Type* is currency-specific **{T2S.03.303}** [61]. An *Event Type* can be linked to one *Category*.

21 – Category

Reference Id	***SDMG.RPS.ENT.21.1***

This entity includes all the information concerning the entity *Category* defined in T2S. Each *Category* is linked to one or many *Event Types*.

ATTRIBUTE	DESCRIPTION
Category Type	It specifies the type of the category. The exhaustive list of possible value is: ▮ Business report ▮ SD report ▮ Liquidity transfer order report
Category Description	It specifies the description of the category.

22 – Operating Day Type

Reference Id	***SDMG.RPS.ENT.22.1***

This entity includes all the information concerning the *Operating Day Types* defined in T2S, so to allow the T2S operator to define different kinds of operating days in T2S (e.g. to define specific settlement day types in which some *Events* must deviate from the standard schedule, or for testing purposes).

60 The actual data to be stored for this attribute is clarified during the detailed specification phase.

61 The T2S operator specifies each scheduled time via a Default Event Schedule link between the event type, an operating day type and a currency (when applicable).

All rights reserved.

ATTRIBUTE	DESCRIPTION
Operating Day Type Code	Code of the operating day type.
Operating Day Type Description	Description of the operating day type.

The *Operating Day Type* entity has a many-to-many association with the *Event Type* entity, which allows the T2S operator to specify, for each *Event Type* of each *Operating Day Type*, one Default Event Schedule Time or several Default Event Schedule Times (in the case of currency-specific *Event Types*).

23 – Closing Day

Reference Id	***SDMG.RPS.ENT.23.1***

This entity defines the set of *Closing Days* **{T2S.11.180}** for the T2S calendar.

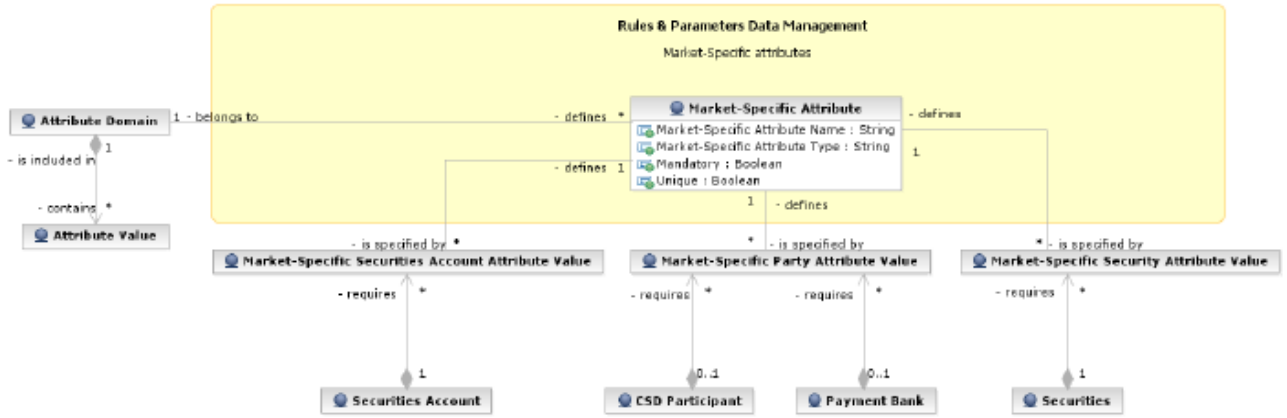
ATTRIBUTE	DESCRIPTION
Date	The specified date is a closing day for the linked currency.

Each occurrence is linked to an occurrence of the Currency entity and defines a specific Closing Day for the linked Currency **{T2S.20.020} {T2S.03.310} {T2S.03.320} {T2S.11.160} {T2S.11.170}**.

3.3.11.9 Market-specific attributes

Reference Id	***SDMG.RPS.DMD.7.1***

The following diagram shows the conceptual data model for market-specific attributes management **{T2S.16.750}**.



It is possible for a CSD to define additional, specific attributes for its *Securities, Parties* and *Securities Accounts*. These attributes give the possibility to add data to *Securities, Parties* and *Securities Accounts* for informational purposes and for the configuration of restriction type (see section 3.3.11.10 [▸ 176]). A CSD can configure and load its own market-specific attributes as part of its reference data migration process to T2S. It is also possible to add further Market-Specific Attributes or changing the configuration of the existing market-specific attributes. When doing so, however, the CSD has to ensure the data integrity of the information stored in T2S. In the same way, a CB can define additional specific attributes for its payment banks.

Only the following validations are possible for these attributes:

All rights reserved.

- Format validation: the value assigned to an attribute is validated against the format definition for that attribute **{T2S.16.780}**;
- Mandatory check: if an attribute is defined as mandatory, each securities, party or securities account for which the attribute is defined must have a value assigned for it **{T2S.16.790}**;
  On this basis:
  - T2S rejects the creation of a new security by the security-maintaining CSD, if it provides no value for a *Market-Specific Attribute* that it has defined as mandatory **{T2S.16.811}**;
  - T2S rejects the creation of a new security-CSD link by an investor CSD, if it provides no value for a *Market-Specific Attribute* that has been defined as mandatory **{T2S.16.812}**;
  - T2S rejects the creation of a new *Securities Account* by a CSD, if it provides no value for a *Market-Specific Attribute* that has been defined as mandatory {**T2S.16.813}**
  - T2S rejects the creation of a new *Party* by a CSD/CB, if it provides no value for a *Market-Specific Attribute* that has been defined as mandatory **{T2S.16.814}**;
  - T2S rejects the creation of a new security by the CSD, if it provides no value for a *Market-Specific Attribute* that it has defined as mandatory **{T2S.16.815}**;
- Uniqueness: if an attribute is defined as unique, each *Securities*, *Party* or *Securities Account* for which the attribute is defined must have a value assigned for it and this value must be unique across all *Securities*, *Parties* or *Securities Account* **{T2S.16.800}**.
- Valid list value: the value assigned to an attribute must belong to a list of valid values for that attribute **{T2S.16.810}**.

24 – Market-Specific Attribute

Reference Id	***SDMG.RPS.ENT.24.1***

This entity includes all the information concerning the market-specific attributes definition.

ATTRIBUTE	DESCRIPTION
Market-Specific Attribute Name	Name of the market-specific attribute.
Market-Specific Attribute Type	Type of the market-specific attribute. The exhaustive list of possible values is as follows: - Security - Party - Securities Account
Mandatory	Boolean attribute specifying whether providing a value for the market-specific attribute type is mandatory.
Unique	Boolean attribute specifying whether the value provided for the market-specific attribute type has to be unique.
Market-Specific Attribute Name	Name of the market-specific attribute.

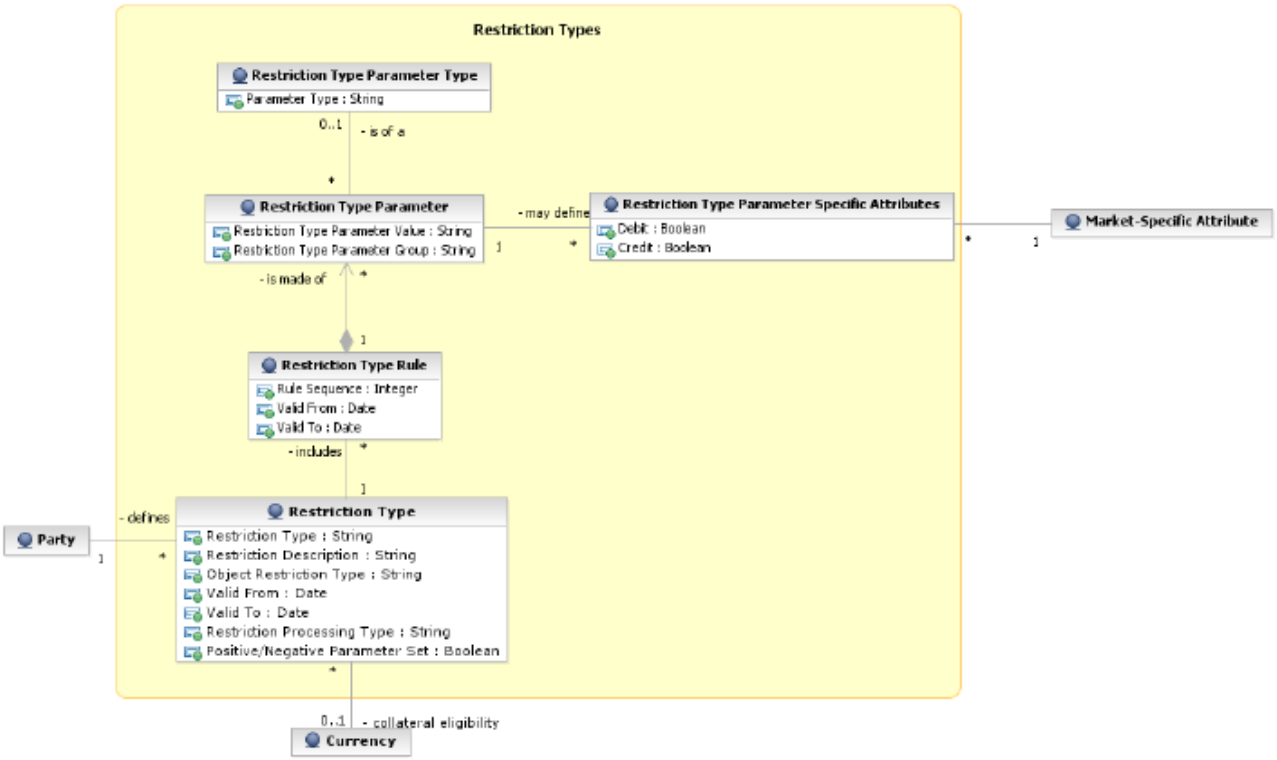
All rights reserved.

Each *Market-Specific Attribute* can be linked to one or many *Securities*, *Parties* or *Securities Accounts* [62], and to the relevant *Attribute Domain* **{T2S.11.420} {T2S.16.760} {T2S.16.770}**.

3.3.11.10 Restriction types

Reference Id	***SDMG.RPS.DMD.8.1***

The following diagram shows the conceptual data model for *Restriction Types* management.



It is possible for CSDs and CBs to define their own restriction types **{T2S.11.670} {T2S.11.680} {T2S.11.690}**. A restriction type is a set of attributes that define the specific processing characteristics for one of the following object types: *System Entities*, *Parties*, *Securities*, *Securities Accounts*, *T2S Dedicated Cash Accounts*, *Settlement Instructions*, *Securities Positions*, *Cash Balances* and *External RTGS Accounts*. T2S supports multiple concurrent restrictions on *Securities*, *Securities Positions*, *T2S Dedicated Cash Accounts*, *Parties*, *Securities Positions* **{T2S.10.030}**, *Cash Balances* and *External RTGS Accounts* **{T2S.16.658} {T2S.16.659}**.

For each *Restriction Type*, optionally, one or many *Restriction Type Rules* define the set of possible actions that can be performed on the restricted object, the set of parties or party types that are allowed or not allowed to perform such actions, and the type of processing T2S shall apply for the restriction type **{T2S.11.661} {T2S.11.693} {T2S.11.694}**. Each *Restriction Type Rule* specifies a matrix of parameters, i.e. a set of groups of parameters, each of which including a list of parameter values for a given set of parameter types **{T2S.11.664}**.

25 – Restriction Type

62 For each link between a market specific attribute and a specific security, party or securities account the value assigned to the security, to the party or to the securities account must be specified.

All rights reserved.

Reference Id	***SDMG.RPS. ENT.25.1***

This entity includes all the information concerning the *Restriction Types*' definition **{T2S.11.651}**. It also includes the harmonised restriction types defined and maintained by the T2S Operator and available to all CSDs and CBs **{T2S.11.666}**.

ATTRIBUTE	DESCRIPTION
Restriction Type	It specifies a code defined by the CSD or the CB to identify the restriction.
Restriction Description	Description of the restriction.
Valid From	▪ It specifies the date from which the restriction type is valid.
Valid To	▪ It specifies the date to which the restriction type is valid.
Object Restriction Type	It specifies a classification for the object type on which the restriction applies. The exhaustive list of possible values is as follows: ▪ Party ▪ Security ▪ Securities Account ▪ Securities Position ▪ T2S Dedicated Cash Account ▪ Cash Balance ▪ External RTGS Account ▪ Settlement Instruction ▪ Intra-Position Movement Instruction ▪ Intra-Balance Movement Instruction
Restriction Processing Type	It specifies a classification for the type of processing T2S shall apply for the restriction **{T2S.11.662}**. The exhaustive list of possible values is as follows: ▪ Rejection: rejection in Settlement Instruction validation **{T2S.05.128}** ▪ CSD Validation Hold: setting the CSD validation status automatically to "Hold" when accepting a Settlement Instruction **{T2S.05.127}** ▪ Reservation: creating a reservation of a Cash Balance or a Securities Position for a specific purpose ▪ Blocking: blocking of a party, Securities Account, Security or T2S Dedicated Cash Account from settlement, blocking of Securities Position or Cash Balance for a specific purpose **{T2S.05.129}** ▪ Balance Type / Earmarking: define and manage position types for Securities Positions. ▪ Earmarking for Auto-collateralisation: specific earmarking for auto-collateralisation

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▪ Deliverable: without any restriction ▪ Collateralised: used for sub-pledged collateral transactions ▪ CoSD Blocking: it does not support any partial execution
Positive / Negative Parameter Set	It specifies whether the rules of the restriction type represent a positive or negative set of parameters. A positive parameter set shall specify the conditions requiring T2S to apply the restriction. A negative parameter set shall specify the conditions for which T2S shall not apply the restriction **{T2S.11.663}**.

Each *Restriction Type* is linked to the specifying Party (i.e. a CSD, a CB or the T2S Operator) and can be linked to one or many *Restriction Type Rules* **{T2S.11.653}**. In addition, in the case of "*Earmarking for Auto-collateralisation*" Restriction Processing Type, it is linked to the relevant *Currency*.

26 – Restriction Type Rule

Reference Id	***SDMG.RPS. ENT.26.1***

This entity defines the *Restriction Type Rules* defined by each CDS or CB or by the T2S Operator **{T2S.11.653}**. Authorised T2S system users can add or delete *Restriction Type Rules* or reorder the sequence of rules within a *Restriction Type* **{T2S.11.664}**.

ATTRIBUTE	DESCRIPTION
Rule Sequence	It specifies the order in which the rules are processed within the relevant rule set.
Valid From	It specifies the date from which the rule is valid.
Valid To	It specifies the date to which the rule is valid.

Each Restriction Type Rule belongs to a single *Restriction Type* and it is linked to one or many *Restriction Type Parameters*.

27 – Restriction Type Parameter

Reference Id	***SDMG.RPS. ENT.27.1***

This entity includes the restriction type parameters defined within each restriction type rule **{T2S.11.654}**. Authorised T2S system users can configure matrices of parameters for their *Restriction Type Rules* by adding, updating or deleting *Restriction Type Parameters* **{T2S.11.664}**.

ATTRIBUTE	DESCRIPTION
Restriction Type Parameter Group	It specifies the group of the restriction type parameter. All the groups within a restriction type rule include the same number of restriction type parameters. A rule is matched when all the restriction type parameters of at least one of its groups are matched.
Restriction Type Parameter Value	It specifies a valid value for the restriction type parameter.

All rights reserved.

Each *Restriction Type Parameter* belongs to a single *Restriction Type Rule* and it is linked either to a specific *Restriction Type Parameter Type or to a Market-Specific Attribute* [63] **{T2S.11.652} {T2S.11.693} {T2S.11.694}**.

28 – Restriction Type Parameter Type

Reference Id	***SDMG.RPS. ENT.28.1***

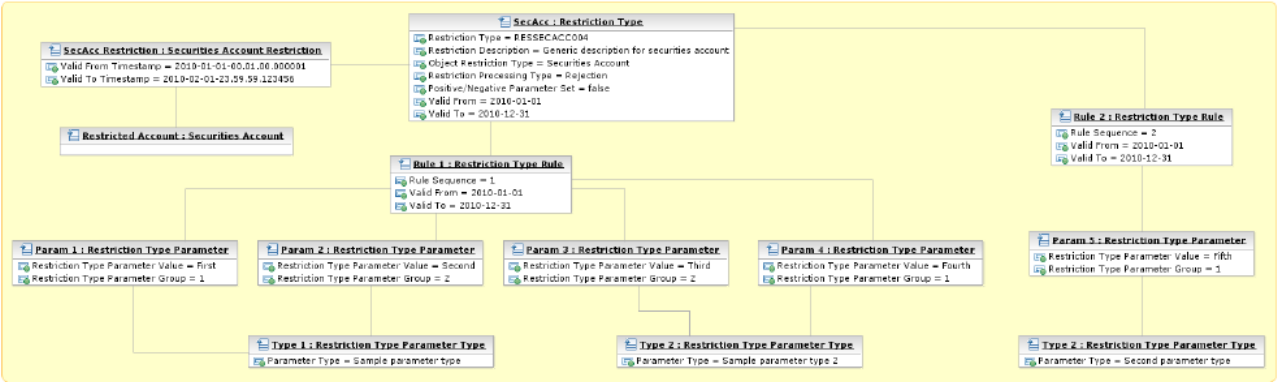
This entity defines all restriction type parameter types **{T2S.11.654}**.

ATTRIBUTE	DESCRIPTION
Parameter Type	It specifies a classification for the restriction type parameter type. The exhaustive list of possible values is as follows: ▪ Securities movement type, whose possible values are RECE (Receive), DELI (Deliver) and ZERO. ▪ Payment, whose possible values are APMT (Against payment) and FREE (Free of payment / Separate payment) ▪ Transaction identification ▪ Party type of the account owner ▪ Party type of the party instructing on behalf of the account owner ▪ Specific party ▪ Security Identifier ▪ Matching status ▪ Debited Position/Balance ▪ Credited Position/Balance ▪ Securities account ▪ T2S dedicated cash account ▪ Credit / Debit indicator ▪ Country of issuance ▪ T2S dedicated cash account type ▪ Settlement currency
Reference Id	***SDMG.RPS.ODI.8.1***

63 For each link to a *Market-Specific Attribute*, two Boolean attributes specify if the relevant parameter value applies to the credit leg, to the debit leg or to both legs of a settlement instruction.

All rights reserved.

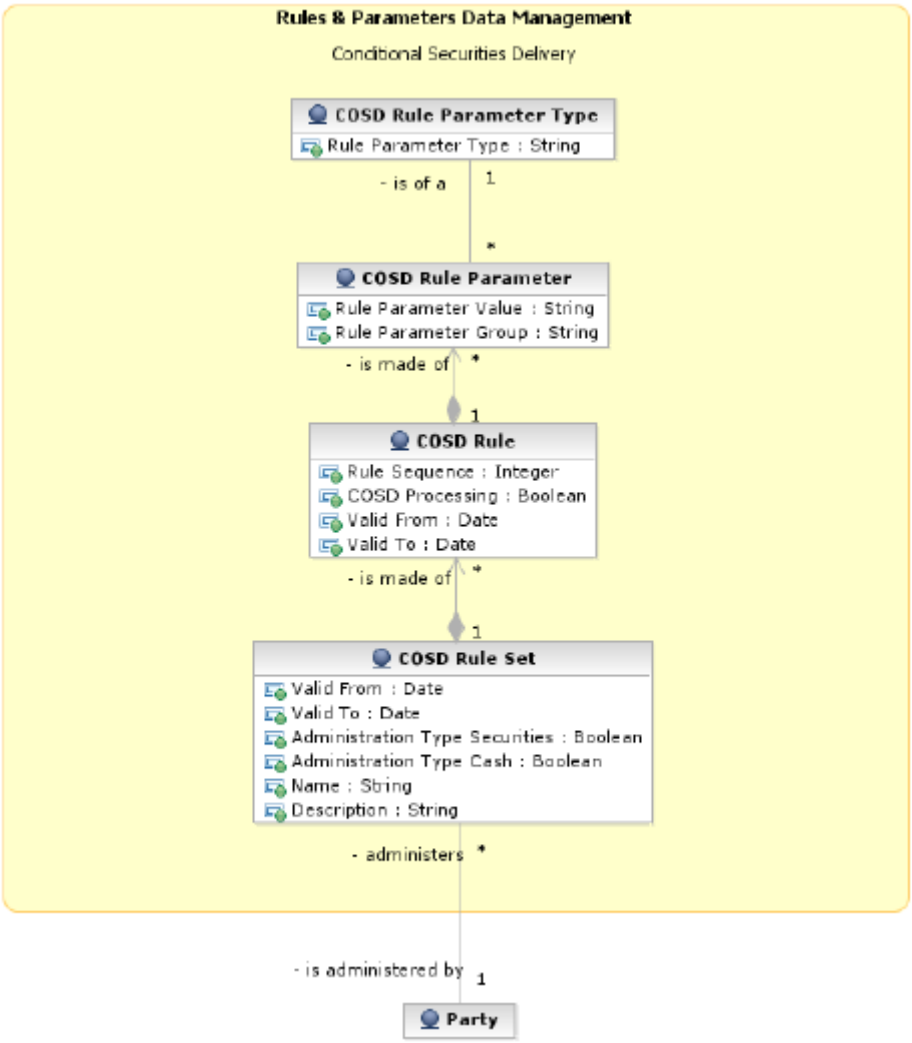
The following object diagram provides an example of the data model for *Restriction Types*:



3.3.11.11 Conditional securities delivery

Reference Id	***SDMG.RPS.DMD.9.1***

The following diagram shows the conceptual data model for *Conditional Securities Delivery* management.



All rights reserved.

Conditional Securities Delivery (CoSD) allows CSDs to set up rules-based, date-dependent configurations of conditional securities delivery **{T2S.11.740}**.

Each CSD can set up its *CoSD Rule Set*. Each *CoSD Rule Set* defines a sequence of *CoSD Rules*. Each *CoSD Rule* specifies the triggering conditions (e.g. settlement currency, transaction type, place of settlement) that have to be met for having a conditional securities delivery and the processing conditions (e.g. securities blocking) that T2S applies in case a conditional securities delivery is triggered.

29 – CoSD Rule Set

Reference Id	***SDMG.RPS.ENT.29.1***

This entity defines the set of CoSD rules defined by each CSD.

ATTRIBUTE	DESCRIPTION
Valid From	It specifies the date from which the rule set is valid.
Valid To	It specifies the date to which the rule set is valid.
Name	The name of the Conditional Securities Delivery Rule Set.
Description	It specifies the description of the Conditional Securities Delivery Rule Set.
Administration Type Securities	It defines whether T2S shall block securities when an instruction is marked for conditional delivery.
Administration Type Cash	It defines whether T2S shall block cash when an instruction is marked for conditional delivery.

Each *CoSD Rule Set* is linked to the relevant CSD and to a set of *CoSD Rules*.

30 – CoSD Rule

Reference Id	***SDMG.RPS.ENT.30.1***

This entity defines the *CoSD Rules* defined by each CSD.

ATTRIBUTE	DESCRIPTION
Rule Sequence	It specifies the order in which the rules are processed within the relevant rule set.
CoSD Processing	Boolean attribute specifying if the settlement instruction that matches with the rule is to settle conditionally.
Valid From	It specifies the date from which the rule is valid.
Valid To	It specifies the date to which the rule is valid.

Each *CoSD Rule* belongs to a single *CoSD Rule Set*, it is linked to a set of *CoSD Rule Parameters* and it may be linked to one or many *CoSD Processing Conditions*.

31 – CoSD Rule Parameter

Reference Id	***SDMG.RPS.ENT.31.1***

This entity includes the CoSD rule parameters defined within each CoSD rule.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Rule Parameter Group	It specifies the group of the rule parameter. All the groups within a CoSD rule include the same number of rule parameters. A rule is matched when all the rule parameters of at least one of its groups are matched.
Rule Parameter Value	It specifies a valid value for the rule parameter

Each *CoSD Rule Parameter* belongs to a single *CoSD Rule* and it is linked to a specific *CoSD Rule Parameter Type*.

32 – CoSD Rule Parameter Type

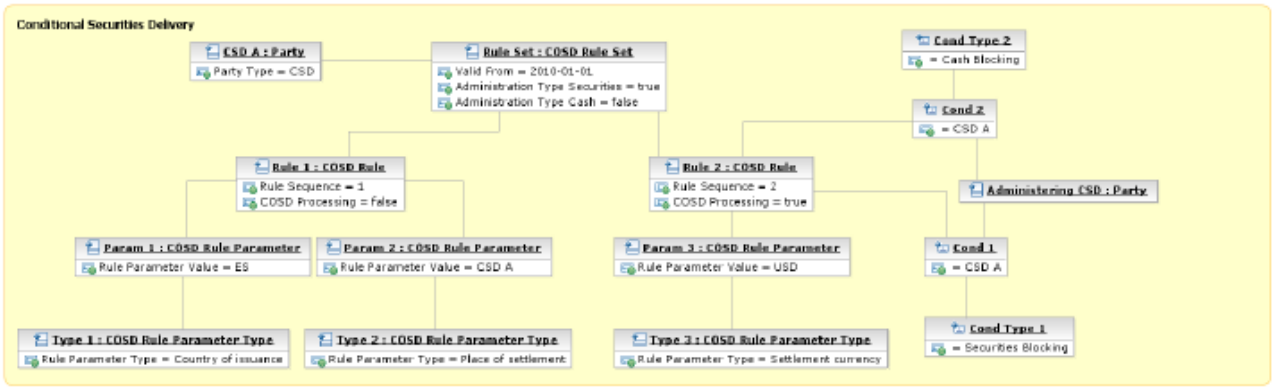
Reference Id	***SDMG.RPS.ENT.32.1***

This entity defines all CoSD rule parameter types **{T2S.13.040}**.

ATTRIBUTE	DESCRIPTION
Rule Parameter Type	It specifies a classification for the CoSD rule parameters. The exhaustive list of possible values is as follows: ▮ ISIN ▮ Settlement currency ▮ CSD ▮ Securities account ▮ Country of issuance ▮ Place of settlement ▮ Transaction type ▮ Issuer CSD in T2S ▮ Delivering CSD in T2S ▮ Receiving CSD in T2S ▮ BIC of issuer CSD in T2S ▮ BIC of delivering CSD in T2S ▮ BIC of receiving CSD in T2S
Reference Id	***SDMG.RPS.ODI.9.1***

All rights reserved.

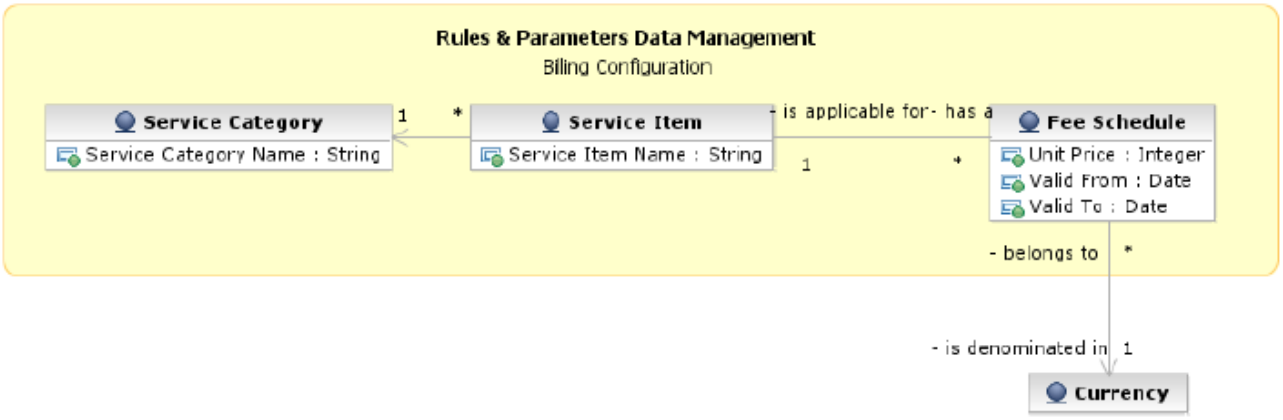
The following object diagram provides an example of the data model for conditional securities delivery:



3.3.11.12 Billing configuration

Reference Id	***SDMG.RPS.DMD.10.1***

The following diagram shows the conceptual data model for *Billing Configuration* **{T2S.15.060}** **{T2S.15.140}**.



Billing Configuration allows the T2S Operator to set up and maintain reference data for the *Billing* module (see section 3.8.5 [▸ 663]).

35 – Service Item

Reference Id	***SDMG.RPS.ENT.35.1***

This entity stores reference data for all billable items of the T2S tariff structure.

ATTRIBUTE	DESCRIPTION
Service Item Name	It specifies a classification for the service items.

Each *Service Item* is linked to all the applicable *VAT Rates* and *Fee Schedules* (see below).

36 – Fee Schedule

Reference Id	***SDMG.RPS.ENT.36.1***

This entity defines *Fee Schedules* for all the *Service Items* defined in T2S, i.e. applicable fees for each range of billable units and each type of *Service Item*.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Valid From	It specifies the date from which the fee schedule is valid.
Valid To	It specifies the date to which the fee schedule is valid.
Unit Price	Price of one billable unit for a given fee schedule.

Each *Fee Schedule* is linked to the relevant *Service Item* and *Currency*.

37 – Service Category

Reference Id	***SDMG.RPS.ENT.37.1***

This entity stores the *Service Categories* applicable for any *Service Items* defined in T2S.

ATTRIBUTE	DESCRIPTION
Service Category Name	It specifies the name of the Service Item Category.

Each *Service Category* is linked to one or several *Service Items*.

3.3.11.13 Other rules and parameters

This section describes all reference data concerning the following rules and parameters:

- Country;
- Currency;
- Limit;
- Partial settlement threshold;
- System entity;
- Tolerance amount;
- T2S BIC directory.

38 – Country

Reference Id	***SDMG.RPS.ENT.38.1***

This entity includes all reference data related to countries defined in T2S.

ATTRIBUTE	DESCRIPTION
Country Code	Numeric code of the country according to the ISO 3166-1 standard.
Country Name	Name of the country according to the ISO 3166-1 standard.

39 – Currency

Reference Id	***SDMG.RPS.ENT.39.1***

All rights reserved.

This entity includes all reference data related to *Currencies* defined in T2S **{T2S.16.320} {T2S.16.330}**. Within T2S, the notion of *Currency* applies to: the currencies eligible for settlement in T2S, the *Currency* in which the cash leg of a *Settlement Instruction* in T2S settles, the currency of the security denomination, and the *Currency* of the *T2S Dedicated Cash Accounts* and limits. T2S stores the currencies eligible for settlement in T2S and the recognised currencies for the settlement of the cash legs outside T2S.

ATTRIBUTE	DESCRIPTION
Currency Code	Unique code of the currency according to the ISO 4217 standard.
Currency Name	Name of the currency.
Number of Decimals	Number of decimals in which the currency is expressed.
T2S Settlement Currency	Boolean attribute specifying whether the currency is eligible for settlement in T2S.

40 – Limit

Reference Id	***SDMG.RPS.ENT.40.1***

This entity includes all reference data related to limits defined in T2S. **{T2S.10.086}** T2S shall allow:

- A CB to define and maintain an auto-collateralisation limit for the maximum net amount of intraday credit a payment bank can get for a T2S dedicated cash account through the collateralisation of securities with the same -CB **{T2S.10.061}**;
- A payment bank to define and maintain an auto-collateralisation limit for the maximum net amount of intraday credit one of its clients can get through the collateralisation of securities with the same payment bank (client collateralisation) for a T2S dedicated cash account from which the client receives liquidity **{T2S.10.062}**;
- A payment bank to define and maintain an external guarantee limit for the cap of credit secured outside T2S that one of its clients can get from the same payment bank for a T2S dedicated cash account from which the client receives liquidity **{T2S.10.063}**;
- A payment bank to define and maintain an unsecured credit limit for the cap of credit unsecured in T2S that one of its clients can get from the same payment bank for a T2S dedicated cash account from which the client receives liquidity **{T2S.10.064}**.

ATTRIBUTE	DESCRIPTION
Limit Type	It specifies a classification for the limit **{T2S.08.630}**. The exhaustive list of possible values is as follows: - External guarantee - Unsecured credit - Auto-collateralisation
Limit Amount	It specifies the limit amount for the credit consumer for the relevant T2S Dedicated cash account. It must be set to zero if the party has no limit for the relevant T2S Dedicated cash account.
Valid From Timestamp	It specifies the date from which the credit limit is valid.

All rights reserved.

Each *Limit* is linked to its relevant *Credit Memorandum Balance* and to the relevant CB or payment bank.

41 – Partial Settlement Threshold

Reference Id	***SDMG.RPS.ENT.41.1***

This entity specifies all reference data required for the definition of partial settlement thresholds in T2S **{T2S.08.290}**. Partial settlement thresholds shall apply for each *Settlement Instruction* eligible for partial settlement. The threshold in quantity for both unit-quoted securities and nominal-quoted securities equals the minimum settlement unit and the settlement unit multiple of the underlying security defined in the securities reference data. Each threshold specifies the minimum cash value and currency (for equity instruments and for debt instruments) that T2S applies to trigger partial settlement.

ATTRIBUTE	DESCRIPTION
Instrument Type	It specifies a classification for the threshold based on the type of instrument. The exhaustive list of possible values is as follows: ▪ Uni-quoted securities ▪ Nominal-quoted securities ▪ Both types are determined by the first character of the ISO10962 Classification of Financial Instruments stored as securities reference data.
Threshold Value	Value of the threshold as an amount in cash.

Each *Partial Settlement Threshold* is linked to the relevant *Currency* **{T2S.11.730}**. The threshold in cash value for uni-quoted securities equals 10,000 EUR. The threshold in cash value for nominal-quoted securities equals 100,000 EUR.

42 – System Entity

Reference Id	***SDMG.RPS.ENT.42.1***

This entity includes all reference data for system entities. System entities define the legal entities (i.e. CSDs, CBs and the T2S Operator) **{T2S.11.110} {T2S.11.120} {T2S.11.130} {T2S.11.140} {T2S.11.150}** by which T2S segregates data **{T2S.11.050} {T2S.11.060} {T2S.11.080} {T2S.11.070}**.

ATTRIBUTE	DESCRIPTION
System Entity Type	It specifies a classification for the system entity. The exhaustive list of possible values is as follows: ▪ T2S Operator ▪ Central Securities Depository (CSD) ▪ National Central Bank (NCB)
System Entity Mnemonic	It specifies a unique short code used to identify the system entity.
System Entity Name	It specifies the full name of the system entity.
Direct Holding CSD	Boolean attribute specifying whether the system entity is a CSD and it operates in a direct holding market.

All rights reserved.

ATTRIBUTE	DESCRIPTION
Direct Holding Technical Offset Account	It specifies the technical offset account that T2S requires for settlement of settlement instructions in a direct holding market.

Every Reference data or dynamic data entity, owned explicitly by a CSD or CB has an association with *System Entity* to inherit the System Entity Identifier attribute. These are:

- *Parties;*
- *Securities Accounts;*
- *T2S Dedicated Cash Accounts;*
- *Settlement Instructions* and other related types of instructions;
- *Securities Positions;*
- *Limits;*
- *Cash Balances;*
- *Cash Postings;*
- *Securities Postings;*
- *Limits;*
- *Party* specific configuration data.

Each *System Entity* is linked to its relevant *Party*, i.e. to the T2S Operator, the CSD or the CB defined as a party in T2S and corresponding to the same system entity.

43 – Tolerance Amount

Reference Id	***SDMG.RPS.ENT.43.1***

This entity defines the set of tolerance amount values for each *Currency* eligible for settlement in T2S **{T2S.11.190}**.

ATTRIBUTE	DESCRIPTION
Cash Value Amount Limit	It specifies the cash value up to (and including) which the tolerance amount is valid.
Tolerance Amount	It specifies the tolerance amount value within the range identified by the cash value amount limit and for the linked currency.
Valid From	It specifies the date from which a given set of tolerance amount values is valid.

The *Tolerance Amount* entity inherits the Currency Identifier attribute from the association to the *Currency* entity to provide the capability to define *Tolerance Amounts* per T2S settlement *Currency*.

44 – T2S BIC Directory

Reference Id	***SDMG.RPS.ENT.44.1***

All rights reserved.

This entity includes all the information needed to identify the legal entities to which SWIFT assigned the BIC **{T2S.11.710}** that is used to validate the input BICs as *Party* and technical address identifier **{T2S.11.700}**. T2S supports the automatic loading and update of the *T2S BIC Directory* based on the BIC Data+ **{T2S.11.720}**. From a *Reference Data Management* domain perspective, this results in the processing of one *Reference Data Maintenance Approval Request* (see section 3.3.2 [108]) per each item of the T2S BIC Directory.

ATTRIBUTE	DESCRIPTION
BIC Source	It specifies a classification for the BIC source. The exhaustive list of possible values is as follows: ▮ Manual input ▮ Automatic loading
BIC Type	It specifies a classification for the BIC type. The exhaustive list of possible values is as follows: ▮ Official BIC ▮ Internal technical BIC
BIC	8-character BIC, consisting of the bank code (financial institution), country code and location code.
BIC Branch Code	3-character branch code for the financial institution.
Financial Institution Name	Three text fields with a length of 35 characters each to store the name of the financial institution.
City Name	35-character name of the city in which the financial institution resides.
Branch Information	Two text fields with a length of 35 characters each to identify the branch of the financial institution.
BIC Technical Identifier	This attribute specifies the unique technical identifier of a BIC in T2S.
Valid From	It specifies the date from which the BIC is valid.
Valid To	It specifies the date to which the BIC is valid.

45 – Error Type

Reference Id	***SDMG.RPS.ENT.45.1***

This entity stores all the error types defined in T2S and the mapping to the relevant ISO 20022 error codes.

ATTRIBUTE	DESCRIPTION
T2S Internal Reason Code	Reason code of the internal T2S error.
ISO 20022 Error Code	ISO 20022 error code corresponding to the T2S error.
ISO 20022 Reason Code	ISO 20022 reason code corresponding to the T2S error.

All rights reserved.

3.3.12 Reference Data Management (SM) Use Cases

Scope

This category of use cases describes the interaction between a T2S system user and the CRDM component.

Criteria

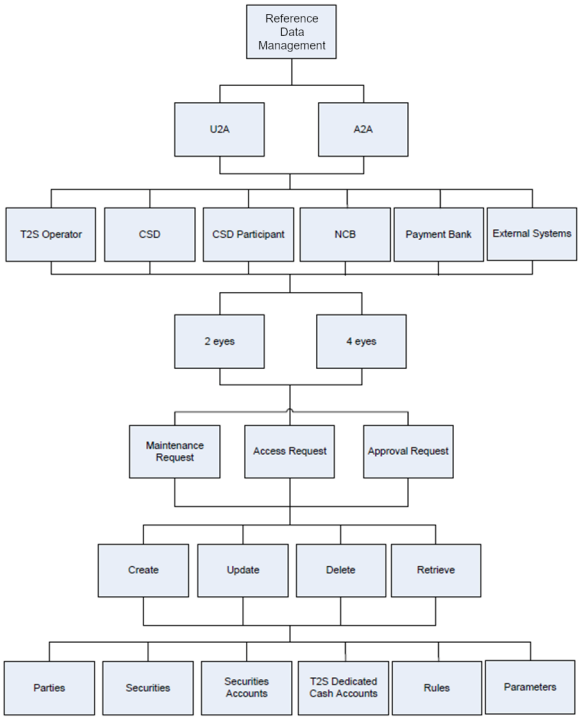
The criteria used to identify the use cases are: the instructing party, the access type, the mode to process the request, the action type and function type of request, and the data type to be considered. The possible criteria values are:

CRITERIA	POSSIBLE VALUES	COMMENT
Instructing Party	T2S Operator, CSD, CSD Participant, CB, Payment Bank, External systems	Type of requestor
Access Type	U2A, A2A	Access to functionality starts from User or from Application
Mode	Two- Eyes, Four-Eyes	Identifies the kind of change approval processes
Action Type	Maintenance Request, Access Request, Approval Request	Type of request to be processed.
Function Type	Create, Update, Delete, Retrieve	Type of functionality
Data Type	Parties, Securities, Securities Accounts, T2S Dedicated Cash Accounts, Rules and Parameters	Type of data to be considered

List of Use Cases

The criteria described above are reported in the following tree:

All rights reserved.



Not all the combinations of values are relevant from a business point of view. The appendix contains the complete use cases for *Reference Data Management*.

3.3.13 Processing of Reference Data Management Use Cases

3.3.13.1 Selection of representative SM use cases

The set of criteria described in the previous section determines the possible use cases for *Reference data management*.

This section specifies the following representative use cases:

- UC-SM-1 Reference Data Maintenance;
- UC-SM-2 Reference Data Approval;
- UC-SM-3 Reference Data Access.

3.3.13.2 Processing of UC-SM-1 Reference Data Maintenance

This use case allows T2S system users to create new and maintain existing Reference data objects according to their specific access rights.

Business assumption

Depending on the type of request, the Reference data object involved and the phase of the settlement day, the CRDM componen processes a *Reference Data Maintenance Request* according to different scenarios.

During the daytime processing, T2S applies a *Reference Data Maintenance Request* in the Two-Eyes principle to a Reference data object immediately. The same principle applies for the night-time processing of such requests unless the requested change updates Reference data information, possibly affecting settle-

All rights reserved.

ment instructions in the ongoing settlement process. *Reference Data Maintenance* sets the approval status of the relevant object to "*Queued*" and, as a consequence, it is not available for processing (see alternative Alt1-SDMRequest and alternative Alt3 in the following sequence diagram).

In addition, if the request relates to a Reference data object that the *Instruction Validation* process of the *LCMM* domain uses, Reference data forwards a *Reference Data Maintenance Notification* to the *Instruction Validation* module of the *LCMM* domain, so that it can trigger a revalidation of the set of all LCMM instructions, possibly affected by the update, against the new data (see option Opt1 in the following sequence diagram). In the case of an update of a limit, CRDM data forwards a *Reference Data Maintenance Notification* to the *Settlement* domain, so that it can check if it must trigger a forced auto-collateralisation reimbursement.

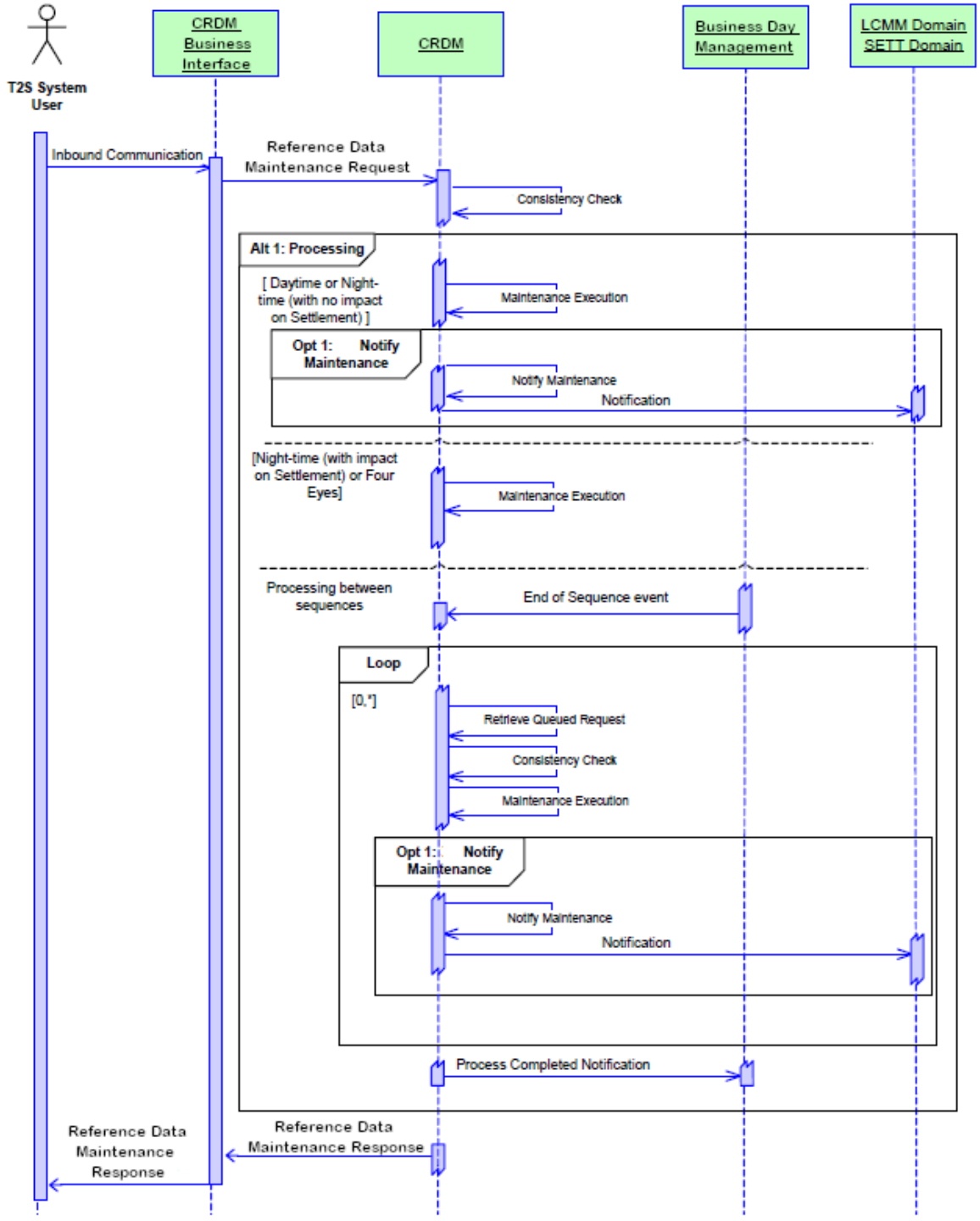
During night-time processing, *Reference Data Maintenance* revalidates all Reference data objects with the approval status "*Queued*" and applies the updates (i.e. their approval status is set to "Approved") at the end of the current sequence and before the beginning of the next sequence (see alternative Alt2 in the following sequence diagram) following a successful revalidation.

The same previously described processing principles also apply in case of a *Reference Data Maintenance Request* according to the Four-Eyes principle. Such a request always is processed immediately (i.e. regardless of the specific request and the type of Reference data object involved), but *Reference Data Maintenance* does not active the requested change until it successfully processes the subsequent *Reference Data Maintenance Approval Request*.

The *consistency check* function rejects *Reference Data Maintenance Requests* unless they meet business validation requirements.

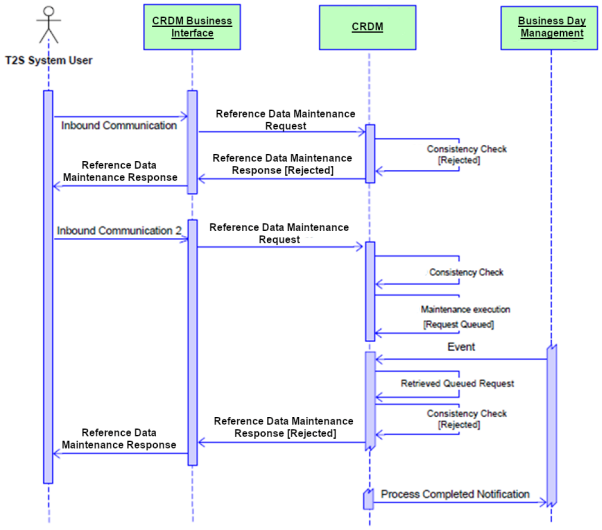
All rights reserved.

Processing (positive scenario)



Processing (negative scenario)

All rights reserved.



3.3.13.3 Processing of UC-SM-2 Reference Data Approval

This use case allows authorised T2S system users to approve or revoke a pending *Reference Data Maintenance Requests* according to their own specific access rights.

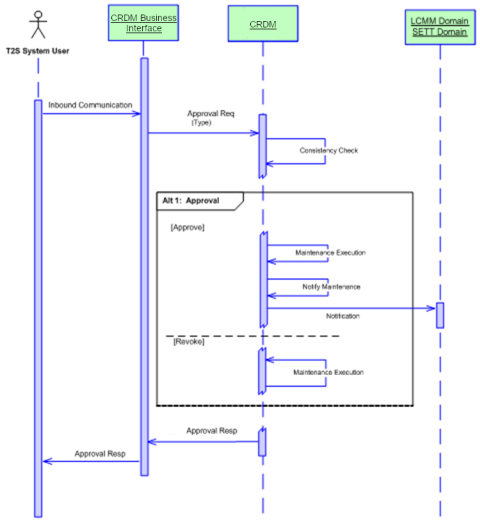
Business assumption

There are two possible scenarios for the *Reference Data Approval Request*. The first scenario represents an approval of a pending Reference data object. The second scenario corresponds to a revocation of a pending Reference data object.

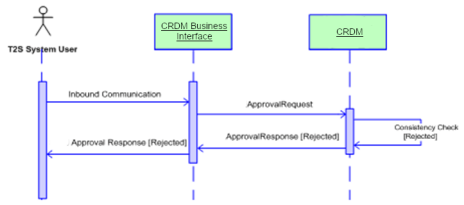
Like *Reference Data Maintenance Requests*, T2S can reject *Reference Data Maintenance Approval Requests* owing to business validation errors, identified in the Consistent Check step of the processing.

Processing

The following sequence diagrams show both possible scenarios:



All rights reserved.



3.3.13.4 Processing of UC-SM-3 Reference Data Access

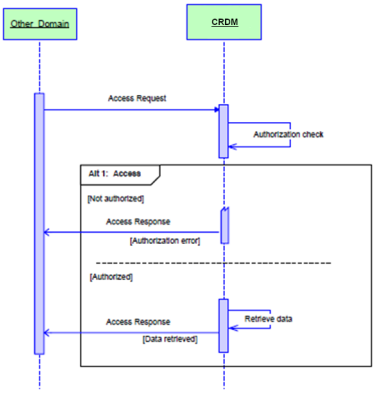
This use case allows authorised T2S system users to access Reference data objects according to their specific access rights.

Business assumption

As far as the Reference data access use case is concerned only one scenario is identified, corresponding to the processing of a generic request of access to one or many Reference data objects.

Processing

The following sequence diagram details the scenario and its two possible (i.e. positive or negative) results:



3.4 Lifecycle Management and Matching

3.4.1 General Introduction

The *Lifecycle Management and Matching* (LCMM) domain is responsible for:

- The business validation of *Settlement Instructions*, maintenance instructions (*Cancellation Instruction, Amendment Instruction, Hold/Release Instruction*) and *Settlement Restrictions* (for reservation, blocking and earmarking);
- The matching of the *Settlement Instructions* (those that need to be matched in T2S), before they can be submitted to the *Settlement* domain;
- The management and execution of maintenance instructions;
- The checking of the possible impact of reference data changes on pending instructions, the revalidation and management of the consequences of such impact when relevant, while keeping tracks of the changes in the lifecycle of instructions

All rights reserved.

- The communication of the status updates and the allegements to the relevant T2S Actors in a consistent way via the *Interface* domain, according to the message subscription service.

The services provided by this domain are available continuously during the whole T2S settlement day, with the exception of the maintenance window.

Additionally, the Lifecycle Management and Matching (LCMM) domain is responsible for:

The detection, calculation, recalculation, maintenance and preparation for reporting of Penalties.

The detection, calculation, recalculation and preparation for reporting of Penalties are triggered by busi-ness events (e.g. Penalties Eligibility Event, Penalties Calculation Event or Penalties Recalculation Event). The service for modifying Penalties is available continuously during the whole T2S settlement day, with the exception of the maintenance window and during the calculation, recalculation and preparation for reporting of Penalties.

The *LCMM* domain comprises Five modules:

- The *Instruction Validation* module (IVA);
- The *Instruction Matching* module (IMH);
- The *Instruction Maintenance* module (IMA);
- The *Status Management* module (STM);
- The Penalty Mechanism module (PEM).

The *Instruction Validation* module checks the consistency of incoming *Inbound LCMM Messages* (Incoming *Settlement Instructions; Settlement Restrictions* and maintenance instructions) sent by a CSD or directly connected T2S Party and LCMM Instructions with T2S Reference Data and Dynamic Data. If an *Inbound LCMM Message*, a *Settlement Instruction* or a *Settlement Restriction* fulfils a Restriction Processing Type, this module is in charge of rejecting the *Inbound LCMM Message* or putting the LCMM Instruction On Hold. Once the instructions are validated, this module stores the outcome of the different validations and reports all errors to the Instructing Parties, if any.

This module is also in charge of the splitting of already matched instructions, of the creation and of the storage of the relevant *Settlement Instructions*, *Settlement Restrictions* and maintenance instructions. It also stores the corresponding *Matching Object* for each pair of matched instructions to be sent, along with the related *Settlement Instructions*, to the *Settlement* domain.

This module is also in charge of revalidating all the instructions at the SOD (Start of Day) of every business day or those instructions that could have been impacted due to a reference data change except those awaiting for approval (Four-Eyes principle). Then, the validated instructions are forwarded to the appropriate subsequent module: *Instruction Matching* module in case the instructions need to be matched in T2S, *Instruction Maintenance* module, or *Settlement* domain, for their further processing.

The *Instruction Matching* module is responsible for matching *Settlement Instructions* that require to be matched in T2S. This module compares the settlement details provided by the buyer and the seller of securities to ensure that both parties agree on the settlement terms of the transaction. Once instructions are matched the *Instruction Matching* module creates the corresponding *Matching Object* for each pair of *Settlement Instructions* matched in T2S, assigning them a T2S Matching Reference. The Match Status of the *Set-*

All rights reserved.

tlement Instruction(s) processed in the module is updated and the *Instruction Matching* module routes each *Matching Object* and the related *Settlement Instructions* to the *Settlement* domain for further processing. If matching fails for the *Settlement instruction*, the *Status Management* module is triggered to send the *Allegement* messages.

The *Instruction Maintenance* module handles maintenance instructions that:

- Cancel, hold, release or partially release a *Settlement Instruction* or amend its process indicator;
- Cancel or release a *Settlement Restriction* or amend its process indicator;
- Release or cancel a *Settlement Instruction* for Conditional Securities Delivery purpose.

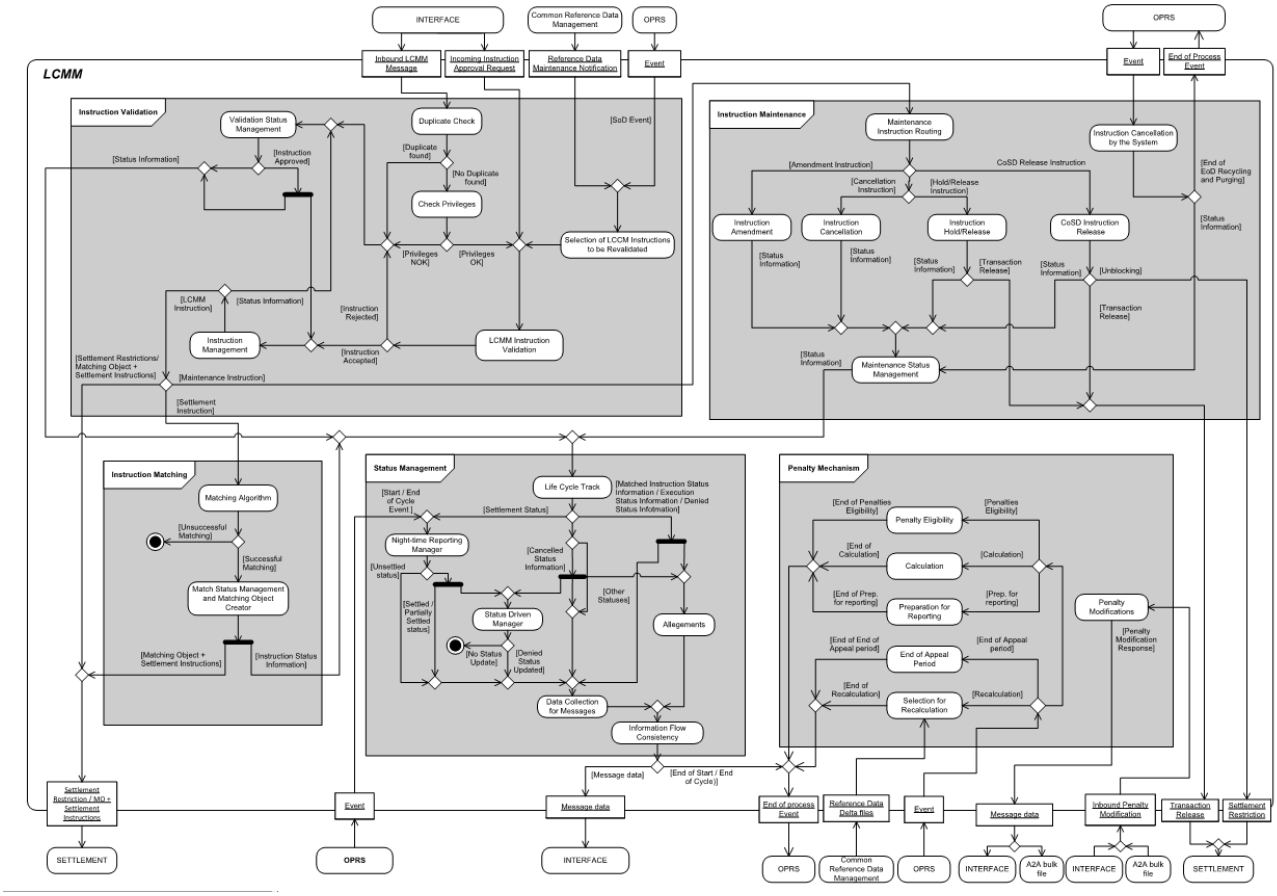
This module is also in charge of the cancellation of unmatched *Settlement Instructions* that remain as such after a standard period beyond their Intended settlement date, or the date of their last status value change, and also those instructions which have reached the end of their recycling period. The module forwards all instruction status value updates to the *Status Management* module.

The *Status Management* module receives status values changes information from other *LCMM* modules and the *Settlement* domain, analyses it, collects the relevant data, and forwards it to the *Interface* domain for transmission to the directly connected T2S Parties and CSDs, as per the message subscription service in a consistent way. All the modules dealing with changes in the status values of the *Settlement Instructions* (and/or the *Settlement Instruction* itself), *Settlement Restrictions* and maintenance instructions, activate this module by providing it with relevant information on the changes carried out as *Instruction Status Information* or *Maintenance Status Information*. This module collects the data for sending *Allegement* messages in case of Unmatched *Settlement Instructions*, when needed to the relevant T2S Parties, via *Interface* domain.

The Penalty Mechanism module is responsible for the daily calculation and preparation for reporting of Penalties for failed settlement instructions as well as the complementary features, which are necessary for T2S Actors in this context. Every business day, the Penalty Mechanism module analyses all the failed Settlement Instructions from previous business day in order to compute the necessary Penalties. This module also

All rights reserved.

handles requests from CSDs to modify an existing Penalty, as well as the recalculation of Penalties that have been modified or impacted by a reference data change. This module prepares the information, including the aggregation of amounts (daily or monthly), for the ad-hoc reports for Penalties.



3.4.2 Dynamic data managed by the domain

3.4.2.1 List of dynamic data managed by the LCMM domain

The LCMM domain manages data related to:

- LCMM Instructions: Those whose purpose is instructing the receipt or delivery of securities with or without payment, the movement of securities within a Securities Account from one position to another, the move-ment of cash within a Dedicated Cash Account from one balance to another, as well as those whose pur-pose is the cancelation or modification of one of the former instructions.

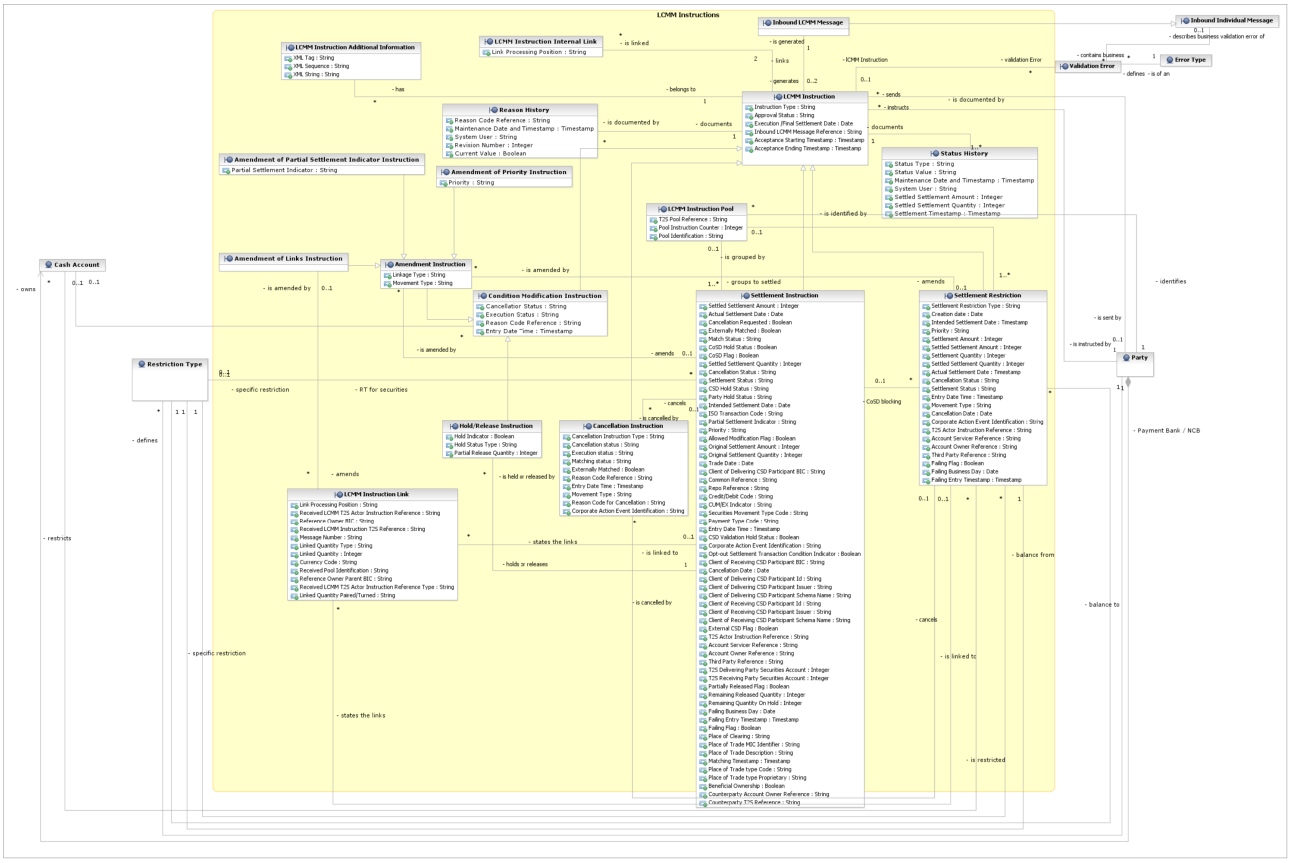
The *LCMM Instructions* managed by the domain and described below in this chapter are the following:

- *Settlement Instructions;*
- *Settlement Restrictions;*
- *Cancellation Instructions;*
- *Amendment Instructions;*
- *Hold/Release Instructions.*

All rights reserved.

- Penalties: Dynamic data for the Penalty Mechanism needed for the detection, calculation, recalculation, management and reporting of Penalties. For the definitions of the attributes of the entities described in the following chapter, please refer to the HTML data model.

3.4.2.2 Description of the data related to LCMM Instructions



Inbound LCMM Message

This entity is the generalisation for the messages that enter into the *LCMM* domain, which the *Inbound Individual Message* class stores. The *Interface* data model provides the detailed specification of this entity.

Validation Error

Validation Error represents an association in the *Interface* domain between an *Inbound LCMM Message* (*Inbound Individual Message*) or the *LCMM Instruction* with the Error Type that specifies one or several errors that LCMM Validation detected during the business validation of an *Inbound LCMM Message* or during the re-validation of a *LCMM Instruction.*

LCMM Instruction

This entity is the generalisation for the following classes related to securities settlement ISO messages:

- *Settlement Instruction;*
- *Settlement Restriction;*
- *Cancellation Instructions;*
- *Condition Modification Instructions.*

All rights reserved.

It specifies all attributes that are common to all *LCMM Instructions*.

This entity is linked to:

- The *Status History* that tracks the statuses during the life cycle of the *LCMM Instruction*;
- The *Reason History*, that tracks the reason codes associated to settlement attempts or non execution of the *LCMM Instruction*;
- The *Party* by the following relationships:
  - Instructing Party: T2S Party who instructs the *LCMM Instruction*,
  - Sending Party: T2S Party who sends the *LCMM Instruction*;
- The *LCMM Instruction Additional Information* that contains the information included in the message not needed for settlement.
- The *LCMM Instruction Internal Link* that contains the information related to the linked instructions.

In case of unsuccessful business revalidation, the *LCMM Instruction* may be linked to the *Validation Error* entity.

LCMM Instruction Additional Information

This entity includes the information of a *LCMM Instruction* compliant with the ISO standard 20022 that is not needed for its processing in T2S and it is not stored in the *LCMM Instruction* entity with the same level of granularity in a structured way.

Each LCMM Additional Information stores blocks of the *Inbound LCMM Message* not stored in the LCMM Instruction data, avoiding redundancies, specifying an XML Tag and an XML Sequence so that the location of the XML String stored in the entity could be recognised against the message schema.

Each *LCMM Instruction Additional Information* is linked to the *LCMM Instruction* it belongs to.

Status History

This entity documents the audit trail history for all changes for all types of processing statuses, managed in the full life cycle of an *LCMM Instruction* that *LCMM* processes [64].

Each *Status History* belongs to a single *LCMM Instruction*.

Reason History

This entity includes the reason codes related to the unsuccessful attempts of settlement of a *Settlement Instruction* or a *Settlement Restriction*, and the reason codes associated to the pending execution of a *Cancellation Instruction*.

Each *Reason History* is linked to one *LCMM Instruction*.

LCMM Instruction Link

This entity specifies the information that allows T2S Actors to link one or more instructions with each other for settlement. The entity supports links for the following types of instructions:

- *Settlement Instruction;*

64 Statuses of *LCMM Instructions* are described in the *Status Management* module of the LCMM domain.

All rights reserved.

- *Settlement Restriction;*
- *Condition Modification Instruction* for amendments of links.

LCMM Instruction Pool

This entity groups the *LCMM Instructions* linked with a common pool reference.

Each *LCMM Instruction Pool* is linked to the *Settlement Instruction* or *Settlement Restriction* that stated the link with a common pool reference and to the Party that sent the instructions included in the pool.

LCMM Instruction Internal Link

This entity stores one link for each pair of linked *LCMM Instructions* (*Settlement Instructions* or *Settlement Restrictions*) as stated in the latest created in T2S.

Upon creation of a *LCMM Instruction*, *LCMM* domain creates one *LCMM Instruction Internal Link* per each Link Processing Position included in the *Inbound LCMM Message*, when such Link Processing Position links the new *LCMM Instruction* with another already existing in the system.

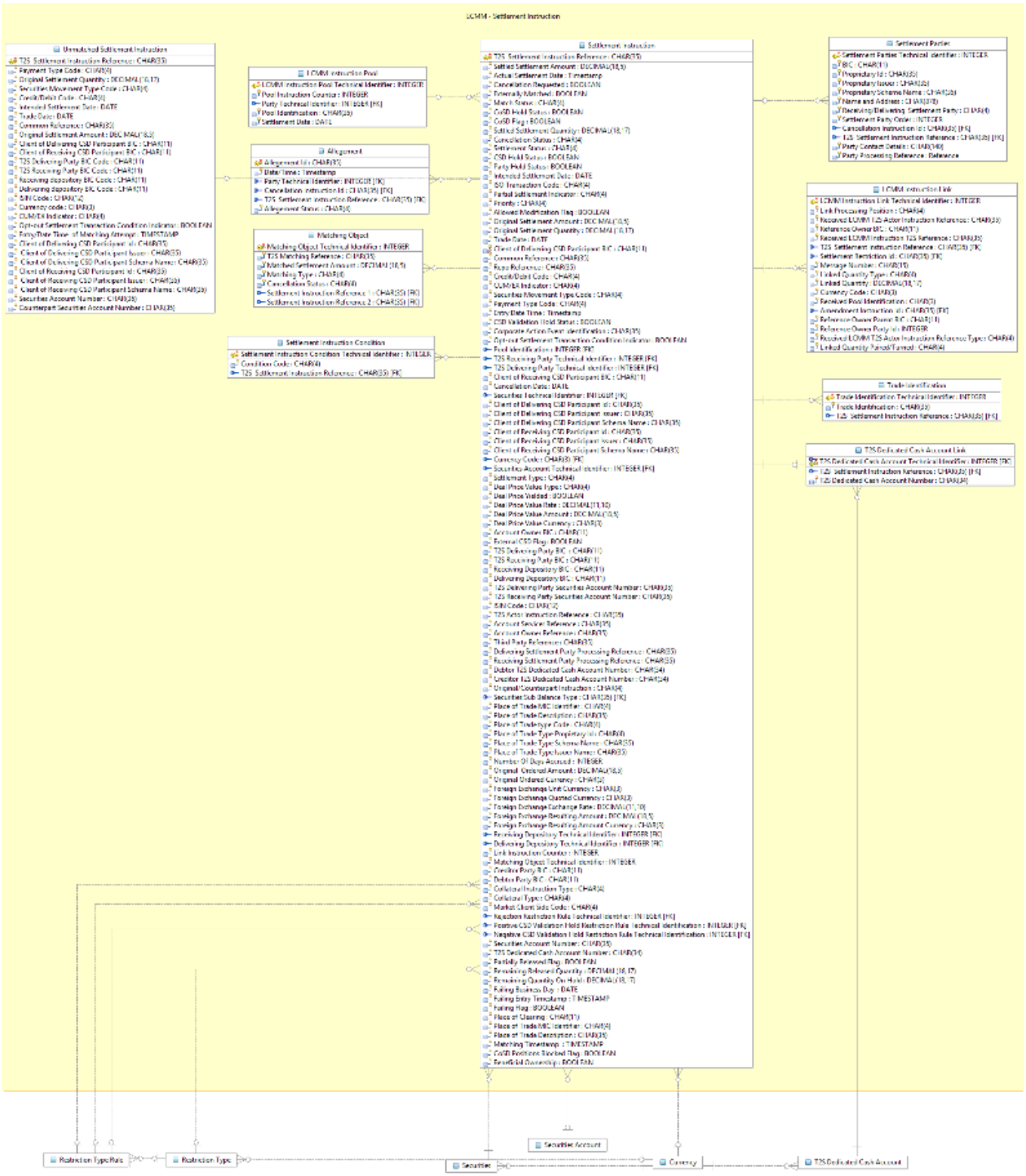
Upon amendment of the links of a *LCMM Instruction*, *LCMM* domain creates one *LCMM Instruction Internal Link* per each new Link Processing Position included in the modified *LCMM Instruction* when such new Link Processing Position links the modified *LCMM Instruction* with another already existing in the system between.

Upon amendment of the links of a *LCMM Instruction*, *LCMM* domain removes the *LCMM Instruction Internal Link* related to the dropped Link Processing Position removed in the modified *LCMM Instruction*.

Each *LCMM Instruction Internal Link* is linked to the two pair of linked *LCMM Instructions*.

All rights reserved.

3.4.2.3 Description of the data related to Settlement Instructions



Settlement Instruction

This entity models the specific data of a Settlement Instruction Transaction Message. It inherits the attributes from the generalisation of *LCMM Instruction*. The entity includes de-normalised status information attributes to ensure efficient retrieval of the *Settlement Instructions*, based on their current status.

The following table maps the acronyms in URD and GFS, as document in the first column, with to the ISO 20022 attributes in the *Settlement Instruction* entity:

All rights reserved.

URD / GFS DESCRIPTION	PAYMENT TYPE CODE	SECURITIES MOVEMENT TYPE CODE	SETTLEMENT QUANTITY	CREDIT / DEBIT CODE
DVP	APMT	DELI	Above zero	CRDT
RVP	APMT	RECE	Above zero	DBIT
PFoD-CRDT	APMT	DELI	Zero	CRDT
PFoD-DBIT	APMT	RECE	Zero	DBIT
DFoP	FREE	DELI	Any	CRDT/DBIT/no value
RFoP	FREE	RECE	Any	CRDT/DBIT/no value
DWP	APMT	DELI	Above zero	DBIT
RWP	APMT	RECE	Above zero	CRDT

T2S allows the indication of a corporate action reference in a *Settlement Instruction*, if the *Settlement Instruction* relates to a corporate action **{T2S.05.235}**.

Each *Settlement Instruction* is linked to:

- The *Party* entity by the following relationships:
  - T2S Delivering Party (the Party that delivers the securities (cash in case of a PFOD) in a *Settlement Instruction*),
  - T2S Receiving Party (the Party that receives the securities (cash in case of a PFOD) in a *Settlement Instruction*),
  - Delivering Depository (the CSD where the T2S Delivering Party has the securities account),
  - Receiving Depository (the CSD where the T2S Receiving Party has the securities account);
- The applicable *T2S Dedicated Cash Account*, when specified by the T2S Actor in the *Inbound LCMM Message* in case of *Settlement Instructions* against payment (Payment Type Code is APMT), through the *T2S Dedicated Account Link* entity;
- The applicable *Securities Account;*
- The applicable *Securities;*
- The relevant *Currency*, in case of *Settlement Instructions* against payment (Payment Type Code is APMT);
- The related *Securities Postings* and *Cash Postings* resulting of settlement;
- The *Matching Object* resulting of matching;
- If the *Settlement Instruction* is identified as a CoSD *Settlement Instruction*:
  - At least one *CoSD Rule* entity,
  - The *Settlement Restriction* entity;
- The *Unmatched Settlement Instructions*, in case the instruction needs to be matched whereas it remains unmatched;
- *Allegement*, when a settlement allegement is created;

All rights reserved.

- The *Restriction Type*, by the following relationships:
  - To identify, according to the securities movement type, either the sub-balance to which the securities are received, or the sub-balance from which the securities are delivered,
  - When the *Settlement Instruction* fulfils the condition stored in that entity;
- The *Restriction Reference* entity, if the *Settlement Instruction* states the use of an already restricted position.
- *LCMM Instruction Link*, when the Settlement *Instruction* has links with other *Settlement Instructions* or *Settlement Restrictions*;
- *LCMM Instruction Pool*, when the *Settlement Instruction* has a collective reference identifying a pool of *Settlement Instructions* or *Settlement Restrictions* to be settled together.

Finally, each *Settlement Instruction* may be linked to maintenance instructions that try to or have updated it:

- *Cancellation Instruction*;
- *Amendment Instruction*;
- *Hold/Release Instruction.*

T2S Dedicated Cash Account Link

This entity specifies the link between each *Settlement Instruction* and the applicable *T2S Dedicated Cash Account*.

LCMM creates a new *T2S Dedicated Cash Account Link* for each accepted *Settlement Instruction* against payment (Payment Type Code is "APMT"), where the *T2S Dedicated Cash Account* was specified by the T2S Actor in the *Inbound LCMM Message*; this *T2S Dedicated Cash Account Link* contains the *T2S Dedicated Cash Account* stated in the message.

Each *T2S Dedicated Cash Account Link* belongs to only one *Settlement Instruction* and to only one *T2S Dedicated Cash Account*.

Unmatched Settlement Instructions

This entity is used as repository to store the values of matching fields from unmatched *Settlement Instructions* in order to optimise the matching process. *LCMM* domain creates an occurrence in the *Unmatched Settlement Instructions* entity when an *Unmatched Settlement Instruction* cannot be matched and deletes it when such *Unmatched Settlement Instruction* is matched.

Any instance of an *Unmatched Settlement Instruction* entity is linked to its *Settlement Instruction*.

Allegement

This entity includes the information of the *Allegement message* sent to advise an account owner that counterparty has alleged a *Settlement Instruction* or a cancellation of *Settlement Instruction* against it, as T2S could not find a corresponding instruction by the account owner. The entity also reflects whether the Allegement message has been already sent, removed (due to an Allegement removal) or Cancelled (due to an Allegement Cancellation)

All rights reserved.

The *Allegement* class has a relationship to either the *Settlement Instruction* entity or *Cancellation Instruction* entity to identify the instruction that originated the allegement. It has a relationship to the *Party* class to identify the recipient of the allegement.

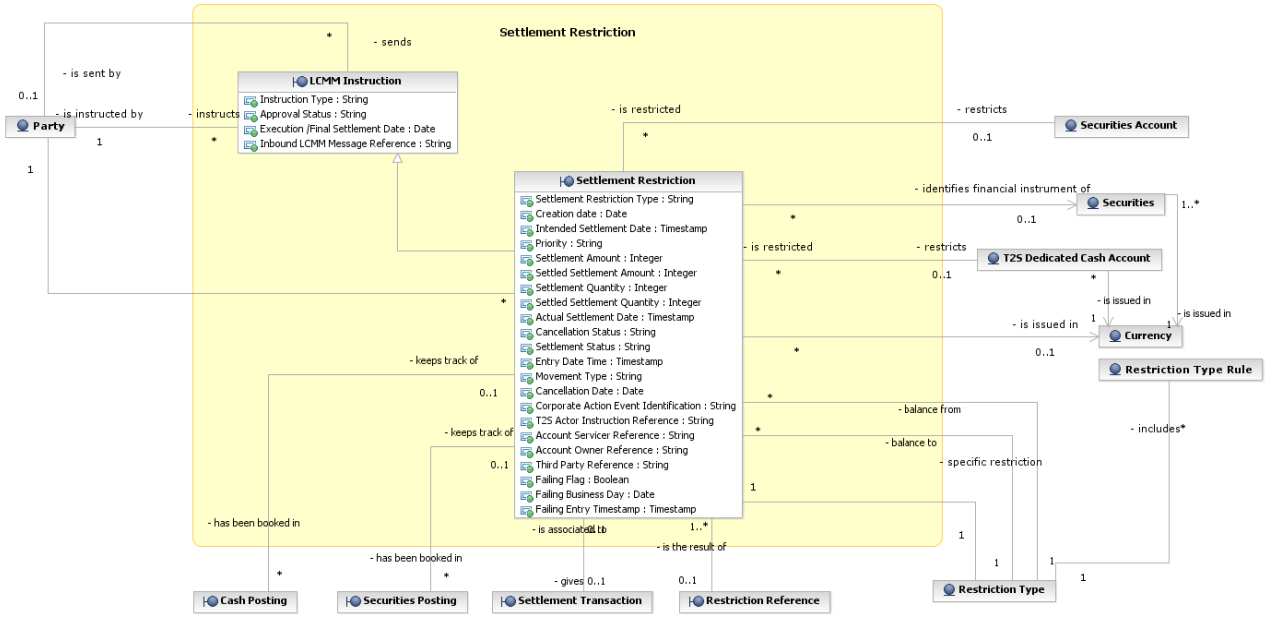
Matching Object

This entity stores one *Matching Object* for each pair of matched *Settlement Instructions* regardless of whether they are received already matched or match inside T2S System.

Each *Matching Object* is linked to the two matched *Settlement Instructions* that belong to it, and the *Settlement Transaction* result of settlement.

When T2S creates T2S generated *Settlement Instructions* in case of cross-CSD instructions or auto-collateralisation, the T2S Matching Reference is the same for all *Matching Objects* linked to these T2S generated *Settlement Instructions* and for the *Matching Object* linked to the original *Settlement Instructions* received from the T2S Actors.

3.4.2.4 Description of the data related to Settlement Restrictions



Settlement Restriction

This entity models the specific data of an Intra-Position Movement Instruction Message or an Intra-Balance Movement Instruction Message. It inherits the attributes from the generalisation of *LCMM Instruction*. The entity includes de-normalised status information attributes to ensure efficient retrieval of the Settlement Restrictions, based on their current status.

Each *Settlement Restriction* is linked to:

- The *Restriction Type*, by the following relationships:
  - "Position from", to identify the type of sub-balance from which the securities are moving;
  - "Position to", to identify the type of sub-balance to which the securities are moving.

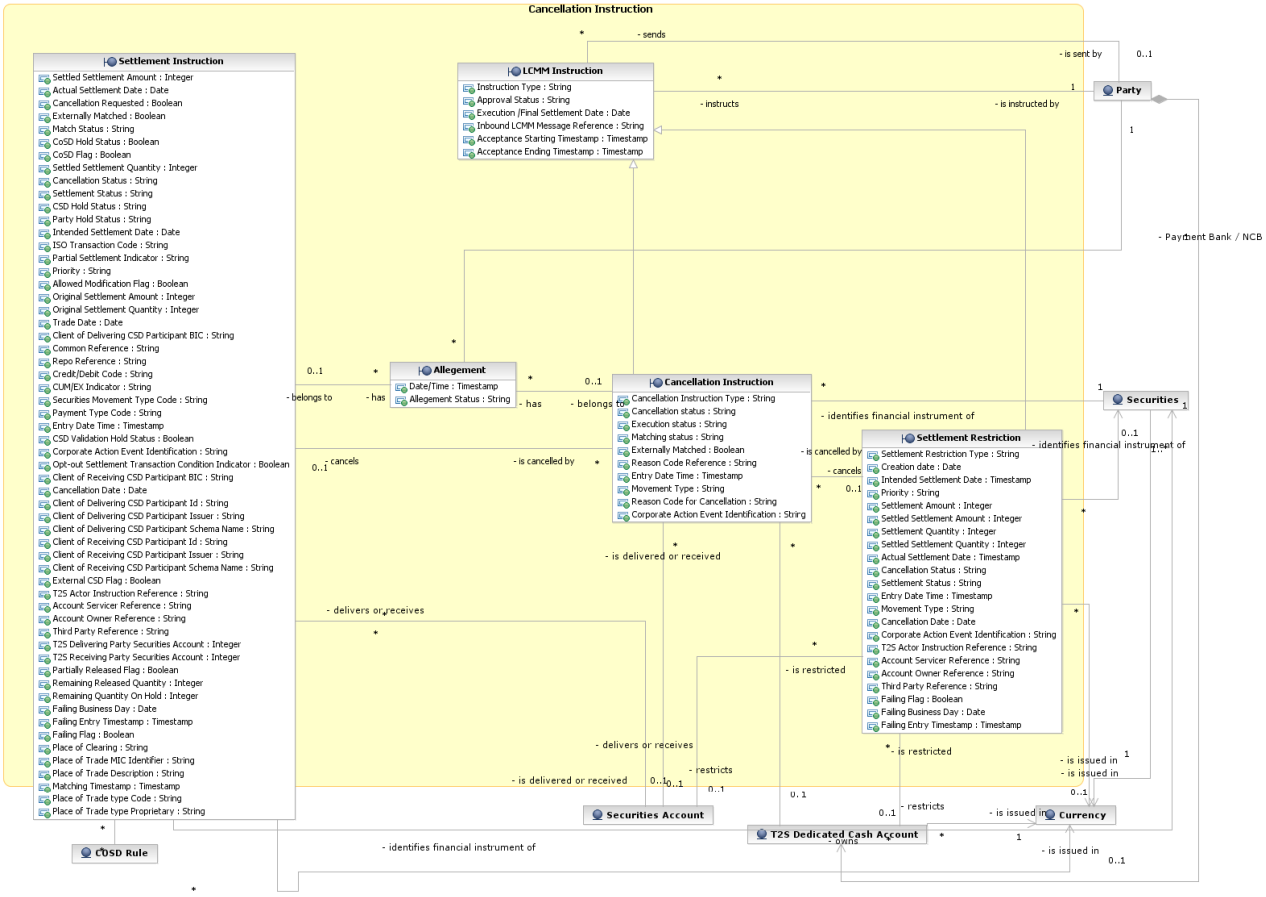
All rights reserved.

– When the *Settlement Restriction* fulfils the condition stored in that entity.

- The restricted *T2S Dedicated Cash Account*, in case of *Settlement Restriction* on Cash (from an Intra-Balance Movement Instruction Message);
- The relevant *Currency*, in case of *Settlement Restriction* on Cash (from an Intra-Balance Movement Instruction Message);
- The restricted *Securities Account*, in case of *Settlement Restriction* on Securities (from an Intra-Position Movement Instruction Message);
- The applicable *Securities* that identifies the financial instrument, in case of *Settlement Restriction* on Securities (from an Intra-Position Movement Instruction Message);
- The *Settlement Transaction* result of settlement;
- *Securities Postings* or *Cash Postings* resulting of settlement;
- The *Restriction Reference* result of settlement in case of blocking or reservation.
- The *Party* entity by the relationship "Depository" (the CSD where the T2S Party has the securities account in case of a Securities *Settlement Restriction*);
- *LCMM Instruction Link*, when the *Settlement Restriction* has links with other instructions;
- *Instruction Pool*, when the *Settlement Restriction* has a collective reference identifying a pool of *Settlement Instructions* or *Settlement Restrictions* to be settled together.

All rights reserved.

3.4.2.5 Description of the data related to Cancellation Instructions



Cancellation Instruction

This entity includes the information of a single maintenance request sent to cancel an existing *Settlement Instruction* or *Settlement Restriction*.

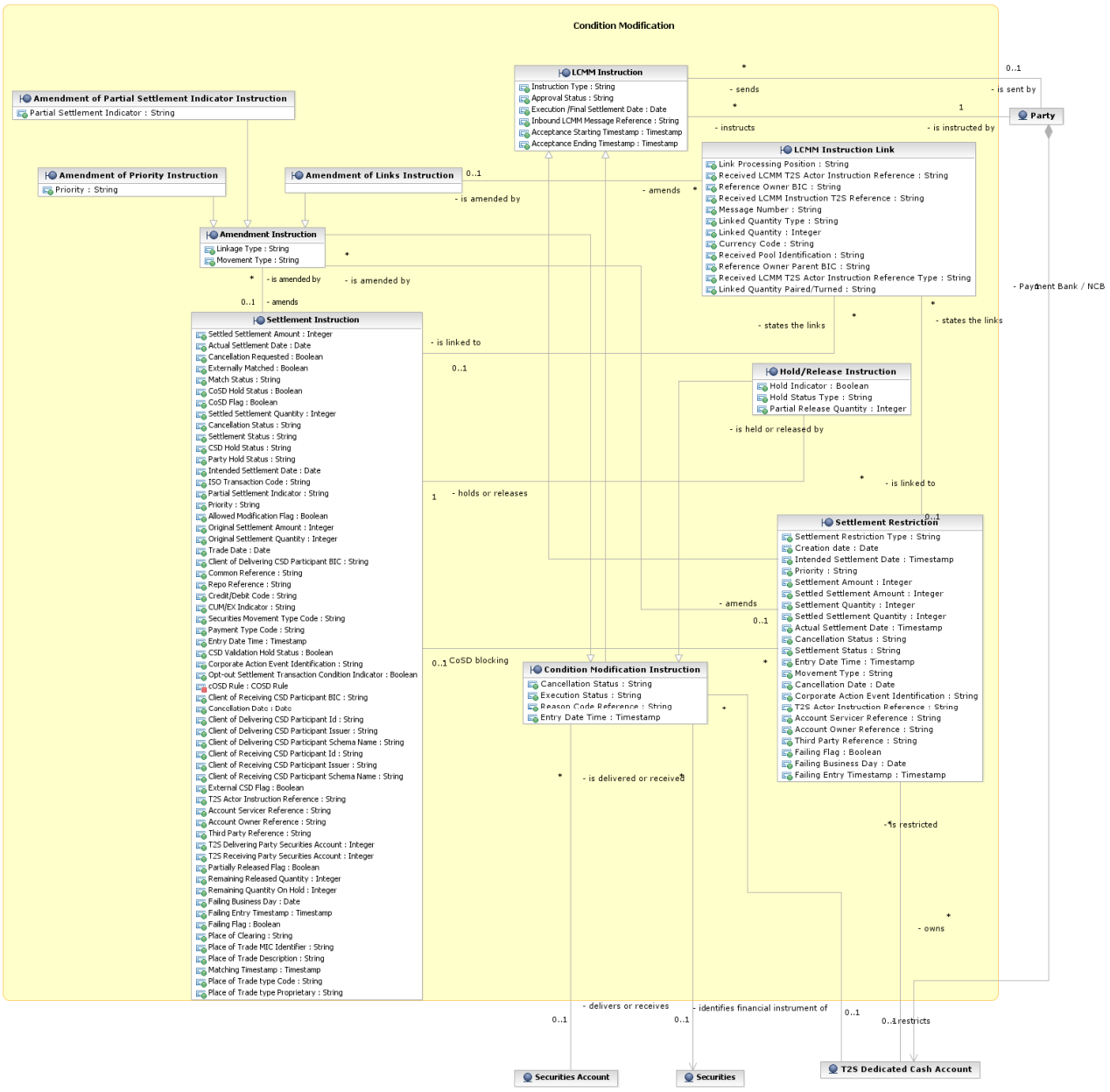
The *Cancellation Instruction* entity extends the information included in *LCMM Instruction* entity.

Each *Cancellation Instruction* is linked to the following entities:

- The applicable Securities Account;
- The applicable Securities that identifies the financial instrument involved in the instruction on which the Cancellation Instruction should apply;
- The Settlement Instruction or Settlement Restriction it is trying to cancel;
- Allegement, when a cancellation allegement is created;

All rights reserved.

3.4.2.6 Description of the data related to Condition Modification Instruction (Hold/Release and Amendment Instructions)



Condition Modification Instruction

The *Condition Modification Instruction* entity models the specific data of a Condition Modification Instruction Message. It inherits the attributes from the generalisation of *LCMM Instruction*. The entity includes de-normalised status information attributes to ensure efficient retrieval of the *Condition Modification Instructions*, based on their current status.

Each *Condition Modification Instruction* is linked to the following entities:

The applicable *Securities Account;*

The applicable *Securities* that identifies the financial instrument involved in the instruction on which the *Condition Modification Instruction* should apply.

All rights reserved.

Amendment Instruction

This entity stores the specific data corresponding to an *Amendment Instruction,* i.e. a *Condition Modification Instruction* for amending the process indicators of a *Settlement Instruction* or a *Settlement Restriction*.

The *Amendment Instruction* entity inherits the attributes from the generalisation of *Condition Modification Instruction*.

Each *Amendment Instruction* is linked to the *Settlement Instruction or Settlement Restriction* it is trying to amend.

Amendment of Partial Settlement Indicator Instruction

This entity stores the specific data corresponding to an *Amendment Instruction* that intends to modify the Partial Settlement Indicator of a *Settlement Instruction*. It inherits the attributes from the generalisation of *Amendment Instruction*.

Amendment of Priority Instruction

This entity stores the specific data corresponding to an *Amendment Instruction* that intends to modify the Priority *of a Settlement Instruction* or *Settlement Restriction*. It inherits the attributes from the generalisation of *Amendment Instruction*.

Amendment of Links Instruction

This entity stores the specific data corresponding to an *Amendment Instruction* that intends to modify the linkage information of a *Settlement Instruction* or *Settlement Restriction*. It inherits the attributes from the generalisation of *Amendment Instruction*.

Each *Amendment of Links Instruction* is linked to the *LCMM Instruction Link* entity.

Hold/Release Instruction

This entity includes the information of a single maintenance request sent to hold, release or partially release an existing *Settlement Instruction* or to release a *Settlement Restriction*.

The *Hold/Release Instruction* entity inherits the attributes from the generalisation of *Condition Modification Instruction*.

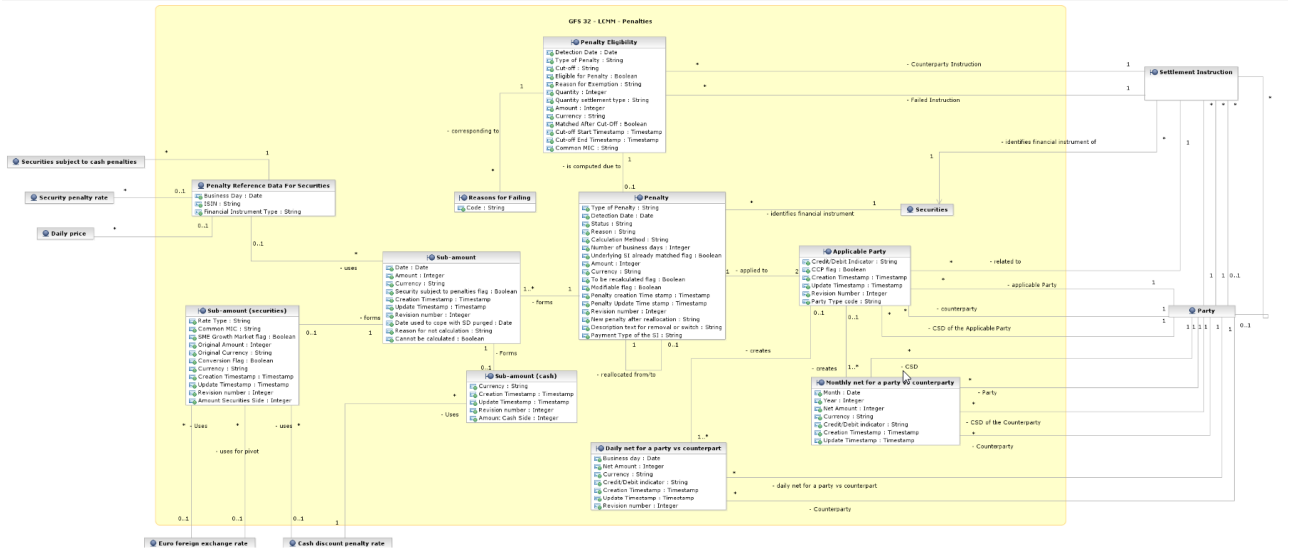
Each *Hold/Release Instruction* is linked to the *Settlement Instruction* it is trying to hold, release or partially release or the *Settlement Restriction* it is trying to release.

When the *Hold/Release Instruction* has Hold Status Type "CoSD", it is linked to the applicable *CoSD Rule* as well.

All rights reserved.

3.4.2.7 Description of the data related to Penalties

The following diagram shows the conceptual data model for the Penalty detection and the Penalty preparation for calculation and re-calculation.



Penalty Eligibility

This entity includes the information needed to identify whether a failed Settlement Instruction is eligible for a penalty or not, depending on its ISO transaction code and, also: the reason(s) for settlement failure in case of Settlement Fail Penalties (SEFPs), or the Acceptance Timestamps in case of Late Matching Fail Penalties (LMFPs).

Each Penalty Eligibility instance is linked to the following entities:

- Settlement Instruction entity whose possible penalisation is analysed, for the following relationships: failed instruction and counterparty instruction;
- Penalty, when the value of the attribute Eligible for a Penalty is "True";
- Reason(s) for Failing of the underlying Settlement Instruction, in case of being eligible for a Settlement Fail Penalty (SEFP).

Reasons for Failing

This entity stores the reason for failing that a Settlement Instruction had at the processing of its relevant cut-off. It is linked to the Penalty Eligibility to which it corresponds.

Penalty

It stores the information about the current version of the Penalty. In case recalculations or modifications occur, previous values will be stored in the corresponding audit trail object for this entity.

There are two types of Penalties:

- "SEFP" - Settlement Fail Penalty: For settlement instructions that i) are matched before the relevant cut-off of a given business day and that ii) having reached their Intended Settlement Date (ISD), fail to settle on that business day.

All rights reserved.

- "LMFP" - Late Matching Fail Penalty: For settlement instructions that are matched in T2S after the relevant cut-off of their Intended Settlement Date (ISD).

Each Penalty is linked to the following entities:

- Penalty Eligibility that identifies the need for its calculation;
- The Penalty entity itself in case of reallocation, in order to identify, through the association "reallocated from/to", the two Penalties involved when a penalty reallocation request is executed, i.e. the original Pen-alty and the new Penalty after reallocation.
- The Securities that identifies the financial instrument involved;
- The two Applicable Parties it applies to: one link for the Failing Party and a second one for the non-failing party (i.e. the two Counterparties of the Penalty);
- Each and every Sub-amount calculated that forms the Penalty.

Applicable Party

This entity stores the specific data corresponding to either the DEBIT side (failing) or the CREDIT side (non-failing) of the Penalty. It stores the information about the current version of the Applicable Party, previous val-ues will be stored in the corresponding audit trail object for this entity.

It is linked to the following entities:

- The Penalty to which the Applicable Party applies;
- The Party for the following relationships: applicable Party (i.e.: the Failing Party if "DEBIT", the non-Failing Party if "CREDIT"), CSD of the applicable Party, Counterparty and CSD of the counterparty;
- The Settlement Instruction related to the Applicable Party, i.e.: the failing Settlement Instruction for the Failing Party, and the non-failing Settlement Instruction for the non-Failing Party.
- The Daily Net for a party vs counterparty for the Party and Counterparty in a given business day;
- The Monthly Net for a party vs counterparty created for the Party and Counterparty within a given month.

Sub-amount

This entity stores the specific data corresponding to the Sub-amount(s) that forms/form a given Penalty.

It stores the information about the current version of the Sub-amount, previous values will be stored in the corresponding audit trail object for this entity together with the Sub-amount (securities) and Sub-amount (cash) that form the Sub-amount.

There will be a Sub-amount per business day for which the Penalty applies. Therefore:

- In case of a "SEFP", there will be a single Sub-amount; whereas
- In case of a "LMFP", there will be as many Sub-amounts as business days when the Penalty applies.

The number of Sub-amounts will be indicated in the Penalty with the value of the attribute Number of business days.

Each Sub-amount is linked to the following entities:

All rights reserved.

- The unique Penalty the Sub-amount relates to;
- The applicable Penalty Reference Data for Securities used in the calculation of the Sub-amount;
- The Sub-amount (securities) that forms the Sub-amount in case the attribute Calculation Method of the re-lated Penalty is "SECU", "MIXE" or "BOTH";
- The Sub-amount (cash) that forms the Sub-amount in case the Calculation Method of the related Penalty is "CASH" or "BOTH".

Sub-amount (securities)

This entity stores specific data of the Sub-amount it forms, i.e. the part of the Sub-amount that is based on the quantity of securities failed to be delivered by the underlying instruction. It exists when the Calculation Method of the related Penalty is "SECU", "MIXE" or "BOTH".

Each Sub-amount (securities) is linked to the following entities:

- The Sub-amount it forms;
- The applicable Cash Discount Penalty Rate if used (i.e. only when the Calculation Method of the Penalty is "MIXE");
- The applicable Euro Foreign Exchange Rate if used (i.e. only when value of the Original Currency differs from the value of the Currency, and hence the attribute Conversion Flag is "True").

Sub-amount (cash)

This entity stores specific data of the Sub-amount it forms, i.e. the part of the Sub-amount that is based on the amount of cash failed to be delivered by the underlying instruction. It exists when the Calculation Method of the Penalty is "CASH" or "BOTH".

Each Sub-amount (cash) is linked to the following entities:

- The Sub-amount it forms;
- The applicable Cash Discount Penalty Rate.

Penalty Reference Data for Securities

This entity stores specific information of the Sub-amount, i.e. the information from CRDM related to the Se-curity of the underlying instruction (Security Subject to penalties and related Security penalty data) and for the corresponding business day.

It is linked to the following entities:

- The Sub-amount it applies to;
- The Securities subject to Penalties it refers to;
- The Security Penalty Rate used;
- The Daily Price used.

Daily net for a party vs counterparty

This entity stores the specific data corresponding to the net amount, due or entitled, from a Party to a given Counterparty for a given business day and currency.

All rights reserved.

It stores the information about the current version of the Daily net for a party vs counterparty, previous values will be stored in the corresponding audit trail object for this entity.

It represents the result of aggregating the Debit amounts due by the party to a counterparty, and the Credit amounts to be received by the party from the counterparty, on a given business day.

Each Daily net for a party vs counterparty is linked to:

- The Applicable Party that creates it;
- The Party entity for the following relationships: Party and Counterparty.

Monthly net for a party vs counterparty

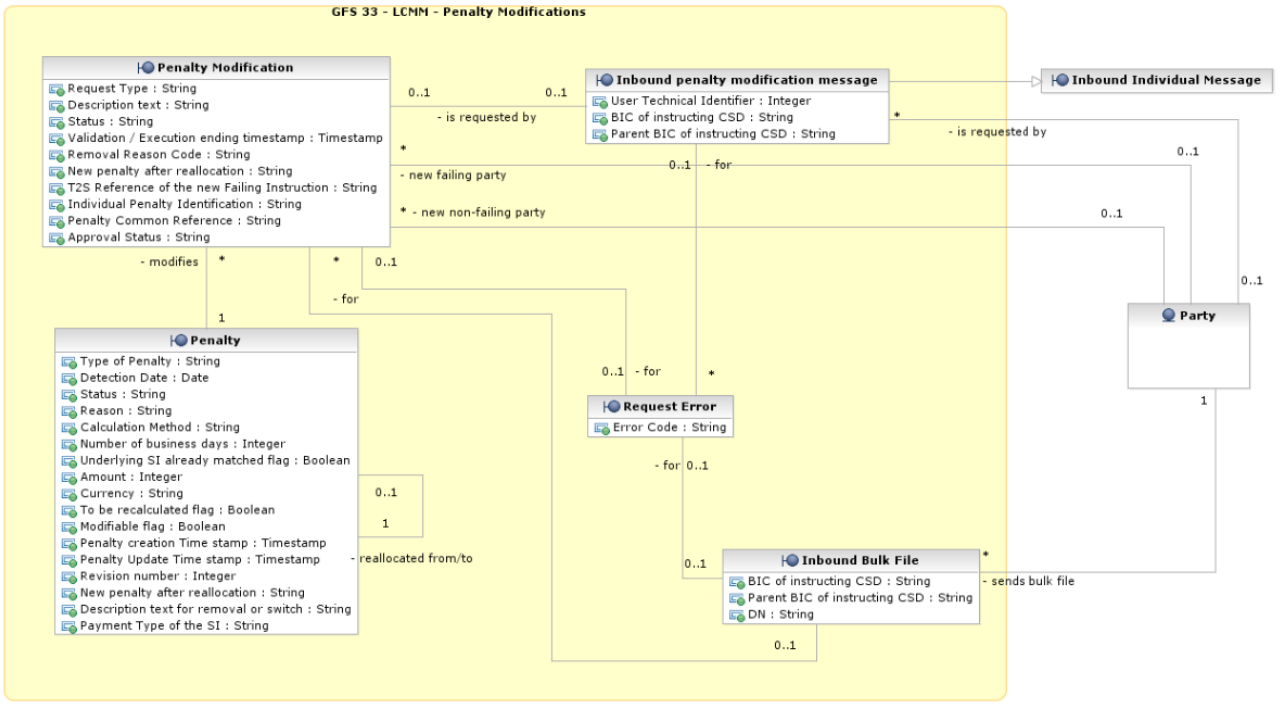
This entity stores the specific data corresponding to the monthly net amount, due or entitled, from a Party to a given Counterparty in a given currency.

It represents the result of aggregating the Debit amounts due by the party to a counterparty, and the Credit amounts to be received by the party from the counterparty, on a given month.

Each Monthly net for a party vs counterparty is linked to:

- The Applicable Party that creates it
- The Party entity for the following relationships: Party, CSD of the party, Counterparty and CSD of the counterparty.

The following diagram shows the conceptual data model for Penalty Modification Requests:



Penalty Modification

This entity stores information concerning Penalty Modification instructions. It is a specialisation of the LCMM Instruction entity, with an additional attribute for storing the current Status of the request (validated, executed or rejected).

All rights reserved.

Each Penalty Modification is linked to the following entities:

- The Penalty whose modification is requested;
- Either the Inbound penalty modification message by means of which a penalty modification can be requested via U2A; or the Inbound Bulk File by means of which one or several Penalty Modification(s) can be also requested via Middleware;
- The Request error(s) that can arise when a Penalty Modification cannot be executed.

Inbound Penalty Modification Message

This entity stores information concerning the Penalty Modification instructions received by the Inbound Processing module of the Interface domain {T2S.16.167}. It is a specialisation of the Inbound Individual Message entity (refer to the dynamic data description of the Interface domain for more information).

Each Inbound Penalty Modification Message is linked to:

- The Penalty Modification whose request it creates;
- The Request Error that can arise while validating the Message;
- Its requestor/sending Party

Inbound Bulk File

This entity stores the messages that enter into the Penalty Mechanism from the Middleware via bulk files.

Each Inbound bulk file is linked to:

- The Penalty Modification(s) whose request it creates: each modification requested in the Bulk File creates one instance in Penalty Modification;
- The Request Error that can arise while validating the Bulk File;
- Its sending Party.

Request error

This entity contains information about errors that may arise either while validating Inbound Penalty Modification Messages, Inbound Bulk File, or Penalty Modifications.

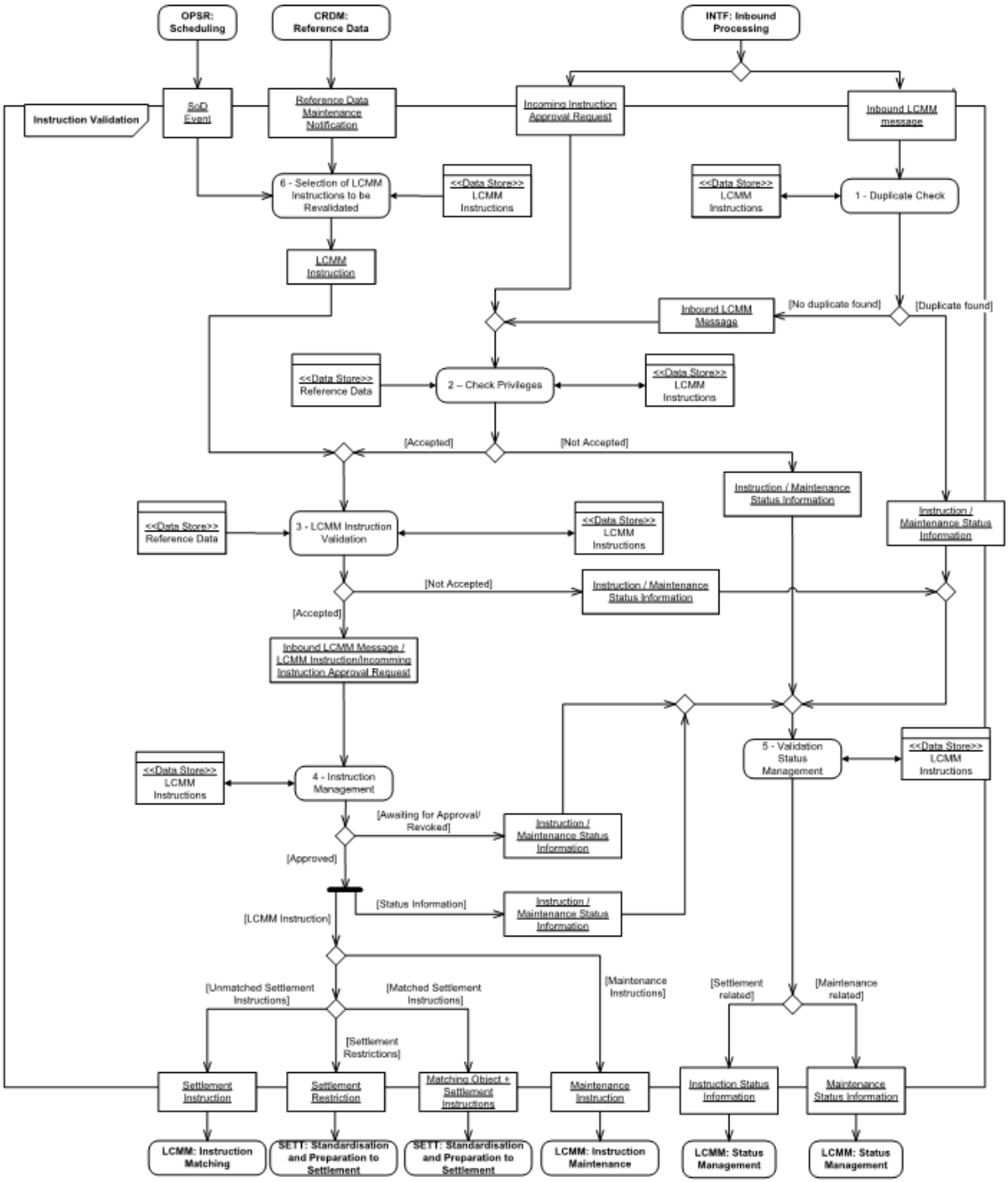
Each Request error is linked to the following entities:

- The Inbound Penalty Modification Message originating the error; or
- The Inbound Bulk File originating the error; or
- The Penalty Modification originating the error.

All rights reserved.

3.4.3 Instruction Validation

3.4.3.1 Diagram of the module



3.4.3.2 Description of the module

LCMM Validation is the process of checking the consistency of *Inbound LCMM Messages* containing information to create *Settlement Instructions*, *Settlement Restrictions* and maintenance instructions (*Hold/Release, Amendment and Cancellation Instructions*) sent to T2S by a *CSD* or directly connected T2S *Party*, with the

All rights reserved.

information stored in the T2S Reference Data and Dynamic Data. It also checks the consistency of already existing *LCMM Instructions* that need to be revalidated. This service is available continuously during the settlement day, except during the maintenance window **{T2S.03.210}**.

The *Instruction Validation* module performs different checks on *Inbound LCMM Messages* or already existing *LCMM Instructions,* depending on the type of instruction received.

After a successful validation of the *Inbound LCMM Message*, the *Instruction Validation* module creates a *LCMM Instruction* that can be one of the following types:

- *Settlement Instruction*;
- *Settlement Restriction*;
- *Hold/Release Instruction*;
- *Amendment Instruction*;
- *Cancellation Instruction*.

Hold/Release, Amendment and Cancellation Instructions are considered as maintenance instructions that aim at modifying an already existing *Settlement Instruction* or *Settlement Restriction*.

After an unsuccessful validation, the *Instruction Validation* module either rejects the *Inbound LCMM Message*, or cancels the *LCMM Instruction* when it already exists in T2S (i.e. revalidation is triggered by a Reference Data update or at the Start of Day).

The *Instruction Validation* module sends information on the validation process outcome to the *Status Management* module.

3.4.3.3 Description of the functions of the module

The *Instruction Validation* module performs the following six functions:

- Duplicate Check;
- Check Privileges;
- LCMM Instruction Validation;
- Instruction Management;
- Validation Status Management;
- Selection of LCMM Instructions to be Revalidated.

1 – Duplicate Check

Introduction

Reference Id	***LCMM.IVA.DUC.1.1***

The function performs a check for the submission of duplicate instructions for all *Inbound LCMM Messages*, prior to its formal validation in the *LCMM Instruction Validation* function*,* avoiding the processing of duplicated *Inbound LCMM Messages* in the system.

Identification of duplicate instructions

All rights reserved.

Reference Id	***LCMM.IVA.DUC.2.1***

The function rejects duplicate or multiple submissions of *Inbound LCMM Messages* based on a combination of the T2S Party identifier and the Instruction reference assigned by the Instructing Party [65] **{T2S.05.030}**.

The duplicate check on *LCMM Instructions* is performed by comparing the T2S Actor Instruction Reference of the *Inbound LCMM Messages* with all the *LCMM Instructions* received from the same T2S Party:

- That are not yet settled, executed or cancelled **{T2S.05.030}**;
- That have been settled, executed or cancelled in a predefined number of days in the past **{T2S.05.030}**.

This predefined number of days is a parameter set in T2S and is the same for the *Settlement Restrictions*, *Settlement Instructions* and all maintenance instructions.

When the function finds a duplicated instruction reference, the function stores the error and sets the Business Validation Status to "*True*" in the *Inbound LCMM Message* entity, and forwards to the *Validation Status Management* function a *Maintenance/Instruction Status Information* with the validation error.

If no duplicate is found, the function forwards the *Inbound LCMM Message* to the *LCMM Instruction Validation* function.

2 – Check Privileges

Reference Id	***LCMM.IVA.DUC.2.1***

The Check Privileges function ensures that a user has the privileges to use a specific function provided by T2S.

This function is the entry point for all authorisation checks for all *Inbound LCMM Messages*. These are performed over the T2S System User to validate the privileges to:

- Send Settlement Instructions / Settlement Restrictions
- Instruct on a specific Securities Account / T2S Dedicated Cash Account
- Link instructions of a specific party
- Other relevant privileges for maintenance instructions

3 – LCMM Instruction Validation

Introduction

Reference Id	***LCMM.IVA.SMIV.1.1***

The LCMM Instruction Validation function receives:

- *Inbound LCMM Messages* for validation when no duplicate has been found;
- *Incoming Instruction Approval Request* from the *Interface* domain;
- Existing *LCMM Instructions* to be revalidated.

65 T2S considers Instructing Party of an LCMM Instruction the originator of the message, if other than the sender.

All rights reserved.

Depending on the Request Type, the function identifies and executes the different necessary validation checks.

When the function receives an *Incoming Instruction Approval Request* (approval or revocation) from the Second T2S system user, this function checks that the referenced *LCMM Instruction* exists in T2S and that its Approval Status value is *"Awaiting for Approval"*.

The different validation checks perform all logically possible validations and communicate the result of the process to the *Validation Status Management* function when the *Inbound LCMM Message* is rejected or the *LCMM Instruction* is cancelled through an *Instruction/Maintenance Status Information*, or to the *Instruction Management* function when it is accepted **{T2S.05.020}**.

The function does not validate non-settlement information added to the *Inbound LCMM Message* by T2S Actors for their own ends **{T2S.05.230}**.

In case of *Inbound LCMM Messages* or *LCMM Instructions* with an Intended Settlement Date in the past, the function performs the corresponding validation checks with the Reference Data valid for the day of the Intended Settlement Date in the past and for the current day [66]. In case of future-dated *Inbound LCMM Messages* or *LCMM Instructions*, the function performs the validation checks with the Reference Data valid for that future date available at the current day.

The *LCMM Instruction Validation* function includes the following sub-functions:

- *Consistency Validation*;
- *Validation Checks*;
- *Validation Result Manager*.
- *Consistency validation*

Reference Id	***LCMM.IVA.SMIV.2.1***

The sub-function checks whether the relevant information available in the *Inbound LCMM Message* or in the *LCMM Instruction* enables the *LCMM Instruction Validation* function to perform all the required checks **{T2S.05.010}**.

The process checks the existence and completeness of harmonised mandatory fields in the *Inbound LCMM Messages*. This means checking the existence of mandatory fields depending on the Payment Type Code and Securities Movement Type **{T2S.05.035}**.

The following table depicts the mandatory fields on a *Settlement Instruction*:

DVP/DWP/RVP/RWP	FOP [67]/DFP
Intended Settlement Date	
Settlement Quantity [68]	

66 During the migration period to T2S, for all the Settlement Instructions marked as "migrated instructions" the validation checks will only be performed at the current day.

67 In line with current market practices, in T2S delivery versus delivery can be performed through two linked FOP instructions (link between a RFOP and a DFOP)

68 Equivalent to Share Quantity (for equities) or Nominal Amount (for fixed income securities) in URD

All rights reserved.

DVP/DWP/RVP/RWP	FOP [67]/DFP
Delivering Party BIC Code	
Receiving Party BIC Code	
Delivering CSD BIC Code	
Receiving CSD BIC Code	
Trade Date	
Credit/Debit Code [69]	N/A
ISIN Code	ISIN code to deliver (DFP) / ISIN Code to Receive (RFP)
Currency	N/A
Settlement Amount	N/A
Deliverer's securities account (to be included by the delivering party)	Deliverer's securities account in case of FOP instructions between securities accounts belonging to the same T2S party within the same CSD (to be included by the delivering party).
Receiver's securities account (to be included only by the receiving party) [70]	Receiver's securities account in case of FOP instructions between securities accounts belonging to the same T2S party within the same CSD (to be included by the receiving party)

The sub-function also checks:

- Whether the Securities Account included in a Settlement Instruction or in a Settlement Restriction on securities exists in T2S and is open on the Intended Settlement Date. This check is performed also for the counterparty´s Securities Account if it is included in a Settlement Instruction regardless if the Settlement Instruction is unmatched or matched.
- If the value of the "Hold Release default" attribute of the Securities Account included in the Settlement Instruction [71] is set to Yes in case the Settlement Instruction has no party hold value set (i.e. neither Party Hold status "Yes" nor "No" have been specified) **{T2S.16.590}**. This check is performed also for the counterparty´s Securities Account only in case of "Already Matched" Settlement Instructions.

Nevertheless, the sub-function does not perform the above mentioned checks over the Securities Account if the relevant CSD is external to T2S.

Validation Checks

Reference Id	***LCMM.IVA.SMIV.3.1***

69 Possible values: debit/credit

70 For already matched instructions, the instruction should include also deliver securities account

71 Internally generated instructions are not considered for the Hold/Release default check

All rights reserved.

The sub-function performs the validation of the *Inbound LCMM Message* or the *LCMM Instruction* according to the instruction type (Settlement Instructions, Settlement Restrictions or maintenance instructions) and the use of cash and/or securities resources. The validation process goes on executing all checks applicable to the *Inbound LCMM Message* or the *LCMM Instruction* even though errors have been encountered in the process. The following paragraphs describe the list of the different validations.

The following descriptions only depict the different types of checks to be performed, but do not aim at specifying the order of these checks.

There is no difference between the initial validation processing of an *Inbound LCMM Message* and later re-validation processing of *LCMM Instructions*.

ISIN Code Validation

Reference Id	***LCMM.IVA.SMIV.3.2***

The sub-function does the following checks related to the ISIN of the instruction processed:

- Eligible ISIN Check
  - The ISIN [72] code exists in T2S Reference Data **{T2S.05.080}**;
  - The ISIN code is eligible for settlement in the corresponding CSD for the Intended Settlement Date, unless the instruction was sent by a CSD, which are allowed to send instructions for non-settlement eligible ISIN(s) as long as they are still active (not logically deleted), or unless the Settlement Instruction **or Settlement Restriction** was sent by other T2S Actors (non-CSD) and involves a non-settlement eligible ISIN that is still within the transformation detection period **{T2S.05.080}**;
  - The CSD (internal or external to T2S) of the instructing T2S Party and the CSD of the counterpart in the *Settlement Instruction* allow settlement with each other for the specified ISIN code **{T2S.05.160}**;
  - The Intended Settlement Date is equal to or later than the Issue Date, unless the *Instructing Party* is the issuer CSD or the Technical Issuer CSD and the Settlement Instruction is not a cross-CSD sent as already matched to T2S **{T2S.05.150}**;
  - When the Instruction Party is not the Investor CSD, the Technical Issuer CSD or the Issuer CSD of the ISIN Code specified in the Settlement Instruction or the Settlement Instruction is a cross-CSD sent as already matched to T2S, the current Business Day should be earlier than the Maturity date of the Security plus the transformation detection period.
  - When the Instructing Party is not the Investor CSD, the Technical Issuer CSD or the issuer CSD of the ISIN Code specified in the Settlement Restriction on securities or the Settlement Instruction is a cross-CSD sent as already matched to T2S, the current Business Day should be earlier than the Maturity date of the Security **plus the transformation detection period**.
- Eligible Issuance Account Check.

72 An update in Reference Data of the ISIN code does not lead to revalidation to instructions related to that ISIN

All rights reserved.

- The ISIN must be eligible for settlement in the Issuer CSD for the Issuance Account specified in the Settlement Instruction on the Intended Settlement Date. Additionally on the Current Business Date if the Intended Settlement Date is in the past.

- Eligible Counterpart CSD Check.
  - The CSD of the T2S Party must allow settlement with the CSD of the counterpart and vice versa **{T2S.05.126}**

Intended Settlement Date Validation

Reference Id	***LCMM.IVA.SMIV.3.3***

This validation verifies that:

- The Intended Settlement Date is a T2S settlement day for the *Settlement currency* as defined in T2S reference data [73] **{T2S.05.120}**;
- The Trade Date is equal to or earlier than the Intended Settlement Date **{T2S.05.110}** except for maintenance instructions;

For *Inbound LCMM Messages* or *LCMM Instructions* sent with an Intended Settlement Date in the past or in the future, the Intended Settlement Date does not exceed the predefined time frame to send past-dated or future-dated *Inbound LCMM Messages* or *LCMM Instructions* [74]**{T2S.05.120}**.

Settlement Quantity Validation

Reference Id	***LCMM.IVA.SMIV.3.4***

The following checks are performed on the Original Settlement Quantity:

- The Original Settlement Quantity is equal or greater than zero;
- In case the Original Settlement Quantity is greater than zero, the following checks are applied:
  - The Original Settlement Quantity is greater than or equal to the Settlement Unit Minimum (in units or nominal) for the ISIN code, as it is defined in T2S Reference Data, unless the instruction is related to corporate actions, market claims or transformations **{T2S.05.090}**,
  - The Original Settlement Quantity is multiple of the securities Settlement Unit Multiple (in units or nominal) or a Deviating Settlement Unit (in units or nominal), unless the instruction is related to corporate actions, market claims or transformations **{T2S.05.100}**,
  - The number of decimals of the Original Settlement Quantity is equal or lower than the number of decimals defined in the Settlement Unit Multiple **{T2S.05.100}**.
- The Settlement Type stated in the Settlement Quantity of a Settlement Instruction with exchange of Securities or a Settlement Restriction on Securities must be the same as the Settlement Type specified T2S Reference Data for the ISIN Code of the Instruction.

---

73 When T2S offers settlement in non-euro currency, the system shall accommodate the relevant opening days for these currencies. In addition T2S shall be open for settlement of FOP transactions from Monday to Friday every week.

74 The current allowed predefined time frame to send past-dated or future-dated *Inbound LCMM Messages* or *LCMM Instructions* in T2S is unlimited

All rights reserved.

T2S Dedicated Cash Account Validation

Reference Id	***LCMM.IVA.SMIV.3.5***

The following checks are performed concerning the *T2S dedicated cash account* for *Settlement Instructions* which Payment Type Code is against payment ("*APMT*"):

- If the T2S Dedicated Cash Account is included:
  - it exists in T2S Reference Data and the Intended Settlement Date [75] is between its Opening Date and Closing Date,
  - an active link exists to the *Securities Account* in T2S Reference Data **{T2S.05.040} {T2S.05.060}**;
  - it's associated Currency must be consistent with the Currency of the Settlement Instruction **{T2S.05.070}**.
  - the relevant party must be authorized to operate the Cash Account on the Intended Settlement Date;
- If the *T2S Dedicated Cash Account* is not included, a default T2S Dedicated Cash Account for the currency of the cash leg of the *Settlement Instruction* is defined for the *Securities Account* in T2S Reference Data **{T2S.05.060}**.

The following check is performed concerning the *T2S Dedicated Cash Account* for *Settlement Restrictions* on Cash:

- The *T2S Dedicated Cash Account* exists and the Intended Settlement Date is between its Opening Date and Closing Date;
- The *T2S Dedicated Cash Account* belongs to the corresponding T2S Party in T2S.

Insolvency Restriction Type Validation

Reference Id	***LCMM.IVA.SMIV.3.6***

When the "Under Insolvency Situation" general configuration parameter has been activated by the T2S Operator, the following checks are performed:

- The Securities Account specified in a Settlement Instruction delivering securities must not be flagged through a Restriction Type as insolvent {T2S.05.065};
- The T2S Dedicated Cash Account specified in a Settlement Instruction debiting cash must not be flagged through a Restriction Type as insolvent {T2S.05.065}. In case the debiting T2S Dedicated Cash Account is not specified by the user in the Settlement Instruction, the default T2S Dedicated Cash Account derived from the Securities Account of the Settlement Instruction must not be flagged through a Restriction Type as insolvent;
- In case of Settlement instructions mandated by the Insolvency Liquidator, in order to bypass the rejection, the Instructing Party of the Settlement instruction must be the CSD where the Securities Account is held and the first four characters of the Common Trade Matching Reference must be the code word "LIQR". Otherwise, the Settlement instruction mandated by the Insolvency Liquidator will be rejected.

75 If the ISD falls into the past it is also checked that the T2S Dedicated Cash Account is open in the current business day, this is not checked for future-dated instructions

All rights reserved.

This means that cross CSD Settlement Instructions sent as already matched to T2S will never bypass the rejection (consequently, the cross CSD Settlement Instructions shall be sent unmatched to T2S by the Insolvency Liquidator).

- This check is not performed at revalidation level for Settlement Instructions already present in the system, i.e. this check is only applicable for new incoming Settlement Instructions.

Linked Instruction Validation

Reference Id	***LCMM.IVA.SMIV.3.7***

When an *Inbound LCMM Message* [76] or a *LCMM Instruction* has a link indicator (After ("AFTE"), Before ("BEFO"), All-or-none ("WITH"), and Non-settlement-related link indicator "INFO") to a *Settlement Instruction* or *Settlement Restriction*, the sub-function performs the following checks if the linked instructions already exist in T2S [77] **{T2S.05.140} {T2S.05.147}**:

- If the *Inbound LCMM Message* or *LCMM Instruction* contains a "WITH" or "BEFORE" link, the existing linked instruction is neither settled nor cancelled **{T2S.05.148}**;
- The *Inbound LCMM Message* or *LCMM Instruction* and the existing *Settlement Instruction* or *Settlement Restriction* links, do not contradict each other regarding the Link Type and the Intended Settlement Date **{T2S.05.148} {T2S.05.186}**;
- The information contained in the *Inbound LCMM Message* or *LCMM Instruction* is consistent with the existing linked *Settlement Instruction* or *Settlement Restriction* (i.e. securities account owner) **{T2S.05.148}**.

When the *Inbound LCMM Message* [78] or *LCMM Instruction* includes a *Pool Reference* the sub-functions performs the following checks.

- The *Inbound LCMM Message* or *LCMM Instruction* does not include more than one *Pool Reference* since that might lead to inconsistent links between the instructions belonging to two or more pools;
- The Total number of linked i*nstructions* of the pool is included in the *Inbound LCMM Message*.
- In case the pool [79] already exists in T2S:
  - The Total number of linked *instructions* stated in the *Inbound LCMM Message* coincides with the Pool Instruction Counter of the relevant *LCMM Instruction Pool* in T2S,
  - The successful validation of the *Inbound LCMM Message* does not lead to exceed the Pool Instruction Counter of the relevant *LCMM Instruction Pool*.

Reference to Settlement Restriction Validation

---

76 Settlement Instructions, Settlement Restrictions, or Amendment Instructions modifying Settlement Instructions. For Amendment Instructions the Intended Settlement Date is understood as the one of the referenced *Settlement Instruction*

77 Since it is not possible to submit to partial settlement Settlement Instructions carrying a link indicator, Inbound LCMM Messages or LCMM Instructions cannot be linked to Settlement Instructions which have already partially settled. For the partial settlement specific case, T2S rejects Inbound LCMM Messages or LCMM Instructions referring to instructions which have already partially settled.

78 Settlement Instructions, Settlement Restrictions, or Amendment Instruction modifying Settlement Instructions. For Amendment Instructions the Intended Settlement Date is understood as the one of the referenced Settlement Instruction

79 The existence of a pool in T2S is derived by referring to the LCMM Instruction Pool entity.

All rights reserved.

Reference Id	***LCMM.IVA.SMIV.3.8***

When an *Inbound LCMM Message* or a LCMM *Instruction* refers to a *Restriction Reference*, the process performs the following consistency checks between the information contained in the *Inbound LCMM Message* or *LCMM Instruction and* the referenced *Restriction Reference*:

- The *Restriction Reference* included in the *Inbound LCMM Message* or *LCMM Instruction* exists in the entity *Restriction Reference* **{T2S.05.148}**, otherwise, the instruction is rejected;
- The Restriction Type (Blocking/Reservation) in the *Inbound LCMM Message* or *LCMM Instruction* is consistent with the Restriction Type of the *Restriction Reference*. The possibility of auto-collateralisation is given by the Restriction Type **{T2S.05.143}**

Restriction Type Validation

Reference Id	***LCMM.IVA.SMIV.3.9***

The validation checks that the Restriction Type included in the *Inbound LCMM Message* or in the *LCMM Instruction* is valid for the Intended Settlement Date according to reference data.

Instructing Party Validation

Reference Id	***LCMM.IVA.SMIV.3.10***

T2S considers as *Instructing Party* the party specified in the Originator of the message field of the *Inbound LCMM Message*. If the originator of the message is not specified, T2S considers as *Instructing Party* the sender of the message.

The *Instructing Party* of the *Inbound LCMM Message* or of the *LCMM Instruction* is checked as follows:

- The *Instructing Party* exists in T2S Reference Data and the Intended Settlement Date is between its Opening Date and Closing Date;
- The *Instructing Party* is authorised to use the *Securities Account* according to the privileges defined in T2S Reference Data **{T2S.05.040}** **{T2S.05.050}**;
- The *Instructing Party* is authorised to use the counterparty's *Securities Account* in case of "*Already Matched*" instruction according to the privileges defined in T2S Reference Data **{T2S.05.040}**.
- In case of Already Matched instructions, the CSD of the Instructing Party of a Settlement Instruction can be different than the CSD of the Securities Account because T2S Actors can send already matched intra-CSD and cross-CSD instructions.
- If the T2S System User does not belong to the Instructing Party of the *Inbound LCMM Message* or of the *LCMM Instruction*, then the T2S System User must be authorised to use this Instructing Party (i.e. to instruct on behalf of the Instructing Party).

In case the *Inbound LCMM Message* or the *LCMM Instruction* relates to a maintenance instruction, the following checks are performed **{T2S.05.220}**:

- *Instructing Party* is allowed to hold, release, amend and cancel instructions **{T2S.05.310}** **{T2S.05.220}**. In case of an "Already Matched" *Cancellation Instruction*, the *Instructing Party* is authorised to cancel both referenced *Settlement Instructions*;

All rights reserved.

- CSD participants do not attempt to hold, release, partially release, amend or cancel CSD *Settlement Instructions* identified by CSDs as non-modifiable by other CSD participants **{T2S.05.300}**.

Settlement Process Indicator Validation

Reference Id	***LCMM.IVA.SMIV.3.11***

The check ensures that the settlement process indicators are valid considering the *Instructing Party and the instruction type*, as defined in T2S Reference Data **{T2S.05.140}**:

- For the Priority Indicator process T2S Actors can assign different levels of priority **{T2S.05.145}** **{T2S.07.150}**. Accepted values are:
  - *"Reserved Priority"***{T2S.07.160}**;
  - *"Top Priority"* **{T2S.07.170}**;
  - *"High Priority"* **{T2S.07.180}**;
  - *"Normal Priority"* **{T2S.07.190}**;
- "*Reserved Priority*" can only be used by CSDs and Central Banks (e.g. intraday corporate actions and monetary policy operations) **{T2S.07.160}**.

Currency Validation

Reference Id	***LCMM.IVA.SMIV.3.12***

The checks ensures for all *Inbound LCMM Message* or *LCMM Instruction*, except *FOP instructions* even if the Original Settlement Amount contains a value **{T2S.05.070}**, that:

- The Settlement currency is a valid *Currency* in accordance with the list of currencies defined by the standard ISO 4217 (codes for the representation of currencies and funds) **{T2S.05.070}**;
- The number of decimals of the Original Settlement Amount does not exceed the number of decimals specified for the given Settlement currency in Reference Data **{T2S.16.320}**;
- The *Currency* of the cash leg of the instruction is a T2S Settlement Currency as defined in T2S Reference Data;

Referenced LCMM Instruction Validation

Reference Id	***LCMM.IVA.SMIV.3.13***

When sending a maintenance instruction, T2S Actors can make use of either the *Instructing Party Reference* with the T2S Actor Instruction Reference or the related T2S Instruction Reference (i.e. T2S internal reference assigned to the *Settlement Instruction* or *Settlement Restriction* by T2S) of the impacted *Settlement Instruction* or *Settlement Restriction*. The following checks are performed:

- The T2S Actor Instruction Reference for the corresponding Instructing Party or the T2S Instruction Reference refers to a *Settlement Instruction* or *Settlement Restriction* that exists in T2S **{T2S.05.210}**;
- The information contained in the maintenance instruction related to the referenced *LCMM Instruction* is consistent with the latter (e.g.: *Instruction Type* of the referenced *LCMM Instruction*).

All rights reserved.

In the specific case of a Release Instruction that intends to Partially Release a Settlement Instruction the following checks are performed:

- The Settlement Quantity indicated in a Party Release Instruction must be higher than zero and equal to or lower than the Settlement Quantity stated in the referenced Settlement Instruction
- The Settlement Quantity indicated in a Party Release Instruction must be a multiple of the Settlement Unit Multiple.
- The number of decimals of the Settlement Quantity of a Party Release Instruction must be equal or lower to the number of decimals of the number of decimals of the Settlement Multiple of the related security
- The Settlement Quantity stated in the Party Release Instruction must be expressed using the same Settlement Type specified in T2S Reference Data.
- The Intended Settlement Day of the referenced Settlement Instruction must have been reached.
- The Securities Movement type of Referenced Settlement Instruction must be 'DELI'
- The referenced Settlement Instruction of a Party Release instruction must not be a T2S generated Settlement Instruction.

Condition Modification Validation

Reference Id	***LCMM.IVA.SMIV.3.14***

When receiving an *Inbound LCMM Message* conveying a *Condition Modification Instruction* for hold, release, partially release or amendment purposes, the process performs the following checks:

- The *Inbound LCMM Message* aims at modifying only one *LCMM Instruction*. Nevertheless, when the referenced instruction is an already matched *Settlement Instruction* received through one single *Inbound LCMM Message*, both *Settlement Instructions* can be modified by a single *Inbound LCMM Message*;
- One *Inbound LCMM Message* only modifies one single processing indicator (i.e. Partial Settlement Indicator, Settlement Priority and linkages) or the Hold/Release Status **{T2S.05.390}**;
- The *Inbound LCMM Message* does not modify a processing indicator (i.e. Partial Settlement Indicator, Settlement Priority and linkages) of a T2S generated realignment *Settlement Instruction*.
- The Inbound LCMM Message does not modify the Partial Settlement Indicator of a Settlement Restriction.

Specific Restriction Validation

Reference Id	***LCMM.IVA.SMIV.3.16***

The check verifies whether the combination of parameters contained in the *Inbound LCMM Message*, the *Settlement Instruction* or the *Settlement Restriction,* fulfils any rule of the different *Restriction Processing Types.Restriction Processing Types.* In case one *Restriction Processing Type* is found, no subsequent checking of any rules for that *Restriction Type* is performed **{T2S.05.125}**.

Each *Restriction Processing Type* defined in Reference Data specifies the set of rules (i.e. restriction rules) including all the criteria or combinations of attributes for the triggering of the Restriction Processing Type.

All rights reserved.

Also each *Restriction Processing Type* specifies a positive/negative parameter according to which T2S should apply/should not apply the restriction if any of its rules is satisfied

A *Restriction Type* can imply that the *Settlement Instruction* has to be put automatically On Hold at the moment of its creation for additional validation or processing **either** by the CSD, through the CSD Validation Hold Status **or by the T2S Party, through the Party Hold** **{T2S.05.127}.{CR-812}** Also, it can imply the rejection of the *Inbound LCMM Message* or the cancellation of the *LCMM Instruction* **{T2S.05.128}**.

This check is only done in case the previous business validations checks have been successfully performed. The check accepts Inbound LCMM Messages or LCMM Instructions even when the party, securities account, dedicated cash account and/or security, referenced in the instruction, are blocked from settlement **{T2S.05.129}**.

Incoming Instruction Approval Request check

Reference Id	***LCMM.IVA.SMIV.3.17***

When receiving an Incoming Instruction Approval Request *(approval or revocation)* from the Second T2S system user, the function performs the following checks:

- That the referenced *LCMM Instruction* exists in T2S;
- That the Approval Status value of the referenced *LCMM Instruction* is "Awaiting for Approval".

Validation Result Manager

Reference Id	***LCMM.IVA.SMIV.4.1***

This sub-function receives the result of the different validations performed by the *LCMM Instruction Validation* function, stores all the errors possibly encountered during the validation processes and:

- Forwards to the *Instruction Management* function *Inbound LCMM Messages*, *LCMM Instructions* and Incoming Instruction Approval Request that successfully passed all the relevant validation checks;
- Forwards to the *Validation Status Management* function the *Status Information* relating to the *Inbound LCMM Message*, *LCMM Instruction or* Incoming Instruction Approval Request that failed at least one validation check with the list of validation errors.

The sub-function sets the Business Validation Status value to "*True*" in the *Inbound LCMM Message* entity after unsuccessful validation. This attribute has the objective of specifying that the business validation of the *Instruction Validation* module has been performed. For those *Inbound LCMM Messages* successfully validated the value of this attribute is set at the moment of the creation of the *LCMM Instruction* [80].

4 – Instruction Management

Introduction

Reference Id	***LCMM.IVA.INM.1.1***

The *Instruction Management* function receives:

- *"Accepted" Inbound LCMM Message*;

80 Please refer to the Instruction Management function

All rights reserved.

- *LCMM Instructions* after successful revalidation;
- *Incoming Instruction Approval Request*.

Instruction Management

Reference Id	***LCMM.IVA.INM.2.1***

When receiving an Inbound LCMM Message after its successful validation, the function creates the corresponding new *LCMM Instruction(s) (Settlement Instruction, Settlement Restriction, Amendment Instruction, Cancellation Instruction or Hold / Release Instruction)* storing the message information not needed for settlement or execution in the additional information classes:

- For already matched instructions (i.e. those Settlement Instructions identified through the Externally Matching field of the *Inbound LCMM Message or those maintenance instructions referring to an already matched Settlement Instruction by its* T2S Actor Instruction Reference):
  - Splits the *Inbound LCMM Message* containing information of already matched instructions, and creates two separate *LCMM Instructions,*
  - Assigns a T2S Instruction Reference to each of them **{T2S.05.170}**,
  - Assigns a T2S Matching Reference to both *Settlement Instructions* **{T2S.05.170}**,
  - Creates simultaneously the *Matching Object* (via *Matching Object Creator* function)*;*
- For unmatched instructions, the function:
  - Creates and stores the *LCMM Instruction* from the *Inbound LCMM Message*,
  - Assigns a T2S Instruction Reference to this *LCMM Instruction*.

In addition, at the moment of creation of the *LCMM Instruction*(s) the function performs the following statuses assignment:

- Hold Status assignment: The Party Hold Status or CSD Hold Status of the instructed leg is set to "Yes" when the Inbound LCMM Message includes hold indicator with the value "PTYH" or "CSDH" accordingly specifying that it is On Hold by the T2S Party or the CSD. In case of an already matched Settlement Instruction, the Party Hold Status of the T2S generated counter-leg is set to "Yes" when the Inbound LCMM Message includes hold indicator with the value "PRCY", and both legs, instructed and T2S generated Counter-leg, are set On Party Hold when the inbound LCMM Message includes the hold indicator with the value "BOTH". The Party Hold Status or CSD Hold Status value **is set** to "*No*" otherwise. Additionally, in case the Inbound LCMM Message comes without the Party Hold Status being informed (i.e. neither Hold Status "Yes" nor "No" have been specified) and the "Hold Release default" value of the Securities Account included in the Settlement Instruction is set to "Yes", **or if it fulfils predefined conditions of a Restriction Processing Type "Party Hold" previously identified during the Specific Restriction Validation,** the Party Hold Status is automatically set to "Yes.**{CR-812}**

All rights reserved.

- Approval Status (Four-Eyes processing) assignment: The value of the Approval Status is set to "*Approved*" as default value, when the *Inbound LCMM Message* is not processed under the Four-Eyes principle, and to "*Awaiting Approval*" when the *Inbound LCMM Message* is processed under the Four-Eyes principle. This principle implies that the *LCMM Instruction* is not routed to the next function until an approval request has been sent by another T2S System User.
- CoSD Hold and Settlement Status assignment: It assigns the CoSD Hold Status value to "*No*" and Settlement Status value to "*Unsettled*".
- Cancellation Status assignment: It sets the default value for the Cancellation Status of a *Settlement Instruction* or a *Maintenance Instruction* to respectively "*Not cancelled*" or "*Not Denied*".
- Execution Status assignment: It sets the default value for the Execution Status of a *Maintenance Instruction* to "*Not Executed*".
- CSD Validation Hold Status assignment: The CSD Validation Hold Status value is set to "*Yes*" when the *Inbound LCMM Message* fulfils predefined conditions of a Restriction Processing Type *"CSD Validation Hold"* previously identified during the Specific *Restriction Validation* and additional validation or processing by the CSD is needed. It sets the CSD Validation Hold Status value to "*No*" as default value, when no Restriction Processing Type *"CSD Validation Hold"* is fulfilled **{T2S.05.127}**.
- Match Status assignment: The Match Status value is set to "*Matched*" for those already matched *LCMM Instructions* or to "*Unmatched*" for those *LCMM Instructions* that require matching **{T2S.05.444}**.
- Settlement Status assignment: This function sets to "True" the Failing flag of the instructions entering T2S with either i) Intended Settlement Day before the Current Business Day, or with ii) Intended Settlement Day equal to the Current Business Day if they enter during the End of Day phase.

It is also responsible for setting the value "True" to the Business Validation Status in the Inbound LCMM Message entity in order to inform that the Instruction Validation module has processed the LCMM Instruction. Performing the update of the Business Validation Status at the moment of the creation of the LCMM Instruction ensures consistency in the data model as there is always a LCMM Instruction for a validated (and not rejected) Inbound LCMM Message.

When receiving successfully revalidated LCMM Instructions that fulfil a new rule of a Restriction Processing Type "CSD Validation Hold", the function sets the value of the CSD Validation Hold Status to "Yes" **{T2S.05.127}**. **When receiving successfully revalidated LCMM Instructions that fulfil a new rule of a Restriction Processing Type "Party Hold", the function sets the value of the Party Hold Status to "Yes".{CR-812}**

When receiving a successfully validated *Incoming Instruction Approval Request*, this function triggers the following processes:

- When the *Incoming Instruction Approval Request* is an approval request, this function triggers the revalidation of the *LCMM Instruction*. Once it is performed, in case the *LCMM Instruction* is successfully validated, this function assigns the *LCMM Instruction* the Approval Status value *"Approved"*. If the *LCMM Instruction* does not pass the revalidation, the instruction is rejected in the *Validation Status Management* function as any other unsuccessfully revalidated instruction setting its Approval Status value to *"Rejected"*.

All rights reserved.

- When the *Incoming Instruction Approval Request* is a revocation request, this function triggers *Instruction Status Information* for the cancellation of the *LCMM Instruction* (for the Approval Status value of the latter to be set to "Revoked").

After the *LCMM Instruction* creation, or after processing of revalidated *LCMM Instructions* or *Incoming Instruction Approval Requests,* the function reports the creation or update of *LCMM Instruction*(s) to *Validation Status Management* function through an *Instruction/Maintenance Status Information and* routes each processed instruction to the corresponding module according to its Match Status and Instruction Type:

- *Settlement Instructions* with Match Status value "*Unmatched*" to the *Instruction Matching* module except for revalidated "*Unmatched*" Settlement Instructions that do not require to be sent again;
- *Settlement Restrictions* to the *Standardisation and Preparation to Settlement* module once the Intended Settlement Date has been reached **{T2S.05.500}**;
- Matching Objects and the related *Settlement Instructions* to the *Standardisation and Preparation to Settlement* module **{T2S.05.500}**;
- Maintenance instructions to the *Instruction Maintenance* module.
- Those instructions with the Approval Status "*Awaiting Approval*" are not forwarded to any module since it is needed an approval from a second T2S System User.

5– Validation Status Management

Introduction

Reference Id	***LCMM.IVA.VSM.1.1***

The *Validation Status Management* function receives *Maintenance/Instruction Status Information*. The function manages them in order to evaluate if any additional action is needed (as status updates after unsuccessful revalidation or Approval Status for Four-Eyes processing) and to forward statuses information to the *Status Management* module.

Status Management description

Reference Id	***LCMM.IVA.VSM.2.1***

The *Validation Status Management* function:

When receiving a *Status Information* from *Instruction Management* function, reports the *Instruction Status Information* to the *Status Management* module with the corresponding T2S Instruction References **{T2S.05.240} {T2S.05.250}.**

When receiving the *Instruction Status Information* for a rejected or duplicated *Inbound LCMM Message,* reports the validation errors to the *Status Management* module **{T2S.05.240} {T2S.05.250}.**

When receiving *Instruction Status Information* for the cancellation after an unsuccessful revalidation of an *LCMM Instruction:*

Updates the status and reason code of the *LCMM Instructions* after an unsuccessful revalidation. These instructions get the Cancellation Status *"Cancelled"* and the reason code *"Cancelled due to* Reference *Data Change"* **{T2S.05.280},**

All rights reserved.

Update the status and reason code of any linked T2S generated *Settlement Instructions* that have been created for realignment purpose by setting their Cancellation Status to *"Cancelled"* and the reason code *"Cancelled due to* Reference *Data change",*

Update the status of all *Settlement Transactions* linked to the *LCMM Instructions* and their relating T2S Generated *Settlement Instructions* that have been created for realignment purpose by setting their status to *"Cancelled",*

Informs the *Status Management* module **{T2S.05.250};**

When receiving *Instruction Status Information* for the cancellation of an LCMM Instruction under Four-Eyes principle after the unsuccessful revalidation upon the reception of the Incoming Instruction Approval Request:

Updates the status and reason code of the *LCMM Instruction,* i.e. Cancellation Status *"Cancelled"* and reason code *"Cancelled due to* Reference *Data Change"* **{T2S.05.280},**

Updates the Approval Status value *"Awaiting Approval"* to *"Rejected",*

Informs the *Status Management module* **{T2S.05.250};**

When receiving *Instruction Status Information* for the cancellation of an LCMM Instruction under Four-Eyes principle due to the revocation by the second T2S system user:

Updates the status and reason code of the *LCMM Instruction,* i.e. Cancellation Status *"Cancelled"* and reason code *"Instruction under Four Eyes principle cancelled due to revocation",*

Informs the *Status Management* module.

6 – Selection of LCMM Instruction to be Revalidated

Introduction

Reference Id	***LCMM.IVA.SIR.1.1***

The function is responsible for selecting validated *LCMM Instructions* that have to be revalidated due to a Reference Data change and because of the Start of Day procedures, including *LCMM Instructions* affected by a night-time Reference Data update during the night-time settlement cycles and at the beginning of day-time. The *Instruction Validation* module receives *Reference Data Maintenance Notifications* that are relevant for *LCMM*.

Selection of LCMM Instructions to be revalidated that are affected by Reference Data update notifications

Reference Id	***LCMM.IVA.SIR.2.1***

The sub-function identifies the *LCMM Instructions* that have to be revalidated due to a Reference *Data Maintenance Notification* from the CRDM domain **{T2S.05.280}.** These maintenance notifications include information about the reference data modified to allow the identification of *LCMM Instructions* that need revalidation.

There are two types of Reference data Updates that LCMM could receive from Reference data domain:

- Reference Data Updates that imply the revalidation of the approved pending Settlement Instructions and Settlement Restrictions regardless their intended settlement date.

All rights reserved.

- Reference Data Updates that imply the revalidation of the approved pending Settlement Instructions and Settlement Restrictions which have an intended settlement date later than the current business date.

The process only considers the *Settlement Instructions* and *Settlement Restrictions* (excluding those with Approval Status value *"Awaiting Approval"*) affected by the reference data update that:

- do not have the Settlement Status *"Settled";*
- do not have the Cancellation Status *"Cancelled".*

The process then forwards the selected *LCMM Instructions* to the *LCMM Instruction Validation* function for their revalidation.

Selection of LCMM Instructions to be revalidated at the Start of Day (SOD)

Reference Id	***LCMM.IVA.SIR.3.1***

The sub-function identifies the *Settlement Instructions* and *Settlement Restrictions* that have to be revalidated at the start of day process upon the reception of a *SOD Event* from the *Scheduling* module of the OPSR domain **{T2S.05.280}.**

To achieve the aforementioned action, the sub-function only selects those *Settlement Instructions* and *Settlement Restrictions* (excluding those with Approval Status value *"Awaiting Approval"*) that do not have:

- the Settlement Status *"Settled";*
- the Cancellation Status *"Cancelled".*

Additionally, it also considers those *Amendment, Hold/Release* and Cancellation Instructions (excluding those with Approval Status value *"Awaiting Approval"*) that do not have:

- the Execution Status *"Executed";*
- the Cancellation Status *"Denied".*

The sub-function then forwards the selected *LCMM Instructions* to the *LCMM Instruction Validation* function to start the revalidation.

3.4.3.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Inbound LCMM Message	IN		INTF: Inbound Processing	
Reference Data Maintenance Notification	IN		CRDM: Reference Data	
Incoming Instruction Approval Request	IN	Answer received for instructions that required a Four-Eyes approval.	INTF: Inbound Processing	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Event SOD	IN	Event sent for the processing of the SOD	OPSR: Scheduling	
Maintenance Status Information	OUT	Update of the maintenance status value. Includes reason code.		LCMM: Status Management
Matching Object + Settlement Instructions	OUT	Matching Object and related Settlement Instructions.		SETT: Standardisation and Preparation to Settlement
Instruction Status Information	OUT	Settlement Instruction and Settlement Restriction statuses values. Includes reason code.		LCMM: Status Management
Settlement Instruction	OUT			LCMM: Instruction Matching
Settlement Restriction	OUT			SETT: Standardisation and Preparation to Settlement
Maintenance Instruction	OUT	Hold/Release instructions, Amendment instructions, Cancellation instructions		LCMM: Instruction Maintenance

3.4.3.5 Data accessed by the module

DATA	DATA ENTITY	ACCESS MODE	COMMENTS
CRDM			
CRDM	Restriction Type	Read	Accessed for checking purposes
	T2S Dedicated Cash Account	Read	Accessed for checking purposes
	Conditional Securities Delivery Rule	Read	Accessed for checking purposes
	Closing Day	Read	Accessed for checking purposes

All rights reserved.

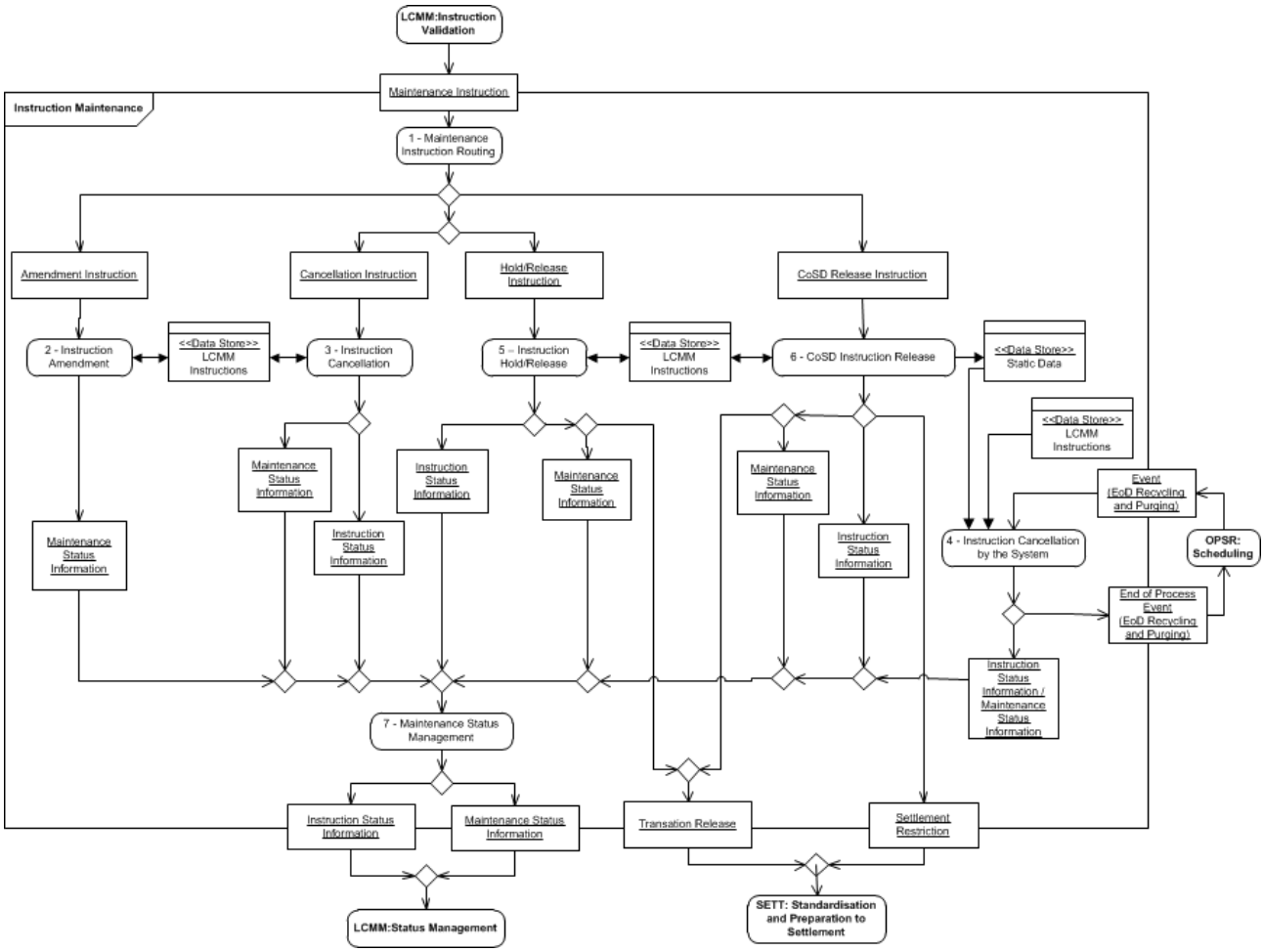
DATA	DATA ENTITY	ACCESS MODE	COMMENTS
	CSD Account Link	Read	Accessed for checking purposes
	Currency	Read	Accessed for checking purposes
	Deviating Settlement Unit	Read	Accessed for checking purposes
	Market-Specific Attribute	Read	Accessed for checking purposes
	Error Type	Read	Accessed for checking purposes
	Party	Read	Accessed for checking purposes
	Party Code	Read	Accessed for checking purposes
	Party Restriction	Read	Accessed for checking purposes
	Securities account	Read	Accessed for checking purposes
	Securities Account Restriction	Read	Accessed for checking purposes
	Securities	Read	Accessed for checking purposes
	Security Code	Read	Accessed for checking purposes
	Security CSD Link	Read	Accessed for checking purposes
	System Entity	Read	Accessed for checking purposes
	Reference Restriction	Read	Accessed for checking purposes
DYNAMIC DATA			
LCMM Instructions	Settlement Instruction	Write/Read	
	Settlement Restriction	Write/Read	
	Amendment Instruction	Write/Read	
	Cancellation Instruction	Write/Read	

All rights reserved.

DATA	DATA ENTITY	ACCESS MODE	COMMENTS
	Hold/Release Instruction	Write/Read	
	Validation Error	Write /Read	
	Instruction Pool	Write/Read	
	LCMM Instruction Link	Write/Read	
	Inbound LCMM Message	Write/Read	

3.4.4 Instruction Maintenance

3.4.4.1 Diagram of the module



3.4.4.2 Description of the module

The *Instruction Maintenance* module processes maintenance instructions, classified in the following types:

- *Amendment Instruction*: maintenance instructions intended to change the process indicators of a *Settlement Instruction*;

All rights reserved.

- *Cancellation Instruction*: maintenance instructions intended to trigger the cancellation of a *Settlement Instruction* or a *Settlement Restriction*. *Cancellation Instructions* also include those sent by *Administering Parties* with the purpose of cancelling *Settlement Instructions* identified as CoSD (*CoSD Cancellation Instructions*);
- *Hold / Release Instruction*: maintenance instructions intended to hold, release or partially release *Settlement Instructions*. *Hold / Release Instructions* also include those sent by *Administering Parties* with the purpose of releasing *Settlement Instructions* identified as CoSD (*CoSD Release Instructions*).

The *Instruction Maintenance* module processes maintenance instructions sent by T2S Actors to hold, release, amend or cancel *Settlement Instructions* or amend or cancel *Settlement Restrictions* until the actual settlement or cancellation of the referenced *Settlement Instruction* or *Settlement Restriction* occurs **{T2S.05.310}**.

The inputs for this module are:

- *Amendment Instruction;*
- *Cancellation Instruction,* including *CoSD Cancellation Instruction;*
- *Hold/Release Instruction,* including *CoSD Release Instruction;*
- *EOD Recycling and Purging Event.*

T2S Actors are allowed to send maintenance instructions regardless of whether the matching of the referenced *Settlement Instruction* was internal or external to T2S **{T2S.05.290}**.

When processing *Amendment Instructions* T2S does not perform any calculation **{T2S.05.400}**.

The module is composed of seven functions:

- *Maintenance Instruction Routing;*
- *Instruction Amendment;*
- *Instruction Cancellation;*
- *Instruction Cancellation by the System;*
- *Instruction Hold/Release;*
- *Conditional Securities Delivery (CoSD) Instruction Release;*
- *Maintenance Status Management.*

3.4.4.3 Description of the functions of the module

1 – Maintenance Instruction Routing

Introduction

This function receives the different types of maintenance instructions and forwards them to the relevant function depending on their type and sub-type.

The table below provides an overview of the *LCMM Instruction* types that are routed to the different functions:

All rights reserved.

LCMM INSTRUCTION	LCMM INSTRUCTION SUB-TYPE	INSTRUCTION MAINTENANCE FUNCTION
Amendment Instruction	N/A	Instruction Amendment
Cancellation Instruction	Party Cancellation	Cancellation of Settlement Instruction
	CoSD Cancellation	Cancellation of Settlement Instruction
	CSD Cancellation	Cancellation of Settlement Instruction
	Cancellation Instruction for Settlement Restriction	Cancellation of Settlement Restriction
Hold/Release Instruction	Hold	Instruction Hold/Release
	Release	
	CoSD Release	CoSD Instruction Maintenance Release

2 – Instruction Amendment

Introduction

Reference Id	***LCMM.IMA.INA.1.1***

This function processes *Amendment Instructions* that modify *Settlement Instructions or Settlement Restrictions in T2S*.

T2S Actors may modify the following process indicators of a *Settlement Instruction*: Partial Settlement Indicator, Priority, and linkages block [81], until actual settlement or cancellation occurs. Nevertheless, T2S Actor may only amend Priority of the pending part of a *Settlement Instruction* with Settlement Status "*Partially Settled*" {**T2S.05.390}**. For *Settlement Restrictions* only priority and linkages block can be modified and the Priority of the pending part of a *Settlement Restriction* with Settlement Status "*Partially Settled*".

The function processes *Amendment Instructions* intended to modify an existing *Settlement Instructions* or *Settlement Restrictions* by performing the following steps:

"Amendment Instructions to be Denied"

Reference Id	***LCMM.IMA.INA.2.1***

This step denies the execution of an *Amendment Instruction* when one of the following business rules is fulfilled **{T2S.05.390}**:

BUSINESS RULES	RESULTING AMENDMENT INSTRUCTION'S REASON CODE REFERENCE
The Settlement Status of the referenced *Settlement Instruction* or *Settlement Restriction* is "*Settled*".	*"Referenced Settlement Instruction / Settlement Restriction already settled"*

81 See Instruction Validation module for additional information.

All rights reserved.

The Settlement Status of the referenced *Settlement Instruction* or *Settlement Restriction* is *"Partially Settled"* and the modification does not refer to Priority.	*"Referenced Settlement Instruction / Settlement Restriction already partially settled"*
The Cancellation Status of the referenced *Settlement Instruction* or *Settlement Restriction* is *"Cancelled"*.	*"Referenced Settlement Instruction / Settlement Restriction already cancelled"*
The Approval Status of the referenced *Settlement Instruction* or *Settlement Restriction* is "*Awaiting Approval*".	*"Referenced Settlement Instruction has not been approved yet"*
The referenced Settlement Instruction is identified as CoSD and the *Amendment Instruction* does not aim to remove a linkage having the CSD as the Instructing Party.	*"Referenced Settlement Instruction identified as CoSD"*
The Approval Status of the referenced *Settlement Instruction* or *Settlement Restriction* is "*Revoked*".	*"Referenced Settlement Instruction / Settlement Restriction has not been approved"*

If the execution of the *Amendment Instruction* is denied, the step sets the Cancellation Status of the *Amendment Instruction* to *"Denied"* with the relevant Reason Code Reference as given in the table above and triggers the *Maintenance Status Management* function [82].

If the execution of the Amendment Instruction is not denied, the step triggers the "*Amendment Instruction to be Executed"* step.

"Amendment Instruction to be Executed"

Reference Id	***LCMM.IMA.INA.3.1***

This step executes the *Amendment Instruction* by:

- Setting the Execution Status of the *Amendment Instruction* to *"Executed"*;
- Modifying the referenced *Settlement Instruction* and *Settlement Transaction*;
- Triggering the *Maintenance Status Management* function.

3 – Instruction Cancellation

Introduction

82 The *Maintenance Status Management* function is triggered when a status is updated in order to inform *Status Management* with the relevant *Maintenance Status Information* or *Instruction Status Information*

All rights reserved.

Reference Id	***LCMM.IMA.INC.1.1***

This function processes *Cancellation Instructions* intended to cancel an existing *Settlement Instruction* or *Settlement Restriction* **{T2S.05.420}**.

Settlement Instructions or *Settlement Restrictions* can be cancelled upon request from the relevant T2S Actors prior to their settlement or their cancellation; regardless of whether matching of the *Settlement Instruction* has taken place or not **{T2S.05.560}**. A T2S Actor may cancel the pending part of the *Settlement Instruction* or *Settlement Restrictions* that have the Settlement Status *"Partially Settled"*"*Partially Settled"* **{T2S.05.420}**.

This function processes all types of *Cancellation Instructions*, i.e. Party *Cancellation Instructions* (those instructed by the T2S Party), CSD *Cancellation Instructions* (those instructed by the CSD) and CoSD *Cancellation Instructions* (those instructed by Administering Parties).

T2S Parties are not allowed to cancel *Settlement Instructions* when they have the CoSD Flag as "Yes", in this case, only CSDs and Administering Parties can request the Cancellation of a *Settlement Instruction*.

When processing CoSD *Cancellation Instructions*, if there is more than one Administering Party, T2S requires all the Cancellation Requests from all Administering Parties in order to cancel the CoSD *Settlement Instruction* **{T2S.05.470} {T2S.09.250}.** In contrast to the CoSD release processing, in order to cancel a CoSD Settlement Instruction the Administering Parties need to send to T2S only one CoSD *Cancellation Instruction* per Administering Party and per instruction regardless the number of conditions fulfilled.

This function is divided into two sub-functions:

- *Cancellation of Settlement Instructions;*
- *Cancellation of Settlement Restrictions.*

Cancellation of Settlement Instructions

Reference Id	***LCMM.IMA.INC.2.1***

This sub-function processes *Cancellation Instructions* intended to cancel an existing *Settlement Instruction*, performing the following steps:

Step "Cancellation Instruction to be Denied"

Reference Id	***LCMM.IMA.INC.2.2***

This step denies the execution of a *Cancellation Instruction* when one of the following business rules is fulfilled:

BUSINESS RULES	Resulting Cancellation Instruction's Reason code reference
The Settlement Status of the referenced *Settlement Instruction* is "*Settled*".	*"Referenced Settlement Instruction already settled"*
The Cancellation Status of the referenced *Settlement Instruction* is "*Cancelled*".	*"Referenced Settlement Instruction already cancelled"*

All rights reserved.

The Approval Status of the referenced *Settlement Instruction* is "*Revoked*".	*"Referenced Settlement Instruction has been Revoked"*
The Approval Status of the referenced *Settlement Instruction* is "*Awaiting Approval*".	*"Referenced Settlement Instruction has not been approved yet"*
The CoSD Flag of the referenced *Settlement Instruction* is "*Yes*" and it is a Party Cancellation.	*"Referenced Settlement Instruction on CoSD process".*
The Cancellation Requested Flag of a *Settlement Instruction* referenced by an unmatched T2S Party *Cancellation Instruction* must not be "*Yes*".	*"Referenced Settlement Instruction already requested for cancellation"*
There is a pending *Cancellation Instruction* from the same Instructing Party (either T2S Party, Administering Party or CSD).	*"Referenced Settlement Instruction already requested for cancellation by the same Instructing Party"*
A *CoSD Rule* applies over a *Realignment Instruction* of the referenced *Settlement Instruction*.	*"Realignment Instruction fulfils a CoSD Rule"*

If the execution of the *Cancellation Instruction* is denied, the step sets the Cancellation Status of the *Cancellation Instruction* to *"Denied"* with the relevant reason code reference as given in the table above and triggers the *Maintenance Status Management* function **{T2S.05.445}**.

If the execution of the *Cancellation Instruction* is not denied, the step triggers the *"Party and CSD Cancellation Instruction to be Processed"* step for Party and CSD *Cancellation Instructions* or "*CoSD Cancellation Instruction to be processed*" step for CoSD Cancellation Instructions **{T2S.05.420}**.

Step "Party and CSD Cancellation Instruction to be Processed"

Reference Id	***LCMM.IMA.INC.2.3***

The step executes the *Cancellation Instruction* or triggers further processing depending on the Match Status of the Settlement Instruction (i.e. if a bilateral Cancellation is needed or not). The result of the analysis is described in the table below **{T2S.05.440} {T2S.05.441} {T2S.05.442} {T2S.05.443} {T2S.05.445}**:

REFERENCED SETTLEMENT INSTRUCTION'S MATCH STATUS	RESULTING ACTION	RESULTING SETTLEMENT INSTRUCTION'S REASON CODE REFERENCE
"Unmatched"	Execute	*"Cancelled by the T2S Party "* or

All rights reserved.

		"Cancelled by the CSD"
"Matched"	Trigger bilateral cancellation step	N/A

If the Resulting Action is "Execute", this step:

- Sets the Execution Status of the *Cancellation Instruction* to *"Executed"*;
- Cancels the *Settlement Instruction* by setting its Cancellation Status to *"Cancelled"* and the Reason Code Reference to the value indicated in the table above.

If the Resulting Action is "Trigger bilateral cancellation step", the step triggers the *"Bilateral cancellation to be processed"* step.

Step "CoSD Cancellation Instruction to be processed"

Reference Id	***LCMM.IMA.INC.2.4***

This step analyses the number of received CoSD Cancellation Instruction(s) for the referenced Settlement Instruction. The result of the analysis is described in the table below [83]

BUSINESS RULES	RESULTING ACTION	RESULTING CANCELLATION INSTRUCTION'S REASON CODE REFERENCE
At least one Administering Party has not yet sent a *Cancellation Instruction*.	Keep Pending	*"Cancellation pending from Administering Party"*
All Administering Parties have sent their respective *Cancellation Instruction*.	Trigger bilateral cancellation step	No resulting reason code

If the Resulting Action is "Keep Pending", this step sets the Reason Code Reference of the Cancellation Instruction to the value indicated in the table above. [84]

If the Resulting Action is "Trigger bilateral cancellation step" this step triggers *"Bilateral cancellation to be processed"* step.

Step "Bilateral cancellation to be processed"

Reference Id	***LCMM.IMA.INC.2.5***

This step analyses the Cancellation requested flag of the counterparty´s Settlement Instruction in order to execute or not the Cancellation Instructions. The result of the analysis is described in the table below

BUSINESS RULES	RESULTING ACTION	RESULTING CANCELLATION INSTRUCTION'S REASON CODE REFERENCE

83 In order to check the number of CoSD Cancellation Instructions from the Administering Parties received, the function counts the number of CoSD Cancellations Instructions with Reason Code "CoSD cancellation pending from Administering Party".

84 The change in the Reason Code of the Execution status of the Cancellation Instruction is informed to relevant Administering Party that instructed the processed Cancellation Instruction (and its CSD) through the sese27 "Securities Transaction Cancellation Request Status Advice". T2S also sends a copy of this Status Advice to all the Administering Parties in order to inform them that this Administering Party wants to cancel the referred CoSD Settlement Instruction.

All rights reserved.

The counterparty's *Settlement Instruction* has the Cancellation Requested Flag as "*No*".	Keep Pending	"*Cancellation pending, awaiting cancellation from the counterparty*"
The counterparty's *Settlement Instruction* has Cancellation Requested Flag as "*Yes*".	Execute	No resulting reason code

If the Resulting Action is "Keep Pending", this step:

- Sets the Reason Code Reference [85] of all the *Cancellation Instructions* to the value indicated in the table above;
- Sets the Cancellation Requested Flag [86] of the referenced Settlement Instruction to "Yes".
- Triggers the *Maintenance Status Management* function, in order to inform *Status Management Module*. In order to allege the counterparty that the cancellation has been requested, T2S (*Status Management Module*) sends a status advice related to the *Settlement Instruction* of the counterparty. T2S sends a sese.024 message to the T2S Party, CSD and Administering Parties of the counterparty's *Settlement Instruction* informing about the Status *"Not Cancelled"* and Reason code *"Cancellation requested from the counterparty"*

If the Resulting Action is "Execute" this step:

- Sets the Execution Status of all the pending *Cancellation Instructions* to *"Executed"*.
- Cancels the referenced *Settlement Instruction* and its counterparty´s *Settlement Instruction* by setting the Cancellation Status to *"Cancelled"* and the Reason Code Reference to *"Cancelled by the T2S Party "*, *"Cancelled by the CSD"* or *"CoSD Cancellation"* [87];
- Sets the Match Status of both *Cancellation Instructions* to *"Matched"*, this and the counterparty´s,
- Cancels the related Realignment *Settlement Instruction*s in case there are any, by setting their Cancellation Status to *"Cancelled"*,
- Cancels the related *Settlement Transactions*, when the Intended Settlement Date has been reached, by setting their Cancellation Status to *"Cancelled"*;
- Cancels the related *Settlement Transaction*(s) of *Realignment Settlement Instructions* in case there are any, when the Intended Settlement Date has been reached, by setting their Cancellation Status to *"Cancelled"*;
- Triggers the *Maintenance Status Management* function.
- Performs the following additional activities in case the referenced *Settlement Instruction* is CoSD **{T2S.05.444}**:

85 The change in the Reason Code Reference of the *Cancellation Instruction* is informed to relevant T2S Actor that instructed it through the sese.027 message.

86 The change in the flag of the referenced *Settlement Instruction* is communicated to its T2S Party, CSD and Administering Parties through a sese.024 message "Securities Settlement Transaction Status Advice" informing about the Status "*Not Cancelled"* and Reason code *"Cancellation pending awaiting for cancellation from the counterparty"*

87 Depending on the type of the relevant *Cancellation Instructions* executed.

All rights reserved.

- Creates an Unblocking *Settlement Restriction* for all settled *Settlement Restrictions* and sends them to the *Settlement* domain (*Standardisation and Preparation to Settlement* module)
- Or cancels the unsettled CoSD *Settlement Restriction*(s), if the blocking has not taken place yet.

Cancellation of Settlement Restrictions

Reference Id	***LCMM.IMA.INC.3.1***

This sub-function processes *Cancellation Instructions* intended to cancel an existing *Settlement Restriction*, performing the following steps:

Step "Cancellation Instruction to be Denied"

Reference Id	***LCMM.IMA.INC.3.2***

This step denies the execution of the *Cancellation Instruction* when one of the following business rules is fulfilled:

BUSINESS RULES	RESULTING ACTION
The Settlement Status of the referenced *Settlement Restriction* is "*Settled*".	*"Referenced Settlement Restriction already settled"*
The Approval Status of the referenced *Settlement Restriction* is "*Awaiting Approval*".	*"Referenced Settlement Restriction has been Revoked"*
The Approval Status of the referenced *Settlement Restriction* is "*Revoked*".	*"Referenced Settlement Restriction has not been approved yet"*

If the execution of the *Cancellation Instruction* is denied, the step sets the Cancellation Status of the *Cancellation Instruction* to *"Denied"* with the relevant Reason Code Reference triggering the *Maintenance Status Management* function.

If the execution of the *Cancellation Instruction* is not denied, the step triggers the "*Cancellation Instruction to be Executed*" step.

Step "Cancellation Instruction to be Executed"

Reference Id	***LCMM.IMA.INC.3.3***

This step executes the *Cancellation Instruction* by:

- Setting the Execution Status of the *Cancellation Instruction* to "Executed";
- Cancelling the referenced *Settlement Restriction* by setting its Cancellation Status to *"Cancelled"* and the Reason Code Reference to *"Cancelled by the T2S Actor"*;
- Cancelling the *Settlement Transaction* if the Intended Settlement Date has been reached;
- Triggering *Maintenance Status Management* function.

4 – Instruction Cancellation by the System

All rights reserved.

Reference Id	***LCMM.IMA.ICS.1.1***

This function cancels *Settlement Instructions*, *Settlement Restrictions and Cancellation Instructions* after a standard period of time as defined in Reference Data **{T2S.11.910}**.

The function is triggered by the *EOD Recycling and Purging Event* received from the *Scheduling* module.

T2S supports two separate recycling periods for all T2S Actors:

- Recycling period for unmatched instructions:

This period defines the minimum time the system keeps an unmatched instruction in the system. For *Settlement Instructions*, the period starts on the Intended Settlement Date or the date of the last status change of the instruction depending on which date is later while, for *Cancellation Instructions*, the period starts on the Entry Date or the date of the last status change, depending on which one is later. . In line with ESSF/ECSDA recommendations T2S uses the default value of 20 working days.

This period applies to *"Unmatched" Settlement Instructions* and *Cancellation Instructions* **{T2S.05.430}**.

- Recycling period for matched instructions:

This period defines the minimum time the system keeps matched instructions in the system. The period starts on the Intended Settlement Date or the date of the last status change of the instructions depending on which date is later. T2S uses the default value of 60 working days.

This period applies to *"Unsettled" "Matched" Settlement Instructions and "Unsettled"Settlement Restrictions* **{T2S.05.460}**.

The function selects *Settlement Instructions and Settlement Restrictions that fulfil the criteria defined below including those instructions with the Approval Status "Awaiting approval", Settlement Instructions and Settlement Restrictions that fulfil the criteria defined below including those instructions with the Approval Status "Awaiting approval",* that have exceeded the corresponding T2S recycling period if any in order to cancel these instructions (and their underlying transactions as well as any linked T2S generated *Settlement Instruction(s)* when relevant).

INSTRUCTION TYPE	RELEVANT STATUSES	RELEVANT RECYCLING PERIOD	RESULTING CANCELLATION STATUS	RESULTING REASON CODE REFERENCE
Settlement Instruction	Match Status: *"Unmatched"* and Cancellation Status: *"Not Cancelled"*	For unmatched instructions	Cancellation Status: *"Cancelled"*	*"Standard Period for unmatched instructions reached"*
Settlement Instruction	Match Status: *"Matched"* and Settlement Status: *"Unsettled"* and Cancellation Status:	For matched instructions	Cancellation Status: *"Cancelled"*	*"Recycling Period reached"*

All rights reserved.

INSTRUCTION TYPE	RELEVANT STATUSES	RELEVANT RECYCLING PERIOD	RESULTING CANCELLATION STATUS	RESULTING REASON CODE REFERENCE
	"Not Cancelled"			
Settlement Restriction	Settlement Status: *"Unsettled"* and Cancellation Status: *"Not Cancelled"*	For matched instructions	Cancellation Status: *"Cancelled"*	*"Recycling Period reached"*
Cancellation Instruction	Match Status: *"Unmatched"* and Cancellation Status: *"Not Denied"* and Execution Status: *"Not executed"*	For unmatched instructions	Cancellation Status: *"Denied"*	*"Standard Period for unmatched instructions reached"*

When the recycling period for a *"Matched" and "Unsettled" Settlement Instruction* or an *"Unsettled" Settlement Restriction* is exceeded, the function cancels the considered instruction and its underlying *Matching Object* and *Settlement Transaction(s) if any*. In case the function cancels a *"Matched" and "Unsettled" Settlement Instruction*, the function analyses if the cancelled *Settlement Instruction* has any T2S generated *Settlement Instruction* for realignment purposes associated, in order to cancel them and its underlying *Matching Objects* and *Settlement Transaction*(s) if any. If there are T2S generated *Settlement Instructions* associated to the initial *Settlement Instruction*, the function cancels all of them (i.e. sets their Cancellation Status to *"Cancelled"*). Finally, the function triggers the *Maintenance Status Management* function.

As an exception, matched pending settlement instructions will not be eligible for automatic cancelation if at least one of the relevant CSDs of the transaction (i.e. the delivering CSD, the receiving CSD, or both) is an external CSD, is defined as the issuer of the security, and is not compliant with the automatic cancelation rule configured in reference data.
When the recycling period for an "Unmatched" *Settlement Instruction* or *Cancellation Instructions* is exceeded, the function cancels the *Settlement Instruction* or *Cancellation Instruction* and triggers the *Maintenance Status Management* function.

After completing the processing this function notifies the *Scheduling* module by sending the *End of Process* Event.

5 – Instruction Hold/Release

Introduction

Reference Id	***LCMM.IMA.HRM.1.1***

The purpose of the Instruction Hold/Release function is to process *Hold or Release Instructions* from T2S Actors **{T2S.05.350}**.

All rights reserved.

The *Instruction Hold/Release* function allows putting on hold or releasing *Settlement Instructions* until their actual settlement or cancellation occurs, even beyond their Intended Settlement Date **{T2S.05.370}**. Under specific conditions, this function also allows to partially release a Settlement Instruction.

The function supports putting On Hold as well as releasing "*Partially Settled*" *Settlement Instructions* **{T2S.05.350}**.

The function processes *Hold/Release Instructions* related to the Party Hold Status or CSD Hold Status of *Settlement Instructions* and Release Instructions related to the Party Hold Status that intend to partially release a Settlement Instruction **{T2S.05.360}**.

The function also processes release instructions related to the CSD Validation Hold **and Party Hold** Status of *Settlement Instructions* automatically put On Hold by T2S for additional validation of the relevant CSD **{T2S.05.127}{CR-812}**.

The function is divided in two sub-functions:

- *Processing of Hold Instructions;*
- *Processing of Release Instructions.*

Processing of Hold Instructions

Reference Id	***LCMM.IMA.HRM.2.1***

This sub-function processes *Hold Instructions* intended to hold existing *Settlement Instructions*, performing the following steps:

"Hold Instruction to be Denied"

Reference Id	***LCMM.IMA.HRM.2.2***

This step denies the execution of the *Hold Instruction* when one of the following business rules is met **{T2S.05.380}**:

BUSINESS RULES	RESULTING HOLD INSTRUCTION'S REASON CODE REFERENCE
The Settlement Status of the referenced *Settlement Instruction* is "*Settled*"	"*Referenced Settlement Instruction already settled*"
The Cancellation Status of the referenced *Settlement Instruction* is "*Cancelled*"	"*Referenced Settlement Instruction already cancelled*"
The Approval Status of the referenced *Settlement Instruction* is "*Awaiting Approval*"	"*Referenced Settlement Instruction has not been approved yet*"
The relevant hold status (Party Hold Status or CSD Hold Status) of the referenced *Settlement Instruction* is already set to "*Yes*"	"*Referenced Settlement Instruction already on hold*"

All rights reserved.

The referenced *Settlement Instruction* is identified as CoSD	"*Referenced Settlement Instruction identified as CoSD*"
The Approval Status of the referenced *Settlement Instruction* is "*Revoked*"	"*Referenced Settlement Instruction has been "Revoked*"

If the execution of the *Hold Instruction* is denied, the step sets the Cancellation Status of the *Hold Instruction* to *"Denied"* with the relevant Reason Code Reference as given in the table above and triggers the *Maintenance Status Management* function.

If the execution of the *Hold Instruction* is not denied, the step triggers the "*Hold Instruction to be Executed"* step.

"Hold Instruction to be Executed"

Reference Id	***LCMM.IMA.HRM.2.3***

This step executes the *Hold Instruction* by:

- setting the Execution Status of the Hold Instruction to *"Executed",*
- putting the *Settlement Instruction* On Hold by setting its relevant hold status (Party Hold Status or CSD Hold Status) to *"Yes"*;
- putting the *Settlement Transaction* On Hold if the Intended Settlement Date has been reached;
- triggering the *Maintenance Status Management* function.

If the Party Hold Status or CSD Hold Status of a *Settlement Instruction* is *"Yes"*, other maintenance instructions can be executed on the same instruction.

If the Party Hold Status or CSD Hold Status of a *Settlement Instruction is "Yes"* at the end of its Intended Settlement Date, the instruction is recycled according to the T2S Recycling Periods for unmatched and matched instructions **{T2S.05.370}**.

Processing of Release Instructions

Reference Id	***LCMM.IMA.HRM.3.1***

This sub-function processes *Release Instructions* intended to release existing *Settlement Instructions already put On Hold* (Party Hold Status, CSD Hold Status, or CSD Validation Hold Status), and Release Instructions that intend to Partially Release a Settlement Instruction already on Party Hold, by performing the following steps:

"Release Instructions to be Denied"

Reference Id	***LCMM.IMA.HRM.3.2***

This step denies the execution of the *Release Instruction* when one of the following business rules is met:

BUSINESS RULES	RESULTING HOLD INSTRUCTION'S REASON CODE REFERENCE

All rights reserved.

The Settlement Status of the referenced *Settlement Instruction* or Settlement Restriction is "*Settled*".	"*Referenced Settlement Instruction or Settlement Restriction already settled*"
The Cancellation Status of the referenced *Settlement Instruction* or Settlement Restriction is "*Cancelled*".	"*Referenced Settlement Instruction or Settlement Restriction already cancelled*"
The *Approval Status* of the referenced *Settlement Instruction* or *Settlement Restriction* is "*Awaiting Approval*".	"*Referenced Settlement Instruction or Settlement Restriction has not been approved yet*"
The relevant hold status of the referenced *Settlement Instruction* (Party Hold Status, CSD Hold Status, or CSD Validation Hold Status) is already set to "*No*".	"*Referenced Settlement Instruction or Settlement Restriction already released*"
The Approval Status of the referenced *Settlement Instruction* is "*Revoked*".	"*Referenced Settlement Instruction has been "Revoked*"
The Settlement Status of the referenced *Settlement Instruction* or *Settlement Restriction* is "*Settled*".	"*Referenced Settlement Instruction or Settlement Restriction already settled*"

In the specific case of a Release Instruction that intends to Partially Release a Settlement Instruction (i.e. a Release Instruction informing a quantity lower than the original settlement quantity stated in the underlying Settlement Instruction), additionally to the business rules described above, this step denies the execution of the Release Instruction when one of the following business rules is met:

BUSINESS RULES	RESULTING HOLD INSTRUCTION'S REASON CODE REFERENCE
If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the referenced Settlement Instruction, neither the referenced Settlement Instruction must be partially released nor a pending Release Instruction of the same Hold type must be present in the system	The referenced Settlement Instruction is already partially released or there is a pending Release Instruction of the same Hold type in the system.
If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the ref-	The referenced Settlement Instruction is unmatched

All rights reserved.

erenced Settlement Instruction, the matching status of the referenced Settlement Instruction must not be 'Unmatched'	
If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the referenced Settlement Instruction, no other holds must apply neither on the referenced Settlement Instruction nor on the counterpart	Other holds apply either on the referenced Settlement Instruction or on the counterpart.
If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the referenced Settlement Instruction, the referenced Settlement Instruction and its counterpart must allow partial settlement	Either the referenced Settlement Instruction or its counterpart do not allow partial settlement.
If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the referenced Settlement Instruction, no settlement links must apply neither on the referenced Settlement Instruction nor on its counterpart	A settlement link applies either on the referenced Settlement Instruction or on its counterpart.
If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the referenced Settlement Instruction, no non-reciprocal links must apply neither on the referenced Settlement Instruction nor on its counterpart.	A non-reciprocal link applies either on the referenced Settlement Instruction or on its counterpart.
If the quantity present in a release instruction is lower than the settlement quantity stated in the referenced settlement instruction, it must comply with partial settlement rules.	Partial release instruction is denied because it does not comply with partial settlement rules.

All rights reserved.

If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the referenced Settlement Instruction, the related cut-off must not have been reached.	Partial release instruction is denied because the relevant cut-off has been reached.
If the quantity present in a Release Instruction is lower than the Settlement Quantity stated in the referenced Settlement Instruction, none of the associated realignment Settlement Instruction must be subject to any hold	At least one associated realignment Settlement Instruction is on hold.

If the execution of the *Release Instruction* is denied, the step sets the Cancellation Status of the *Release Instruction* to *"Denied"* with the relevant Reason Code Reference as stated in the table above and triggers the *Maintenance Status Management* function.

If the execution of the *Release Instruction* is not denied, the step triggers the "*Release Instruction to be Executed"* step.

"Release Instruction to be Executed"

Reference Id	***LCMM.IMA.HRM.3.3***

The step executes the *Release Instruction* by:

- setting the Execution Status of the *Release Instruction* to *"Executed"*;
- releasing the *Settlement Instruction* by setting its relevant Hold Status (Party Hold Status, CSD Hold Status, or CSD Validation Hold Status) to *"No"*. If a release instruction that intends to partially release a Settlement Instruction is executed, the Party Hold Status of the Settlement Instruction will remain as "Yes" during the partial release process and the Settlement Instruction will be partially released for the quantity stated in the release instruction.
- If the Intended Settlement Date of the *Settlement Instruction* has been reached, performing the following additional activities in case there is no remaining Party Hold Status or CSD Hold Status or CSD Validation Hold Status on the *Settlement Instruction* and on the *Settlement Instruction* of the counterparty in case of *Settlement Instruction* with the Match Status *"Matched"*:
  - releasing the *Settlement Transaction,*
  - sending a *Transaction release* flow to the *Standardisation and Preparation to Settlement* module of the *Settlement* domain with the reference number of the *Settlement Transactions* released;
- triggering the *Maintenance Status Management* function.

6 – Conditional Securities Delivery (CoSD) Instruction Release

Introduction

All rights reserved.

Reference Id	***LCMM.IMA.CRC.1.1***

This function processes the Administering Parties' maintenance instructions intended to release or cancel a *Settlement Instruction* identified as a CoSD. A CoSD *Settlement Instruction* is a *Settlement Instruction* that has been identified by T2S as a CoSD on its Intended Settlement Date based on criteria rules defined in T2S Reference Data.

When a *Settlement Instruction* is identified as CoSD on its Intended Settlement Date, T2S also identifies an Administering Party for each condition applying on the *Settlement Instruction*.

In order to settle a CoSD *Settlement Instruction* put on CoSD hold, after the blocking of securities, cash or securities and cash, the Administering Party/ies must release a *Settlement Instruction* by sending a release per condition (*i.e. Administering Parties only need to send their releases for the leg that triggered the CoSD rule)*. T2S must receive all the releases requests for all the conditions associated in order to release the CoSD *Settlement Instructions*, e.g.: Two CoSD rules of two Administering Parties (Rule 1 of Administering Party 1 and Rule 2 of Administering Party 2) are fulfilled by two matched Settlement Instructions independently (Rule 1 of Administering Party 1 is fulfilled by Settlement Instruction 1 and Rule 2 of Administering Party 2 is fulfilled by Settlement Instruction 2), then each Administering Party has to send one Release Instructions (i.e. T2S has to receive 2 Release Instructions).

The function processes *CoSD Release Instructions* intended to release a Settlement Instruction on CoSD Hold by performing the following steps:

This function is divided in two sub-functions:

- *Processing of CoSD Cancellation Instructions*
- *Processing of CoSD Release Instructions*

Processing of CoSD Release Instructions

Reference Id	***LCMM.IMA.CRC.3.1***

This sub-function processes CoSD *Release Instructions* intended to release existing *Settlement Instructions* identified as CoSD, performing the following steps:

"CoSD Release Instruction to be Denied"

Reference Id	***LCMM.IMA.CRC.3.2***

This step denies the execution of the CoSD *Release Instruction* when one of the following business rules is met:

BUSINESS RULES	RESULTING HOLD INSTRUCTION'S REASON CODE REFERENCE
The Settlement Status of the referenced *Settlement Instruction* is "*Settled*"	*"Referenced Settlement Instruction already settled"*
The Cancellation Status of the referenced *Settlement Instruction* is "*Cancelled*"	*"Referenced Settlement Instruction already cancelled"*

All rights reserved.

The Administering Party has already released the CoSD Rule for the referenced *Settlement Instruction*	*"Referenced CoSD Rule already released"*
The cash balance of a *Settlement Instruction* referenced by a *CoSD Release Instruction* must be blocked.	*"Waiting for CoSD cash blocking"*
The securities position of a *Settlement Instruction* referenced by a *CoSD Release Instruction* must be blocked.	*"Waiting for CoSD securities blocking"*
The cash balance and the securities position of a *Settlement Instruction* referenced by a *CoSD Release Instruction* must be blocked.	*"Waiting for CoSD securities and cash blocking"*
The Hold statuses, except the CoSD Hold Status, of a *Settlement Instruction* referenced by a *CoSD Release Instruction* or its matched *Settlement Instruction*, must be "No"	*"Any other Hold Status apart from the CoSD Hold Status is set to "Yes"."*

If the execution of the CoSD *Release Instruction* is denied, the step sets the Cancellation Status of the *Release Instruction* to *"Denied"* with the relevant Reason Code Reference as indicated in the table above and triggers the *Maintenance Status Management* function.

If the execution of the *Release Instruction* is not denied, the step triggers "*CoSD Release Instruction to be Processed"* step.

"CoSD Release Instruction to be Processed"

Reference Id	***LCMM.IMA.CRC.3.3***

This step analyses the number of received CoSD *Release Instruction(s)* for the referenced *Settlement Instruction and on the Settlement Instruction of the counterparty* in order to execute the CoSD *Release Instruction* or trigger a different processing. The result of the analysis is described in the table below.

BUSINESS RULES	RESULTING ACTION	RESULTING COSD RELEASE INSTRUCTION'S REASON CODE REFERENCE
At least one Administering Party CoSD *Release Instruction* has not been received.	Keep Pending	*"CoSD release pending from Administering Party"*

All rights reserved.

All Administering Parties have released their respective CoSD Rules (i.e. sent a *Release Instruction* for each relevant CoSD Rule fulfilled they administer.	*Execute*	No resulting reason code

If the Resulting Action is "Keep Pending", this step:

- sets the Reason Code Reference of the CoSD *Release Instruction* to the value indicated in the table above;
- triggers the *Maintenance Status Management function*.

If the Resulting Action is "*Execute*" the step triggers the CoSD *Release Instruction* for all the *Release Instructions* received to be executed.

"CoSD Release Instruction to be Executed"

Reference Id	***LCMM.IMA.CRC.3.4***

This step executes the CoSD *Release Instruction* by:

- setting the Execution Status of the CoSD *Release Instructions* to *"Executed"*;
- perform the following additional activities:
  - releasing the *Settlement Instructions* by setting its CoSD Hold Status to *"No"*;
  - releasing the *Settlement Transaction*;
  - sending a *Transaction release* flow to the *Standardisation and Preparation to Settlement* module of the *Settlement* domain with the reference number of the *Settlement Transactions* released;
- triggering the Maintenance Status Management function.

7 – Maintenance Status Management

Reference Id	***LCMM.IMA.MSM.1.1***

This function forwards all *Instruction Status Information* and *Maintenance Status Information* to the *Status Management* module in order to inform the relevant T2S Parties about the updates of the statuses of referenced instructions and maintenance instructions processed in this module {**T2S.05.385**} {**T2S.05.405**} {**T2S.05.480**}.

3.4.4.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Maintenance Instruction	IN	Amendment Instruction, Cancellation Instruction, Hold/Release Instruction, CoSD Release Instruction	LCMM: Instruction Validation	
Event (EOD Recycling and Purging)	IN	End of Day	OPSR: Scheduling	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
End of Process Event (EOD Recycling and Purging)	OUT	Instruction Cancellation has finished		OPSR: Scheduling
Settlement Restriction	OUT	Unblocking Settlement Restriction for the cancellation of a CoSD Settlement Instruction		SETT: SPS
Maintenance Status information	OUT	Update of any maintenance instruction's status value. Includes reason code.		LCMM: Status Management
Transaction Release	OUT	Includes the reference number of the Settlement Transactions released		SETT: SPS
Instruction Status Information	OUT	Update of the Settlement Instruction or Settlement restriction's status value. Includes reason code		LCMM: Status Management

3.4.4.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
CRDM			
Reference Data	CoSD Rule	Read	
	Party	Read	
	Attribute Domain	Read	Contains information on the Recycling period to be applied on Settlement Instructions, Settlement Restrictions and Cancellation Instructions
DYNAMIC DATA			
LCMM Instructions	Settlement Instruction	Read / Write	Settlement Instruction referenced in a maintenance instruction

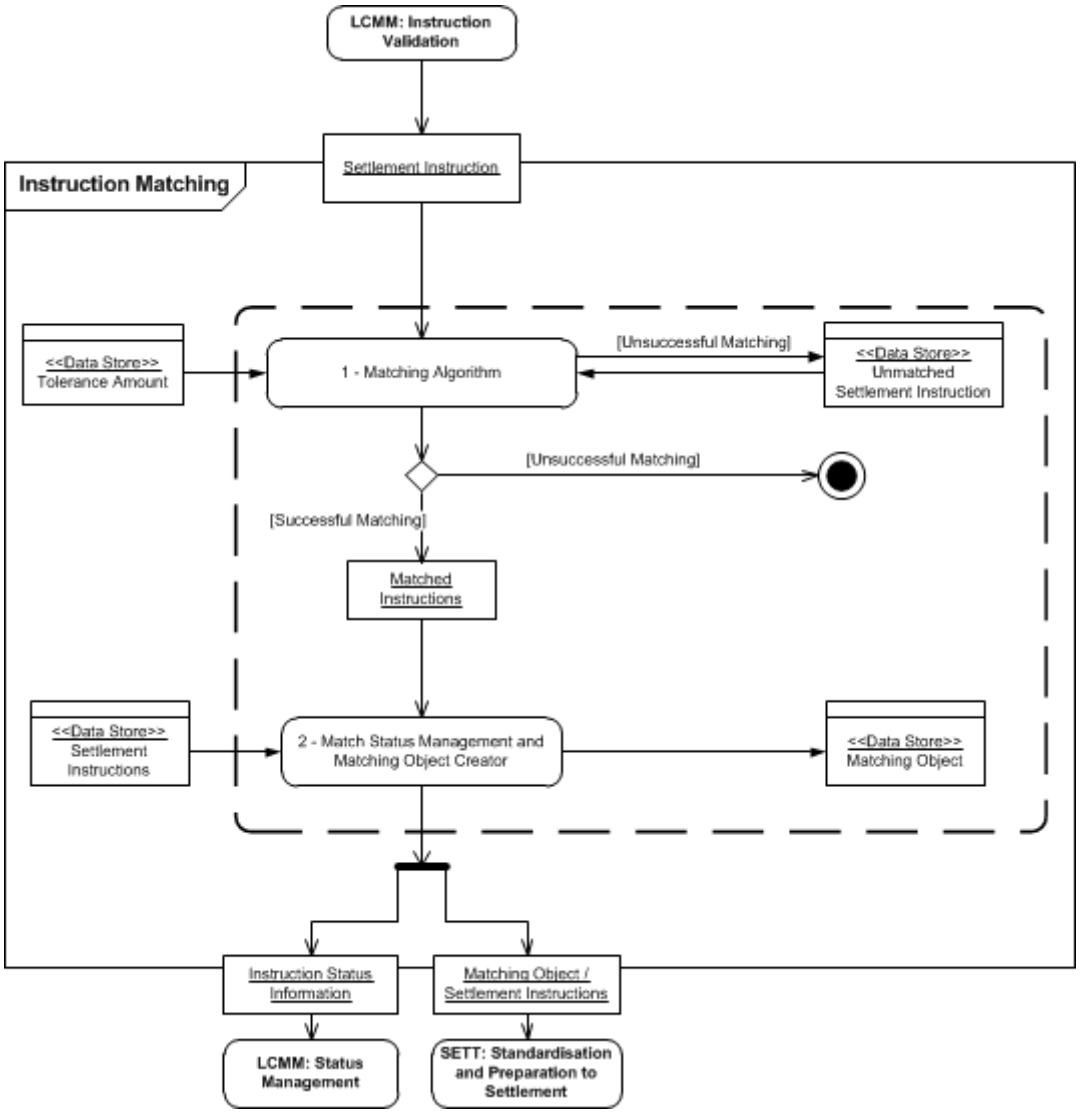
All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
	Settlement Restriction	Read / Write	Settlement Restriction referenced in a maintenance instruction
	Condition Modification Instruction	Read / Write	Modify the Status or Reason code of a Condition Modification Instruction (Amendment Instruction or Hold/Release Instruction)
	Cancellation Instruction	Read / Write	Modify Status or reason code of a Cancellation Instruction
	LCMM Instruction Link	Read/Write	
	Amendment Instruction	Read	
	Hold/Release Instruction	Read	

All rights reserved.

3.4.5 Instruction Matching

3.4.5.1 Diagram of the module



3.4.5.2 Description of the module

The *Instruction Matching* module matches *Settlement Instructions* that require to be matched in T2S (i.e. *Settlement Instructions* with Match status *"Unmatched"*) {**T2S.05.500}**. The module creates a *Matching Object* for each pair of matched *Settlement Instructions* after matching. Afterwards the module forwards the *Matching Object* and the relating *Settlement Instructions* to the *Standardisation and Preparation to Settlement* module for settlement. In addition, the module forwards the *Instruction Status Information* of the matched *Settlement Instructions* to the *Status Management* module in order to inform the relevant T2S Actor/s about the matching of their *Settlement Instructions* {**T2S.05.530}**.

When a cancellation of *Settlement Instructions* requires bilateral agreement, the validation that both *Cancellation Instructions* have been received is ensured by the *Instruction Maintenance* module and not by the *Instruction Matching* module.

All rights reserved.

3.4.5.3 Description of the functions of the module

The module is composed of two functions:

- *Matching Algorithm;*
- *Match Status Management and Matching Object Creator.*

1 – Matching Algorithm

Reference Id	***LCMM.IMH.MAA.1.1***

The *Matching Algorithm* function compares the *Settlement Instruction*'s fields provided by the buyer and the seller of securities in real-time to validate the equality of the settlement terms of the respective *Settlement Instructions* **{T2S.05.490}**.

When *Settlement Instructions* are set on hold, matching can still occur.

After receiving a *Settlement Instruction*, the function searches in the repository of *Unmatched Settlement Instructions* for those *Settlement Instructions* that have:

- the same mandatory matching field values;
- the same additional matching field values;
- the same optional matching field values only if the optional matching field value is not blank.

The following tables identify the exhaustive matching fields per instruction type [88] that are compared by this function.

Mandatory matching fields **{T2S.05.570}**:

DVP/DWP	FOP
Payment Type Code [89]	
Securities Movement Type [90]	
Intended Settlement Date	
Trade Date	
Settlement Quantity [91]	
T2S Security Identifier derived from the ISIN Code [92]	
Delivering Party BIC Code	
Receiving Party BIC Code	
Delivering CSD BIC Code	
Receiving CSD BIC Code	

88 Instruction types and its definition under ISO standard is described in the introduction of the *LCMM* domain.

89 The combination of Payment Type Code and Securities Movement Type is equivalent to Instruction Type Code in URD.

90 Equivalent to Deliver/Receive (for FOP) and Buy/Sell (for DVP/DWP) in URD.

91 Equivalent to Share Quantity (for equities) or Nominal Amount (for fixed income securities) in URD.

92 T2S Security Identifier is the internal identification of the Security in T2S

All rights reserved.

DVP/DWP	FOP
Currency	N/A
Original Settlement Amount	N/A
Credit/Debit Code	N/A

For each *Settlement Instruction* the *Matching Algorithm* function performs the matching comparison process of all matching field values except for the Original Settlement Amount field [93] for which the matching rules allow for a certain tolerance {**T2S.05.580**}.

Non-mandatory matching fields **{T2S.05.590}:**

The function verifies the two types of non-mandatory matching fields, if applicable:

- Additional matching fields
- Additional matching fields become mandatory when at least one of the *Settlement Instructions* includes a value in these fields.

DVP/DWP	FOP
Opt-out Settlement Transaction Condition Indicator	
CUM/EX Indicator	
n/a	Currency
n/a	Settlement Amount
n/a	Credit/Debit

Setting the Opt-out Settlement Transaction Condition Indicator allows T2S Actors to specify in the *Settlement Instructions* that no claim should be raised on the particular transaction and to indicate deviation from the standard transformation procedures.

The CUM/EX Indicator enables T2S Actors to indicate if they wish to override the standard market practice for market claim detection.

- Optional matching fields
- An optional matching field becomes mandatory field only if both *Settlement Instructions* include values for the field.

DVP/DWP	FOP
Common trade Reference	
Client of delivering CSD participant [94]	
Client of receiving CSD participant [95]	
Securities Account of the delivering Party	

93 The Settlement Amount check is the last one to be performed, as this field allows a certain tolerance, while all other matching fields do not allow this.

94 The ESF/ECSDA standards refer to "second layer market participant (sub-account/customer of counterparty)".

95 The ESF/ECSDA standards refer to "second layer market participant (sub-account/customer of counterparty)".

All rights reserved.

DVP/DWP	FOP
Securities Account of the receiving Party	

As a consequence of this comparison process, there are only two possible results:

- No *Unmatched Settlement Instruction* is found: the matching process is stopped and the *Settlement Instruction* is stored in the repository of *Unmatched Settlement Instructions*;
- One or several *Unmatched Settlement Instructions* are found in the repository of *Unmatched Settlement Instructions* with the same matching field values as the processed *Settlement Instruction*.

In case one or several possible *Unmatched Settlement Instructions* were identified, the function then compares the value of the Original Settlement Amount field of the processed *Settlement Instruction* with the value of this field of the above-identified *Unmatched Settlement Instructions*.

Among the identified *Unmatched Settlement Instructions*, the function selects the one having the smallest Original Settlement Amount difference with the processed *Settlement Instruction*, provided that the difference is within the band of the tolerance amount established for the specific currency in Reference Data **{T2S.05.580}**. The tolerance amount set up in T2S has two different bands per currency, depending on the Original Settlement Amount **{T2S.05.580}**. The tolerance suggested by ECSDA (European Central Securities Depository Association) is 25€ when the Original Settlement Amount is above 100.000€ or 2€ when it is less or equal to 100.000€. In case the relevant tolerances of two Original Settlement Amounts are not equal, the more restrictive one should apply.

When there is more than one identified *Unmatched Settlement Instruction* with equal values for the matching fields and the same values for the Original Settlement Amount within the band of the tolerance amount, the function selects the *Unmatched Settlement Instruction* with the closest entry time in T2S (based on the T2S instruction reference) to the processed *Settlement Instruction*.

Matched *Settlement Instructions* with different Original Settlement Amounts are submitted for settlement with a Matched settlement amount that equals the Original Settlement Amount of the Delivering Party's *Settlement Instruction* **{T2S.05.580}**.

e.g.: The Settlement Instruction "A" (Delivering Party) is processed in the Matching Algorithm and the function identifies three Settlement Instructions ("B", "C" and "D") with the same matching fields (except for the Original Settlement Amount, that is not checked at this point) in the repository of Unmatched Instructions.

All Settlement Instructions have equal difference of Original Settlement Amount and they are within the band of the tolerance amount of "A" (25€). Instructions "B" and "C" are as well within their own band of tolerance with "A" but not "D" because its tolerance is 2€. Then the Matching Algorithm checks between "B" and "C" the one with the closest time to "A".

SETTLEMENT INSTRUCTION	ORIGINAL SETTLEMENT AMOUNT	INSTRUCTING PARTY	ENTRY TIME
Settlement Instruction "A"	100.025,00 €	Delivering Party	13:00 20/01/2012
Settlement Instruction "B"	100.050,00 €	Receiving Party	19:00 19/01/2012
Settlement Instruction "C"	100.050,00 €	Receiving Party	13:52 19/01/2012

All rights reserved.

SETTLEMENT INSTRUCTION	ORIGINAL SETTLEMENT AMOUNT	INSTRUCTING PARTY	ENTRY TIME
Settlement Instruction "D"	100.000,00 €	Receiving Party	15:15 19/01/2012

In this example Settlement Instruction "B" is selected because it has the lowest difference of Original Settlement Amount with "A" (difference is in line with the tolerance amount for euros) and has the closest entry time to "A".

Then T2S matches "A" with "B" and the Matched settlement amount is equal to 100.025,00 € (Original Settlement Amount of the Delivering Party).

The following could result from the Matching Algorithm:

- An Unmatched Settlement Instruction is found: The selected *Unmatched Settlement Instruction* is removed from the repository of *Unmatched Settlement Instructions* and the *Matching Object* is created simultaneously by the *Match Status Management and Matching Object Creator* function.
- No corresponding *Unmatched* Settlement Instruction is found: The processed *Settlement Instruction* is stored in the repository of *Unmatched Settlement Instructions*.

2 – Match Status Management and Matching Object Creator

Match Status Management and Matching Object Creator:

Reference Id	***LCMM.IMH.MST.1.1***

Once the comparison process is successful, this function sets the Match status attribute of the matched *Settlement Instructions* to *"Matched"*, assigns a T2S Matching Reference and creates the *Matching Object* linked to both *Settlement Instructions* **{T2S.05.520}**. It is ensured that the creation of the *Matching Object* and the removal of the corresponding entry from the *Unmatched Settlement Instruction* the data store are performed simultaneously.

The *Matching Object* includes the T2S Matching Reference, the Matched settlement amount, the *Matching Type* [96] and the *Cancellation Status* of both *Settlement Instructions*.

The function forwards the *Matching Object* with its associated *Settlement Instructions*, to the *Standardisation and Preparation to Settlement* module. The function also sends an *Instruction Status Information*, relating to the *Match status "Matched"*, to the *Status Management* module.

When T2S creates T2S generated *Settlement Instructions* in case of cross-CSD instructions or auto-collateralisation, the T2S Matching Reference is the same for all *Matching Objects* linked to these T2S generated *Settlement Instructions* and for the *Matching Object* linked to the original *Settlement Instructions* received from the T2S Actors **{T2S.05.174} {T2S.05.528}**.

96 The Matching Type allows specifying whether the Matching Object links Settlement Instructions generated in T2S (i.e. realignments instructions) [Technical] or Settlement Instructions sent by T2S Actors [Business].

All rights reserved.

3.4.5.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Settlement Instruction	IN	Once the Settlement Instruction is generated and its Match status is "Unmatched"	LCMM: Instruction Validation	
Instruction Status Information	OUT	Update of the Settlement Instruction status		LCMM: Status Management
Matching Object + Settlement Instructions	OUT	After all the matching procedures have successfully been completed (pair of matched Settlement Instructions found)		SETT: Standardisation and Preparation to Settlement

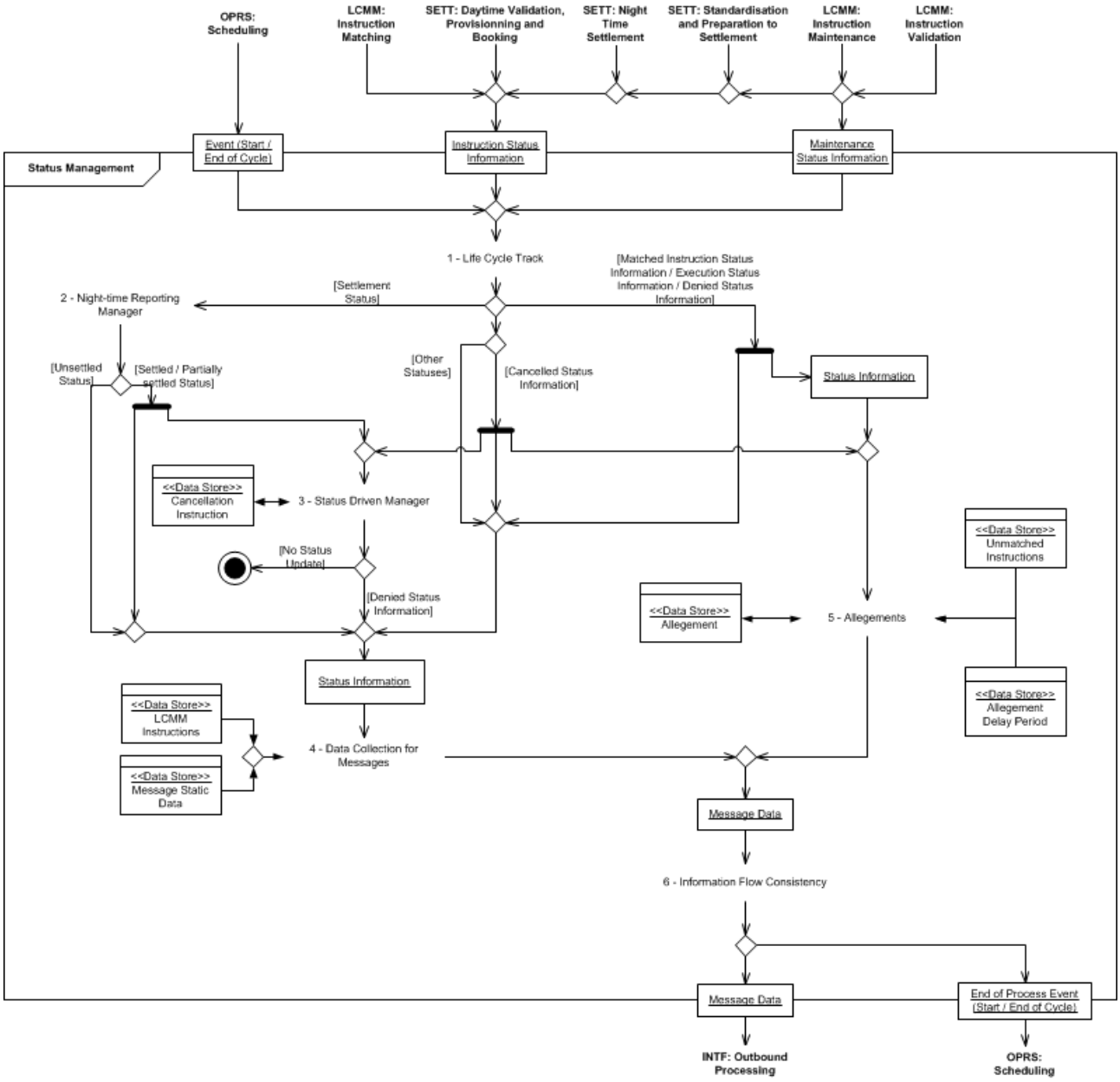
3.4.5.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
CRDM			
Tolerance Amount	Tolerance Amount	Read	For each currency there are two different bands
DYNAMIC DATA			
Unmatched Settlement Instruction	Unmatched Settlement Instruction	Read/Write/Delete	Settlement Instructions to be matched
Matching Object	Matching Object	Write	For each pair of matched Settlement Instructions
Settlement Instructions	LCMM Instruction	Read	For reading matching fields of the processed Settlement Instruction
	Settlement Instruction	Read/Write	For reading matching fields of the processed Settlement Instruction and for updating the Match status

All rights reserved.

3.4.6 Status Management

3.4.6.1 Diagram of the module



3.4.6.2 Description of the module

Introduction

The *Status Management* module receives all the status updates of all *LCMM Instructions* (*Settlement Instructions*, *Settlement Restrictions*, *Amendment Instructions, Hold/Release Instructions* and *Cancellation Instructions*) performed in *LCMM* and *Settlement* domain. It processes the data related to these updates, collecting if needed additional information in order to forward them in a consistent order to the *Interface* domain (*Outbound Processing* module).

Settlement Instruction status attributes

All rights reserved.

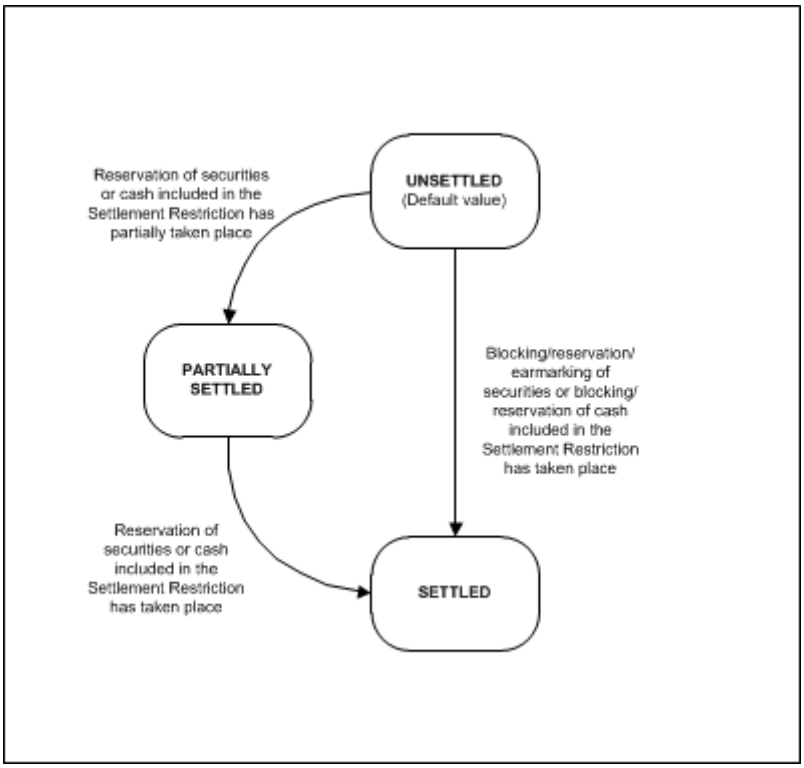
Every *Settlement Instruction* in T2S has a value for all the following statuses.

- Settlement Status;
- Match Status;
- Cancellation Status;
- CSD Hold Status;
- Party Hold Status;
- CSD Validation Hold Status;
- CoSD Hold Status;
- Approval Status.

The definition of the different status and the transitions among the status values are depicted below.

Settlement Status (Settlement Instruction):

Indicates the Settlement Status of the *Settlement Instruction*. Each status value reflects in which step of the settlement process an instruction can be.



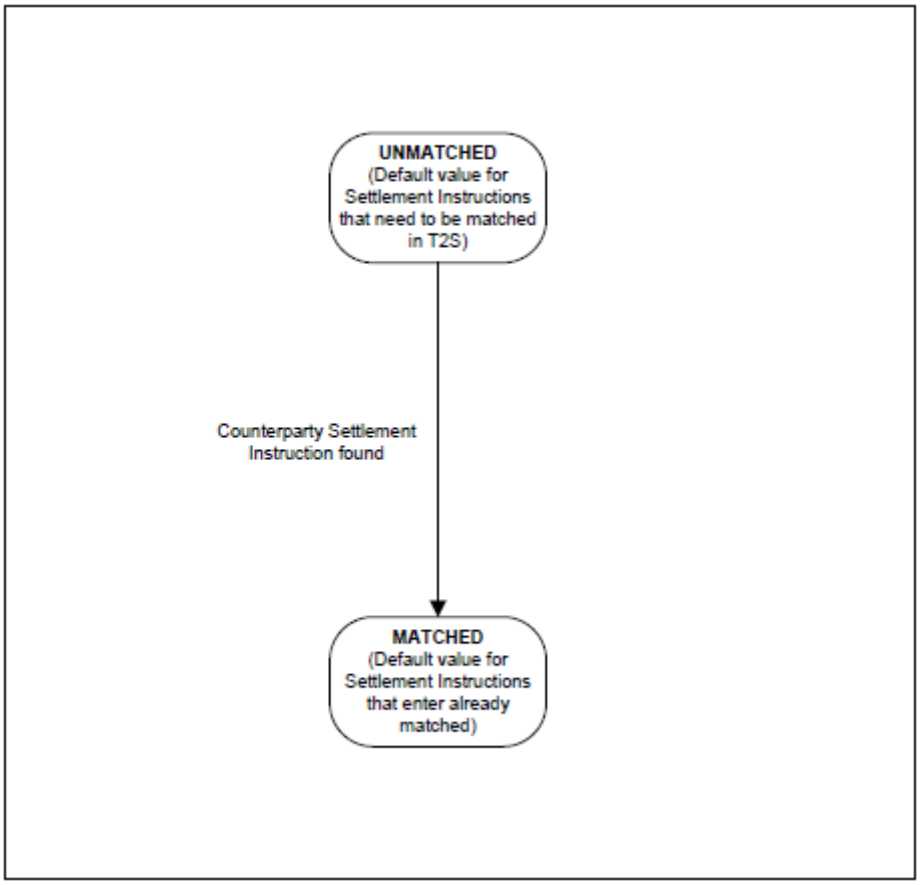
STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Unsettled	Default value. The Settlement Instruction is not settled.	IVA	IMA / IVA / SQRA / STM

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Partially Settled	T2S settled a partial quantity of securities and/or cash. T2S continues to attempt settlement of the Settlement Instruction until it fulfils full quantity and/or amount.	VPB / NTS	IMA / IVA / SQRA / STM
Settled	T2S transferred the full quantity of securities and/or cash. T2S performs no further processing on the Settlement Instruction.	VPB / NTS	IMA / IVA / SQRA / STM

Match Status (Settlement Instruction):

Indicates the Match Status of the *Settlement Instruction*. There are only two possible Match Status values that indicate whether an instruction needs to be matched in T2S or not.



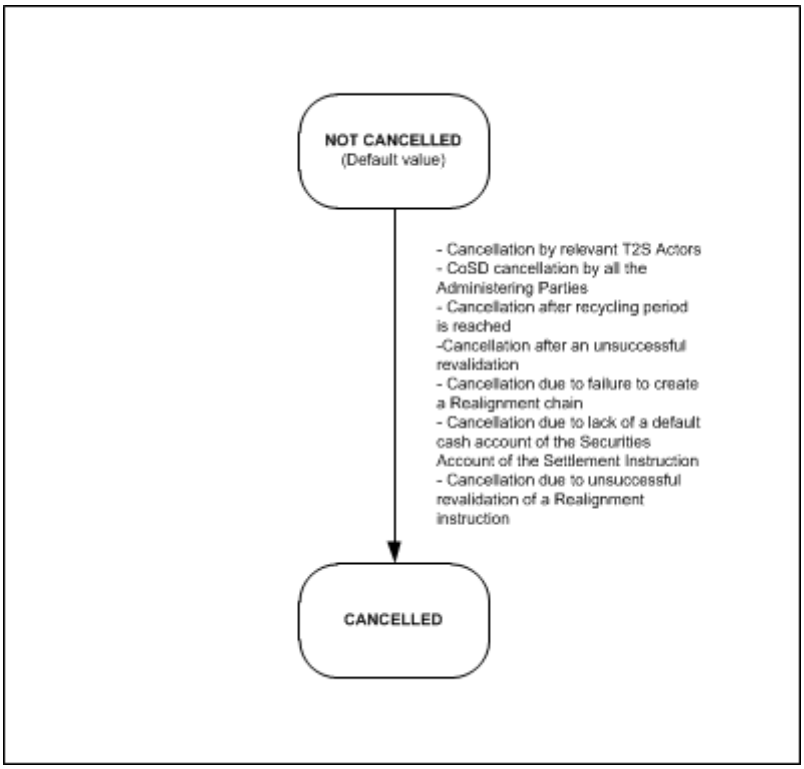
STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Unmatched	Default value when T2S receives a Settlement Instruction that needs to match.	IVA	IMH / SQRA IMA/ IVA / STM

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	The Settlement Instruction has no corresponding counterpart instruction identified in T2S.		
Matched	Default value when a Settlement Instruction enters T2S as already matched. The Settlement Instruction has a corresponding counterpart instruction identified in T2S. .	IVA / IMH	IMA / SQRA/ IMH / STM

Cancellation Status (Settlement Instruction):

Indicates the Cancellation Status of the *Settlement Instruction.* Depending on the status, the instruction can be further processed in T2S or not. There is no possible revocability of this status.



STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Not Cancelled	Default value. Settlement Instruction is available for further processing.	IVA	IMA / IVA / SQRA / STM

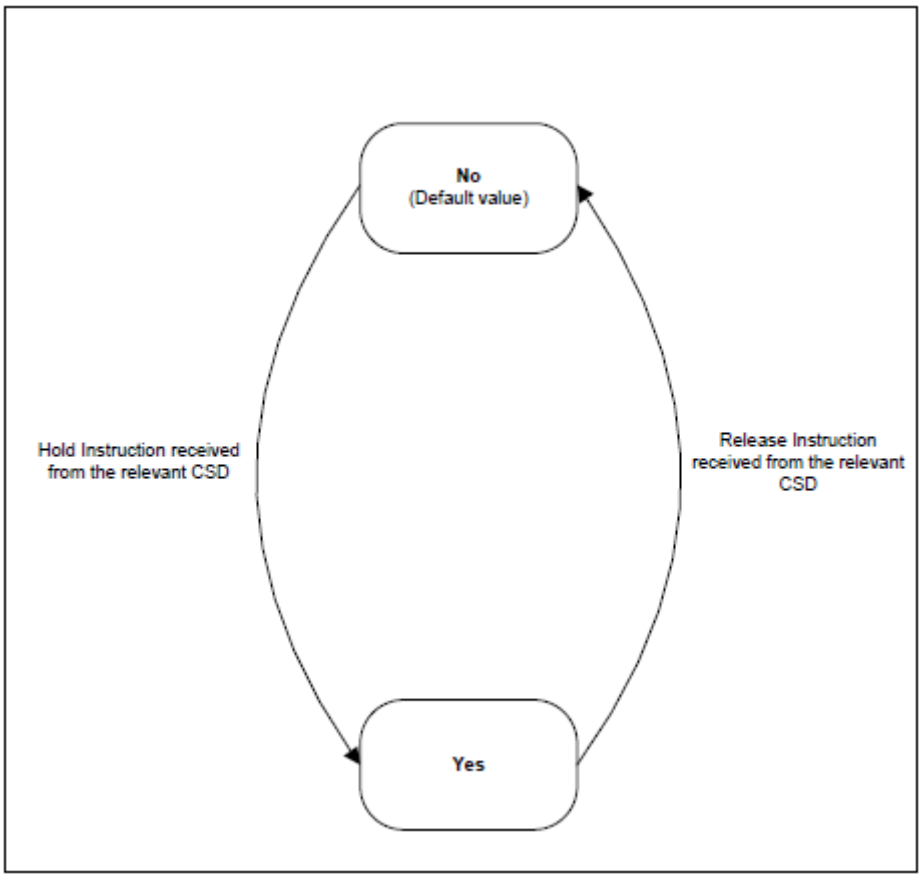
All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Cancelled	Settlement Instruction is cancelled and cannot be further processed: ▌ The Settlement Instruction was cancelled upon request from the relevant T2S Actors. ▌ The CoSD Settlement Instruction was cancelled upon request by all the involved Administering Parties. ▌ The Settlement Instruction was cancelled due to an excess of the recycling period. ▌ The Settlement Instruction was cancelled after an unsuccessful revalidation. ▌ Cancellation due to failure to create a Realignment Chain. ▌ Cancellation due to lack of a default cash account of the Securities Account of the Settlement Instruction. Cancellation due to unsuccessful revalidation of a Realignment Instruction.	IMA / IVA	IMA / IVA / SQRA / STM

CSD Hold Status (Settlement Instruction):

All rights reserved.

Indicates if the *Settlement Instructio*n is set On Hold or Released by the corresponding CSD. Settlement Instructions with the CSD Hold Status "Yes" are not submitted for settlement.

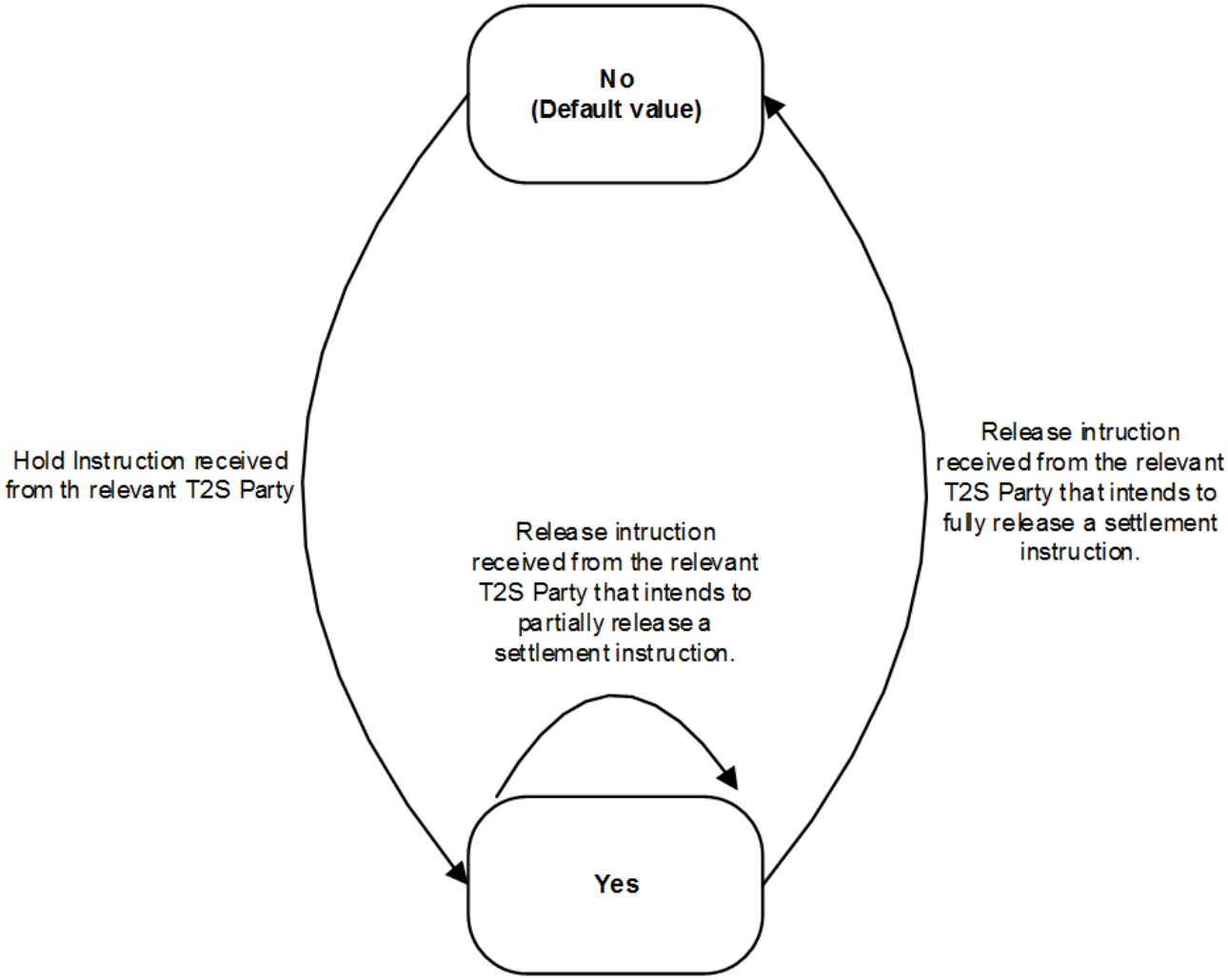


STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
No	Default value. The relevant CSD has not instructed the Settlement Instruction On Hold. The value is also set when a Settlement Instruction, which had been previously put On Hold by the relevant CSD, is released by this CSD.	IVA / IMA	IMA / IVA / SQRA/ SPS / STM
Yes	The relevant CSD has instructed the Settlement Instruction On Hold. When a Hold request, originated from the relevant CSD, on the Settlement Instruction is executed.	IVA / IMA	IMA / IVA / SQRA/ SPS / STM

All rights reserved.

Party Hold Status (Settlement Instruction):

Indicates if the *Settlement Instruction* is set On Hold **(either by the corresponding Party or because it fulfilled predefined conditions of the Restriction Processing Type "Party Hold")** or Released by the corresponding *Party*.**{CR-812}** Settlement Instructions with the Party Hold Status "Yes" are not submitted for settlement with the exception of those Settlement Instructions Partially Released which are submitted for partial settlement.



STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
No	Default value. The relevant T2S Party has not instructed the Settlement Instruction On Hold**, and there is not a Restriction Processing Type "Party Hold" applicable for it.**	IVA / IMA / VPB / NTS	IMA / IVA / SQRA / SPS / STM

All rights reserved.

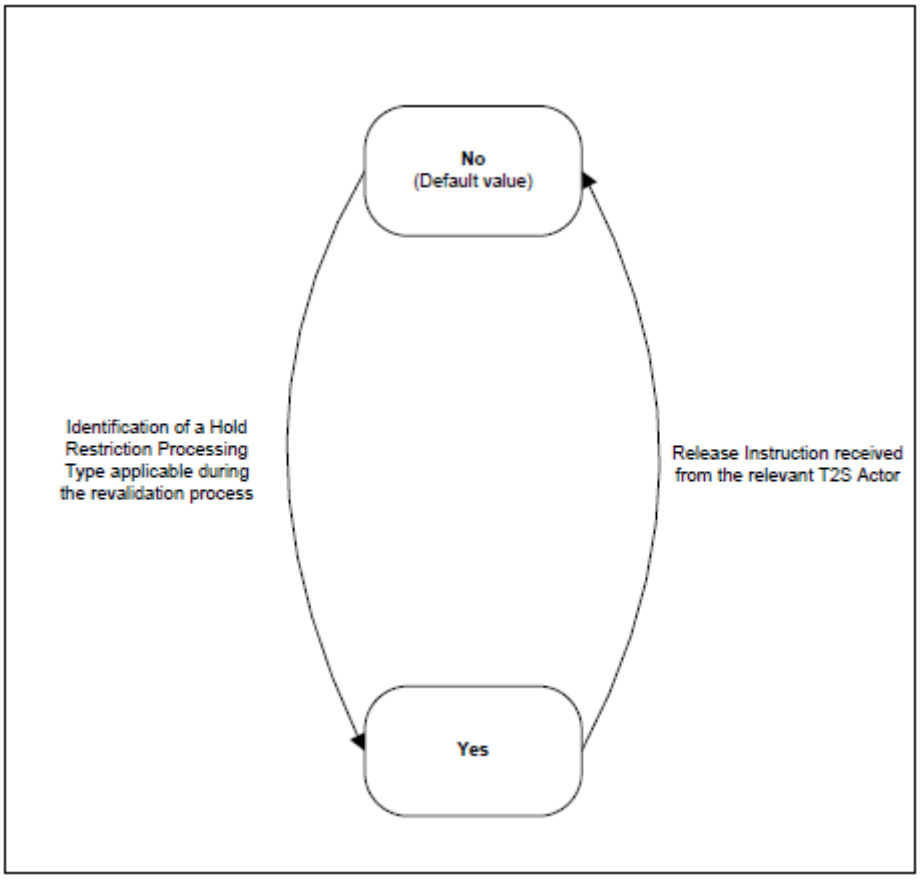
STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	The relevant T2S Party releases the corresponding Settlement Instruction which was previously On Hold.		
Yes	The relevant T2S Party has instructed the Settlement Instruction On Hold. When a Hold request on the Settlement Instruction, originated by the corresponding T2S Party, is executed. The relevant T2S Actor instructs the Settlement Instruction and the "hold Release default" value of the Securities Account included in the Instruction is set to "Yes". When a Settlement Instruction is under a partial release process. The remaining quantity is still On Hold. When the Settlement Instruction comes without Hold Indicator being informed (i.e. neither Hold Status "Yes" or "No" have been specified) and the "Hold Release default" value of the Securities Account included in the Instruction is set to "Yes", the Party Hold Status is automatically set to "Yes". **Settlement Instruction has been set On Hold by T2S when the Settlement Instruction fulfils predefined conditions of the**	IVA / IMA / VPB / NTS	IMA / IVA / SQRA / SPS / STM

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	Restriction Processing Type "Party Hold". **{CR-812}**		

CSD Validation Hold Status (Settlement Instruction):

Indicates if the *Settlement Instruction* fulfils predefined conditions of the Restriction Processing Type *"CSD Validation Hold"* and if the relevant T2S Actor (i.e. CSD) has released it. Settlement Instructions with the CSD Validation Hold Status "Yes" are not submitted for settlement.



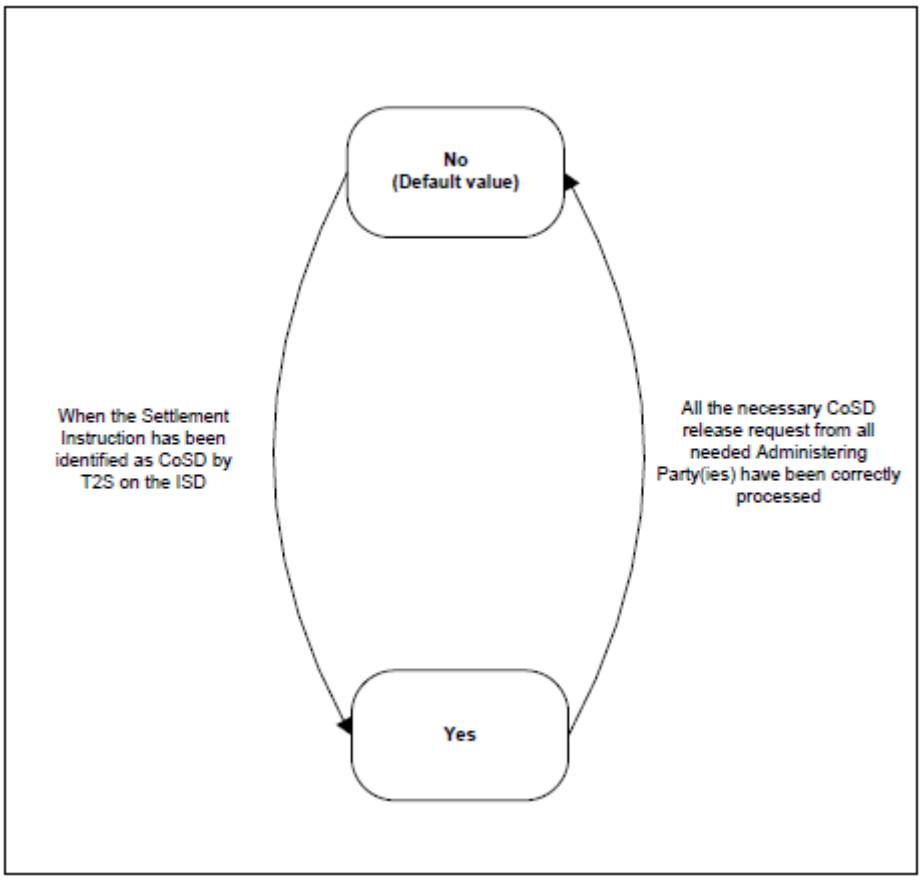
STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
No	Default value. T2S Actor (i.e. CSD) has released the Settlement Instruction, which had been previously on CSD Validation Hold.	IVA / IMA	IMA / IVA / SQRA / SPS / STM

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	The value is also set when there is not a Restriction Processing Type "CSD Validation Hold" applicable for the Settlement Instruction.		
Yes	Settlement Instruction has been set On Hold by T2S when the Settlement Instruction fulfils predefined conditions of the Restriction Processing Type "CSD Validation Hold".	IVA	IMA / IVA / SQRA / SPS / STM

CoSD Hold Status (Settlement Instruction):

Indicates if the *Settlement Instruction* has been identified as a *CoSD instruction* by T2S and if it has been released by the Administering Parties. *Settlement Instructions* with the CoSD Hold Status "*Yes*" are not submitted for settlement.



STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
No	Default value.	IVA / IMA	IMA / STM

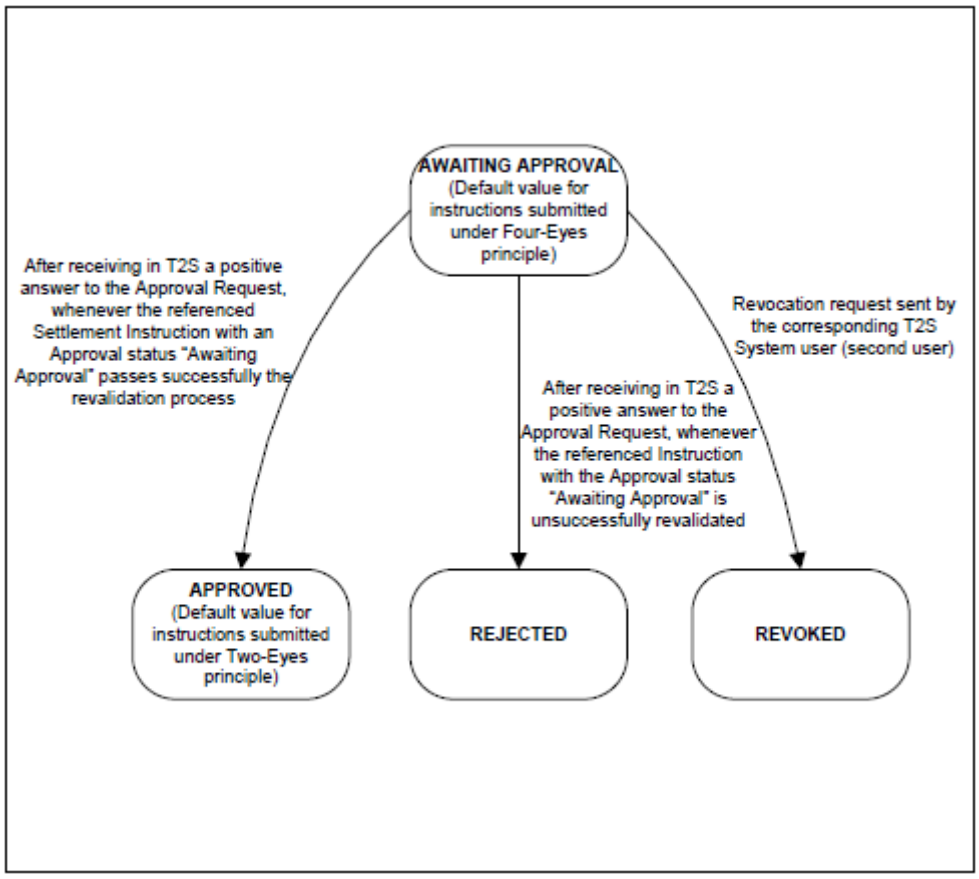
All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	Settlement Instruction has not been identified as a CoSD instruction by T2S on the ISD. T2S sets the value when all relevant Administering Parties release the Settlement Instruction by sending a release for all applicable CoSD Rules.		
Yes	Settlement Instruction has been identified as a CoSD instruction by T2S on the ISD. It remains unchanged while there is no release request from all relevant Administering Parties for all applicable CoSD Rules.	SPS	IMA / STM

Approval Status (Settlement Instruction):

All rights reserved.

Indicates the Approval Status of *Settlement Instructions* submitted under Four-Eyes principle. This status is only communicated in U2A mode [97], and it is not communicated in any A2A queries.



STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Awaiting Approval	Default value for those Settlement Instructions that are submitted under Four-Eyes principle.	IVA	IVA / STM
Approved	As a result of the revalidation process (triggered by the reception of a positive answer to the Approval Request), whenever a Settlement Instruction with an Approval Status "Awaiting Approval" passes successfully the revalidation process. De-	IVA	IVA / STM

97 T2S does not report the Approval Status via a Status Advice message

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	fault value for those Settlement Instructions submitted under Two-Eyes principle.		
Revoked	Settlement Instruction submitted to T2S under Four-Eyes principle that is revoked by the second T2S System user.	IVA	IVA / STM
Rejected	As a result of the revalidation process (triggered by the reception of a positive answer to the Approval Request), whenever a Settlement Instruction with an Approval Status "Awaiting Approval" does not pass successfully the revalidation process and is cancelled.	IVA	IVA / STM

Settlement Restriction status attributes

Every *Settlement Restriction* in T2S has a value for all the following statuses.

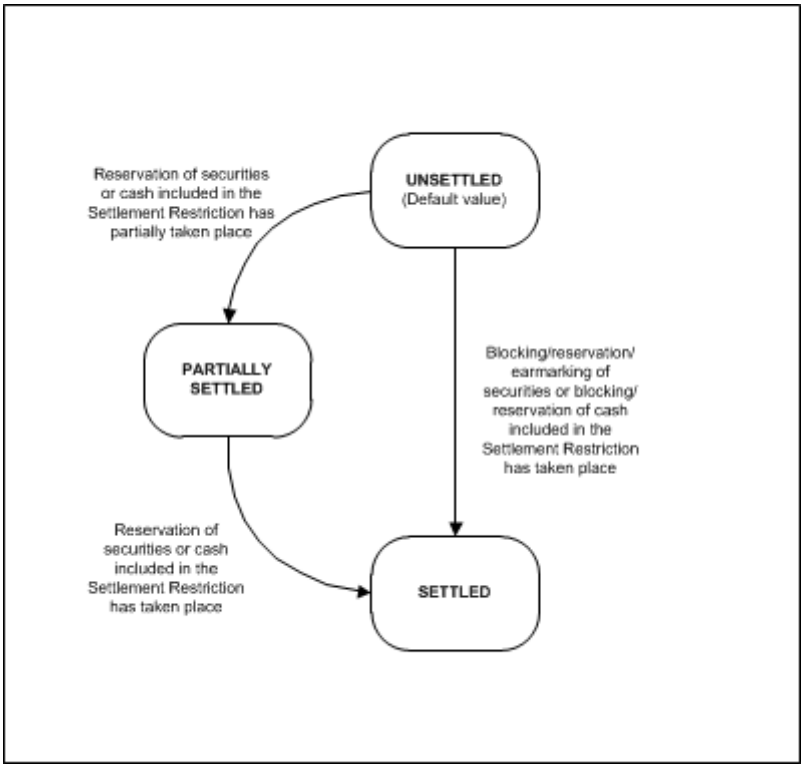
- Settlement Status;
- Cancellation Status;
- Approval Status.

The definition of the different status and the transitions among the status values are depicted below.

Settlement Status (Settlement Restriction):

Indicates the Settlement Status of a *Settlement Restriction*. Each status value reflects in which step of the settlement process a restriction can be.

All rights reserved.



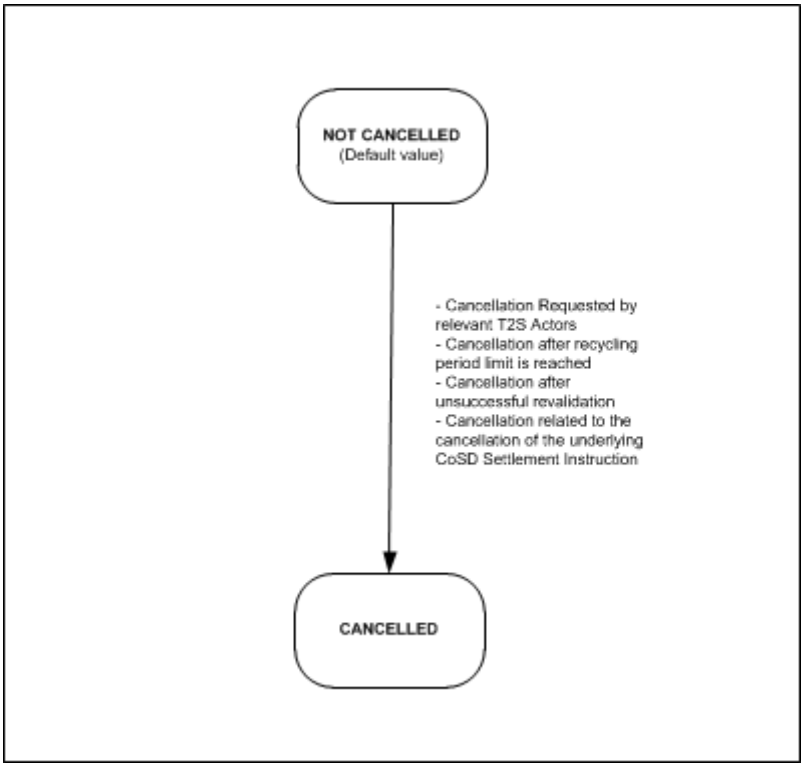
STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Unsettled	Default value. Blocking/reservation/earmarking of securities or Blocking/reservation of cash included in the Settlement Restriction has not yet taken place.	IVA	IMA / IVA / STM
Partially Settled	Reservation of securities or cash included in the Settlement Restriction has partially taken place. Additional settlement attempts are needed to fulfil original quantity or amount.	VPB / NTS	IMA / IVA / STM
Settled	Earmarking of securities included in the Settlement Restriction has taken place (partially or fully). No additional settlement attempts	VPB / NTS	IMA / IVA / STM

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	occur even if the Settlement Restriction has not been fully settled. Reservation/Blocking of securities or cash included in the Settlement Restriction has been fully settled. The Settlement Restriction has been successfully settled and does not require further processing.		

Cancellation Status (Settlement Restriction):

Indicates the Cancellation Status of the *Settlement Restriction*. Depending on the status, the restriction can be further processed in T2S or not. There is no possible revocability of this status.



STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Not Cancelled	Default value. Settlement Restriction is available for further processing.	IVA	IMA / IVA / STM

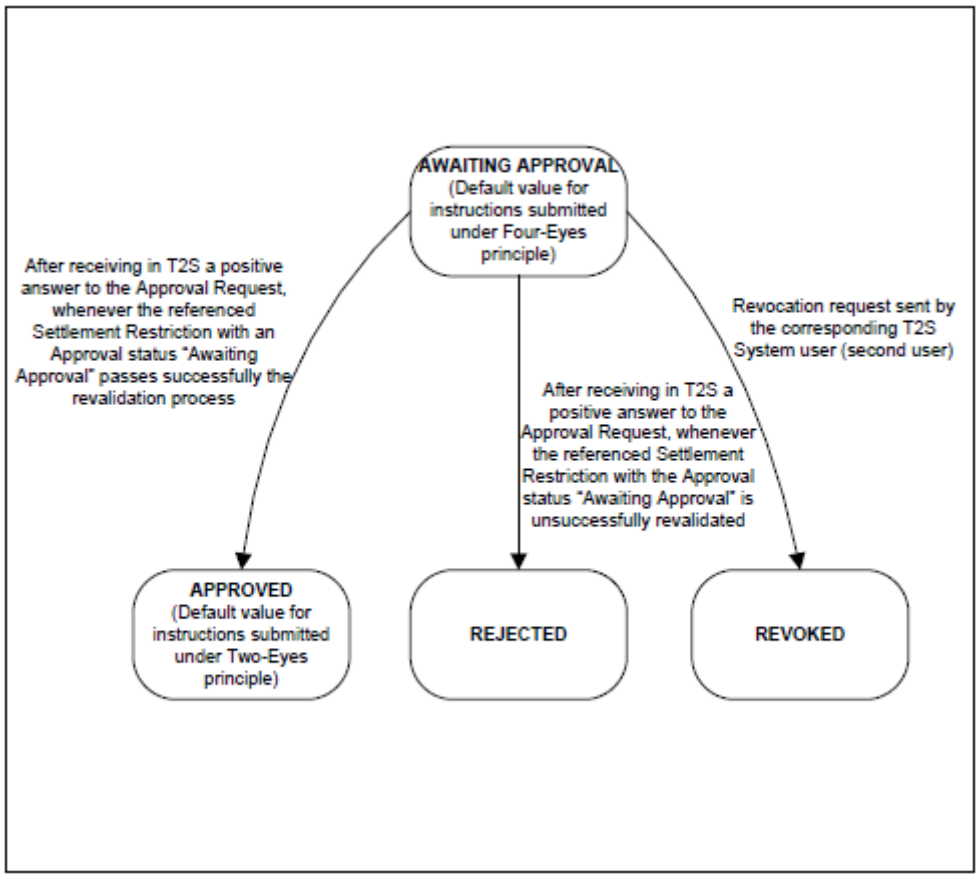
All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Cancelled	Settlement Restriction is cancelled and cannot be further processed. The Settlement Restriction was cancelled upon request from the relevant T2S Actor prior to its settlement. The Settlement Restriction was cancelled after the recycling period limit is reached. Settlement Restriction was cancelled after an unsuccessful revalidation. Cancellation related to the cancellation of the underlying CoSD Settlement Instruction.	IMA / IVA	IMA / IVA/ STM

Approval Status (Settlement Restriction):

All rights reserved.

Indicates the Approval Status of *Settlement Restriction* submitted under Four-Eyes principle. This status is only communicated in U2A mode [98] and it is not communicated in any A2A queries.



STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Awaiting Approval	Default value for those Settlement Restrictions that are submitted under Four-Eyes principle.	IVA	IVA / STM
Approved	As a result of the revalidation process (triggered by the reception of a positive answer to the Approval Request), whenever a Settlement Restriction with an Approval Status "Awaiting Approval" passes successfully the revalidation process. De-	IVA	IVA / STM

98 T2S does not report the Approval Status via a Status Advice message

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	fault value for those Settlement Restrictions submitted under Two-Eyes principle.		
Revoked	Settlement Restriction submitted to T2S under Four-Eyes principle that is revoked by the second T2S System User.	IVA	IVA / STM
Rejected	As a result of the revalidation process (triggered by the reception of a positive answer to the Approval Request), whenever a Settlement Restriction with an Approval Status "Awaiting Approval" does not pass successfully the revalidation process and is cancelled.	IVA	IVA / STM

Maintenance instruction (Amendment, Hold/Release and Cancellation Instructions) status attributes

Every maintenance instruction in T2S has a value for all the following statuses, except *Amendment Instructions* and *Hold / Release Instructions* that do not have Match Status.

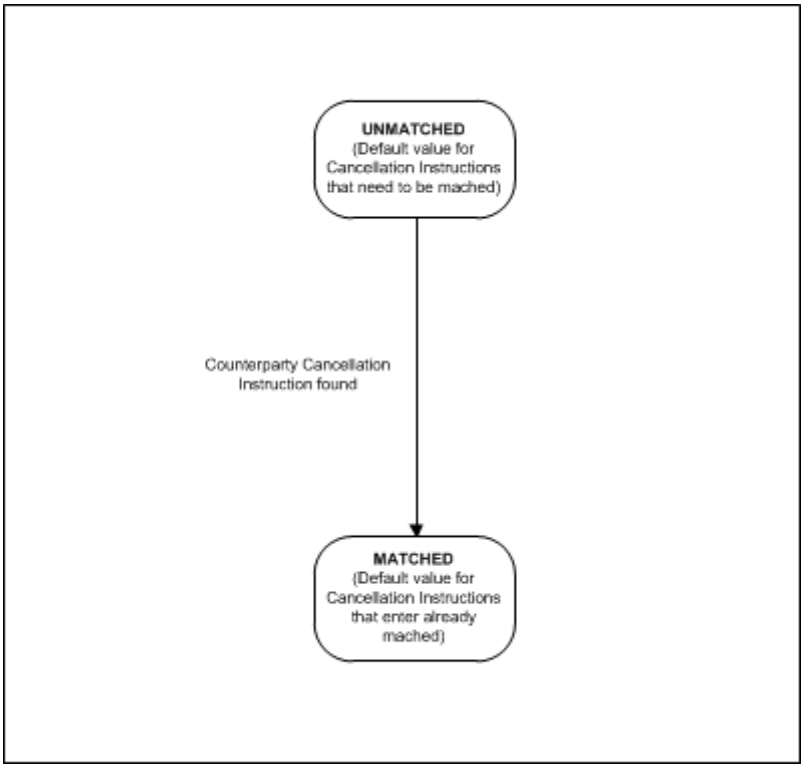
- Matching Status (For Cancellation Instructions);
- Cancellation Status;
- Execution Status;
- Approval Status.

The definition of the different status and the transitions among the status values are depicted below.

Match Status (Cancellation Instruction)

Indicates the Match Status of the *Cancellation Instruction*. There are only two possible Match Status values that indicate whether an instruction needs to be matched in T2S or not.

All rights reserved.

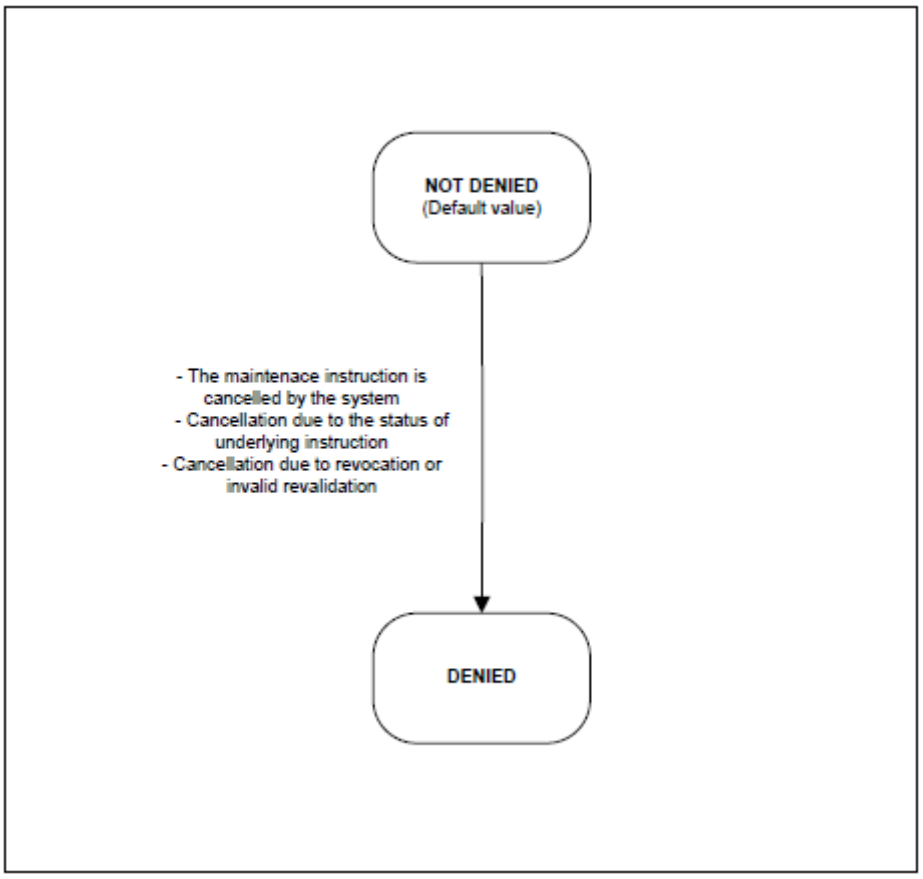


STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Unmatched	Default value when T2S receives a Cancellation Instruction that needs to match. The Cancellation Instruction has no corresponding counterpart instruction identified in T2S.	IVA	IMA / STM
Matched	Default value when a Cancellation Instruction enters T2S as already matched. The Cancellation Instruction has a corresponding counterpart instruction identified in T2S.	IVA / IMA	IMA / STM

Cancellation Status (Maintenance instruction):

All rights reserved.

Indicates the Cancellation Status of the maintenance instruction. Depending on the status, the instruction can be further processed in T2S or not. There is no possible revocability of this status.



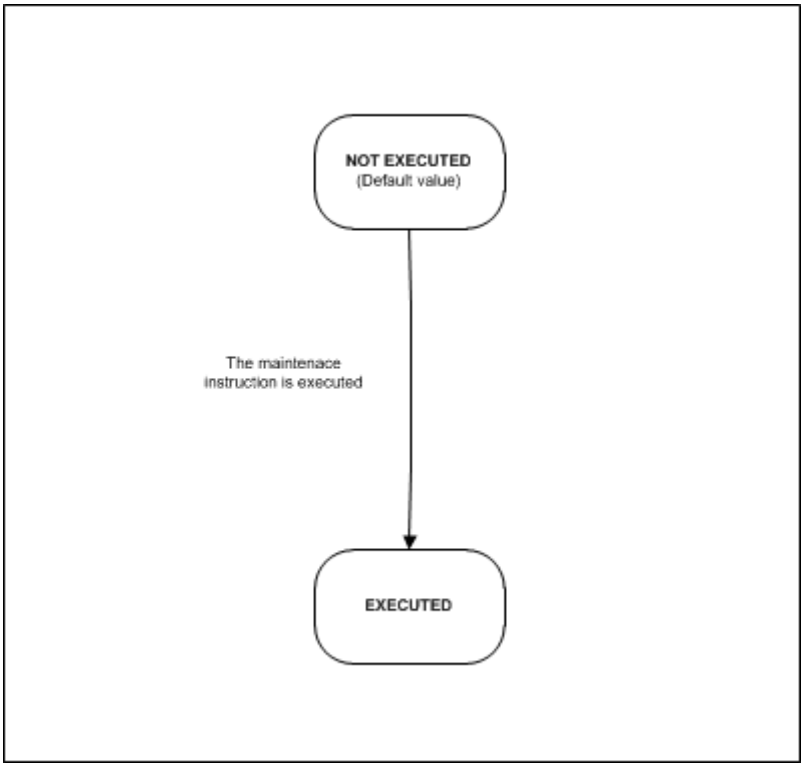
STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Not Denied	Default value. Maintenance instruction is available for further processing.	IVA	IMA / STM
Denied	Maintenance instruction is cancelled and cannot be further processed. Instruction Maintenance module has denied the execution of a maintenance instruction due to the status of the referenced instructions.	IMA / STM	IMA / STM

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	Maintenance instruction was cancelled by the system or was cancelled after revocation or unsuccessful revalidation.		

Execution Status (Maintenance instruction):

Indicates whether a maintenance instruction has been successfully processed in T2S.

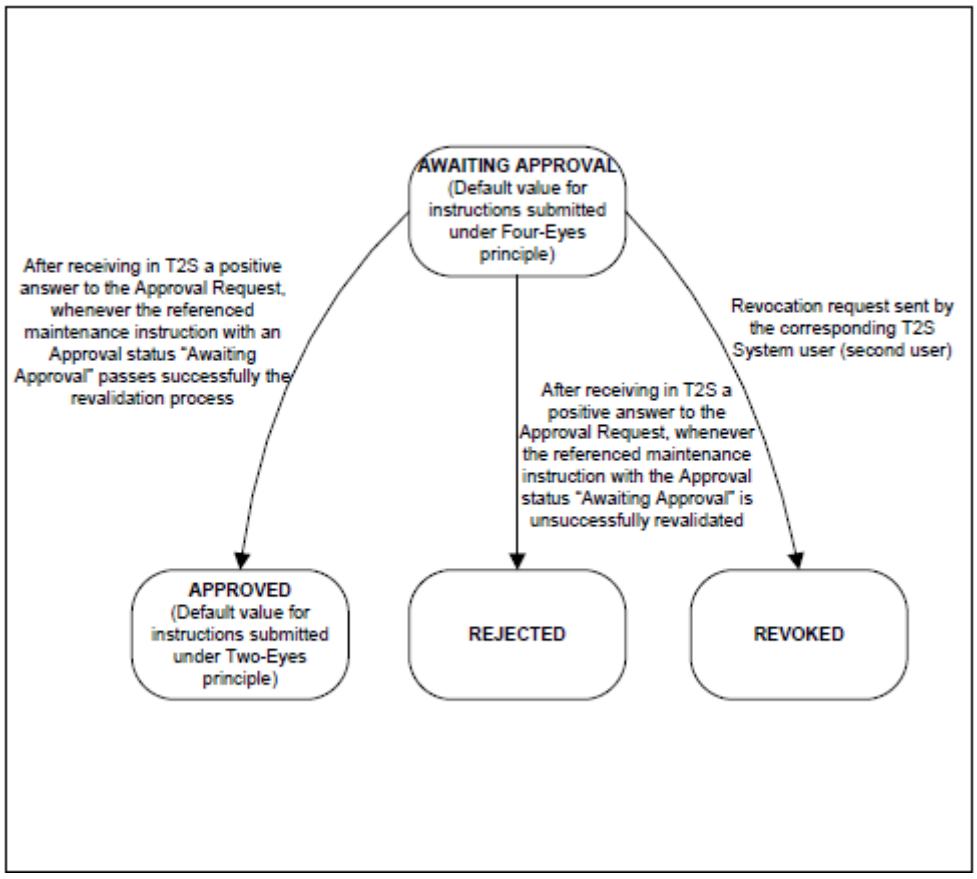


STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Not Executed	Default value. Maintenance instruction can be further processed.	IVA	IMA / STM
Executed	The maintenance instruction has been successfully executed.	IMA	IMA / STM

Approval Status (Maintenance instruction):

All rights reserved.

Indicates the Approval Status of maintenance instructions submitted under Four-Eyes principle. This status is only communicated in U2A mode [99], and it is not communicated in any A2A queries.



STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Awaiting Approval	Default value for those maintenance instructions that are submitted under Four-Eyes principle.	IVA	IVA / STM
Approved	As a result of the revalidation process (triggered by the reception of a positive answer to the Approval Request), whenever a maintenance Instruction with an Approval Status "Awaiting Approval" passes successfully the revalidation process..	IVA	IVA / STM

99 T2S does not report the Approval Status via a Status Advice message

All rights reserved.

STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
	Default value for those maintenance instructions submitted under Two-Eyes principle.		
Revoked	Maintenance instruction submitted to T2S under Four-Eyes principle that is revoked by the second T2S System User.	IVA	IVA / STM
Rejected	As a result of the revalidation process (triggered by the reception of a positive answer to the Approval Request), whenever a whenever a maintenance instruction with an Approval Status "Awaiting Approval" does not pass successfully the revalidation process and is cancelled.	IVA	IVA / STM

Pending and Failing Instructions

Settlement Instructions and Settlement Restrictions can be classified as "Pending" or as "Failing" instructions depending whether their settlement is still possible in T2S on their Intended Settlement Date or not:

- Pending Instructions: Settlement Restrictions, unmatched and matched Settlement Instructions for which settlement is still possible on their Intended Settlement Date.
- Failing Instructions: Settlement Restrictions, unmatched and matched Settlement Instructions for which T2S has identified that they cannot settle on their Intended Settlement Date.

This classification is linked to the T2S settlement status, but it does not represent a T2S status as such (i.e. the "Failing" and "Pending" are ISO statuses for the reporting in relevant messages but both relate to the Settlement Status in T2S). Therefore, this information will be a complement to the Settlement status values "Unsettled" and "Partially settled".

As described below, instructions can change form "Pending" to "Failing", as well as be created directly as "Failing" at its acceptance. However, it is important to note that once T2S identifies a Settlement Instruction or a Settlement Restriction as "Failing", the instruction will no longer be classified as "Pending" again.

T2S identifies as "Failing" a Settlement Instruction or Settlement Restriction as "Failing", which is already in the system, in the following scenarios:

- In the completion of the processing of each cut-off, for Settlement Restrictions or matched Settlement Instructions applicable to such cut-off:

All rights reserved.

- All Settlement Restrictions or matched Settlement Instructions, which are eligible to the cut-off and therefore have had at least one settlement attempt, remaining unsettled after the completion of the cut-off, will be identified as "Failing".

- After the completion of the processing of the last cut-off (FOP cut-off), for all Unmatched Settlement Instructions:
  - Upon completion of the last cut-off, all unmatched settlement instructions having Intended Settlement Date equal to the current business day will be identified as "Failing".
- After unsuccessful eligibility check of a Settlement Instruction or Settlement Restriction because its applicable cut-off has been reached.
  - Any matched Settlement Instruction and Settlement Restriction that fails its eligibility check because its applicable cut-off has been reached, will be identified as "Failing".

Additionally, T2S identifies as "Failing" any new incoming Settlement Instruction or Settlement Restriction, directly upon their successful validation, in the following scenarios:

- During the whole Settlement Day except for the End of day phase:
  - T2S will identify as "Failing" all incoming Settlement Instructions and Settlement Restrictions having their Intended Settlement Date before the Current Business Day.
- During the End of Day phase:
  - T2S will identify as "Failing" all incoming Settlement Instructions and Settlement Restrictions having their Intended Settlement Date before or equal to the Current Business Day.

Once a Settlement Instruction has been identified as "Failing" upon completion of its cut-off, the Penalty Mechanism Module will retrieve the Settlement Instruction status and reasons causing the settlement failure in order to identify if the instruction is eligible for a Penalty.

Reason Codes

At each moment in time, each of the status attributes of an *LCMM Instruction* has one single value. Negative status values such as *"Not Executed"*, *"Denied"*, *"Cancelled"* or *"Unsettled"* might be linked to a specific reason code, which provides T2S Actors information about the underlying cause (e.g. unsettled due lack of securities). The following list reflects a non-exhaustive list of possible reason codes for each status:

Settlement Instruction reason codes:

- Unsettled Status reason codes:
  - *Settlement Instruction « On Hold »*
  - *Linked Settlement Instruction / Settlement restriction missing*
  - *Lack of cash*
  - *Lack of securities*
  - *Intraday restriction (security, currency, security account, T2S dedicated cash account or T2S party blocked)*
  - *Net buying limit reached*

All rights reserved.

- *Cut-off reached*
- *Counterparty´s instruction on CoSD process*

- Cancelled Status reason codes:
  - *Recycling period reached*
  - *CoSD Cancellation*
  - *Cancelled due to Reference Data change*
  - *Instruction under Four-Eyes principle cancelled due to revocation*
  - *Standard period for unmatched instructions reached*
  - *Cancelled by the T2S Actor*

Settlement Restriction reason codes:

- Unsettled Status reason codes:
  - *Linked Settlement Instruction / Settlement Restriction missing*
  - *Lack of cash*
  - *Lack of securities*
  - *Net buying limit reached*
  - *Cut-off reached*
- Cancelled Status reason codes:
  - *Recycling period reached*
  - *Cancelled by the T2S Actor*
  - *Cancelled due to Reference Data change*
  - *Instruction under Four-Eyes principle cancelled due to revocation*

Cancellation instruction reason codes:

- Not Executed Status reason codes:
  - *Cancellation pending, awaiting cancellation from the counterparty*
  - *Cancellation pending, awaiting CoSD cancellation from Administering Party*
- Denied Status reason codes:
  - *Referenced Settlement Instruction / Settlement Restriction already settled*
  - *Referenced Settlement Instruction / Settlement Restriction already cancelled*
  - *Referenced Settlement Instruction already requested for cancellation*
  - Referenced Instruction CoSD On Hold
  - *Referenced Settlement Instruction / Settlement Restriction has not been approved yet*
  - *Instruction under Four-Eyes principle cancelled due to revocation*
  - *Standard period for unmatched instructions reached*

All rights reserved.

Amendment Instruction reason codes:

- Denied Status reason codes:
  - *Referenced Settlement Instruction / Settlement Restriction already settled*
  - *Referenced Settlement Instruction / Settlement Restriction already cancelled*
  - *Referenced Settlement Instruction identified as CoSD*
  - *Referenced Settlement Instruction / Settlement Restriction has not been approved yet*
  - *Referenced Settlement Instruction / Settlement Restriction already partially settled*
  - *Instruction under Four-Eyes principle cancelled due to revocation*

Hold Instruction reason codes:

- Denied Status reason codes:
  - *Referenced Settlement Instruction / Settlement Restriction already settled*
  - *Referenced Settlement Instruction / Settlement Restriction already cancelled*
  - *Referenced Settlement Instruction identified as CoSD*
  - *Referenced Settlement Instruction already On Hold*
  - *Referenced Settlement Instruction has not been approved yet*
  - *Instruction under Four-Eyes principle cancelled due to revocation*

Release Instruction reason codes:

- Denied Status reason codes:
  - *Referenced Settlement Instruction / Settlement Restriction already settled*
  - *Referenced Settlement Instruction / Settlement Restriction already cancelled*
  - Referenced Instruction CoSD On Hold
  - *CoSD Blocking of the referenced Settlement Instruction not settled yet*
  - *Referenced Settlement Instruction CoSD rule already released*
  - *Referenced Settlement Instruction / Settlement Restriction already Released*
  - *Referenced Settlement Instruction / Settlement Restriction has not been approved yet*
  - *Instruction under Four-Eyes principle cancelled due to revocation.*

    Additionally, if the release instruction intends to Partially Release a settlement instruction, the following Denied Status reason codes are applicable:
  - The referenced Settlement Instruction is already partially released or there is a pending Release Instruction of the same Hold type in the system.
  - The referenced Settlement Instruction is unmatched
  - Other holds apply either on the referenced Settlement Instruction or on the counterpart.
  - At least one associated realignment Settlement Instruction is on hold.
  - Either the referenced Settlement Instruction or its counterpart do not allow partial settlement.

All rights reserved.

- A settlement link applies either on the referenced Settlement Instruction or on its counterpart.
- A non-reciprocal link applies either on the referenced Settlement Instruction or on its counterpart.
- Partial release instruction is denied because the quantity to be released does not comply with the Minimum Settlement Unit or with the Settlement Unit Multiple.
- Partial release instruction is denied because the relevant cut-off has been reached.

The date, the time of each status update, the reason code and a unique identifier of the system user originating the change are recorded in the *Status History*, *Reason History, Maintenance Status History* and Maintenance Reason History entities **{T2S.05.270}**.

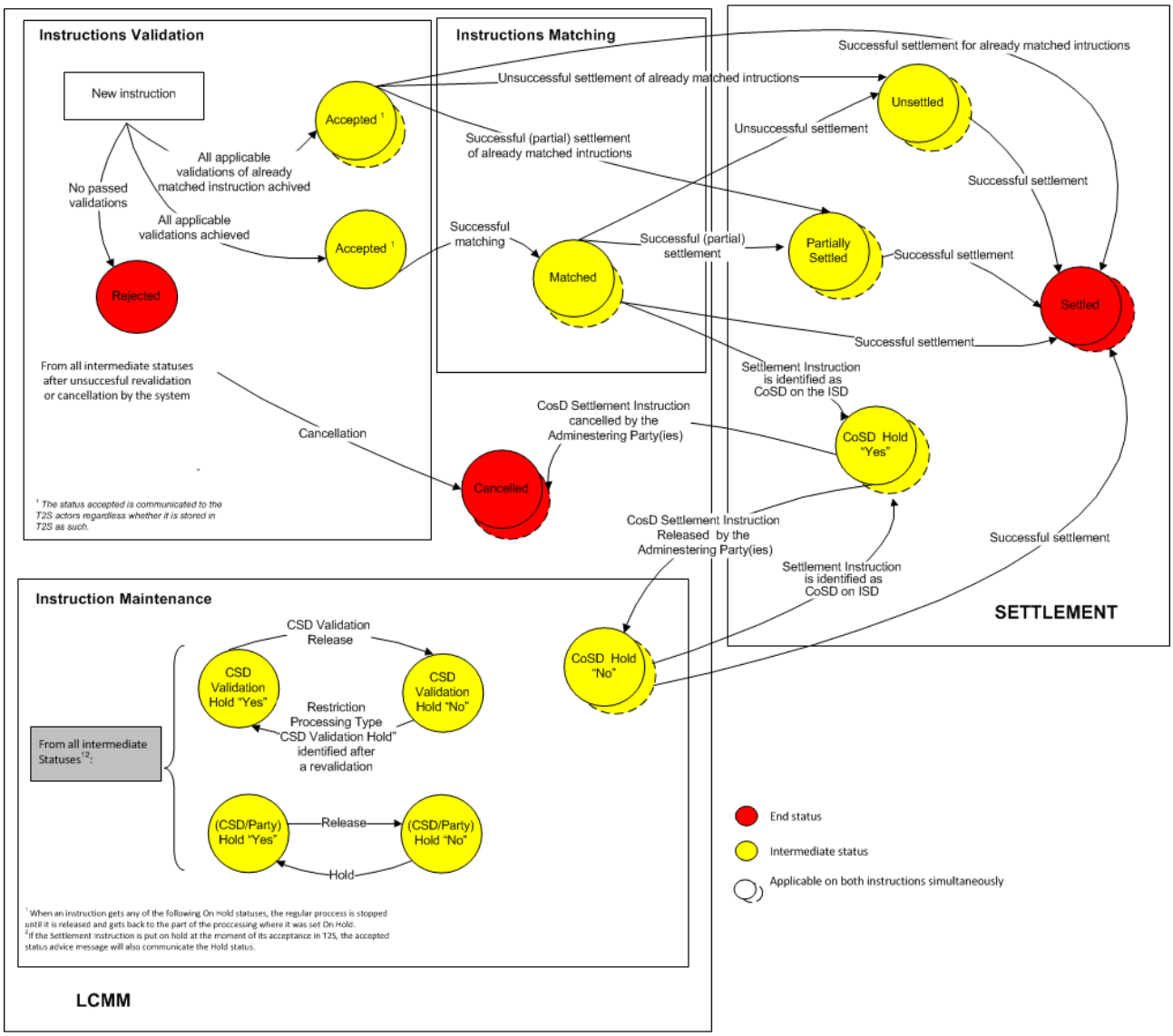
All the status values and reason codes are available, at any point in time, upon request in pull mode. In addition, based on the information provided by the *Status Management* module, the *Interface* domain communicates in push mode to the T2S Actors (according to their message subscription set up) all status attributes updates with their reason codes, whenever there is a change of a status or reason code value (except for the Approval Status values which are available in U2A only).

Status Transition Diagrams

The following diagrams reflect all the possible transitions from one status to another (sequences of status attributes changes) that are communicated to the *T2S Party* for each type of *LCMM Instruction* depending on the message subscription service. The diagram also reflects in which module the status is updated.

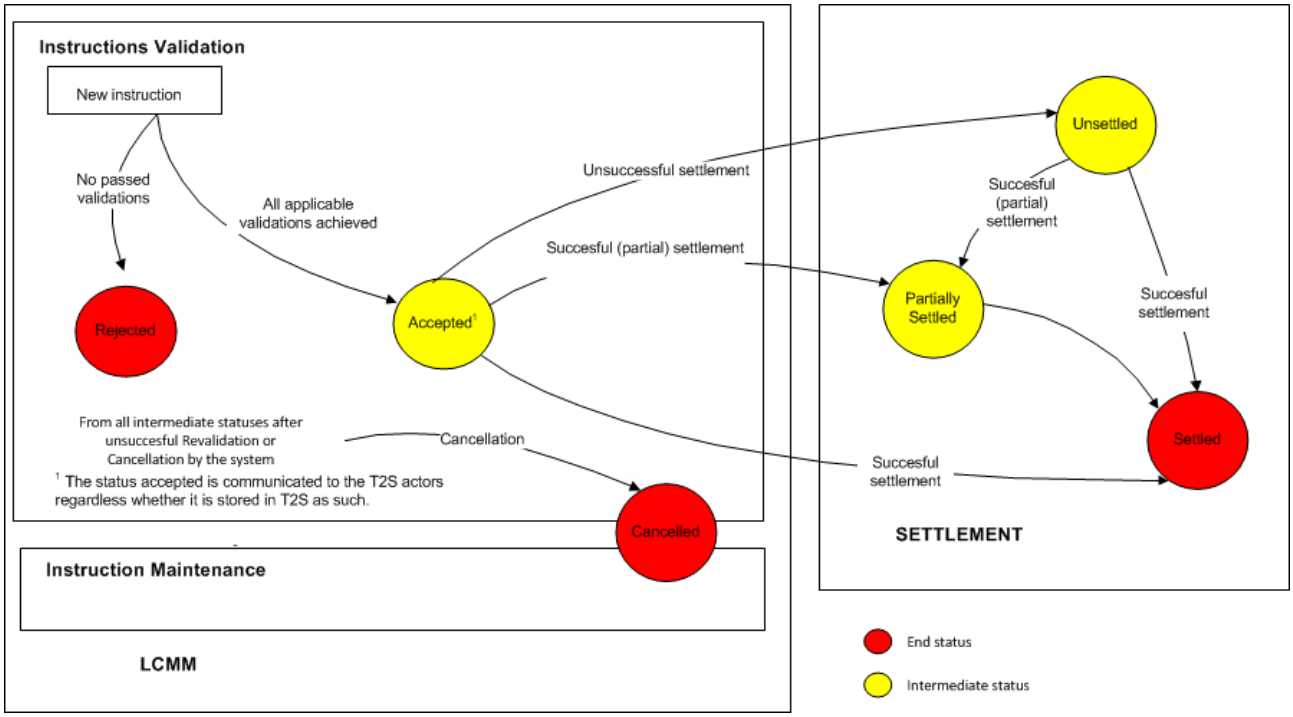
All rights reserved.

Settlement Instruction Status transition diagram



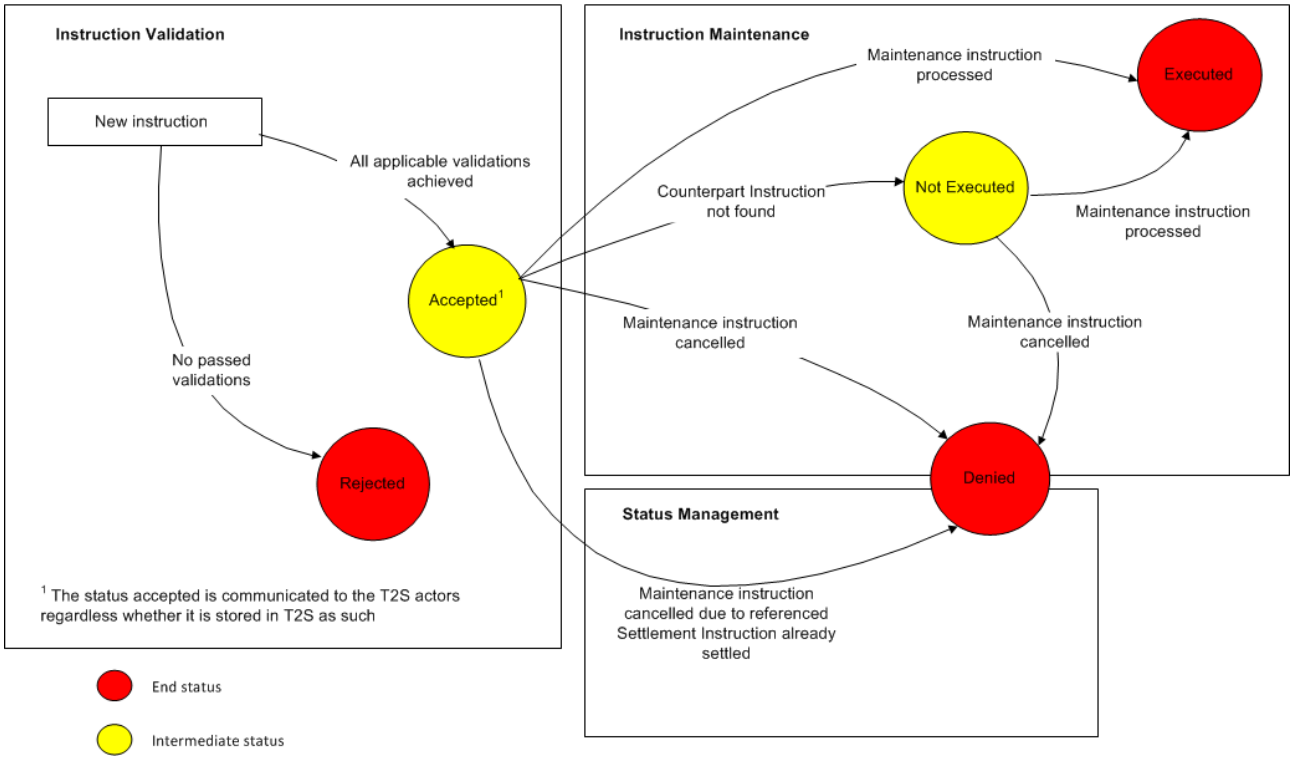
All rights reserved.

Settlement Restrictions status transition diagram



The status attributes that are common to *Settlement Instruction* and *Settlement Restriction* follow the same status transitions. The Match Status attribute, CSD Hold Status attribute, Party Hold Status attribute and CoSD Hold Status attribute do not apply to *Settlement Restrictions*.

Maintenance instructions status transition diagram



All rights reserved.

3.4.6.3 Description of the functions of the module

The *Status Management* module is composed of the following six functions dealing with status information:

- Life Cycle Track;
- Night-time Reporting Manager;
- Status Driven Manager;
- Data Collection for Messages;
- Allegements;
- Information Flow Consistency.
- Update Unmatched Settlement Instructions

1 – Life Cycle Track

Life Cycle Track

Reference Id	***LCMM.STM.LCT.1.1***

The *Life Cycle Track* function receives the status information (i.e. *Instruction Status Information and Maintenance Status Information*) and forwards it to either the *Status Driven Manager* function, the *Allegements* function, the *Night-time Reporting Manager* function or the *Data Collection for Messages* function according to the content of each status information, which includes the status attribute value, the reason code and the T2S Instruction Reference. According to the information contained in the status information, this function forwards it to the corresponding function, as the following table reflects:

TYPE	STATUS ATTRIBUTE	VALUE	TO ALLEGEMENT	TO STATUS DRIVEN MANAGER	TO NIGHT-TIME REPORTING MANAGER	TO DATA COLLECTION FOR MESSAGES
Instruction Status Information	Match Status	Matched [102]	X			X
	Cancellation Status	Cancelled	X	X		X
	Settlement Status	Settled		X [103]	X [104]	
		Unsettled [105]				X
		Partially Settled				X
	Approval Status	Awaiting Approval				X

102 There is no need to send instructions status information related to the "Unmatched" value for the Match status of Settlement Instructions since the Allegement function identifies which instructions are stored in the repository of Unmatched Instructions, to check if an Allegement shall be sent or not with a delay

103 Settled Statuses Information not corresponding to Night Time Settlement process

104 Settled Statuses Information corresponding to Night Time Settlement process

105 Instruction Status Information related to the settlement status "Unsettled" always have a new reason code attached **{T2S.13.110}**

All rights reserved.

TYPE	STATUS ATTRIBUTE	VALUE	TO ALLEGEMENT	TO STATUS DRIVEN MANAGER	TO NIGHT-TIME REPORTING MANAGER	TO DATA COLLECTION FOR MESSAGES
		Approved				X
		Revoked				X
	Validation [106]	Accepted				X
		Rejected				X
	Party / CSD Hold Status	Yes				X
		No				X
	CoSD Hold Status	Yes				X
		No				X
	CSD Validation Hold Status	Yes				X
		No				X
Maintenance Status Information	Execution Status	Executed				X
		Not Executed [107]	X [108]			X
	Cancellation Status	Denied				X
	Match Status	Matched				X
	Validation [109]	Accepted				X
		Rejected				X
	Approval Status	Awaiting Approval				X
		Approved				X
		Revoked				X

Instructions Status Information related to "*Accepted*" and "*Rejected*" status identifies when an *LCMM Instruction* has been (un)successfully validated, with the related information for its acceptance or rejection. The status "*Accepted*" or "*Rejected*" is communicated to the T2S Actors, but it is materialised in T2S by the existence or not of the *LCMM Instruction* and not by a status attribute as such.

2 – Night-time Reporting Manager

106 This kind of Instruction Status Information is not stored as a status in T2S

107 Not Executed maintenance status information "cancellation pending awaiting cancellation from the counterparty" or "cancellation pending awaiting CoSD cancellation from remaining Administering Party's received to inform the corresponding T2S Actor and its counterparty about the not execution of a Cancellation Instruction

108 Only applicable for Maintenance Status Information of Cancellation Instructions

109 This kind of Instruction Status Information is not stored as a status in T2S

All rights reserved.

Reference Id	***LCMM.STM.NRM.1.1***

The purpose of the *Night-time Reporting Manager* function is to generate the information relating to the night-time settlement period according to the following rule:

- During the night-time cycle, when receiving an *Event (Start of Sequence)* the function stops forwarding the *Instruction Status Information* relating to settlement to the *Data Collection for Messages* function;
- After receiving an *Event (End of Sequence),* the function sends the latest valid statuses attributes (from the last sequence) together with the associated reason code to the *Data Collection for Messages* function **{T2S.13.070}**;
- Then, the function starts forwarding the *Instruction Status Information* relating to settlement to the *Data Collection for Messages* function again.

3 – Status Driven Manager

Reference Id	***LCMM.STM.SDM.1.1***

The function receives status information relating to cancellation and settlement of *Settlement Instruction* and *Settlement Restrictions* and checks whether the settled or cancelled instruction has a pending *Cancellation Instruction*, so that the function cancels it.

For this purpose the *Status Driven Manager* function receives status information and checks if it relates to the settlement or cancellation of a *Settlement Instruction or Settlement Restriction* that has the Cancellation Requested Boolean set to "*True*", which means that the settled or cancelled *Settlement Instruction* or *Settlement Restriction* has a pending *Cancellation Instruction*. If that is the case, the sub-function sets the Cancellation Status attribute of the pending *Cancellation Instruction* to "*Denied*" with the Reason Code Reference "*Referenced Settlement Instruction | Settlement Restriction already settled*" or "*Referenced Settlement Instruction | Settlement Restriction already cancelled*" and forwards the related Maintenance Status Information to Data Collection for Messages **{T2S.05.450}**.

4 – Data Collection for Messages

Data Collection for Messages

Reference Id	***LCMM.STM.DCM.1.1***

Each *Instruction Status Information* or *Maintenance Status Information* processed in this function contains some key fields related to the status update, in order to allow its processing.

The *Data Collection for Messages* function is the responsible to enrich this flow with the relevant data in order to be able to fill in all fields of the ISO message or the communication that the *Outbound Processing* module sends to the user regardless if A2A or U2A communication is needed.

For this purpose the function determines which kind of message should be sent and collects all the data from the data stores of T2S, for composing a specific flow called *Message Data* considering all the potential interested parties identified {**T2S.13.060} {T2S.05.445} {T2S.05.630}**. The list of interested parties varies depending on the instruction type. E.g. possible interested parties are the following:

- Sender of the instruction;
- Originator of the instruction;

All rights reserved.

- Account owner of the Securities Account affected by the Settlement Instruction;
- CSD operating the Securities Account;
- CB operating the Dedicated Cash Account;
- Account Owner of the Dedicated Cash Account affected by the Settlement Instruction;
- Counterpart CSD Participant
- Counterpart CSD
- European Collateral Management System (ECMS) CSD Participant (as defined by the MSA on the securities account)

The relevant T2S Actors are informed accordingly with the internal and/or external reference assigned to the instruction therefore the corresponding references are collected (i.e. T2S Actor Instruction Reference and T2S Instruction Reference).

The function performs additional processes depending of the status information being processed:

- For *"Awaiting Approval"* Status Information, the function collects the relevant information related to the referenced instruction submitted under Four-Eyes principle, to inform the corresponding T2S System user to validate it **{T2S.16.170}** **{T2S.16.230}**) and forwards it to the *Outbound Processing* module as *Message Data*. This status is only communicated in U2A mode [110], and it is not communicated in any A2A queries.
- For unsuccessful validation of an *Inbound LCMM Message*, the function collects all validation errors associated to the rejection **{T2S.05.020}**. The function composes a *Message Data* to inform the T2S Actors accordingly.
- For "*Denied*" *Maintenance Status Information* related to *Amendment Instructions*, *Cancellation Instructions* or *Hold/Release Instructions*, the function collects the data related to the relevant T2S Actors, (i.e. those T2S Actors whose instructions may be affected because of the cancellation of the maintenance instruction) and forwards this *Message Data* to the *Outbound Processing* module **{T2S.05.405}** **{T2S.05.385}**.
- For "*Executed*" *Maintenance Status Information* related to a CoSD *Cancellation Instruction* or CoSD *Hold/Release Instruction* the function collects the data related to the T2S Actors and the Administering Parties, and includes it in the *Message Data* with the information received from the status information, in order to inform all relevant Administering Parties and T2S Actors about the cancellation/release of the *Settlement Instructions*.
- For "Hold" Instruction Status Information, the function collects the data related to the relevant T2S Actors, and includes it in the Message Data, in order to compose an On Hold Status Advice **{T2S.05.385}**.
- For a Restriction Type that implies the rejection of the Inbound LCMM Message or the LCMM Instruction, the function collects the information related to the corresponding T2S Actors, including the CSD or CB that set the rule, in order to compose the Message Data to inform them.

110 T2S does not report the Approval Status via a Status Advice message

All rights reserved.

- For Instruction Status Information related to setup of a CSD Validation Hold Status **or a Party Hold Status**, because of the identification of a Hold Restriction Type Rule applicable, the function collects, in addition, the information related to the CSD that set the rule, in order to compose the Message Data to inform both, the Instructing Party and the CSD accordingly.**{CR-812}**
- For Instruction Status Information of "Matched" Settlement Instructions and Settlement Restrictions for which T2S has identified that they cannot settle on their Intended Settlement Date, this function updates them from "Pending" to "Failing" by setting to "True" their "Failing flag".
  Upon completion of each relevant cut-off, this function will receive a communication from Settlement Domain with the information of the relevant instructions that remain unsettled or partially settled after the cut-off. This function will update from "Pending" to "Failing" (i.e. by setting to "True" the "Failing flag") those instructions not already identified as Failing before. The function composes the relevant Message Data and forwards it to the Outbound Processing module.
  Additionally, this function also updates from "Pending" to "Failing" (i.e. by setting to "True" the "Failing flag") those instructions that fail their eligibility check because its applicable cut-off has been reached and that have not been already identified as Failing before.

5 – Allegements

Introduction

Reference Id	***LCMM.STM.MTA.1.1***

The function processes the received Status Information (*Instruction Status Information* and *Maintenance Status Information*) and generates when relevant *Message Data to allege the counterparty*, depending on the status attribute value received **{T2S.05.540}**.

The function is divided into four different sub-functions:

- *Creation of Allegements;*
- *Cancellation of Allegements;*
- *Removal of Allegements;*
- *Alleged Cancellation Instructions.*

Creation of Allegements

Reference Id	***LCMM.STM.MTA.2.1***

The Creation of *Allegements* sub-function composes *Message Data* to allege the counterparty after an unsuccessful matching attempt, because of a missing counterpart instruction, for *Settlement Instructions* [111].

The Creation of *Allegements* sub-function creates an occurrence in the entity *Allegement* and sends a *Message Data* flow to the *Information Flow Consistency* function. The creation of Allegements is triggered when the earliest Allegement Delay Period is reached (this period is calculated from the moment an "Unmatched" Settlement Instruction is stored in the repository of unmatched instructions) **{T2S.05.540}**.

111 ISO schema files are composed in the Interface domain

All rights reserved.

There are two Allegement Delay Periods [112] which the T2S Operator is able to specify **{T2S.11.545}**:

- "*Allegement from first unsuccessful matching attempt*" is the standard delay period from the first unsuccessful matching attempt of a *Settlement Instruction* (Allegement from first unsuccessful matching attempt standard period set to 1 hour);
- "*Allegement before Intended Settlement Date*" is the standard delay period measured backwards from the FOP Cut-off time on the Intended settlement date (Allegement before Intended Settlement Date set to 5 hours).

Once the earliest Allegement Delay Period for an *"Unmatched" Settlement Instruction* is reached, the sub-function selects all the information related to the *"Unmatched" Settlement Instruction* for sending an *Allegement Message Data* and forwards it to the *Information Flow Consistency* function. The sub-function also creates an occurrence in the entity *Allegement* linked to the *Settlement Instruction* with the Allegement Status *"Sent"*.

Cancellation of Allegements

Reference Id	***LCMM.STM.MTA.3.1***

The Cancellation of *Allegements* sub-function composes and sends *Message Data* of an *Allegement* cancellation to the *Information Flow Consistency* function, to inform that the Unmatched *Settlement Instruction*, for which an *Allegement* was sent, has been cancelled and that the *Allegement* previously received is not relevant anymore {**T2S.05.540}**.

The sub-function is triggered whenever a Settlement Instruction with the Match Status attribute "*Unmatched*" is cancelled. The sub-function checks if an *Allegement* record with a Allegement Status *"Sent"* is linked to the processed Instruction, if this is the case, upon the receipt of a *Instruction Status Information* containing the information about the cancellation of an instruction, the sub-function composes the *Message Data* of the *Allegement* cancellation, and sends it to the *Information Flow Consistency* function.

The sub-function also updates the Allegement Status of the *Allegement* record linked to the *Settlement Instruction* to *"Cancelled"*, according to the nature of the status information processed for settlement {**T2S.05.540}**.

Removal of Allegements

Reference Id	***LCMM.STM.MTA.4.1***

The Removal of *Allegements* sub-function composes and sends *Message Data* of an *Allegement* removal to the *Information Flow Consistency* function, to inform that the Unmatched *Settlement Instruction*, for which an *Allegement* was sent, has been matched and that the *Allegement* previously received is not relevant anymore **{T2S.05.540}**.

After the receipt of an *Instruction Status Information* relating to matching, this sub-function checks if an Allegement record with an Allegement Status "Sent" is linked to the Settlement Instruction involved. In that case, the sub-function composes the *Message Data* of the *Allegement* removal, and sends it to the *Information Flow Consistency* function.

112 T2S shall calculate the standard delay period in hours and minutes

All rights reserved.

The sub-function also updates the Allegement Status of the *Allegement* record linked to the *Settlement Instruction* to *"Removed"*, according to the nature of the status information processed for settlement.

Alleged Cancellation Instructions

Reference Id	***LCMM.STM.MTA.5.1***

The Alleged Cancellation Instructions sub-function is in charge of dealing with the Message Data to be sent to the counterparty whenever a Cancellation Instruction needs to be matched. The sub-function does one of the following three types of processes accordingly:

- Whenever the sub-function receives a "Not Executed" with a reason code "Cancellation pending, awaiting for cancellation from the counterparty" *Maintenance Status Information* related to a Cancellation Instruction, it identifies the counterparty's *Settlement Instruction* of the referenced *Settlement Instruction* to be cancelled. Once the *Settlement Instruction* has been identified, the sub-function selects all the information for sending a *Message Data and forwards it* to the *Information Flow Consistency* function *informing about the* "Cancellation request from the counterparty is awaiting for your cancellation request status". The sub-function also creates an occurrence in the entity *Allegement* linked to the *Cancellation Instruction* with the Allegement Status "Sent" **{T2S.05.446}**.
- At the receipt of an *"Executed" Maintenance Status Information* related to a *Cancellation Instruction*, the sub-function checks if there is an Allegement Status *"Sent"* record on the *Allegement* entity linked to the *Cancellation Instruction*. If that is the case, the sub-function updates the Allegement Status record of the entity *Allegement* to *"Removed"*.
- For a *"Denied" Maintenance Status Information* related to a *Cancellation Instruction*, the sub-function checks if there is an *Allegement Status* *"Sent"* record on the *Allegement* entity linked to the *Cancellation Instruction*, and updates it to *"Cancelled"*. The sub-function also checks if the referenced *Settlement Instruction* remains *"Unsettled"*. If that is the case, the sub-function selects all the information to compose a *"Cancellation request from the counterparty is no longer valid" Message Data* and forwards it to the Information Flow Consistency function.

6 – Information Flow Consistency

Reference Id	***LCMM.STM.IFC.1.1***

The function ensures the consistency of the order in which the Status Information is processed and therefore determines the order in which the *Message Data* is sent to the *Interface* domain. The function processes all status updates received from the different modules or domains, and checks whether the data for the related message should be forwarded or not, based on the timestamp of each instruction status update, and on the *Message Data* already sent to the *Outbound processing* module, for the referenced instruction.

For this purpose once a new *Message Data* is sent to the *Outbound Processing* module this function checks if the status that has to be communicated is an allowed status transition according to the transitions explained in the introductory part of the *Status Management* module. The function also considers all the status updates of the referred instruction in its life-cycle by accessing to its status history and status update timestamp ensuring that no status information is missing. If the transition is not allowed or Status Information is missing the function stores the *Message Data* until all the logically needed *Message Data* has been sent.

All rights reserved.

This function sends an End of Process Event (End of Cycle) to the Scheduling module when all the corresponding Message Data related to the reporting of the relevant night-time cycle has been completed **{T2S.13.070}**.

7 – Update Unmatched Settlement Instructions

Upon completion of the last settlement cut-off of the business day, this function identifies as "Failing" the "Unmatched" Settlement Instructions that can no longer settle on their Intended Settlement Date, sets to "True" the "Failing" flag, and informs the T2S Actor if T2S has not already communicated the "Failing" before.

Once all the relevant instructions have been updated to failing, this function must inform the scheduler so the event for the reporting (End of Day reporting) can start.

3.4.6.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Instruction Status Information	IN	Update of the Settlement Instruction status value. Includes reason code.	SETT: Instruction Validation LCMM: Instruction Matching LCMM: Instruction Maintenance SETT: Standardisation and Preparation for Settlement SETT: Night-time Settlement SETT: Daytime Validation Provisioning and Booking	
Event (Start / End of Cycle)	IN	This event is sent in order to trigger the preparation of reporting corresponding to the night-time cycle.	OPRS: Scheduling	
Maintenance Status Information	IN	Update of the maintenance instruction status value. Includes reason code.	LCMM: Instruction Validation LCMM: Instruction Maintenance	
Message Data	OUT	Data for composing a status message		INTF: Outbound Processing

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
End of Process Event (End of Cycle)	OUT	This end of process event is sent when the preparation of reporting corresponding to the night-time cycle is completed.		OPRS: Scheduling

3.4.6.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
CRDM			
Allegement Delay Period	Attribute Domain	Read	In order to identify which Allegement Delay Period should apply.
Message Reference Data	Auto-collateralisation Rule	Read	To collect the necessary information for the sending of Message Data to the Interface
	CSD	Read	
	CSD Participant	Read	
	Eligible Counterpart CSD	Read	
	Market-Specific Party Attribute Value	Read	
	NCB	Read	
	Party	Read	
	Party Address	Read	
	Party Auto-collateralisation Eligibility	Read	
	Party Code	Read	
	Party Name	Read	
	Securities Code	Read	
	Securities	Read	
	Securities Name	Read	
	External RTGS Account	Read	

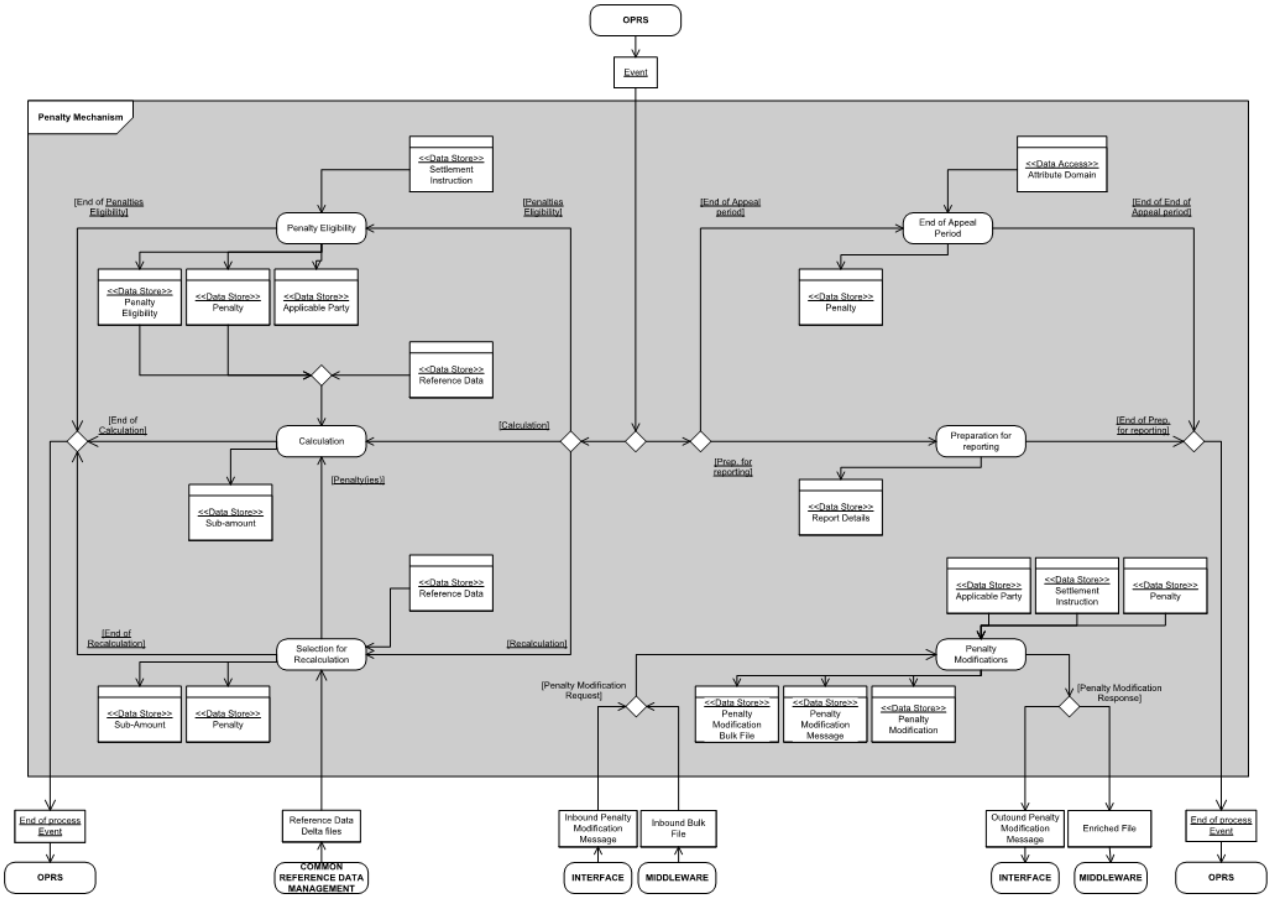
All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
	External RTGS Account Restriction	Read	
	RTGS System	Read	
	T2S BIC CMB	Read	
	T2S Dedicated Cash Account	Read	
	T2S Dedicated Cash Account Restriction	Read	
DYNAMIC DATA			
Cancellation Instruction	Cancellation Instruction	Read/Write	In order to update the Cancellation status to "Cancelled"
Unmatched Instructions	Unmatched Instructions	Read	
Allegement	Allegement	Read / Write	For the identification and recording of Allegements
LCMM Instructions	LCMM Instruction	Read	To collect the necessary information for the sending of Message Data to the Interface
	Settlement Instruction	Read	
	Cancellation Instruction	Read	
	Settlement Restriction	Read	
	Conditions Modification Instruction	Read	
	Amendment Instruction	Read	
	Hold/Release Instruction	Read	
	LCMM Instruction Additional Information	Read	
	Matching Object	Read	

All rights reserved.

3.4.7 Penalty Mechanism

3.4.7.1 Diagram of the module



3.4.7.2 Description of the module

The Penalty Mechanism module manages the detection, calculation, recalculation, modification and preparation for reporting of Penalties.

The Penalty Mechanism module detects and calculates two types of Penalties:

- Settlement Fail Penalties (SEFP): For Settlement Instructions that i) are "Matched" before the end of processing of the relevant cut-off of a given business day and that ii) having reached their Intended Settlement Date (ISD), fail to settle on that business day. These Penalties are calculated for each business day the instruction fails to settle in T2S (i.e. for each and every business day that an instruction keeps failing in T2S, a new Penalty will be computed).

Late Matching Fail Penalties (LMFP): For Settlement Instructions that are "Matched" in T2S after the relevant cut-off of their Intended Settlement Date (ISD). These Penalties are calculated only once per pair of matched instructions (i.e. on the business day when they are matched), but do consider all the previous business days when the instructions failed to settle due to the late matching i.e. due to their late sending.

All rights reserved.

The recalculation of Penalties is the process consisting on calculating again and updating existing Penalties (i.e. that have already been calculated on previous business days) after changes of values of the related T2S Reference Data or after the execution of a modification requested by a T2S Actor.

The modification of Penalties consists on processing requests sent by T2S Actors to remove, re-include, re-allocate or switch existing Penalties that are allowed until they reach the end of their appeal period.

The preparation for reporting consists on calculating the aggregation of amounts (daily or monthly) and preparing the information to be included in the corresponding predefined reports.

The inputs for this module are:

- Penalties Eligibility Event;
- Calculation Event;
- Recalculation Event;
- End of Appeal Period Event;
- Preparation for reporting Events;
- Inbound Penalty Modification Message;
- Inbound Penalty Modification via bulk file.

3.4.7.3 Description of the functions of the module

The Penalty Mechanism module is composed of six functions:

- Penalty Eligibility;
- Calculation;
- Selection for Recalculation;
- Penalty Modifications;
- End of Appeal Period;
- Preparation for Reporting.

1 – Penalty Eligibility

Introduction

Reference Id	**LCMM.PEM.ELI.1.1**

This function is triggered every T2S settlement day **{T2S.03.365} {T2S.03.370} {T2S.03.380}** by the Start Cash Penalty Eligibility Event (SCPE), received from the Scheduling Module. It analyses whether a Settlement Instruction is eligible either for a "SEFP" or for a "LMFP" Penalty, i.e.: if a Penalty has to be calculated either for a settlement fail or for a late matching on the previous business day.

The function is divided in the following sub-functions:

- Eligibility Start
- SEFP Penalty Eligibility

All rights reserved.

- LMFP Penalty Eligibility
- Calculation Method Identification
- Create Penalty
- Get and Store Applicable Party Data
- Eligibility End

Eligibility Start

Reference Id	**LCMM.PEM.ELI.2.1**

Upon reception of the Start Cash Penalty Eligibility (SCPE) Event, the Eligibility Start sub-function sets the Detection Date for all potential Penalties as the previous business date **{T2S.03.380}.** It also defines the Detection Date beginning and ending Timestamps as follows:

- The Timestamp associated with Change of Business Day two days before (start of d-1)
- The Timestamp associated with the previous Change of Business Day (end of d-1)

Then, the sub-function triggers SEFP Penalty Eligibility and LMFP Penalty Eligibility sub-functions.

SEFP Penalty Eligibility

Reference Id	**LCMM.PEM.ELI.3.1**

The SEFP Penalty Eligibility sub-function retrieves the Settlement Instructions which failed to settle during the Detection Date set in the Eligibility Start, i.e. those matched Settlement Instructions identified as failed by the Standardisation and Preparation to Settlement Module at the processing of their cut-off.

Internally generated realignment instructions are out of scope of the T2S Penalty Mechanism **{T2S.22.040}**, hence, Settlement Instructions with ISO Transaction Code "REAL" and with T2S Actor reference "empty" are not retrieved.

For each Settlement Instruction retrieved, a Penalty Eligibility instance is created.

Corporate actions on stock are not subject to Penalties, hence:

- If the Settlement Instruction has ISO Transaction Code "CORP" its Penalty Eligibility instance is recorded with Eligible for a Penalty "False" and Reason for Exemption "ISOT" (i.e. ISO Transaction Code) **{T2S.22.050}**, and the process ends for this Settlement Instruction.
- Otherwise, the process continues. To continue the processing of the Penalty Eligibility [113] , the sub-function retrieves the Statuses and Reasons that caused the settlement failure of the Settlement Instruction {T2S.22.110} i.e. the "Unsettled" Reason Codes and Hold Statuses that the Settlement Instruction had at the end of processing of its cut-off.

Then, the sub-function checks the "Unsettled" Reason Codes and Hold Statuses versus the Failing Reasons Dictionary in order to identify whether the Settlement Instruction is eligible for a "SEFP" Penalty or not {T2S.22.080}, as described in the following – non-exhaustive – table [114]:

113 Or to reprocess an existing Penalty Eligibility with Reason for Exemption "REPR" sent by the Eligibility End sub-function

114 The exhaustive list is provided in the UDFS.

All rights reserved.

SETTLEMENT INSTRUCTION'S UNSETTLED REASON CODES AND HOLD STATUSES	PENALTY ELIGIBLE
Unsettled with LACK – Lack of Securities	"True"
Unsettled with CLAC – Counterparty Insufficient Securities	"False"
Unsettled with MONY – Insufficient Money	"True"
Unsettled with CMON – Counterparty Insufficient Money	"False"
Unsettled and on Party Hold	"True"
Unsettled and on CSD Hold	"True"
Unsettled and on COSD Hold	"True"
Unsettled and on CSD Validation Hold	"True"
Unsettled without Holds and with counterpart instruction on Hold	"False"

Each "Unsettled" Reason Code and Hold Status is analysed one by one [115]:

As soon as one "Unsettled" Reason Code or Hold Status is defined as Penalty Eligible "True" in the Failing Reasons Dictionary, the sub-function stops checking and:

- Sets to "True" the attribute Eligible for a Penalty in the Penalty Eligibility entity,
- Retrieves the data of the Counterparty's Settlement Instruction, which will be the non-failing leg of the Penalty,
- Determines the Common MIC (Place of Trade informed equally in both Settlement Instructions);
- Calculates and stores the remaining Quantity and/or Amount to be settled at the Cut-off timestamp of the Detection date, by subtracting to the original quantity of securities or to the original amount of the Settlement Instruction, the quantity of securities or the amount of cash settled at the Cut-off; and
- Sends the Penalty Eligibility to the Calculation method identification sub-function to continue with the process.

In case that after analysing all the "Unsettled" Reason Codes and all Hold Statuses of the Settlement Instruction, none of them was defined as Penalty Eligible "True" in the Failing Reasons Dictionary, then the sub-function:

- Sets to "False" the attribute Eligible for a Penalty in the Penalty Eligibility entity,
- Sets the attribute Reason for Exemption in the Penalty Eligibility entity to:

115 In case a Reason Code is missing in the "Failing reasons Dictionary" an alarm is raised to the T2S Operator

All rights reserved.

- Either "RECO" if all "Unsettled" Reason Codes were defined in the Failing Reasons Dictionary,
- Or "REPR" if at least one of the "Unsettled" Reason Codes was missing in the Failing Reasons Dictionary.

- Ends the process for this Settlement Instruction.

LMFP Penalty Eligibility

Reference Id	**LCMM.PEM.ELI.4.1**

The LMFP Penalty Eligibility sub-function retrieves the Settlement Instructions that i) have an Intended Settlement Date equal to or before the Detection Date set in the Eligibility Start and ii) were matched on the previous business day (i.e. those with Matching Timestamp between the Detection Date beginning and ending Timestamps).

Internally generated realignment instructions are out of scope of the T2S Penalty Mechanism {T2S.22.040}, hence Settlement Instructions with ISO transaction code "REAL" and with T2S Actor reference "empty" are not retrieved.

For each Settlement Instruction retrieved, in order to determine if it is eligible for a "LMFP" Penalty, the sub-function checks:

1. If it was matched at a point in time when it was no longer possible to settle it on that business day (i.e. on the Detection Date). For this purpose, the sub-function checks if the Settlement Instruction fulfils any of the following:
   - Either the Settlement Instruction's Reason History has a "LATE" Reason Code; or
   - The Matching Timestamp is after the start of the last cut-off of that business day (i.e. after the "EFOP" Event Start Time timestamp of the Detection Date);
   - In case the Settlement Instruction fulfils any of the two conditions, a Penalty Eligibility instance is created with Matched After Cut-Off flag "True"
2. Provided the previous check is not met (the instruction was matched at a point in time when it was still possible to settle it on the Detection Date), the sub-function checks:
   - If the Intended Settlement Date of the Settlement Instruction is before the Detection Date.

In case the Settlement Instruction fulfils this condition, a Penalty Eligibility instance is created with Matched After Cut-Off flag "False";

In case none of the checks is met, no instance is created in Penalty Eligibility for the Settlement Instruction (i.e. the instruction did not match late at all) **{T2S.22.160}**, **{T2S.22.170}**.

For each Penalty Eligibility instance created, the sub-function determines its Eligible for Penalty attribute as follows:

- If the underlying Settlement Instruction has ISO transaction code "CORP" the Eligible for Penalty is set to "False" and the Reason for Exemption to "ISOT" (i.e. ISO Transaction Code) **{T2S.22.050}**.

All rights reserved.

- If the underlying Settlement Instruction has Settlement Transaction Condition Code "BSSP", the Settlement Transaction Condition Code of the counterparty Settlement Instruction is checked. If it is "BSSP" too, then the Eligible for Penalty is set to "False" and the Reason for Exemption to "BSSP" {T2S.22.172}.
- Otherwise, the counterparty's Settlement Instruction is retrieved through the Matching Object and:
  - If the underlying Settlement Instruction was matched in T2S, the Validation Ending Timestamp of both Settlement Instructions is compared:

    (i) In case the underlying Settlement Instruction has a later Validation Ending Timestamp than the one of its counterparty, then the Eligible for Penalty is set to "True";

    (ii) Otherwise (i.e. the underlying Settlement Instruction has an earlier Validation Ending Timestamp than the one of its counterparty), then the Eligible for Penalty is set to "False" and the Reason for Exemption to "RECO" (reason for failing is associated to the counterparty instruction because it entered later) **{T2S.22.160}.**
  - If the underlying Settlement Instruction was sent as already matched, it has the same Validation Ending Timestamp as its counterpart´s, therefore, the sub-function analyses the Securities Movement Type:

    (i) In case the underlying Settlement Instruction is "DELI", then the Eligible for Penalty is set to "True";

    (ii) Otherwise (i.e. the underlying Settlement Instruction is "RECE"), then the Eligible for Penalty is set to "False" and the Reason for Exemption to "RECO" (reason for failing is associated to the counterparty instruction because it entered later) **{T2S.22.170}.**

For each Penalty Eligibility instance with Eligible for a Penalty "True", the sub-function retrieves and stores the Common MIC (Place of Trade informed equally in the Settlement Instruction and in its counterpart´s), the original Settlement Quantity and the Matched Settlement Amount i.e. at the moment of the matching; and triggers the Calculation method identification sub-function to process them.

Calculation method identification

Reference Id	**LCMM.PEM.ELI.5.1**

The Calculation method identification sub-function determines the Calculation Method that has to be applied to calculate the Penalty.

The possible values for the Calculation Method are the following:

- "SECU": The Penalty is based on the quantity of securities failed to be delivered, the price and the penalty rate of the relevant Asset Type;
- "MIXE": The Penalty is based on the quantity of securities failed to be delivered, the price of the security and the discount rate of the relevant Currency;
- "CASH": Penalty is based on the amount of cash failed to be delivered and the discount rate of the relevant Currency;
- "BOTH": Penalty is the sum of "SECU" and "CASH". I.e. the sum of:

All rights reserved.

- The Penalty based on the quantity of securities failed to be delivered, the price and the penalty rate of the relevant Asset Type ("SECU"); and
- The Penalty based on the amount of cash failed to be delivered and the discount rate of the relevant currency ("CASH").

The Calculation Method is determined based on the Instruction Type of the underlying Settlement Instruction that generated the Penalty Eligibility instance, according to the following logic **{T2S.22.120} {T2S.22.121} {T2S.22.130} {T2S.22.140} {T2S.22.210} {T2S.22.220} {T2S.22.230} {T2S.22.231}:**

Instruction type	Calculation method
PFOD ("DPFOD" or "CPFOD") ▪ Quantity = 0	"CASH"
FOP ("DFOP" or "RFOP") ▪ Payment Type = "FREE" and Quantity > 0	"SECU"
DVP ▪ Securities Movement Type = "DELI", Quantity > 0, Payment Type = "APMT" and Credit/ Debit Code = "CRDT"	"SECU"
DWP or RWP ▪ Either Securities Movement Type = "DELI", Quantity > 0, Payment Type = "APMT" and Credit/Debit Code = "DBIT"; ▪ Or Securities Movement Type = "RECE", Quantity > 0, Payment Type = "APMT" and Credit/Debit Code = "CRDT"	"BOTH"
RVP [116] ▪ Securities Movement Type = "RECE" , Quantity > 0, Payment Type = "APMT" and Credit/Debit Code = "DBIT"	"MIXE"

Once the Calculation Method is determined, the sub-function triggers the Get Penalty Data sub-function.

116 According to {T2S.22.232}, in the case of a RVP received as an already matched instruction, the Penalty Calculation Method applied should be "SECU". However, in case of already matched instructions, the LMFP Penalty Eligibility sub-function analyses the Securities Movement Type of the Settlement Instruction, and in case it is "DELI", the Settlement Instruction is deemed eligible for a Penalty; otherwise, i.e. it is "RECE", the Settlement Instruction is deemed as not eligible for a Penalty. Hence, the Calculation Method identification sub-function won't have to determine the Calculation Method for a Penalty whose underlying Settlement Instruction is RVP and sent as already matched.

All rights reserved.

Create Penalty

Reference Id	**LCMM.PEM.ELI.6.1**

The sub-function creates the corresponding Penalty with Status "PEND" associated to the Penalty Eligibility, and assigns it a Common Penalty ID (composed of 15 characters).

Additionally, the following information already retrieved by the previous sub-functions is stored in the Penalty:

- Type of Penalty (LMFP or SEFP)
- Detection Date
- Calculation Method (SECU, CASH, MIXE or BOTH)
- Security
- Underlying SI Already Matched flag (True or False)

Then, the sub-function determines the attributes Payment Type of the SI and, when possible, the Currency of the Penalty as follows **{T2S.22.060}:**

- In case the underlying Settlement Instruction Payment Type Code is "APMT" (i.e. against payment), then the Payment Type of the SI is recorded as "APMT" and the Currency is set with the same value as the Currency of the Amount of the Settlement Instruction;
- In case the underlying Settlement Instruction Payment Type Code is "FREE" (i.e. free of payment), then the Payment Type of the SI is recorded as "FREE" and the Settlement Type of the Security is checked to determine the Currency:
  - If the Settlement Type is "FAMT" (i.e. Face amount), the Currency of the Security is retrieved from Reference Data and
    i) if it is a T2S Settlement Currency different from "EUR" (e.g. "DKK"), then it is checked whether for this Currency, the BIC of the CSD of any of the Applicable Parties of the Penalty (the Failing Party or the Non-Failing Party) is in the List of CSDs with penalty computation in non-Euro T2S settlement currencies for FOPs. If this is the case, the Currency of the Penalty is set with the same value as the Currency of the Security; otherwise, the Currency of the Penalty is set to "EUR";
    ii) if it is either "EUR" or it is not a T2S Settlement Currency, then the Currency of the Penalty is set to "EUR".
  - If the Quantity Settlement Type is "UNIT" (i.e. Units), the Currency of the Penalty is not determined and left empty until it is determined by the Calculation function.

Once all activities of this sub-function are finished, the Get and Store Applicable Party Data sub-function is triggered.

Get and Store Applicable Party Data

Reference Id	**LCMM.PEM.ELI.7.1**

The sub-function determines the Failing Party and the Non-Failing Party of the Penalty, i.e.: the Party debited with the Penalty, and the Party credited with the Penalty, respectively.

All rights reserved.

- In case of a "LMFP" with Underlying SI Already Matched flag "True", the Instructing Party of the underlying Settlement Instruction will be both the Failing and the Non-Failing Party **{T2S.22.180}.**
- Otherwise, the sub-function checks the Depositories of the underlying Settlement Instruction for which the Penalty is computed:
  - In case the Depositories involved in the Settlement Instruction are both CSDs in T2S, the Failing Party will be the account owner of the Securities Account of the instruction, whereas the Non-Failing Party will be the account owner of the Securities Account of the counterparty **{T2S.22.090} {T2S.22.180}.**
  - In case of External CSD settlement scenarios (either Delivering Depository is an External CSD or Receiving Depository is an External CSD), T2S will consider the "External CSD" as the relevant securities account owner and, consequently, as the Failing Party or the Non-Failing Party **{T2S.22.100} {T2S.22.190}.**

With the information above, the sub-function generates two Applicable Party instances (i.e.: one for the Failing Party and another one for the Non-failing Party) and, for each one of them, retrieves and stores its necessary attributes, i.e.:

- Individual Penalty ID: Composed of 16 characters (i.e.: F+Common Penalty ID for the Failing Party; N+Common Penalty ID for the Non-Failing Party).
- The CSD of the Applicable Party (of the Failing Party or of the Non-Failing Party) [117].
- Credit / Debit indicator: "Debit" for the Failing Party or "Credit" for the Non-Failing Party.
- T2S Reference of the related Settlement Instruction (of the Failing Party or of the Non-Failing Party). In case the Penalty is a "LMFP" with Underlying SI Already Matched flag "True", and the Instructing Party is both the Failing and the Non-failing party, T2S assumes that the Delivering leg is the Settlement Instruction of the Failing Party and the Receiving leg is the Settlement Instruction of the Non-Failing Party **{T2S.22.180}.**

Eligibility End

Reference Id	**LCMM.PEM.ELI.8.1**

Once the SEFP Penalty Eligibility and LMFP Penalty Eligibility sub-functions have processed all the corresponding Settlement Instructions and all Penalties and their Applicable Parties have been created, this sub-function checks if there are instances in Penalty Eligibility from previous business days (i.e. with Detection Date before the previous business day) with Reason for Exemption "REPR". In case there is any, it is sent to SEFP Penalty Eligibility sub-function to reprocess it.

Once all Penalty Eligibility instances with Reason for Exemption "REPR" have been reprocessed, the sub-function sends the End of Process Event (Start Cash Penalty Eligibility) to the Scheduling Module.

2 – Penalty Calculation

Introduction

117 In case of an External CSD settlement scenario, the CSD of the Applicable Party is the parent CSD of the External-CSD.

All rights reserved.

Reference Id	**LCMM.PEM.CAL.1.1.**

This function is triggered every T2S settlement day **{T2S.03.370} {T2S.03.380**} by the Calculation of Penalties (CALC) Event, received from the Scheduling Module. Additionally, this function is also triggered by the function Selection for Recalculation in order to calculate again existing Penalties.

The Penalty Calculation function calculates the Amount of each Penalty.

The Amount of a Penalty can be formed by one or several Sub-amounts, depending on the Number of business days when the Penalty applies:

- In case of "SEFP" the Number of business days is always one (and therefore there is one Sub-amount) **{T2S.22.115}**;
- In case of "LMFP" the Number of business days (and therefore Sub-amounts) can be one or several ones **{T2S.22.200}**.

Additionally, depending on the applicable Calculation Method (previously identified by the Penalty Eligibility function) the Amount of each Sub-amount will be formed by:

- Either an Amount Cash Side, based on the amount failed to be delivered, if the Calculation Method of the Penalty is "CASH"; or
- An Amount Securities Side, based on the quantity of securities failed to be delivered, if the Calculation Method of the Penalty is "SECU" or "MIXE"; or
- Both, an Amount Cash Side based on the amount failed to be delivered and an Amount Securities Side based on the quantity of securities failed to be delivered, if the Calculation Method of the Penalty is "BOTH".

The function is divided in the following sub-functions:

- Fetch New Penalties to be calculated
- Penalty Preparation
- Sub-amount Preparation
- Calculate Sub-amount

Fetch New Penalties to be calculated:

Reference Id	**LCMM.PEM.CAL.2.1.**

Upon reception of the Calculation of Penalties (CALC) Event, this sub-function retrieves all the new Penalties (with Detection Date the previous business day and Status "PEND"), which have to be calculated in the current business day. Then it triggers the Penalty Preparation sub-function.

Penalty Preparation:

Reference Id	**LCMM.PEM.CAL.3.1.**

The sub-function is triggered either by the Fetch New Penalties to be calculated sub-function for new Penalties to be calculated, or by the function Selection for Recalculation for recalculation of existing Penalties.

All rights reserved.

For each Penalty to be (re)calculated, this sub-function identifies the Number of business days, and generates the corresponding Sub-amounts:

- For "SEFP" Penalties, the Number of business days is always one, i.e. the applicable business day is the Detection date of the Penalty {T2S.22.115}.
- For "LMFP" Penalties, the sub-function counts the applicable business days from the Intended Settlement Date [118] of the underlying Settlement Instruction, until the following date:
  - The business day before the Detection date of the Penalty, if the Settlement Instruction was matched before its cut-off (i.e. if the corresponding Penalty Eligibility has Matched After Cut-Off flag "False");
  - The Detection date of the Penalty, if the Settlement Instruction was matched after its cut-off (i.e. if its corresponding Penalty Eligibility has Matched After Cut-Off flag "True") {T2S.22.200}.

    In order to count the applicable business days, the sub-function considers T2S Settlement Days, i.e. days for which the instruction can settle in T2S. Therefore:

    (i) In case the Payment Type of the SI is "APMT" (against payment instructions), T2S Settlement Days are business days that are not closing days for the relevant Currency of the cash amount of the Settlement Instruction;

    (ii) In case the Payment Type of the SI is "FREE" (free of Payment instructions), T2S Settlement Days are business days that are not T2S common closing days for all currencies.

Then, the sub-function generates one Sub-amount for each of the identified applicable business day(s) of the Penalty (i.e. creates one Sub-amount with Date equal to the applicable business day, for each of the applicable business days). Depending on the Calculation Method, each Sub-amount can be formed by:

- Either one Sub-amount (securities) if the Calculation Method of the Penalty is "SECU" or "MIXE"; or
- One Sub-amount (cash) if the Calculation Method of the Penalty is "CASH";
- One Sub-amount (securities) and one Sub-amount (cash) if the Calculation Method of the Penalty is "BOTH".

The Sub-Amounts of the Penalty are sent to the Sub-amount Preparation sub-function. Once the Sub-amount Preparation and Sub-amount Calculation sub-functions have processed all the Sub-Amounts of the Penalty:

- If the Security is not Subject to Penalties **{T2S.22.010}** in any of the applicable business days (i.e. all the Sub-amounts of the Penalty have the Security subject to penalties flag equal to "False"), the Status of the Penalty is updated to "Not computed" ("NCOM") and, therefore, the Penalty will not be reported **{T2S.22.200}** [See Penalties Preparation for Reporting function].
- The sub-function informs the Selection for Recalculation function that the Penalty has been recalculated.

Finally, once all the new Penalties have been calculated, the sub-function sends the End of Process Event (Calculation of Penalties) to the Scheduling Module.

118 In case the Intended Settlement Date of the underlying Settlement Instruction is prior the CSDR Entry into force Date, for "LMFP" Penalties, the sub-function counts the applicable business days from the CSDR Entry into force Date.

All rights reserved.

Sub-amount Preparation:

Reference Id	**LCMM.PEM.CAL.4.1.**

Upon reception of the Sub-amounts of a Penalty, either from the Penalties Preparation sub-function or from the Selection for Recalculation function, this sub-function retrieves all the necessary information for each of them.

First, the sub-function checks if the Security is Subject to Penalties for each Sub-amount (i.e. for the Date of each Sub-amount) and stores the result in the Security subject to penalties flag of the relevant Sub-amount.

- For Sub-amounts with Security subject to penalties flag is "False" the processing ends.
- For each Sub-amount with Security subject to penalties flag "True", the sub-function retrieves and stores the necessary Penalty Reference Data for Securities attributes and Reference data information to calculate the Penalty. Note that in case of LMFPs, if any of the Sub-amounts has a Date before 92 days in the past, the Penalty Reference Data retrieved and stored for calculation is the first available one **{T2S.22.240**} i.e.: data from "d-92".

The sub-function performs two steps:

- Retrieve Price, Currency and Euro FX
- Determine rate

Retrieve Price, Currency and Euro FX:

Reference Id	**LCMM.PEM.CAL.4.2.**

For Sub-amounts whose Penalty has Calculation Method "SECU", "MIXE" or "BOTH", this step performs the following actions:

The step first retrieves the Daily Price of the Security for the relevant Date of the Sub-amount **{T2S.22.150} {T2S.22.240**}. Then:

In case the Daily Price is not available in Reference Data, the Sub-amount is flagged as Cannot be calculated "True".

Otherwise, the step performs different checks depending on the Settlement Type of the Security.

In case it is "UNIT":

- The step verifies that the Settlement Type of the quantity stored in Penalty Eligibility from the underlying Settlement Instruction is also "UNIT". In case the condition is not fulfilled, the relevant Sub-amount is flagged as Cannot be calculated "True", because the Price would be in "UNIT" (Price Amount) whereas the failed quantity would be in "FAMT" (i.e. the Settlement Type of the Security has been updated and the Penalty cannot be calculated for this business day).
- The step checks whether the Payment Type of the SI of the Penalty is "FREE", and if so, the Currency of the Penalty is determined as follows:
  i) If Currency of the Daily Price of the Security is a T2S Settlement Currency different from "EUR" (e.g. "DKK"), it is checked whether for this Currency the BIC of the CSD of any of the Applicable Parties of the Penalty (the Failing Party or the Non-Failing Party) is in the List of CSDs with penalty computation in non-Euro T2S settlement currencies for FOPs. If this is the case, the Currency of the Penalty is set

All rights reserved.

with the same value as the Currency of the Daily Price; otherwise, the Currency of the Penalty is set to "EUR".

ii) If Currency of the Daily Price of the Security is either "EUR" or it is not a T2S Settlement Currency, then the Currency of the Penalty is set to "EUR".

- Finally, in case the Sub-amount can be calculated, the step checks if an FX conversion is needed. For this purpose, the process verifies whether the Currency of the Penalty is different from the Currency of the Daily Price. If they are equal, the Conversion flag of the Sub-amount (securities) is set to "False" and the step Determine rate is triggered; if they are different, the Conversion flag of the Sub-amount (securities) is set to "True" and:
  - If the Currency of the Penalty is "EUR", only one Euro Foreign Exchange Rate is retrieved and stored for the relevant Currency of the Daily Price and Date.

    (i) In case the Euro Foreign Exchange Rate is not available in Reference Data, the Sub-amount is flagged as Cannot be calculated "True" and the processing ends.

    (ii) Otherwise, the step Determine rate is triggered.
  - If the Currency of the Penalty is not "EUR" (i.e. it is "DKK"), an additional Euro Foreign Exchange Rate is necessary to pivot from "EUR" to "DKK", thus retrieved and stored.

    (i) In case any of the two Euro Foreign Exchange Rates is not available in Reference Data, the Sub-amount is flagged as Cannot be calculated "True" and the processing ends.

    (ii) Otherwise, the step Determine rate is triggered.

In case it is "FAMT":

- The step verifies that the Settlement Type of the quantity stored in Penalty Eligibility from the underlying Settlement Instruction is also "FAMT".
  - In case the condition is not fulfilled, the relevant Sub-amount is flagged as Cannot be calculated "True" because the Price would be in "FAMT" (Coefficient) whereas the failed quantity would be in "UNIT" (i.e. the Settlement Type of the Security has been updated and the Penalty cannot be calculated for this business day) and the processing ends.
- In case the above condition is fulfilled and the Sub-amount can be calculated, the step verifies that the Currency of the Security is the same as the Currency of the quantity stored in Penalty Eligibility from the underlying Settlement Instruction.
  - In case the condition is not fulfilled, the relevant Sub-amount is flagged as Cannot be calculated "True" (i.e. the Currency of the Security has been updated and the Penalty cannot be calculated for this business day) and the processing ends.
- Finally, in case the Sub-amount can be calculated, the step checks if a FX conversion is needed. For this purpose, the process verifies whether the Currency of the Penalty is different from the Currency of the Security. If they are equal, the Conversion flag of the Sub-amount (securities) is set to "False" and the step Determine rate is triggered; if they are different, the Conversion flag of the Sub-amount (securities) is set to "True" and:

All rights reserved.

– If the Currency of the Penalty is "EUR", only one Euro Foreign Exchange Rate is retrieved and stored for the relevant Currency of the Security and Date.

(i) In case the Euro Foreign Exchange Rate is not available in Reference Data, the Sub-amount is flagged as Cannot be calculated "True" and the processing ends.

(ii) Otherwise, the step Determine rate is triggered.

– If the Currency of the Penalty is not "EUR" (i.e. it is "DKK"), an additional Euro Foreign Exchange Rate is necessary to pivot from "EUR" to "DKK", thus retrieved and stored.

(i) In case any of the Euro Foreign Exchange Rates is not available in Reference Data, the Sub-amount is flagged as Cannot be calculated "True" and the processing ends.

(ii) Otherwise, the step Determine rate is triggered.

Determine rate:

Reference Id	**LCMM.PEM.CAL.4.3.**

Depending on the Calculation Method of the Penalty the step performs the following actions:

- For "CASH" and "MIXE" Calculation Methods, the step retrieves the Cash Discount Penalty Rate (expressed as a daily flat rate) of the corresponding Currency of the Penalty and Date of the Sub-amount. Then:
  – In case the Cash Discount Penalty Rate is not available in Reference Data, the Sub-amount is flagged as Cannot be calculated "True" and processing ends.
  – Otherwise, the Sub-amount is sent to the Calculate Sub-amount sub-function.
- For "SECU" Calculation Method, the step determines whether the Common MIC (Market Identifier Code) is an SME Growth Market or not. Then, for the relevant Date and Security the sub-function requests Reference Data the Financial Instrument Type, Liquidity (if the Financial Instrument Type is equal to "SHRS") and the Security Penalty Rate (for SME or for non-SME depending on the SME Growth Market flag) **{T2S.22.145}**:
  – In case either the Financial Instrument Type, Liquidity or Security Penalty Rate is not available in Reference Data, the Sub-amount is flagged as Cannot be calculated "True" and the processing ends.
  – Otherwise, the Sub-amount is sent to the Calculate Sub-amount sub-function.
- For "BOTH" Calculation Method, the above described actions for "CASH" and "SECU" are performed.

Once the rate has been determined, the sub-function sends the Penalty and all Sub-amounts flagged as Cannot be calculated "False" to the Calculate Sub-amount sub-function.

Calculate Sub-amount:

Reference Id	LCMM.PEM.CAL.5.1.

For each Sub-amount received from the Sub-amount Preparation sub-function, the sub-function calculates the Amount of the Sub-amount, applying the steps required for each Calculation Method:

- For "CASH" Calculation Method, the Amount of the Sub-amount is the same as Amount Cash side, which is calculated as **{T2S.22.130}** **{T2S.22.220}**:

All rights reserved.

Amount (failed to be delivered) * Cash Discount Penalty Rate

- For "SECU" Calculation Method, the Amount of the Sub-amount is the same as Amount Securities side, which is calculated in two steps, first the sub-function calculates the Original Amount of Sub-amount (securities) and then, if necessary it is converted applying the previously retrieved Euro foreign exchange rates. More in detail:
  - The Original Amount of Sub-amount (securities) is calculated as **{T2S.22.120} {T2S.22.210} {T2S.22.232}**:
  - Securities Penalty Rate * Daily Price of the Security * Quantity (failed to be delivered)
  - The Amount Securities side is calculated as:

    (i) If the Conversion flag of the Sub-amount (securities) is "False", the Amount Securities side is equal to the Original Amount;

    (ii) If the Conversion flag of the Sub-amount (securities) is "True", depending on the number of Euro foreign exchange rate stored, the Amount Securities side is:

    Original Amount * First Euro foreign exchange rate, if Sub-amount Currency is "EUR"; or

    Original Amount * First Euro foreign exchange rate * Second Euro foreign exchange rate.
- For "MIXE" Calculation Method, the Amount of the Sub-amount is the same as Amount Securities side, which is calculated in two steps, first the sub-function calculates the Original Amount of Sub-amount (securities) and then, if necessary it is converted applying the previously retrieved Euro foreign exchange rates. More in detail:
  - The Original Amount of Sub-amount (securities) is calculated as **{T2S.22.121} {T2S.22.231}**:
  - Cash Discount Penalty Rate * Daily Price of the Security * Quantity (failed to be delivered)
  - The Amount Securities side is calculated as:

    (i) If the Conversion flag of the Sub-amount (securities) is "False", the Amount Securities side is equal to the Original Amount

    (ii) If the Conversion flag of the Sub-amount (securities) is "True", depending on the number of Euro foreign exchange rate stored, the Amount Securities side is:

    Original Amount * First Euro foreign exchange rate, if Sub-amount Currency is "EUR"; or

    Original Amount * First Euro foreign exchange rate * Second Euro foreign exchange rate.
- For "BOTH" Calculation Method, the Amount of the Sub-amount will be the sum of the Amount Securities side and the Amount Cash side, which are calculated as described above for "SECU" and "CASH" Calculation Method respectively **{T2S.22.140} {T2S.22.230}**.

Once the Amount of the Sub-amount has been calculated, the Amount of the Penalty is updated, summing the Amounts of all of its Sub-amounts, and the sub-function informs either the Penalty Preparation sub-function or the Selection for Recalculation function that the Sub-amount and the Penalty have been processed.

3 - Penalty Modification

Introduction

All rights reserved.

Reference Id	**LCMM.PEM.MOD.1.1**

The Penalty Modification function is triggered either by the Inbound Penalty Modification Messages or by the Inbound Bulk Files sent to LCMM, either from the Interface when sent U2A by the T2S Actor or from the Middleware when sent A2A by the T2S Actor, respectively {T2S.22.450}.

This function analyses whether an existing Penalty can be modified expost (i.e. "removed", "re-included", "re-allocated" or "switched") by the CSD when either its corresponding participant or the CSD itself have been imposed with it, provided the end of the Appeal period of the Penalty has not been reached yet **{T2S.22.440}**.

A Penalty Modification can have the following Request Type:

- "Penalty Removal" or "REMO" **{T2S.22.300}**;
- "Penalty Re-inclusion" or "REIN" **{T2S.22.340}**;
- "Penalty Switch" or "SWIC" **{T2S.22.400}**;
- "Penalty Re-allocation" or "RALO" **{T2S.22.370}**.

In case the required conditions are fulfilled, the Penalty Modification is executed and the Penalty is updated accordingly; otherwise, the Inbound Penalty Modification Message or the Penalty Modification included in the Inbound Bulk File is rejected.

This function is divided in the following sub-functions:

- Validation of Penalty Modifications sent via Bulk File
- Validation of Inbound Penalty Modification Messages and Incoming Instruction Approval Requests
- Perform Removal
- Perform Re-inclusion
- Perform Switch
- Perform Re-allocation.

Validation of the Penalty Modifications sent via Bulk File

Reference Id	**LCMM.PEM.MOD.2.1**

Upon reception of the Inbound Bulk File, which may contain one or several Penalty Modifications, the sub-function checks if the sender is authorised to perform the action of sending the file on behalf of the instructing CSD. For this purpose the function checks:

- If the BIC and Parent BIC of the Instructing Party [119] of the Inbound Bulk File correspond to an existing Party defined as "CSD" in T2S Reference Data, and if so
- If the Distinguished Name (DN) of the sender, the BIC and Parent BIC of the CSD are defined in the "white list" configuration table stored in the Attribute Domain in T2S Reference Data.

119 BICs informed in the Header of the Bulk File ("BIC of the CSD" and "Parent BIC of the CSD" respectively)

All rights reserved.

In case either the BICs of the Instructing Party do not correspond to a Party defined as "CSD", or the combination of Distinguished Name (DN), Parent BIC and BIC are not defined in the "white list", the relevant Request Error is stored and an XML response is sent back to the Sender informing about the rejection of the whole Bulk File.

Otherwise, the sub-function checks whether each and every Penalty Modification of the Inbound Bulk File complies with the business requirements **{T2S.22.310} {T2S.22.320} {T2S.22.350} {T2S.22.380} {T2S.22.410} {T2S.22.420}**.

This sub-function rejects the execution of a Penalty Modification when one of the conditions in the table below is fulfilled:

Conditions for rejection
The CSD requesting the Penalty Modification with Request Type "REMO", "REIN" or "SWIC" is not the failing CSD of the Penalty
The CSD requesting the Penalty Modification with Request Type "RALO" is not the same as the CSD of the LMFP (i.e. it is not the CSD of the Instructing Party of the underlying Settlement Instruction).
The Penalty Modification with Request Type "REMO", "REIN" or "SWIC" does not refer to the Individual Penalty ID of the Failing Party.
A Common Penalty ID is provided in the Penalty Modification with Request Type "REMO", "REIN" or "SWIC".
The Individual Penalty ID of the Penalty provided in the Penalty Modification with Request Type "REMO", "REIN" or "SWIC" is missing.
No Penalty exists with the Individual Penalty ID provided in the Penalty Modification with Request Type "REMO", "REIN" or "SWIC".
An Individual Penalty ID is provided in the Penalty Modification with Request Type "RALO"
The Common Penalty ID of the Penalty provided in the Penalty Modification with Request Type "RALO" is missing
No Penalty exists with the Common Penalty ID provided in the Penalty Modification with Request Type "RALO"
The Penalty provided in the Penalty Modification is not on its Appeal Period.
The Penalty provided in the Penalty Modification with Request Type "REMO", "SWIC" or "RALO" does not have Status "Active" ("ACTV").
The Penalty provided in a Penalty Modification with Request Type "REIN" does not have Status "Removed" ("REMO").
The Penalty provided in a Penalty Modification with Request Type "REIN" is removed because of a previous Re-allocation.
A Penalty Modification with Request Type "REIN", "RALO" and "SWIC" has a Reason for removal.
A Penalty Modification with Request Type "REMO" does not have Reason for removal ("INSO", "SEMP", "SESU", "SUSP", "TECH", "OTHR") or the Reason has an unexpected value.
The 'Description text' is missing in a Penalty Modification with Request Type either "SWIC" or "REMO" and Reason "OTHR".
A Penalty Modification with Request Type "REIN", "RALO" or "REMO" with Reason different from "OTHR" has the 'Description text'.

All rights reserved.

Conditions for rejection
The Type of the Penalty provided in a Penalty Modification with Request Type "RALO" is not "LMFP".
The Penalty provided in a Penalty Modification with Request Type "RALO" has been re-allocated before.
The underlying Settlement Instruction provided in a Penalty Modification with Request Type "RALO" is not an instruction sent as already matched to T2S.
In a Penalty Modification with Request Type "RALO", the New Failing Party is neither the delivering nor the receiving party of the underlying Settlement Instruction sent already matched.
In a Penalty Modification with Request Type "RALO" the New Failing Party BIC corresponds to the delivering party of the underlying Settlement Instruction but the New Non-Failing Party BIC does not correspond to the receiving party, and vice versa.
In a Penalty Modification with Request Type "RALO" the T2S reference of the failed Settlement Instruction is neither the delivering nor the receiving Settlement Instruction created in T2S.
In a Penalty Modification with Request Type "RALO" the New Failing Party BIC is equal to the New Non-Failing Party BIC but the T2S Reference of the failed Settlement Instruction is missing.
In a Penalty Modification with Request Type "RALO" the New Failing Party BIC and the New Non-Failing Party BIC are different but the T2S reference of the failed Settlement Instruction is informed.
In a Penalty Modification with Request Type "RALO" the New failing Party BIC is missing.
In a Penalty Modification with Request Type "RALO" the New non-failing Party BIC is missing.
In a Penalty Modification with Request Type different from "RALO" the New failing Party BIC is informed.
In a Penalty Modification with Request Type different from "RALO" the New non failing Party BIC is informed.
In a Penalty Modification with Request Type different from "RALO" the T2S reference of the failed Settlement Instruction is informed.

In case any of the above described conditions is met, the Penalty Modification is rejected (its Status is set to "Rejected") and the relevant Request Error associated to the condition met is stored.

In case none of the above described conditions is met (and the Penalty Modification is not rejected), then the Penalty Modification is executed (its Status is set to "Executed"), and the Penalty is modified as described for each Type of Request in the following sub-functions:

- Perform Removal: if the Request Type is "REMO";
- Perform Re-inclusion: if the Request Type is "REIN";
- Perform Switch: if the Request Type is "SWIC";
- Perform Re-allocation: if the Request Type is "RALO".

Finally, the necessary data informing about the rejection or the execution of the Penalty Modification is stored in the Enriched File used to respond to the related Inbound Bulk File. The Enriched File will be sent to the Sender of the Inbound Bulk File once the Enriched File has the information about the rejection or the execution of all the Penalty Modifications included in the Inbound Bulk File.

Validation of Inbound Penalty Modification Messages and Incoming Instruction Approval Requests

All rights reserved.

Reference Id	**LCMM.PEM.MOD.3.1**

The sub-function receives from the Interface domain:

- Inbound Penalty Modification Messages;
- Incoming Instruction Approval Requests.

Upon reception of an Inbound Penalty Modification Message, the sub-function stores it and checks that it complies with the business requirements needed to execute the Penalty Modification **{T2S.22.310} {T2S.22.320} {T2S.22.350} {T2S.22.380} {T2S.22.410} {T2S.22.420}.**

When the sub-function receives an Incoming Instruction Approval Request (approval or revocation) from the Second T2S system user, this sub-function retrieves the referenced Penalty Modification (with Approval Status value "Awaiting for Approval") and:

- In case it is a revocation request, it sets the Approval Status to "Revoked";
- In case it is an approval request, it retrieves the Inbound Penalty Modification Message associated to the Penalty Modification and checks again that it complies with the business requirements needed to execute the Penalty Modification **{T2S.22.310} {T2S.22.320} {T2S.22.350} {T2S.22.380} {T2S.22.410} {T2S.22.420}.**

This sub-function rejects the Inbound Penalty Modification Message when one of the conditions in the table below is fulfilled:

Conditions for rejection
The CSD requesting the Penalty Modification with Request Type "REMO", "REIN" or "SWIC" is not the failing CSD of the Penalty
The CSD requesting the Penalty Modification with Request Type "RALO" is not the same as the CSD of the LMFP (i.e. it is not the CSD of the Instructing Party of the underlying Settlement Instruction).
The Penalty Modification with Request Type "REMO", "REIN" or "SWIC" does not refer to the Individual Penalty ID of the Failing Party.
A Common Penalty ID is provided in the Penalty Modification with Request Type "REMO", "REIN" or "SWIC".
The Individual Penalty ID of the Penalty provided in the Penalty Modification with Request Type "REMO", "REIN" or "SWIC" is missing.
No Penalty exists with the Individual Penalty ID provided in the Penalty Modification with Request Type "REMO", "REIN" or "SWIC".
An Individual Penalty ID is provided in the Penalty Modification with Request Type "RALO"
The Common Penalty ID of the Penalty provided in the Penalty Modification with Request Type "RALO" is missing
No Penalty exists with the Common Penalty ID provided in the Penalty Modification with Request Type "RALO"
The Penalty provided in the Penalty Modification is not on its Appeal Period.
The Penalty provided in the Penalty Modification with Request Type "REMO", "SWIC" or "RALO" does not have Status "Active" ("ACTV").
The Penalty provided in a Penalty Modification with Request Type "REIN" does not have Status "Removed" ("REMO").

All rights reserved.

Conditions for rejection
The Penalty provided in a Penalty Modification with Request Type "REIN" is removed because of a previous Re-allocation.
A Penalty Modification with Request Type "REIN", "RALO" and "SWIC" has a Reason for removal.
A Penalty Modification with Request Type "REMO" does not have Reason for removal ("INSO", "SEMP", "SESU", "SUSP", "TECH", "OTHR") or the Reason has an unexpected value.
The 'Description text' is missing in a Penalty Modification with Request Type either "SWIC" or "REMO" and Reason "OTHR".
A Penalty Modification with Request Type "REIN", "RALO" or "REMO" with Reason different from "OTHR" has the 'Description text'.
The Type of the Penalty provided in a Penalty Modification with Request Type "RALO" is not "LMFP".
The Penalty provided in a Penalty Modification with Request Type "RALO" has been re-allocated before.
The underlying Settlement Instruction provided in a Penalty Modification with Request Type "RALO" is not an instruction sent as already matched to T2S.
In a Penalty Modification with Request Type "RALO", the New Failing Party is neither the delivering nor the receiving party of the underlying Settlement Instruction sent already matched.
In a Penalty Modification with Request Type "RALO" the New Failing Party BIC corresponds to the delivering party of the underlying Settlement Instruction but the New Non-Failing Party BIC does not correspond to the receiving party, and vice versa.
In a Penalty Modification with Request Type "RALO" the T2S reference of the failed Settlement Instruction is neither the delivering nor the receiving Settlement Instruction created in T2S.
In a Penalty Modification with Request Type "RALO" the New Failing Party BIC is equal to the New Non-Failing Party BIC but the T2S Reference of the failed Settlement Instruction is missing.
In a Penalty Modification with Request Type "RALO" the New Failing Party BIC and the New Non-Failing Party BIC are different but the T2S reference of the failed Settlement Instruction is informed.
In a Penalty Modification with Request Type "RALO" the New failing Party BIC is missing.
In a Penalty Modification with Request Type "RALO" the New non-failing Party BIC is missing.
In a Penalty Modification with Request Type different from "RALO" the New failing Party BIC is informed.
In a Penalty Modification with Request Type different from "RALO" the New non failing Party BIC is informed.
In a Penalty Modification with Request Type different from "RALO" the T2S reference of the failed Settlement Instruction is informed.

In case any of the above described conditions is met, then:

- When it is a validation of an Inbound Penalty Modification Message, the relevant Request Error associated to the condition met is stored and the information about the rejection of the Outbound Penalty Modification Message is forwarded to the Interface Domain.

All rights reserved.

- When it is a revalidation due to an Incoming Instruction Approval Request, the Approval Status of the related Penalty Modification is set to "Rejected", the relevant Request Error associated to the condition met is stored and the information about the rejection of the Approval Request Response is forwarded to the Interface Domain.

In case none of the above described conditions is met, then:

- When it is a validation of an Inbound Penalty Modification Message, then:
  - If it is submitted under Four-Eyes Principle, the Penalty Modification is created with Status "Validated" and Approval Status "Awaiting for Approval" and forwards this information to the Interface Domain.
  - If it is nor submitted under Four-Eyes Principle, the Penalty Modification is created and executed (its Status is set to "Executed"), informs Interface Domain about the execution, and the Penalty is modified accordingly.
- When it is a revalidation due to an Incoming Instruction Approval Request, the sub- function sets the Approval Status of the related Penalty Modification as "Approved" and informs Interface Domain, executes the Penalty Modification (its Status is set to "Executed") and modifies the Penalty:

The modification of the Penalty is described for each Type of Request in the following sub-functions:

- Perform Removal: if the Request Type is "REMO";
- Perform Re-inclusion: if the Request Type is "REIN";
- Perform Switch: if the Request Type is "SWIC";
- Perform Re-allocation: if the Request Type is "RALO".

Perform Removal

Reference Id	**LCMM.PEM.MOD.4.1**

The removal of a Penalty consists of performing the following actions over it **{T2S.22.330}**:

- Set the Status to "Removed" ("REMO");
- Set the Reason with the value of the Removal Reason Code provided in the Penalty Modification, i.e. one of the following:
  - "'INSO": Penalty was removed because insolvency proceedings are opened against the failing participant;
  - "SESU": Penalty was removed because of ISIN suspension from settlement due to a reconciliation issue under Article 65 (2) and (6) of the RTS on CSD Requirements;
  - "SUSP": Penalty was removed because of ISIN suspension from trading;
  - "SEMP": Penalty was removed because the Settlement Instructions involved cash settlement outside the securities settlement system operated by the CSD if, on the respective day, the relevant payment system is closed for settlement;
  - "TECH": Penalty was removed because there were technical impossibilities at the CSD level that prevent settlement, such as: a failure of the infrastructure components, a cyber-attack, network problems; or

All rights reserved.

– "OTHR": Removed Penalty due to a non-standard reason by the CSD.

In case the Removal Reason Code is "OTHR", store the Description Text provided in the Penalty Modification;

- Set the Amount of the Penalty to zero;
- Update the Update Timestamp;
- Set the To be recalculated flag to "False";
- Flags the Penalty for reporting due to its modification but without providing information about its Sub-amounts (Calculation Data) that will not be visible any more.

Once a Penalty is removed, it can only be subject to a "Re-inclusion" modification (provided its Appeal Period has not finished yet); any other maintenance request will be rejected.

Perform Re-inclusion

Reference Id	**LCMM.PEM.MOD.5.1**

The re-inclusion of a Penalty consists of performing the following actions over it **{T2S.22.360}**:

- Set the Status to "Active" ("ACTV");
- Set the Reason to "Updated" ("UPDT");
- Set the Amount of the Penalty as the existing one before the Penalty was removed;
- Update the Update Timestamp;
- Set the To be recalculated flag to "True"; and
- Select the Penalty for reporting due to its modification.

A re-included Penalty (provided its Appeal Period has not finished yet) can be subject to any further type of modification request.

Perform Switch

Reference Id	**LCMM.PEM.MOD.6.1**

The switch of a Penalty consists of performing the following actions over it **{T2S.22.430}**:

- Set the Reason to "Switched" ("SWIC");
- Store the Description Text provided in the Penalty Modification;
- Update the Debit/Credit Indicator of both Applicable Parties of the Penalty as follows: if the value was "Debit" it is updated to "Credit" and vice versa. It is worth mentioning that, as the only attribute updated of the Applicable Parties is the Debit/Credit Indicator, after the first switch is performed, the Individual Penalty ID for the (new) Failing Party will start by "N", and the Individual Penalty ID for the (new) Non-Failing Party will start by "F".

Original penalty		Penalty switched
Failing Party ("Debit")	-	Non-Failing Party ("Credit")
Non-Failing Party ("Credit")	⇨	Failing Party ("Debit")

All rights reserved.

- Updates the Calculation Method when necessary, i.e.:
  - If the Penalty Calculation Method is "MIXE", it is updated to "SECU".
  - If the Penalty Calculation Method is "SECU", the Payment Type of the underlying SI is "APTM" and either the Type of Penalty is "SEFP", or it is "LMFP" with Underlying SI already matched flag "True", it is updated to "MIXE";
- Update the Update Timestamp;
- Set the To be recalculated flag to "True"; and
- Select the Penalty for reporting due to its modification.

A switched Penalty (provided its Appeal Period has not finished yet) can be subject to any further type of modification request.

Perform Re-allocation

Reference Id	**LCMM.PEM.MOD.7.1**

The re-allocation of a Penalty consists of performing a removal of the existing Penalty and its substitution by creating a new re-allocated Penalty (for the New Failing and New Non-Failing parties), as described below.

The existing Penalty is removed **{T2S.22.390**} performing the following actions over it:

- Set the Status to "Removed" ("REMO");
- Set the Reason to "Re-allocated" ("RALO");
- Set the Amount of the Penalty to zero;
- Store the Common Penalty ID of the new Penalty re-allocated (created as described below) in the New Penalty After Reallocation;
- Update the Update Timestamp;
- Set the To be recalculated flag to "False"; and
- Select the Penalty for reporting due to its modification.

The new re-allocated Penalty is created **{T2S.22.370}** with:

- A new Common Penalty ID with 15 characters;
- Status "Active" ("ACTV");
- Reason "Re-allocated" (RALO);
- Detection Date, Number of Business Days, Currency and Modifiable Flag of the existing Penalty (removed as described above);
- The applicable Calculation Method: If the new failing instruction is the receiving leg instead of the delivering one ("RECE" instead of "DELI") and the Payment Type of the underlying Settlement Instruction is "Against Payment" ("APMT") it is set to "MIXE" (whereas the existing Penalty removed was "SECU"); otherwise, it is the same as the one of the existing Penalty removed;
- To be recalculated flag "True";

All rights reserved.

- Penalty reallocated from equal to the Common Penalty ID of the existing Penalty (removed as described above);
- Two new Applicable Party instances associated to the new Penalty are generated (i.e.: one for the new Failing Party and another one for the new Non-failing Party as provided in the Penalty Modification) and, for each one of them, retrieves and stores its necessary attributes, i.e.:
  - Individual Penalty ID: Composed of 16 characters (i.e.: F+Common Penalty ID for the new Failing Party; N+Common Penalty ID for the new Non-Failing Party);
  - The CSD of the Applicable Party (of the new Failing Party or of the new Non-Failing Party [120]);
  - Credit / Debit indicator: "Debit" for the new Failing Party or "Credit" for the new Non-Failing Party;
  - T2S Reference of the related Settlement Instruction (of the Failing Party or of the Non-Failing Party).
- Select the new reallocated Penalty for reporting due to its modification.

The Penalty removed due to a re-allocation cannot be subject to any Penalty Modification request. However, the new re-allocated Penalty (provided its Appeal Period has not finished yet) can be subject to any penalty modification request, except to a new "Re-allocation".

4 - Selection for Recalculation

Introduction

Reference Id	**LCMM.PEM.SFR.1.1**

The Selection for Recalculation function is triggered every T2S settlement day by the Recalculation of Penalties (RECA) Event, received from the Scheduling Module **{T2S.03.390} {T2S.03.400}**.

Every business day the CRDM Domain provides the Penalty Mechanism Module with the Flow containing the updates (i.e.: insertions, deletions and modifications) occurred since the previous business day deadline (i.e.: since the last file provided with data updates) on the **static reference** data used to calculate Penalties. The Flow includes Security Penalty Data updates, Cash Discount Penalty Rate updates, Euro Foreign Exchange Rate updates and List of SME Growth Market updates.

This function processes the updates informed, selects the Penalties affected (directly or via their Sub-amounts) and sends them to the Calculation function **{T2S.22.470}**.

The function is divided in the following sub-functions:

- Updates on Security Subject to Penalties and related data;
- Updates on Cash Discount Penalty Rates;
- Updates on Euro Foreign Exchange Rates;
- Updates on the List of SME Growth Markets;
- Recalculation Management.

Updates on Security Subject to Penalties and related data

Reference Id	**LCMM.PEM.SFR.2.1**

120 In case of an External CSD settlement scenario, the CSD of the Applicable Party is the parent CSD of the External-CSD.

All rights reserved.

For each Security Subject to Penalties and related data update received in the Flow from CRDM Domain, this sub-function processes it and searches for any affected Penalty.

A Security Subject to Penalties and related data update contains the following information:

Date	Relevant Business Day [121] to which the updates refer
Type of update	Insertion or Modification [122]
Security	Relevant Security subject to penalties to which all the below attributes refer
Financial Instrument Type	Possible values: "SHRS", "SOVR", "DEBT", "SECU", "ETFS", "UCIT", "MMKT", "EMAIL", "OTHR".
Liquidity	Only provided if Financial Instrument Type is "SHRS"
Daily Flat Penalty Rate (for non-SMEs)	Applicable Security Penalty Rate in case the SME Growth Market flag is "FALSE".
Daily Flat Penalty Rate (for SMEs)	Applicable Security Penalty Rate in case the SME Growth Market flag is "TRUE".
Daily Price (Amount)	Provided if the Security has Settlement Type "UNIT"
Currency of the Price	Only provided if Price Amount is informed
Daily Price (Coefficient)	Provided if the Security has Settlement Type "FAMT".
Currency of the Security	Only provided if Price Coefficient is informed

Independently from the type of update received, in order to select a Penalty, the function checks that it has:

- The Modifiable Flag "True";
- The Status different from "REMO"; and
- To be recalculated "False".

Depending on the possible type of update received, the sub-function performs as follows:

- Insertion of a new Security Subject to Penalties (and consequent provision of related data):
  - The sub-function sets to "True" the To be recalculated of those Penalties that have:

    a Sub-amount with Security and Date as the ones informed in the update.
  - Additionally, depending on the status of the Penalty:

    (i) If the Penalty has Status Not Computed "NCOM", the Status is set to Active "ACTV" and the Reason is set to New Penalty "NEWP";

    (ii) If the Penalty has Status Active "ACTV", the Reason of the Penalty is set to Updated "UPDT".
- Modification of the related data of a Security Subject to Penalties (Daily Price, Financial Instrument Type, Liquidity of the Security and Security Penalty Rate):
  - The sub-function selects for recalculation, and sets their Reason to Updated "UPDT", the Penalties that have:

121 It must be a day before the previous business day

122 There is no deletion because a Security cannot be removed from the Security Subject to Penalties in the past

All rights reserved.

(i) A Calculation Method different from "CASH";

(ii) A Sub-amount with Security and Date as the ones informed in the update [123].

Once all the updates Security Subject to Penalties and related data have been processed, the Updates on Cash discount penalty rates sub-function is triggered.

Updates on Cash discount penalty rates

Reference Id	**LCMM.PEM.SFR.3.1**

For each Cash Discount Penalty Rate update received in the Flow from CRDM Domain, this sub-function processes it and searches for any affected Penalty.

A Cash Discount Penalty Rate update contains the following information:

Date	Relevant Business Day [124] to which the updates refer
Currency	Relevant Currency to which all the below attributes refer
Type of Update	Insertion, Deletion or Modification
Daily Flat Penalty Rate	Applicable Cash Discount Penalty Rate

The sub-function performs in the same way for any type of update received i.e.: insertion, modification or deletion of the Cash Discount Penalty Rate used to calculate a Penalty.

The sub-function selects for recalculation, and sets their Reason to Updated "UPDT", the Penalties that have:

- The Modifiable Flag "True";
- The Status different from "REMO";
- To be recalculated "False";
- A Calculation Method different from "SECU"; and
- A Sub-amount with Currency and Date equal to the ones indicated in the update, as long as the Sub-amount flag Security Subject to Penalties is "True".

Once all the updates on Cash Discount Penalty Rates have been processed, the Updates on Euro Foreign Exchange Rates sub-function is triggered.

Once the process is finished, the sub-function informs the Recalculation Management sub-function.

Updates on Euro Foreign Exchange Rates

Reference Id	**LCMM.PEM.SFR.4.1**

For each Euro Foreign Exchange Rate update received in the Flow from CRDM Domain, this sub-function processes it and searches for any affected Penalty.

A Euro Foreign Exchange Rate update flow contains the following information:

123 Given that the update is a modification of the related data of a Security Subject to Penalties, the Sub-amounts affected would have the Security Subject to Penalties flag set to "True"

124 It must be a day before the previous business day

All rights reserved.

Date	Relevant Business Day [125] to which the updates refer
Currency	Relevant Currency to which all the below attributes refer
Type of Update	Insertion, Deletion or Modification
Daily Exchange Rate	Applicable Euro Foreign Exchange Rate

The sub-function performs in the same way for any type of update received i.e.: insertion, modification or deletion of the Euro Foreign Exchange Rates used to calculate a Penalty:

The sub-function selects for recalculation, and sets their Reason to Updated "UPDT", the Penalties that have:

- The Modifiable Flag "True";
- The Status different from "REMO"
- To be recalculated "False";
- A Calculation Method different from "CASH";
- Conversion flag "True";
- A Sub-amount with Currency and Date equal to the ones of the update, as long as the Sub-amount flag Security Subject to Penalties is "True".

Once all the updates on Euro Foreign Exchange Rates have been processed, the Updates on List of SME Growth Markets sub-function is triggered.

Updates on List of SME Growth Markets

Reference Id	**LCMM.PEM.SFR.5.1**

For each update in the List of SME Growth Markets received in the Flow from CRDM Domain, this sub-function processes it and searches for any affected Penalty.

A List of SME Growth Markets update contains the following information:

MIC Code	Market Identifier Code
Type of Update	Insertion, Deletion or Modification [126]

Upon any type of update received, the function selects all the Penalties that have:

- The Modifiable Flag "True";
- The Status different from "REMO";
- To be recalculated "False";
- Calculation Method "SECU" or "BOTH"; and
- A Sub-amount with Common MIC equal to the one indicated in the update, as long as the Sub-amount flag Security Subject to Penalties is "True".

125 It must be a day before the previous business day

126 In this case a modification is treated as a deletion and an insertion

All rights reserved.

Depending on the possible type of update received (insertion and deletion), the sub-function performs as follows:

- Insertion of a new SME Growth Market:
  - The sub-function sets to "True" the flag SME Growth Market of the affected Sub-amounts.
  - Additionally, if the Financial Instrument Type of the Security is different from "SOVR" the flag To be recalculated and the Reason of the Penalty are set to "True" and to "UPDT" respectively.
- Deletion of SME Growth Markets:
  - The sub-function sets to "False" the flag SME Growth Market of the affected Sub-amounts.
  - Additionally, if the Financial Instrument Type of the Security is different from "SOVR" the flag To be recalculated and the Reason of the Penalty are set to "True" and to "UPDT" respectively.

Once all the updates on the List of SME Growth Markets have been processed, the Recalculation Management sub-function is triggered.

Recalculation Management

Reference Id	**LCMM.PEM.SFR.6.1**

After the previous sub-functions have finished their processing, this sub-function sends:

- The Penalties that are either flagged as To be recalculated "True" and that do not have Sub-amounts (i.e. it is a new reallocated Penalty or a Penalty with Status "ACTV" and Reason "NEWP") or Penalties with Detection Date before the previous business day and Status "PEND" to the sub-function Penalty Preparation in the Calculation function;
- The Penalties and their Sub-amount(s), when the Penalty is flagged as To be recalculated "True" to the sub-function Sub-amount Preparation in the Calculation function; and
- The Penalties and their Sub-amount(s), when the Penalty is flagged as To be recalculated "False" but has been selected for recalculation by the previous sub-functions, to the sub-function Sub-amount Preparation of the Calculation function.

Once the Calculation function has processed all the Penalties, the sub-function the End of Process Event (Recalculation of Penalties) to the Scheduling Module.

5 - Appeal Period End Process

Reference Id	**LCMM.PEM.APE.1.1.**

The Appeal Period End Process function is triggered every T2S settlement day by the End of Appeal Period Process (EAPP) Event, but it only performs the update of Penalties when the Appeal Period of the previous month finishes.

This function is responsible for updating the Penalties detected on the previous month that have reached the last day of their corresponding Appeal Period, by setting their Modifiable flag to "False", which means that:

- These Penalties and their Sub-amounts will not be selected for recalculation **{T2S.22.470}**; and
- No further modification requests will be accepted on these Penalties **{T2S.22.440}**.

All rights reserved.

Upon reception of the End of Appeal Period Process (EAPP) Event, this function compares the value of the Appeal Period End Day stored in Reference Data (13th business day of every month) with the value of the system parameter, Current Nth business day of the month, and if they are the same, the function selects all the Penalties whose Detection Date is in the previous calendar month and updates their Modifiable flag from "True" to "False".

Finally this function sends the End of Process Event (End of Appeal Period Process - EAPP) to the Scheduling Module.

6 - Preparation for Reporting

Introduction

Reference Id	**LCMM.PEM.PPR.1.1.**

This function is triggered every T2S settlement day by the Report Preparation for New Penalties (RPDA) Event, the Report Preparation for Modified Penalties (RPMD) Event and the Report Preparation of Monthly Aggregated Amounts (SRPM) Event **{T2S.03.430}.**

This function is in charge of preparing the information to be reported in the Daily Penalty List, the List of Modified Penalties and the Monthly Aggregated Amounts.

REPORT NAME	REPORT CONTENT	URD REFERENCES
Daily Penalty List	It reports per Currency, T2S Party and Counterparty all the new Penalties computed for the previous T2S settlement day (i.e. with Detection Date equal to "current Business day -1")	**{T2S.13.320}** **{T2S.13.340}**
List of Modified Penalties	It reports per Currency, T2S Party and Counterparty, the updates performed, since the previous report, in Penalties with Detection Date before "current Business day -1"	**{T2S.13.390}** **{T2S.13.410}**
Monthly Aggregated Amounts of Penalties	It reports per Currency and T2S Party and Counterparty, the aggregated bilateral net amount of Penalties computed for the T2S settlement day of the previous month (i.e. with Detection Date within the previous month).	**{T2S.13.460}** **{T2S.13.470}**

Once the Preparation for Reporting function has prepared the information, the relevant report can be generated upon reception of the corresponding Event. For the creation of these reports, T2S relies on CRDM Domain. The information specifying which event triggers which report for the data of a party, is stored in the Report Configuration in Reference Data. T2S will create only reports pre-defined in the Report Configuration.

The Preparation for Reporting function is divided in the following sub-functions:

- Preparation for Daily Penalty List

All rights reserved.

- Preparation for List of Modified Penalties
- Preparation for Monthly Aggregated Amounts Penalties

Preparation for Daily Penalty List Report

Reference Id	**LCMM.PEM.PPR.2.1.**

Upon reception of the Report Preparation for New Penalties (RPDA) Event, this sub-function retrieves and prepares the necessary data for the possible generation of any pre-defined Daily Penalty List **{T2S.13.340} {T2S.13.350}**, i.e.:

- The sub-function prepares ad-hoc tables with the necessary information of the Penalties with Detection Date equal to "current Business day -1" and Status "ACTV", their Applicable Parties, Sub-amounts and underlying Settlement Instruction, as well as the Daily Nets for a party vs counterpart.
- For this purpose, the sub-function calculates the Daily Nets for a party vs counterpart as follows:
  - From all the Applicable Parties related to Penalties with Detection Date equal to "current Business day -1" and Status "ACTV", the sub-function creates a Daily Net for a party vs counterpart for each pair of "Applicable Party and Counterpart".
  - Then, the sub-function adds or subtracts the Amount of the Penalty of each Applicable Party with the related pair of "Party and Counterparty", i.e. if the Credit/Debit Indicator of the Applicable Party is "Debit", the Amount of the Penalty is subtracted to the Net Amount of the Daily Aggregated Net Amount; whereas if it is "Credit", the Amount of the Penalty is added to the Net Amount of the Daily Aggregated Net Amount.
  - If the sign of the resulting Net Amount of the Daily Aggregated Net Amount is positive, the Credit/ Debit Indicator attribute is set as "Credit"; whereas if it is negative, the Credit/Debit Indicator attribute is set as "Debit".

Finally, this sub-function sends the End of Process Event (Report Preparation for New Penalties) to the Scheduling Module.

Preparation for List of Modified Penalties

Reference Id	**LCMM.PEM.PPR.3.1.**

Upon reception of the Report Preparation for Modified Penalties (RPMD) Event, this sub-function retrieves and prepares the necessary data for the possible generation of any pre-defined List of Modified Penalties {**T2S.13.410} {T2S.13.420}**, i.e.:

- The sub-function prepares ad-hoc tables with the necessary information of the updated Penalties (flagged for reporting due to its recalculation or modification, hence they must have Detection Date before "current Business day -1" and Status "ACTV" or "REMO"), their Applicable Parties , Sub-amounts (only for Penalties with Status "ACTV") and underlying Settlement Instruction (only for Penalties with Reason "NEWP"), as well as the updated Daily Nets for a party vs counterpart.
- For this purpose, the sub-function calculates the updated Daily Net for a party vs counterpart as follows:
  - Selects all the different Detection Dates of the updated Penalties, and for each one:

All rights reserved.

(i) From all the Applicable Parties related to Penalties with the relevant Detection Date and Status "ACTV", the sub-function calculates the new Daily Net for a party vs counterpart for each pair of "Party and Counterpart".

(ii) The sub-function adds or subtracts the Amount of the Penalty of each Applicable Party with the related pair of "Party and Counterpart", i.e. if the Credit/Debit Indicator of the Applicable Party is "Debit", the Amount of the Penalty is subtracted to the Net Amount of the Daily Aggregated Net Amount; whereas if it is "Credit", the Amount of the Penalty is added to the Net Amount of the Daily Aggregated Net Amount.

(iii) If the sign of the resulting Net Amount of the Daily Aggregated Net Amount is positive, the Credit/Debit Indicator attribute is set as "Credit"; whereas if it is negative, the Credit/Debit Indicator attribute is set as "Debit".

Finally, this sub-function sends the End of Process Event (Report Preparation for Modified Penalties) to the Scheduling Module.

Preparation for Monthly Aggregated Amounts of Penalties Report

Reference Id	**LCMM.PEM.PPR.4.1.**

Upon reception of the Report Preparation of Monthly Aggregated Amounts (SRPM) Event, this function compares the value of the Appeal Period End Day stored in Reference Data (13th business day of every month) with the value of the system parameter, Current Nth business day of the month, and if they are the same, this sub-function retrieves and prepares the necessary data for the possible generation of any pre-defined Monthly Aggregated Amounts of Penalties on the relevant business day defined by the Monthly Reporting Day stored in Reference Data (14th business day of the month) **{T2S.13.460} {T2S.13.470}**, i.e.:.

- The sub-function prepares ad-hoc tables with the necessary information of the Penalties with Detection Date within the previous month and Status "ACTV" and their Applicable Parties, as well as the Monthly Nets for a party vs counterpart.
- For this purpose, the sub-function calculates the Monthly Nets for a party vs counterpart as follows:
  - Selects all the different Detection Dates of the previous month, and for each one:

    (i) From all the Applicable Parties related to Penalties with the relevant Detection Date and Status "ACTV", the sub-function creates a Monthly Net for a party vs counterpart for each pair of "Party and Counterparty".

    (ii) Then, the sub-function adds or subtracts the Amount of the Penalty of each Applicable Party with the related pair of "Party and Counterparty", i.e. if the Credit/Debit Indicator of the Applicable Party is "Debit", the Amount of the Penalty is subtracted to the Net Amount of the Monthly Aggregated Net Amount; whereas if it is "Credit", the Amount of the Penalty is added to the Net Amount of the Monthly Aggregated Net Amount.

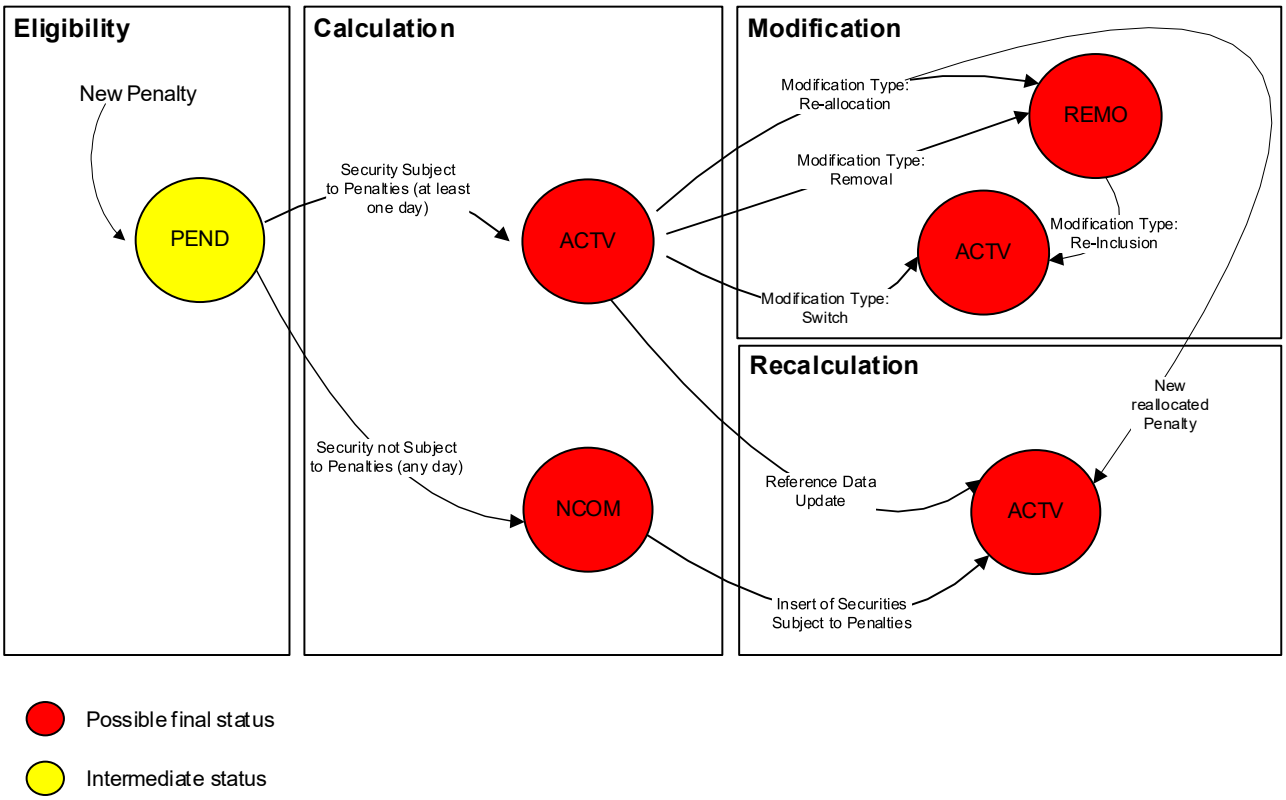
    (iii) If the sign of the resulting Net Amount of the Monthly Aggregated Net Amount is positive, the Credit/Debit Indicator attribute is set as "Credit"; whereas if it is negative, the Credit/Debit Indicator attribute is set as "Debit".

All rights reserved.

– Additionally, for all the Monthly Aggregated Net Amount, the sub-function verifies that their Net Amount is the same as the sum of the Net Amounts of the Daily Aggregated Net Amount for Business days within the related month and the same Party and Counterparty.

Finally, this sub-function sends the End of Process Event (Report Preparation of Monthly Aggregated Amounts) to the Scheduling Module

3.4.7.4 Description of Penalties status transactions



The Penalty Eligibility function creates a Penalty with Status Pending ("PEND") [127]. This is an intermediate status until the Calculation function retrieves and processes the Penalty.

The Calculation function checks whether the Security of the Underlying Settlement Instruction is subject to pen-alties or not on each of the applicable business days of the Penalty:

- If the Security is subject to penalties at least in one of the applicable business days, the Penalty Status is updated to Active ("ACTV") and without any Reason;
- If the Security is not subject to penalties in any of the applicable business days, the Penalty Status is up-dated to Not Computed ("NCOM") and without any Reason.

During the Appeal Period of the Penalty, it can be subject to modifications and recalculations.

127 When the Penalty Eligibility function is unable to determine whether a Settlement Instruction is eligible for a penalty or not, because its reason for failing is not included in the Failing Reasons Dictionary, the same Settlement Instruction is further analysed at a later date, once its reason for failing is included in the Failing Reasons Dictionary. If, during this analysis, the Penalty Eligibility function determines that the Settlement Instruction is eligible for a penalty, it creates a Penalty directly with Status Active ("ACTV") and Reason New Penalty ("NEWP"). The Penalty is then retrieved and processed by the Penalty Recalculation function, which checks whether the Security of the Underlying Settlement Instruction is subject to penalties or not on each of the applicable business days of the Penalty

All rights reserved.

The Penalty Modification function can update the Status of the Penalty depending on the type of the Penalty Modification executed:

- In case of a removal, the Penalty Status is updated to Removed ("REMO") with the corresponding Reason ("INSO", "SEMP", "SESU", "SUSP", "TECH" or "OTHR") indicated in the Penalty Modification;
- In case of a re-inclusion (which can only be performed on Penalties with Status "REMO" and a Reason dif-ferent from Re-allocated "RALO"), the Penalty Status is updated to Active ("ACTV") with Reason Up-dated ("UPDT");
- In case of a switch, the Penalty Status remains as Active ("ACTV") but the Reason is changed to Switched ("SWIC"); and
- In case of a re-allocation, the Status of the Penalty is updated to Removed ("REMO") with Reason Re-allocated ("RALO"), and a new re-allocated Penalty is generated with Status Active ("ACTV") and Reason Re-allocated ("RALO").

The Selection for Recalculation function can update the Status of the Penalty depending on the type of the Reference Data update received:

- In case an insertion of Securities subject to penalties impacts a Penalty with Status Not Computed ("NCOM"), the Penalty Status is updated to Active ("ACTV") with Reason New Penalty ("NEWP"); and
- In any other case, the Penalty Status remains as Active ("ACTV") but the Reason is changed to Up-dated ("UPDT") [128].

The statuses depicted in red in the above diagram are possible final statuses. They become final upon the end of their appeal period, i.e.: when the Appeal Process End function updates the Penalty Modifiable Flag to "False", the Status that the Penalty has at this point in time becomes its final Status.

3.4.7.5 Description of the Input / Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Event (Start Cash Penalty Eligibility)	IN	This event is sent in order to trigger the start of the Penalty Eligibility.	OPRS: Scheduling	
End of Process Event (Start Cash Penalty Eligibility)	OUT	This end of process event is sent when the Penalty Eligibility is completed.		OPRS: Scheduling

128 In very concrete scenarios, Penalties can also get the Status Active ("ACTV") and Reason New Penalty ("NEWP") when they are created and calculated in a contingency scenario, i.e. when:

- The Reason for failing of the Penalty underlying Settlement Instruction was not included in the Failing Reasons Dictionary. Once the Reason for failing is included in the dictionary on a subsequent business day, the corresponding Penalty Eligibility process can identify if a penalty has to be calculated, and if so, it is calculated by the Recalculation function
- The Calculation function is skipped and run at a later business day. Outstanding penalties are calculated by the Recalculation function

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Event (Calculation of Penalties)	IN	This event is sent in order to trigger the calculation of new Penalties.	OPRS: Scheduling	
End of Process Event (Calculation of Penalties Detected)	OUT	This end of process event is sent when the calculation of new Penalties is completed.		OPRS: Scheduling
Inbound Bulk File	IN	Flat file received for bulk loading Penalty Modifications	ICM: A2A Mode	
Inbound Penalty Modification Message	IN	Flow received for communicating Penalty Modifications	INTF: Inbound processing	
Incoming Instruction Approval Request	IN	Flow received to update the Approval Status of Penalty Modification requests requiring 4 eyes processing.	INTF: Inbound processing	
Penalty Modification Response Flat File	OUT	Flat sent with the responses to the Penalty Modifications requested via flat file.		ICM: A2A Mode
Penalty Modification Response U2A	OUT	Flow sent with the response to a Penalty Modification requested via U2A.		INTF: Outbound processing
Event (Recalculation of penalties)	IN	This event is sent in order to trigger the Selection for Recalculation.	OPRS: Scheduling	
Reference data Delta File	IN	Flow with the Reference Data updates over Security Penalty Data, Cash Discount Penalty Rates, Euro	CRDM: Reference Data	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
		Foreign Exchange Rates and in the List of SME Growth Markets		
End of Process Event (Recalculation of penalties)	OUT	This end of process event is sent when the Selection for Recalculation is completed.		OPRS: Scheduling
Event (End of Appeal Period)	IN	This event is sent in order to trigger the Appeal Period End Process.	OPRS: Scheduling	
End of Process Event (End of Appeal Period)	OUT	This end of process event is sent when the Appeal Period End Process is completed.		OPRS: Scheduling
Event (Report Preparation for New Penalties)	IN	This event is sent in order to trigger the preparation for reporting of new Penalties in the Daily Penalty List.	OPRS: Scheduling	
Event (Report Preparation for Modified Penalties)	IN	This event is sent in order to trigger the preparation for reporting of updated Penalties in the List of Modified Penalties.	OPRS: Scheduling	
Event (Report Preparation of Monthly Aggregated Amounts)	IN	This event is sent in order to trigger the preparation for reporting of the Monthly Aggregated Amounts.	OPRS: Scheduling	
End of Process Event (Report Preparation for New Penalties)	OUT	This end of process event is sent when the preparation for reporting of new Penalties in the Daily Penalty List is completed.		OPRS: Scheduling

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
End of Process Event (Report Preparation for Modified Penalties)	OUT	This end of process event is sent when the preparation for reporting of updated Penalties in the List of Modified Penalties is completed.		OPRS: Scheduling
End of Process Event (Start Report Preparation of Monthly Aggregated Amounts)	OUT	This end of process event is sent when the preparation for reporting of Monthly Aggregated Amounts of Penalties is completed.		OPRS: Scheduling

3.4.7.6 Data accessed by the module

DATA	DATA ENTITY	ACCESS MODE	COMMENTS
STATIC DATA			
Static Data	Closing Day	Read	Accessed for checking purposes
	Securities	Read	Accessed for checking purposes
	Securities subject to Penalties	Read	Accessed for checking purposes
	SME Growth Market	Read	Accessed for checking purposes
	Security Penalty Rate	Read	Accessed for checking purposes
	Cash Discount Penalty Rate	Read	Accessed for checking purposes
	Euro Foreign Exchange Rate	Read	Accessed for checking purposes
	Currency	Read	Accessed for checking purposes
	Daily Price	Read	Accessed for checking purposes
	Party	Read	Accessed for checking purposes

All rights reserved.

DATA	DATA ENTITY	ACCESS MODE	COMMENTS
	Party Code	Read	Accessed for checking purposes
	Attribute Domain- "white list"	Read	Accessed for checking purposes
	System parameter- Appeal Period End Day	Read	Accessed for checking purposes
	System parameter- Monthly Reporting Day	Read	Accessed for checking purposes
	System parameter- Nth BD of the month	Read	Accessed for checking purposes
	List of CSDs with penalty computation in non-Euro T2S settlement currencies for FOPs	Read	Accessed for checking purposes
DYNAMIC DATA			
LCMM Instructions	Settlement Instruction	Read	
	Matching Object	Read	
	Reason History	Read	
	Status History	Read	
Penalties	Penalty Eligibility	Read/Write	
	Penalty	Read/Write	
	Applicable Party	Read/Write	
	Failing Reasons Dictionary	Read	
	Reasons for failing	Write	
	Penalty Reference Data for Securities	Read/Write	
	Sub-amount	Read/Write	
	Sub-amount (securities)	Read/Write	
	Sub-amount (cash)	Read/Write	
	Penalty Modification	Read/Write	
	Inbound Bulk File	Read/Write	
	Inbound Penalty Modification message	Read/Write	
	Request Error	Write	

All rights reserved.

DATA	DATA ENTITY	ACCESS MODE	COMMENTS
	Daily net for a Party vs Counterparty	Read/Write	
	Monthly net for a Party vs Counterparty	Read/Write	

3.4.8 Settlement Instructions (SI) Use Cases

Scope

This category of use cases includes all T2S instructions that may have an impact on the securities positions, cash balances and credit memorandum balances through their processing inside the *LCMM* and the *Settlement* domains, i.e. *Settlement Instructions* derived from:

- Trade operations, securities transfers and repo operations;
- *Settlement Restrictions* and *Settlement Transactions* on corporate events.

Criteria

The criteria that characterise a *Settlement Instruction* use case include all the external parameters determining processing of *Settlement Instructions* the *LCMM* and the *Settlement* domains.

These criteria are summarised in the table below with their possible values:

CRITERIA	POSSIBLE VALUES	COMMENT
Communication mode	U2A, A2A	T2S supports access via A2A or U2A communication mode
Four-Eyes principle	Yes , No	Identifies which requests under the Four-Eyes principle require approval from the approver
CSD configuration	Intra-CSD, Cross-CSD, in/out T2S	Derives from the CSD of the T2S Parties and/or the Issuer CSD
Instruction Type	Settlement Instruction: DVP, DWP, FOP, PFOD Settlement Restriction (Blocking, reservation, earmarking, remove a reservation or a blocking) Settlement Instruction deriving from corporate actions	Settlement Instructions related to corporate actions are processed similarly to DVP/DWP/FOP /PFOD instructions
Linked instructions	Yes, No	The "Yes" value derives from instructions with Link Type : 'AFTE', 'BEFO' and 'WITH' or with a pool reference

All rights reserved.

CRITERIA	POSSIBLE VALUES	COMMENT
Matching	Unmatched, Already Matched	The "Already Matched" value derives from a dedicated attribute in the instruction
Partial settlement	Yes, No	The "Yes" value derives from the flag set at instruction level
Auto-collateralisation	Balance type	Balance type for earmarking
Restriction Processing Type	Rejection, CSD Validation Hold, **Party Hold,** none**{CR-812}**	Derived from the Restriction Type to be applied to the Settlement Instruction

It should be noted that for the sake of simplification, the following parameters are taken into account as part of the description of the use cases but not as parameters that impact this list:

- Instructing Party Type with values: CSD Participant, T2S Operator, Payment Bank, Central Securities Depository (CSD), External CSD, Central Bank (CB),;
- Connectivity with values: direct connection, indirect connection;
- ISO Transaction Codes: with the relevant values of the ISO 20022.

List of Use Cases

The criteria described above are reported in the following tree:

Not all of the combinations of values are relevant from a business point of view: the complete list of use cases is presented in appendix.

3.4.9 Processing of Settlement Instructions Use Cases

3.4.9.1 Selection of representative Settlement Instructions use cases for LCMM

This selection aims at limiting the description of processing to the use cases representing a variation of each criterion (defined as the list of representative Settlement Instruction use cases in the table below) and at illustrating use cases relevant for the *LCMM* domain among a common table to *LCMM* and *Settlement* domains. Use cases already to be presented in the *Settlement* domain are mentioned on grey background. The current section only deals with the remaining three use cases (in white in the following table).

All rights reserved.

REPRESENTATIVE USE CASE	PROCESS DESCRIPTION FOCUS	CRITERIA								
		COMMUNICATION MODE	FOUR-EYES PRINCIPLE	CSD CONFIG	COSD	INSTRUCTION TYPE	LINKED	MATCHING	PARTIAL	AUTO-COLLAT
UC-SI-1	Intra-CSD standard settlement	A2A	No	Intra-CSD	No	Settlement Instruction	No	Already Matched	No	Deliverable
UC-SI-2	Linked instructions	U2A/A2A	Yes/No	Intra-CSD	No	Settlement Instruction	Yes	Unmatched	No	Deliverable
UC-SI-3	Matching	A2A	No	Intra-CSD	No	Settlement Instruction	No	Unmatched	No	Deliverable
UC-SI-4	Partial settlement	A2A	No	Intra-CSD	No	Settlement Instruction	No	Already Matched	Yes	Deliverable
UC-SI-5	Auto-collateralisation	A2A	No	Intra-CSD	No	Settlement Instruction	No	Already Matched	No	Deliverable/ Earmarked
UC-SI-6	Conditional Security Delivery	A2A	No	Intra-CSD	Yes	Settlement Instruction	No	Already Matched	No	Deliverable
UC-SI-7	Blocking	A2A	No	Intra-CSD	No	Blocking Instruction	No	Not applicable	No	Deliverable
UC-SI-8	Reservation	A2A	No	Intra-CSD	No	Reservation Instruction	No	Not applicable [129]	No	Deliverable
UC-SI-9	Settlement of a corporate action with as-	A2A	No	Intra-CSD	No	Settlement Instruction	No	Already Matched	No	Deliverable

129 The Not applicable criterion is covered by the Not required criterion in the list of Use cases criteria.

All rights reserved.

REPRESENTATIVE USE CASE	PROCESS DESCRIPTION FOCUS	CRITERIA								
		COMMUNICATION MODE	FOUR-EYES PRINCIPLE	CSD CONFIG	COSD	INSTRUCTION TYPE	LINKED	MATCHING	PARTIAL	AUTO-COLLAT
	sociated liquidity transfer									
UC-SI-10a	Cross CSD settlement	A2A	No	Cross-CSD	No	Settlement Instruction	No	Unmatched	No	Deliverable
UC-SI-10b	In-out T2S settlement	A2A	No	In-out CSD	No	Settlement Instruction	No	Unmatched	No	Deliverable

3.4.9.2 Common features for the following processes

The following processing descriptions follow several common features except:

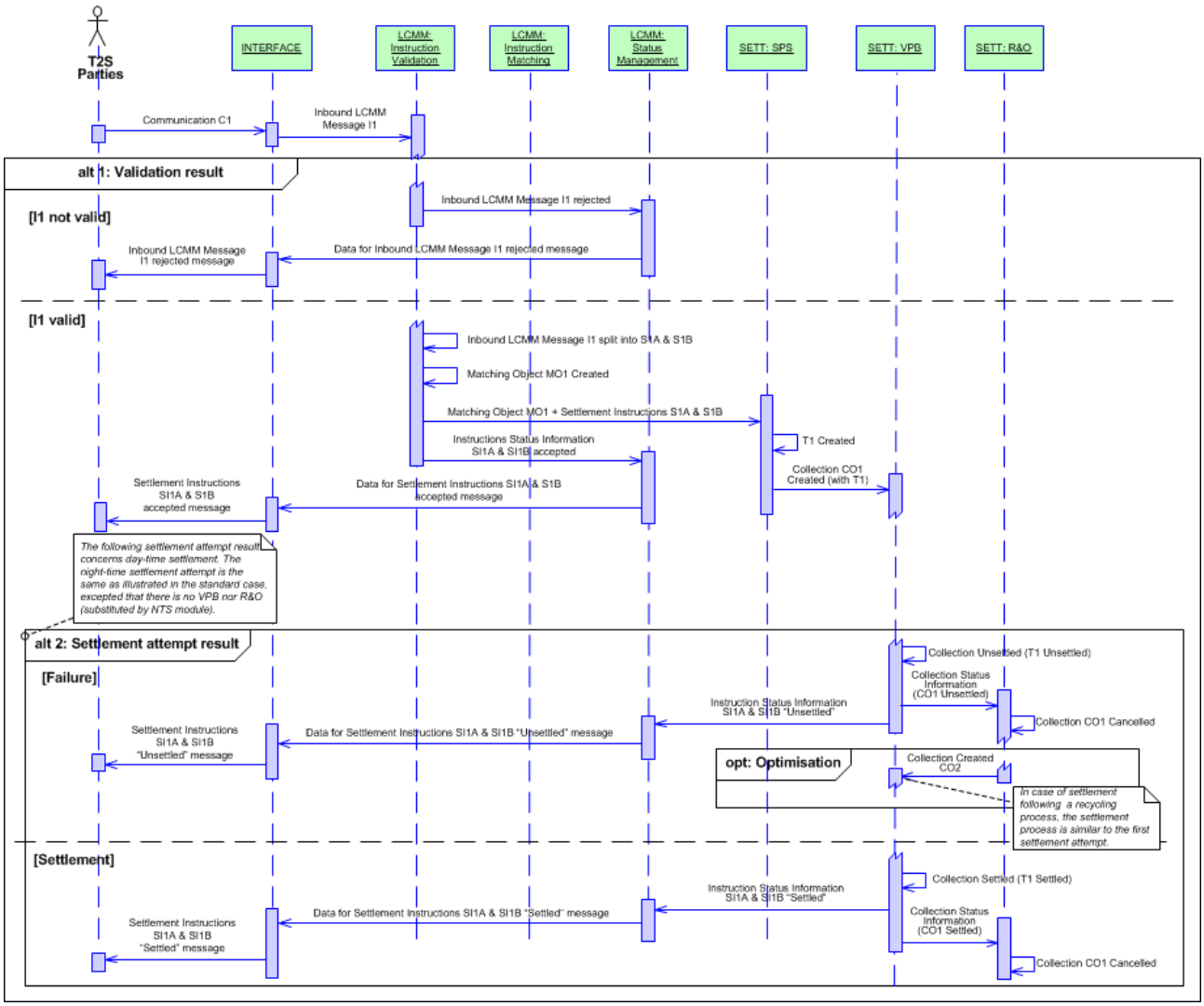
- The processing described is during the daytime settlement window;
- The underlying settlement instructions:
  - Corresponds to an "intra" scenario,
  - Are received at their Intended Settlement Date.

When necessary to the description of the processing, other features can be followed. In this case, the business assumption section mentions it explicitly.

All rights reserved.

3.4.9.3 Processing of UC-SI-1: Settlement of a standard Settlement Instruction- Already Matched

DIAGRAM 15 - SETTLEMENT OF A STANDARD SETTLEMENT INSTRUCTION ALREADY MATCHED



Business assumption

This description relates to the processing of *Settlement Instructions* where *T2S Party* A delivers securities to *T2S Party* B. Both *T2S Parties* belong to the same *CSD*. The instructions have already been matched in the corresponding *CSD*. In this case, *CSD* A sends a single message (*Communication* C1) to T2S, containing the data of the two instructions for both counterparties (*T2S Parties* A and B), with the same and unique common reference. The diagram above represents settlement during the daytime period.

Processing

Once the format and syntax validations are successfully checked by the *Interface* domain, the *Inbound LCMM Message* I1 enters the *Instruction Validation* module which executes the different validation processes to I1.

Alternative 1 - Validation result: I1 not valid

All rights reserved.

The *Inbound LCMM Message* I1 does not pass the business validations. It is rejected and all the errors are communicated to the *Status Management* module. CSD A receives a message from the *Interface* domain, depending on its message subscription preferences, about the rejection of the *Inbound LCMM Message*, together with the reason codes for the rejection.

Alternative 1- Validation result: I1 valid

The *Inbound LCMM Message* I1 successfully passes the validations. Therefore in the *Instruction Validation* module:

- The *Inbound LCMM Message* I1 is split into two separate *Settlement Instructions* SI1A and SI1B;
- Triggers the creation of the corresponding *Matching Object* MO1, including the SI1A and SI1B references and assigning a unique T2S Matching Reference to the *Settlement Instructions* SI1A and SI1B;
- Assigns the corresponding statuses to the *Settlement Instructions* SI1A and SI1B getting the Match Status *"Matched"*;
- Stores the *Settlement Instructions* SI1A and SI1B with all the above mentioned information;
- Communicates to the *Status Management* module the Instructions Status Information, so it can inform through *Interface* domain the *T2S Parties* A and B and CSD A about the acceptance of the instructions, depending on their subscription preferences.

The *Matching Object* MO1 and the related *Settlement Instructions* SI1A and SI1B are immediately forwarded to the *Standardisation and Preparation to Settlement* module. On the Intended Settlement Date, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 with all the requested attributes.

During the daytime, the *Standardisation and Preparation to Settlement* module creates a *Collection* CO1 including the settlement transaction T1 and sends it to the *Daytime Validation, Provisioning and Booking* module for a settlement attempt.

During night-time, the *Night-time Settlement* module selects the settlement transaction T1 according to the considered cycle/sequence for a settlement attempt.

Alternative 2 - Settlement attempt result: Failure

Following a settlement attempt in the *Daytime Validation, Provisioning and Booking* module, if the Settlement Transaction T1 is unsettled:

- The Settlement Status value of *Settlement Instructions* SI1A and SI1B remains *"Unsettled"*;
- An *Instruction Status Information* flow is sent to the *Status Management* module;
- *Collection* CO1 is sent to the *Daytime Recycling and Optimisation* module;
- Upon reception of *Instruction Status Information* flow, the *Status Management* sends a Message Data to the *Interface* domain, which communicates to the Party B about the failure in the settlement attempt of the *Settlement Instruction* SI2, together with the appropriate reason codes.

Option: Optimisation

All rights reserved.

Following an optimisation in the *Daytime Recycling and Optimisation* module, CO1 is cancelled and the *Settlement Transaction* T1 is submitted to the *Daytime Validation, Provisioning and Booking* module in a new *Collection* CO2 for a new settlement attempt, as the optimisation process that has assessed that the *Settlement Transaction* T1 may be settled successfully (following the standard settlement process described below).

Alternative 2 - Settlement attempt result: Settlement

Following a settlement attempt in the *Daytime Validation, Provisioning and Booking* module, if the *Settlement Transaction* T1 is settled:

- The *Daytime Validation, Provisioning and Booking* module sets the *Settlement Transaction* T1 Settlement Status to *"Settled"* and the *Settlement Instructions* SI1A and SI1B Settlement Status to *"Settled";*
- An *Instruction Status Information* flow is sent to the *Status Management* module to confirm that the *Settlement Instructions* SI1A and SI1B have their Settlement Status value set to *"Settled";*
- Upon reception of Instruction Status Information flow, the Status Management module sends to the *Interface* domain the necessary data in order to inform the involved T2S Parties (depending on their message subscriptions).

3.4.9.4 Processing of UC-SI-2: Intra-CSD Standard Linked DVP Instructions

Business assumption

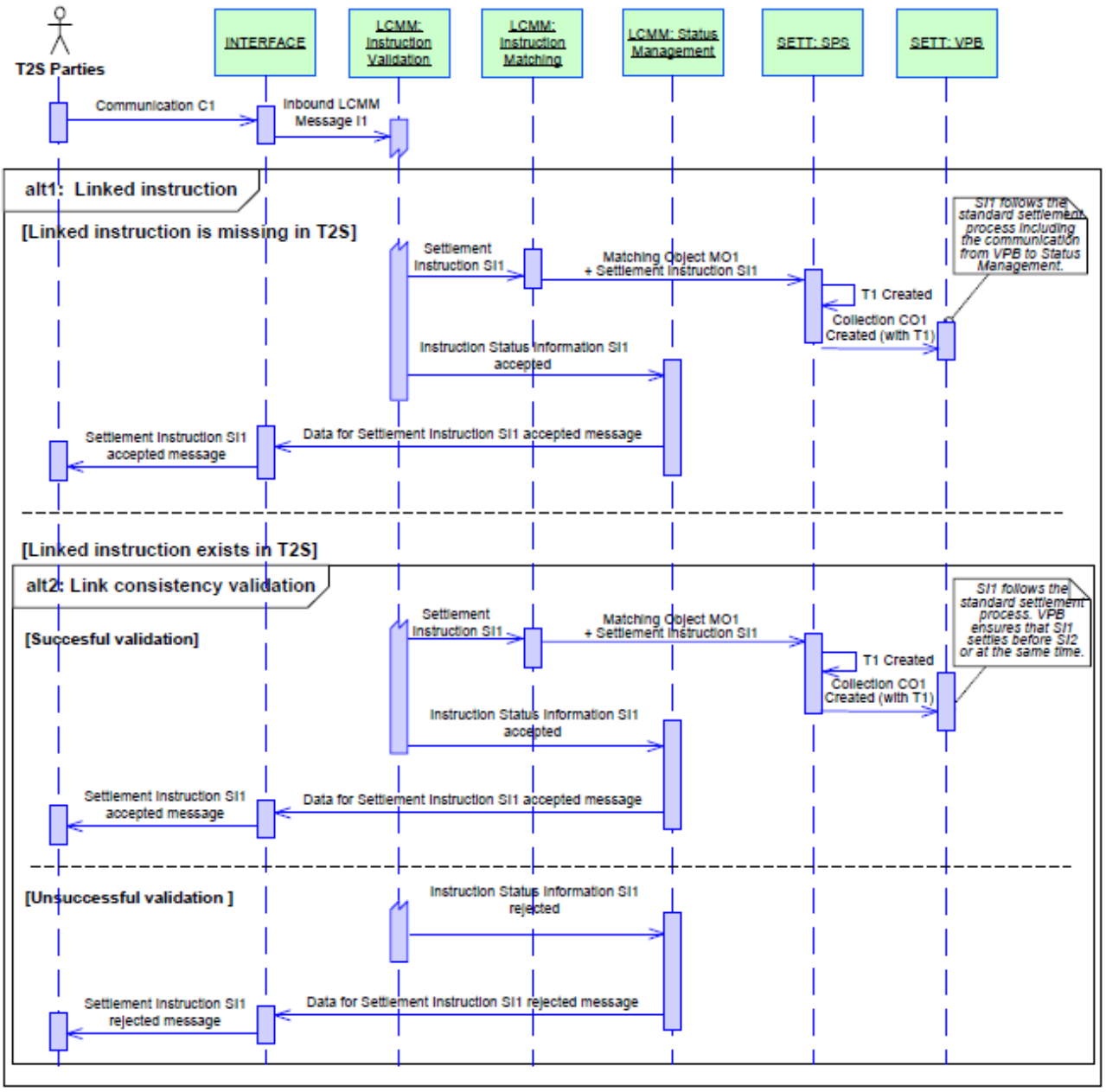
This description relates to the processing of linked *Settlement Instructions* in T2S where:

- *T2S Party* A links its *Settlement Instruction* (SI1) with its *Settlement Instruction* (SI2) by an ISO settlement Link Type Before (*"BEFO"*), After (*"AFTE"*), All-or-none (*"WITH"*);
- All the *T2S Parties* belong to the same *CSD*, and send their *Settlement Instructions* directly to T2S, without the intervention of the *CSD*;
- The *T2S Parties* do not make use of the pool reference and number count;
- This use case focuses on the peculiarities of the processing of linked instructions and do not intend to describe the standard process for the validation and matching processes that are included in the use case UC-SI-3 (Intra-CSD standard DVP with matching).

Settlement link indicator "BEFO" (before)

All rights reserved.

Diagram 16 - Intra-CSD Standard Link DVP Instruction - Settlement link indicator "BEFO"



T2S Party A first sends a *Communication* C1 that includes the Link Type *"BEFO"* with the *Settlement Instruction* reference SI1. As part of the standard validation process, the *LCMM Instruction Validation* module identifies that the *Settlement Instruction* SI1 is linked with another *Settlement Instruction* by the Link Type *"BEFO"* and checks whether the linked instruction already exists in T2S.

Alternative 1- Linked instruction: Linked instruction is missing in T2S

- T2S cannot apply any additional validation related to the link consistency between both linked instructions, as the second instruction is not yet in T2S;
- The *Settlement Instruction* SI1 follows the standard *LCMM* processing;

All rights reserved.

- The *Standardisation and Preparation to Settlement* module forwards the *Settlement Transaction* T1 related to the *Settlement Instruction* SI1 to the *Daytime Validation Provisioning and Booking* module that may settle the transaction individually.

Alternative 1- Linked instruction: Linked instruction exists in T2S

- T2S does apply validations related to the link consistency since the *T2S Party* A has already sent the *Settlement Instruction* SI2 to T2S. An unsuccessful linked instruction validation would trigger the rejection of the *Settlement Instruction* SI1.

Alternative 2 – Link consistency validation: Successful validation

- If the *Settlement Instruction* SI1 is consistent with the *Settlement Instruction* SI2 (e.g. no contradictory links, same securities account owner), then the *Settlement Instruction* SI1 follows the relevant settlement process, taking into account that the *Settlement Instruction* SI1 has to be settled before or at the same time as the *Settlement Instruction* SI2;
- The *Settlement Instruction* SI1 follows the standard *LCMM* processing;
- The *Standardisation and Preparation to Settlement* module forwards the *Settlement Transaction* T1 related to the *Settlement Instruction* SI1 to the *Daytime Validation Provisioning and Booking* module that may settle the *Settlement Transaction* before or at the same time (i.e. within the same collection) as the *Settlement Transaction* T2 related to the *Settlement Instruction* SI2.

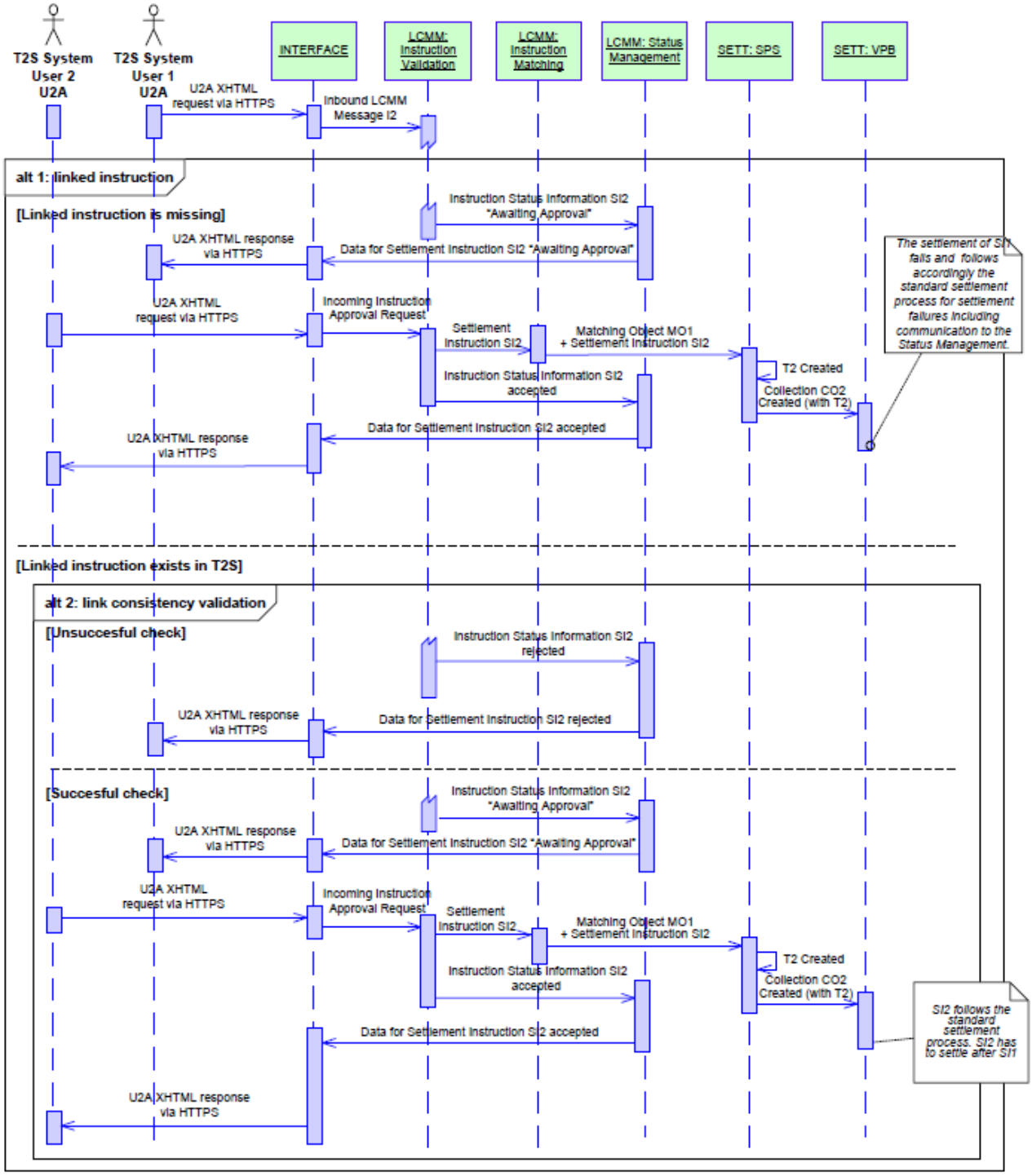
Alternative 2 –Link consistency validation: Unsuccessful validation

- The *Instruction Validation* module rejects the *Settlement Instruction* SI1 after an unsuccessful consistency validation due to contradictory links (i.e. the *Settlement Instructions* SI1 and SI2 are mutually linked with each other with a Link Type *"BEFO"*), Intended Settlement Date and Securities Account Owners;
- The rejection is communicated to the *Status Management* module. *T2S Party* A receives a message from the *Interface* domain, depending on its message subscription preferences, about the rejection of the *Settlement Instruction* SI1, together with the appropriate reason codes for the rejection.

Settlement link indicator "AFTE" (after)

All rights reserved.

DIAGRAM 17 - INTRA-CSD STANDARD LINKED DVP INSTRUCTIONS - SETTLEMENT LINK INDICATOR "AFTE"



All rights reserved.

This use case is described under the Four-Eyes principle. An authorised T2S System User from Party A sends an incoming initial request in U2A Communication mode that includes the Link Type *"AFTE"*, with the *Settlement Instruction* reference SI2. As part of the standard validation process, the *Instruction Validation* module identifies that the *Settlement Instruction* SI2 is linked with *another Settlement Instruction* by the Link Type *"AFTE"* and checks whether the linked *Settlement Instruction* SI1 already exists in T2S [130].

Alternative 1 –Linked Instruction: Linked instruction is missing

In the case of successful business validation of the Inbound LCMM Message, the Instruction Validation module:

- Does not apply any validation related to the link consistency, as the second *Settlement Instruction* with reference SI1 is not yet in T2S;
- Creates the Settlement Instruction SI2, sets its value of the Approval Status to "Awaiting Approval" and sends to the *Status Management* module the instruction status information, so it can inform through *Interface* domain the first T2S System User of the *T2S Party* A that its request is subject to the approval of a second T2S System User who has to confirm or revoke. The message is processed under the Four-Eyes principle which implies that, once created, the LCMM Instruction cannot be further processed until an approval request is sent by another T2S System User. Therefore, the *Settlement Instruction* SI2 remains in the Instruction Validation module and is not forwarded to any module;
- Stores the *Settlement Instruction* SI2 with all the above mentioned information;
- Once the second T2S System User sends its confirmation to approve the Settlement Instruction, the Instruction Validation module receives an Incoming Instruction Approval Request, and validates again the Settlement Instruction that is pending approval (i.e. Approval Status value "*Awaiting Approval*" to ensure it is still valid). In the case of successful revalidation, the Instruction Validation module assigns the value "Approved" to its Approval Status and communicates to the *Status Management* module the Instruction Status Information, so it can inform through *Interface* domain the second T2S System User about the acceptance of the instruction;
- After getting the Approval Status "Approved", the Settlement Instruction SI2 follows the standard LCMM processing.

Standardisation and Preparation to Settlement module forwards the *Settlement Transaction* T2 related to the *Settlement Instruction* SI2 to the *Validation Provisioning and Booking* module on the Intended Settlement Date. At the end of the Intended Settlement Date, if the *Settlement Instruction* SI1 is not received, then the *Settlement Instruction* SI2 remains unsettled, since it can only settle after or at the same time than the *Settlement Instruction* SI1.

Alternative 1 – Linked instruction: Linked instruction exists

T2S does apply validations related to the link consistency since the *T2S Party* A has already sent the *Settlement Instruction* SI1 to T2S. The *Instruction Validation* module checks that the information contained in the *Settlement Instruction* SI2 is consistent with the *Settlement Instruction* SI1. A fail to pass any of the validations would trigger the rejection of the *Settlement Instruction* SI2.

130 In the context of a Repo operation, T2S does not have any different processing than the other links (e.g. "AFTE", "BEFO"). If the T2S party decides to use a common repo reference, T2S will ensure that the closing leg only settles after or at the same time as the starting leg as T2S.

All rights reserved.

Alternative 2 –Link consistency validation: Unsuccessful validation

The *Instruction Validation* module rejects the *Settlement Instruction* SI2 after an unsuccessful consistency validation due to contradictory links (i.e. the *Settlement Instructions* SI1 and SI2 are mutually linked with each other with a Link Type *"AFTE"*), Intended Settlement Date and securities account owner.

The rejection is communicated to the *Status Management* module. T2S Party A receives a XHTML response in U2A from the *Interface* domain, about the rejection of the *Settlement Instruction*.

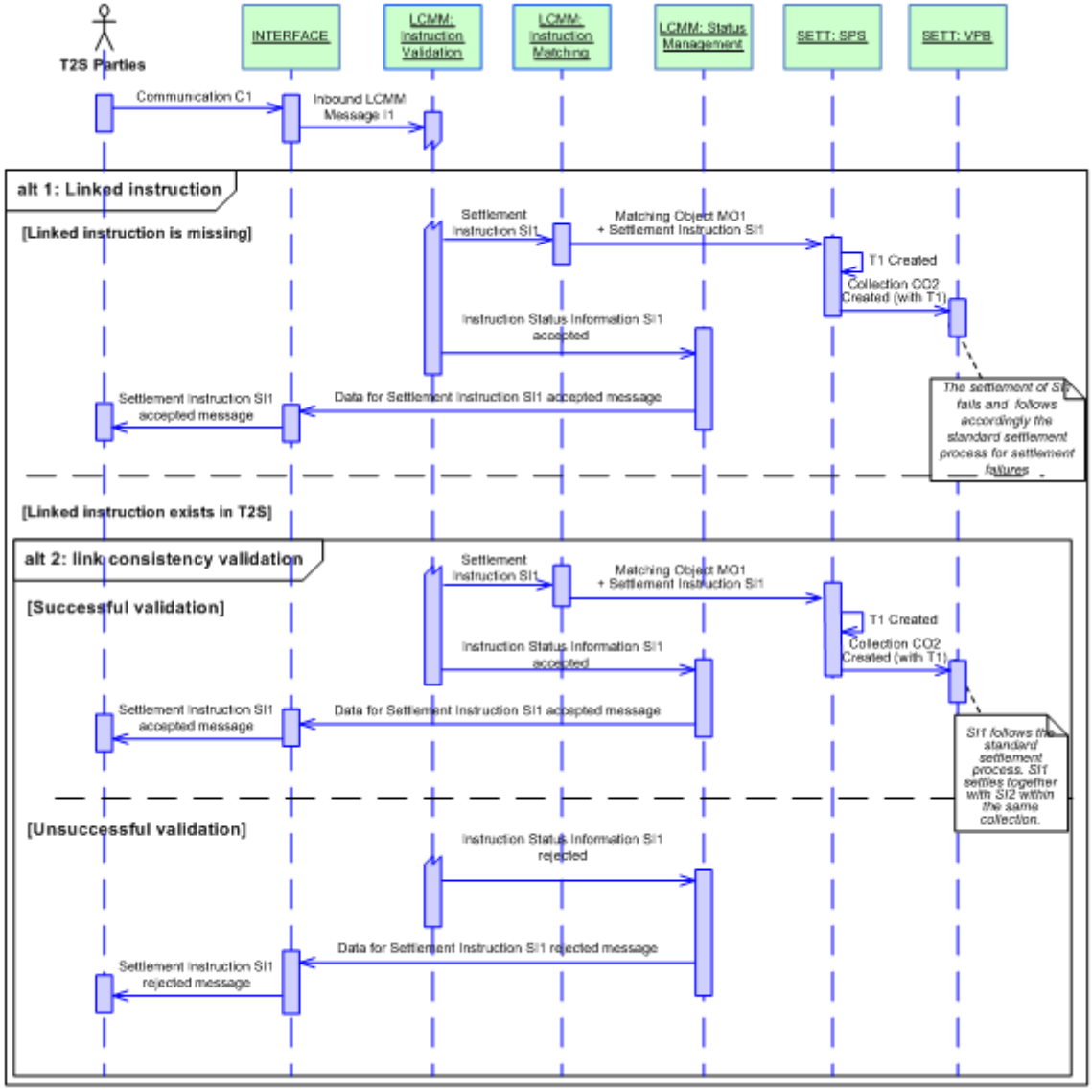
Alternative 2 - Link consistency validation: Successful validation

If the consistency check is successful, the Settlement Instruction SI2 is created and follows the standard LCMM processing once it gets the Approval Status to "Approved", as explained before in the case the linked instruction is missing. The *Settlement Instruction* SI1 and the *Settlement Instruction* SI2 follow the relevant settlement process (the *Settlement Instruction* SI2 is settled after or at the same time as the *Settlement Instruction* SI1).

Settlement link indicator "WITH" (all-or-none)

All rights reserved.

DIAGRAM 18 - INTRA-CSD STANDARD LINKED DVP INSTRUCTION - SETTLEMENT LINK INDICATOR "WITH"



T2S Party A sends the *Communication* C1 which contains the *Settlement Instruction* SI1 that includes the Link Type *"WITH"* with the *Settlement Instruction* reference SI2.

As part of the standard validation process, the *Instruction Validation* module identifies that the *Settlement Instruction* SI1 is linked to another *Settlement Instruction* by the Link Type *"WITH"* and checks whether the linked *Settlement Instruction* SI2 already exists in T2S.

Alternative 1 – Linked Instruction: Linked instruction is missing

- T2S does not apply any validation related to the link consistency, as the SI2 is not yet in T2S;
- The *Settlement Instruction* SI1 follows the standard *LCMM* processing;
- *Standardisation and Preparation to Settlement* module does not forward the *Settlement Transaction* T1 related to the *Settlement Instruction* SI1 to the *Validation Provisioning and Booking* module as it cannot be settled, given that SI1 and SI2 are to be settled simultaneously.

All rights reserved.

Alternative 1 – Linked instruction: Linked instruction exists in T2S

T2S does apply validations related to the link consistency since the *T2S Party* B has already sent the *Settlement Instruction* SI2 to T2S. A fail to pass any of the additional validation would trigger the rejection of the *Settlement Instruction* SI1.

Alternative 2 – Link consistency validation: Successful validation

- If the consistency check is successful, the *Settlement Instructions* SI1 and SI2 follow the relevant settlement process (both to be settled at the same time);
- The *Settlement Instruction* SI1 follows the standard *LCMM* processing;
- When both instructions have arrived to *Standardisation and Preparation to Settlement* module, the corresponding Settlement Transactions are forwarded in a single collection to the *Daytime Validation Provisioning and Booking* module for settlement attempt. The *Daytime Validation Provisioning and Booking* module tries to settle both *Settlement Transactions* in all-or-none basis.

Alternative 2 –Link consistency validation: Unsuccessful validation

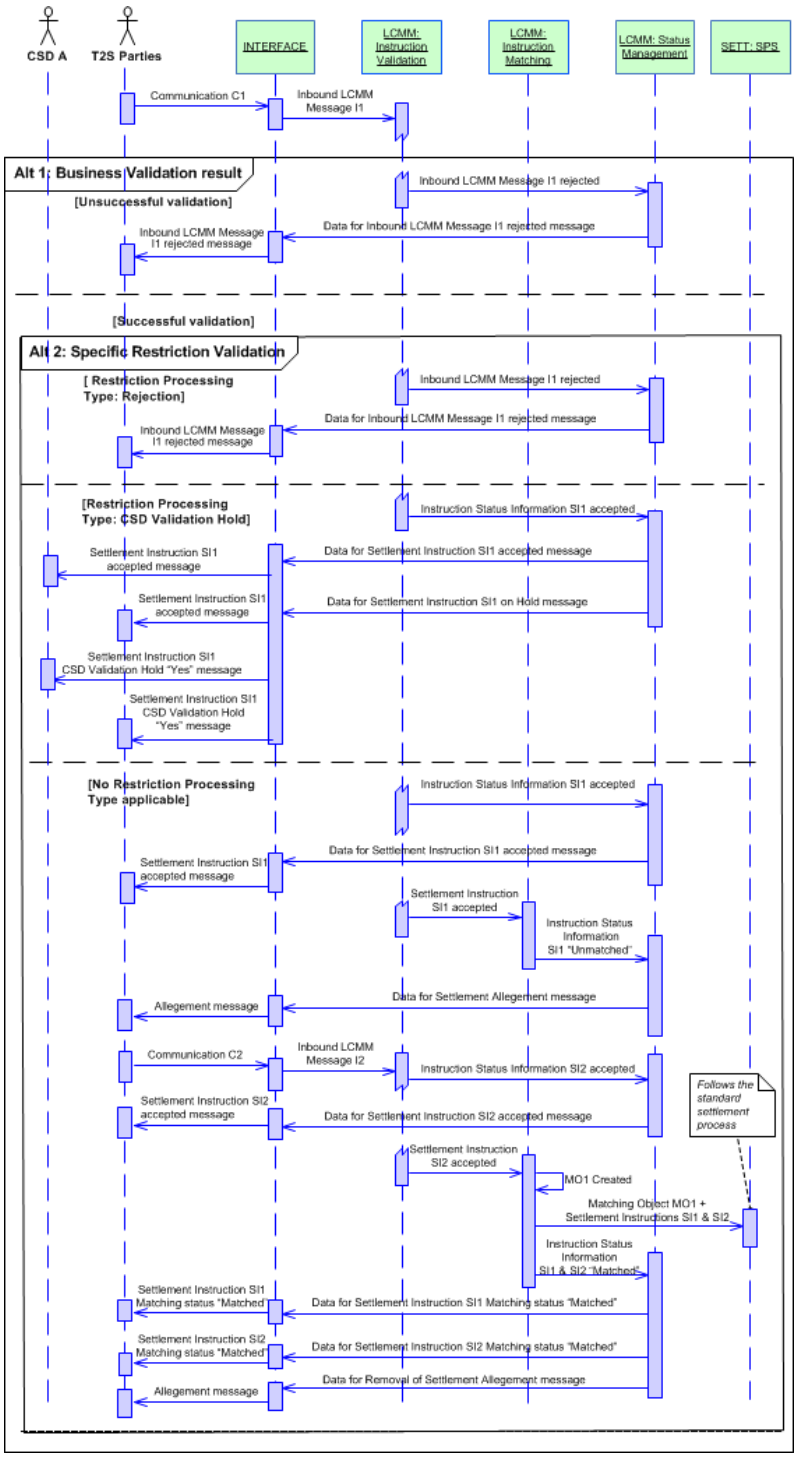
The *Instruction Validation* module rejects the *Settlement Instruction* SI1 after an unsuccessful consistency validation due to contradictory links, Intended Settlement Date and securities account owners.

The rejection is communicated to the *Status Management* module. *T2S Party* A receives a message from the *Interface* domain, depending on its message subscription preferences, about the rejection of the *Settlement Instruction*, together with the appropriate reason codes for the rejection.

All rights reserved.

3.4.9.5 Processing of UC-SI-3: Intra CSD-Standard DVP with Matching

DIAGRAM 19 - INTRA-CSD STANDARD DVP WITH MATCHING



Business assumption

This description relates to the settlement of two standard *Settlement Instructions*, which require matching in T2S. The Use Case also describes the check performed by T2S to verify if there is a Restriction Processing Type applied for the combination of parameters contained in the Inbound LCMM Message.

In this description, the assumptions are:

All rights reserved.

- Both *T2S Parties* A and B belong to the same *CSD*, and send their instruction message directly to T2S, without intervention of a *CSD*;
- *T2S Party* A sends its instruction message (i.e. *Communication* C1) which contains the *Settlement Instruction* SI1 to T2S before T2S Party B;
- *T2S Party* B sends its instruction message (i.e. *Communication* C2) which contains the *Settlement Instruction* SI2 after receiving an Allegement message from T2S related to the *T2S Party* A instruction message;
- CSD A sets the Restriction Processing Type.

Processing

Once the format and syntax validations are successfully passed through the *Interface* domain, the *Inbound LCMM Message* I1 enters the *Instruction Validation* module that validates it according to the validation checks.

Alternative 1- Validation result: Unsuccessful validation

If the *Inbound LCMM Message* I1 does not pass the validations, the *Instruction Validation* module rejects it and communicates all errors to the *Status Management* module. The *Interface* domain informs the *T2S Party* A about the rejection of the *Inbound LCMM Message* I1, together with the reason codes for the rejection, depending on its message subscription preferences.

Alternative 1- Validation result: Successful validation

After the *Inbound LCMM Message* I1 successfully passes the required business validations, T2S checks whether there is any applicable Restriction for the combinations of parameters contained in the Inbound LCMM Message I1.

Alternative 2- Restriction Processing Type: Rejection

If the *Inbound LCMM Message* I1 fulfils predefined conditions of a Restriction Processing Type of the CSD A requiring T2S to reject the *Settlement Instruction*, the *Instruction Validation* module rejects it. After rejection, *Instruction Validation* module communicates the reason of this rejection as well as all remaining errors to the *Status Management* module so that it can inform the T2S Party A via the *Interface* domain about the rejection of the *Inbound LCMM Message* I1, together with the reason codes for the rejection, depending on its message subscription preferences.

Alternative 2- Restriction Processing Type: CSD Validation Hold

If the Inbound LCMM Message I1 fulfils predefined conditions of a Restriction Processing Type of a CSD requiring T2S to hold the Settlement Instruction, then the Instruction Validation module:

- Creates the Settlement Instruction SI1 with the CSD Validation Hold status "Yes";
- Stores the *Settlement Instruction* SI1 with all the above mentioned information;
- Sends the Instruction Status Information to the Status Management module so it can inform the T2S Party A via the Interface domain about the acceptance of its instruction, depending on its message subscription preferences;

All rights reserved.

- Sends the *Instruction Status Information* to the Status Management module so that it can inform the CSD A that set the Restriction Type conditions fulfilled by the Inbound LCMM Message I1 and the T2S Party A about the status update of the instruction via the *Interface* domain. T2S Party B is not informed about the CSD Validation Hold status "Yes" until the Intended Settlement Date is reached.

Alternative 2- Restriction Processing Type: Party Hold

If the Inbound LCMM Message I1 fulfils predefined conditions of a Restriction Processing Type of a CSD requiring T2S to set the Settlement Instruction on Party Hold, then the Instruction Validation module:

- **Creates the Settlement Instruction SI1 with the Party Hold status "Yes";**
- **Stores the Settlement Instruction SI1 with all the above mentioned information;**
- **Sends the Instruction Status Information to the Status Management module so it can inform the T2SParty A via the Interface domain about the acceptance of its instruction, depending on its message subscription preferences;**
- **Sends the Instruction Status Information to the Status Management module so that it can inform the CSD A that set the Restriction Type conditions fulfilled by the Inbound LCMM Message I1 and the T2SParty A about the status update of the instruction via the Interface domain. T2S Party B is not informed about the Party Hold status "Yes" until the Intended Settlement Date is reached.{CR-812}**

Alternative 2- No Restriction Processing Type applicable

If the Inbound LCMM Message I1 does not match any of the rules or parameters defined in any of the Restriction Types, then:

- The *Instruction Validation* module creates and stores a *Settlement Instruction*;
- The *Instruction Validation* module informs the *Status Management* module so that it can inform the T2S Party A about the acceptance of its instruction via the *Interface* domain, depending on their subscription preferences;
- The *Instruction Validation* module informs the *Instruction Matching* module that it can attempt matching the *Settlement Instruction* SI1. Since the counterparty *Settlement Instruction* SI2 is not yet in the system, no matching is possible and the *Settlement Instruction* SI1 remains unmatched;
- The *Instruction Matching* module collects all the relevant data and forwards it to the *Status Management* module in order to create an Settlement *Allegement* message that is sent, after a defined period of time, via the *Interface* domain (depending on the message subscription preference) to the *T2S Party* B.

After receiving the *Settlement Allegement* message, *T2S Party* B sends its *Communication* C2 to T2S.

Once the *Inbound LCMM Mess*age I2 has successfully passed all the validation procedures in the *Interface* domain, the *Instruction Validation* module creates a *Settlement Instruction* SI2, informs *Status Management* module as described above for the *Settlement Instruction* SI1 and forwards the *Settlement Instruction* SI2 to the *Instruction Matching* module.

All rights reserved.

The *Instruction Matching* module searches for a matching counterparty *Settlement Instruction* (i.e. the *Settlement Instruction* SI1) and verifies that all the mandatory matching fields match.

If the result of the comparison is a successful matching then the following processes are performed:

- The *Matching Object* MO1 is created, and a common T2S Matching Reference is assigned to the *Settlement Instructions* SI1 and SI2;
- The Match Status value of the *Settlement Instructions* SI1 and SI2 are set to *"Matched"* and are communicated to the *Status Management* module. *T2S Parties* A and B are informed, through the *Interface* domain, that the *Settlement Instructions* SI1 and SI2 are *"Matched"* according to their message subscription preferences;
- The *Status Management* module also collects the relevant data in order to send a removal of the previously sent *Allegement* message to T2S Party B, to inform that this message is no longer valid, according to T2S Party B message subscription preferences;
- The *Matching Object* MO1 and the related *Settlement Instructions* SI1 and SI2 needed to create the *Settlement Transaction* T1 are sent to the *Standardisation and Preparation to Settlement* module in the *Settlement* domain.

After this point, the description of the Use Case follows the standard settlement process.

3.4.10 Maintenance Instruction (MI) Use Case

Scope

This category of use cases includes all maintenance instructions that aim to amend, cancel, hold or release *Settlement Instructions* (SI) already existing in T2S.

Criteria

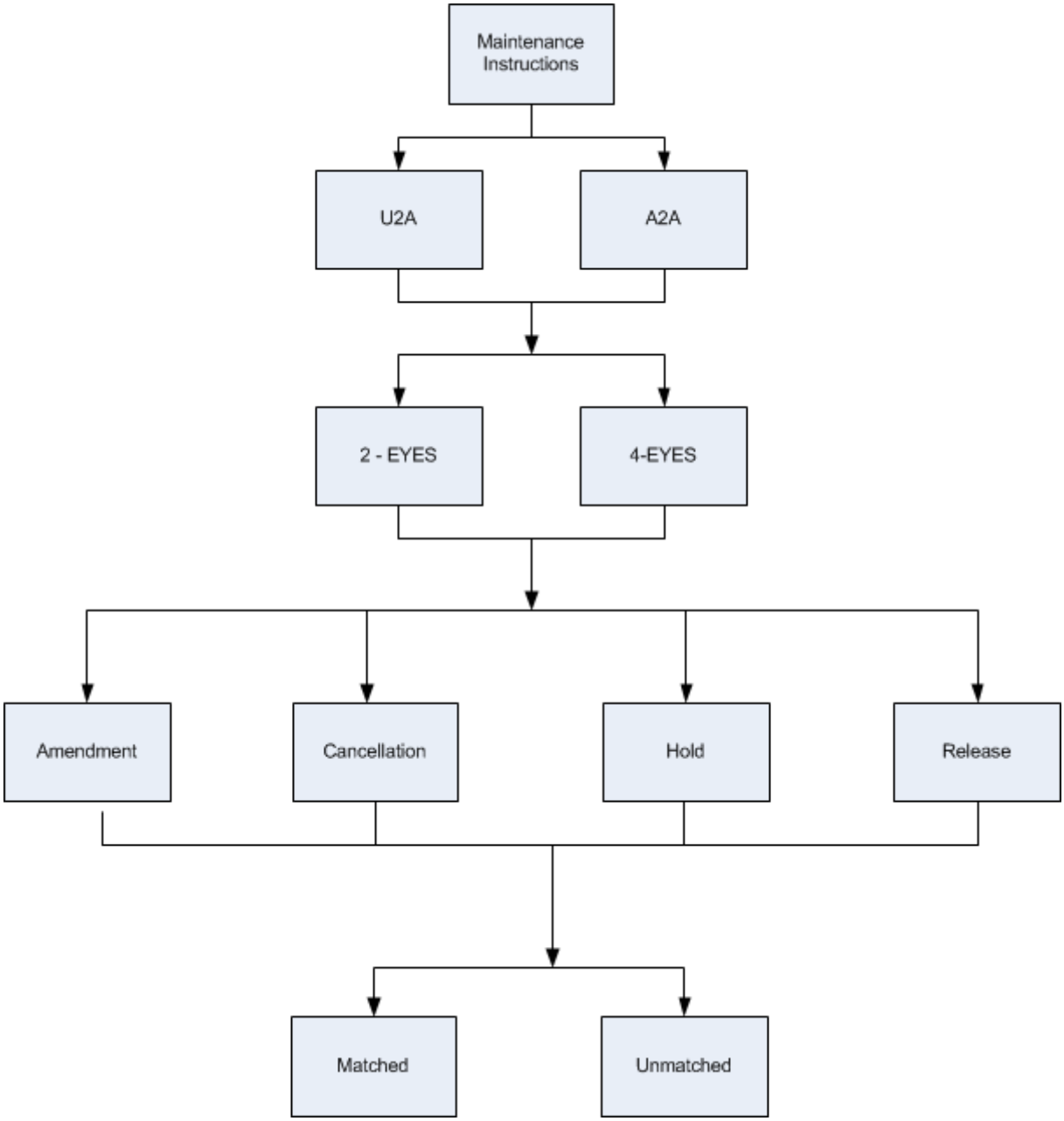
The criteria comprise all the external parameters determining the processing of the maintenance instructions in the *LCMM* and the *Settlement* domains. These criteria are summarised in the table below:

CRITERIA	POSSIBLE VALUES	COMMENT
Communication mode	U2A, A2A	T2S supports access via A2A mode or U2A mode
Four-Eyes principle	Yes , No	Identifies requests under the Four-Eyes principle (requiring approval from a second pair of eyes)
Maintenance Instruction Type	Amendment, Hold, Release, Cancellation	
Status of the original Settlement Instruction	Matched, Unmatched.	

List of Use Cases

All rights reserved.

The criteria described above are reported in the following tree:



Not all the combinations of values are relevant from a business point of view: the complete list of use cases is presented in an appendix.

3.4.11 Processing of Maintenance Instructions Use Cases

3.4.11.1 Selection of representative Maintenance Instructions Use Cases

The following representative use cases are described:

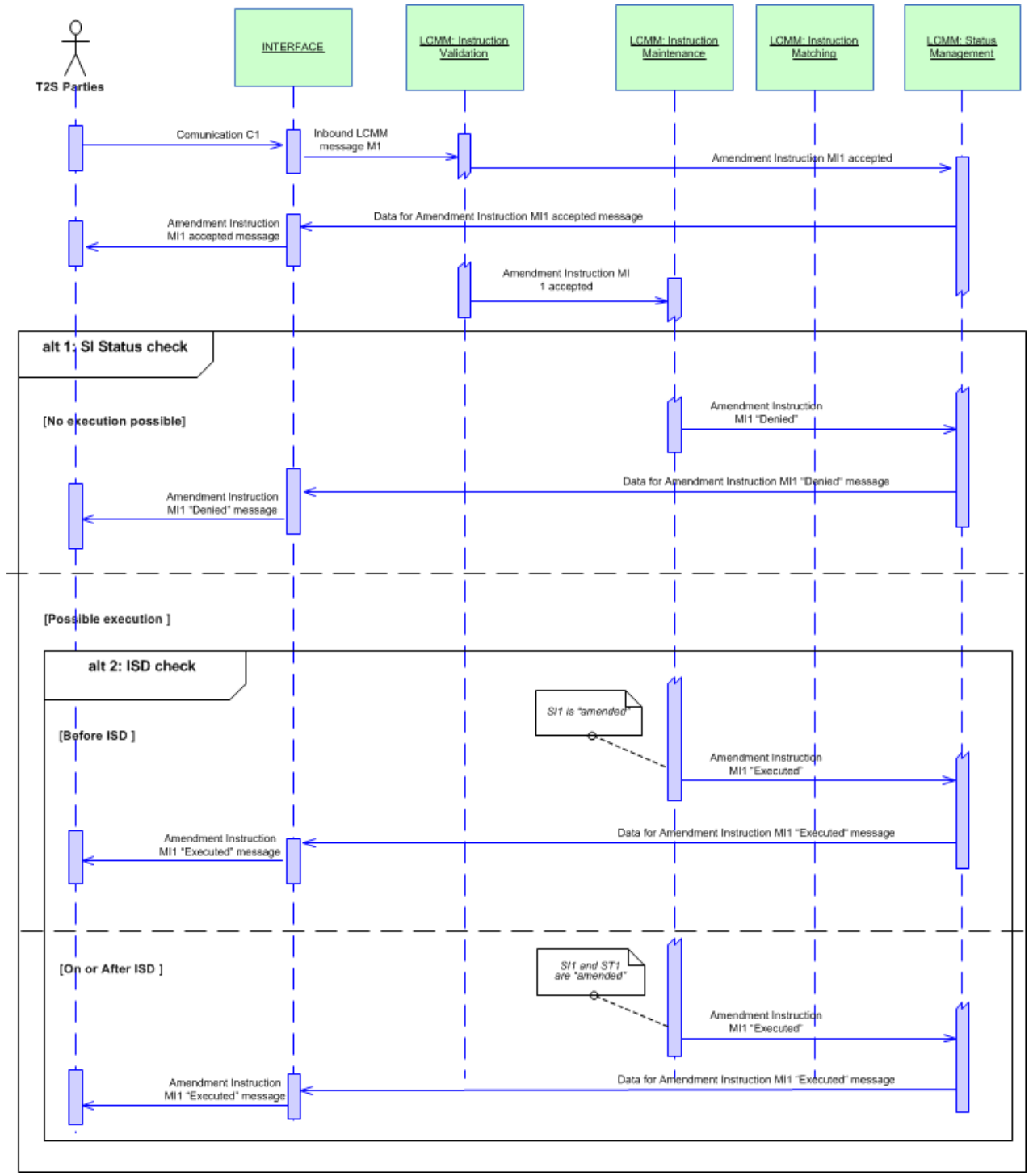
- UC-MI-1: *Amendment Instruction*;
- UC-MI-2: *Hold / Release Instruction*;

All rights reserved.

- UC-MI-3: *Cancellation Instruction*.

3.4.11.2 Processing of UC-MI-1: Amendment Instruction.

DIAGRAM 20 - AMENDMENT INSTRUCTION



Business assumption

All rights reserved.

This description relates to an *Amendment Instruction*, communicated A2A by a T2S *Party*, *Communication* C1, which contains an *Amendment Instruction* MI1 requesting the amendment of a previously sent *Settlement Instruction* SI1 (Intra-CSD instruction). The instruction does not fulfil any CoSD Rule and is not submitted under Four-Eyes principle. The amendment concerns a process indicator.

Processing

Once the format and syntax validations for Communication C1 are successfully passed, the *Inbound LCMM Message* M1 enters the *Instruction Validation* module which validates it. When it is successfully validated, the *Amendment Instruction* MI1 is created, and the *Status Management* module is informed on the acceptance of MI1 to be reported to T2S Party, via the *Interface* domain, depending on the message subscription preferences of the *Party*.

The *Amendment Instruction* MI1 is then routed to the *Instruction Maintenance* module. This module performs the following checks in order to identify if the *Amendment Instruction* can be executed:

- The Settlement Status of the referenced *Settlement Instruction* should not be *"Settled"*. Nevertheless, if the Settlement Status of the referenced Settlement Instruction is "Partially Settled", it is only possible to modify the settlement priority.
- The Cancellation Status of the referenced *Settlement Instruction* should not be *"Cancelled"*.
- The Approval Status of the referenced *Settlement Instruction* should not be *"Awaiting Approval"*.
- The CoSD Hold Status of the referenced *Settlement Instruction* should be *"No"*.

Alternative 1 – SI Status check: No execution possible (e.g. the Settlement Instruction is "Cancelled")

The *Settlement Instruction* SI1 Cancellation Status value is *"Cancelled"*, then the *Amendment Instruction* cannot be executed and the *Settlement Instruction* SI1 is not amended. The processing is the same for the statuses *"Settled", "CoSD hold"* and *"Awaiting approval".*

- The Cancellation Status of the *Amendment Instruction* MI1 is set to "Denied" with Reason Code Reference *"Referenced Settlement Instruction already cancelled";*
- The referenced *Settlement Instruction* SI1 is not modified;
- The *Status Management* module is informed by the *Maintenance Instruction* module of the status of the *Amendment Instruction* MI1 to be reported to T2S *Party* via the *Interface* domain (depending on the T2S *Party*'s message subscription preferences).

Alternative 1 – SI Status check: Possible execution

The *Settlement Instruction* SI1 fulfils the conditions described above, then the *Amendment Instruction* can be executed, but depending whether the Intended Settlement Date has been reached or not *Instruction Maintenance* module performs differently.

Alternative 2 – ISD check: Before the ISD

- The *Amendment Instruction* MI1 Execution status is set to *"Executed"*;
- The amendment is applied on the referenced *Settlement Instruction* SI1;

All rights reserved.

- The *Status Management* module is informed by the *Maintenance Instruction* module of the status of the *Amendment Instruction* MI1 to be reported to T2S *Party* via the *Interface* domain (depending on the T2S *Party*'s message subscription preferences).

Alternative 2 – ISD check: On or After ISD

- The *Amendment Instruction* MI1 Execution Status is set to *"Executed"*;
- The amendment is applied on the referenced *Settlement Instruction* SI1;
- The *Settlement Transaction* of the amended Settlement Instruction is modified;
- The *Status Management* module is informed by the *Maintenance Instruction* module of the status of the *Amendment Instruction* MI1 to be reported to T2S Party via the *Interface* domain (depending on the T2S *Party*'s message subscription preferences).

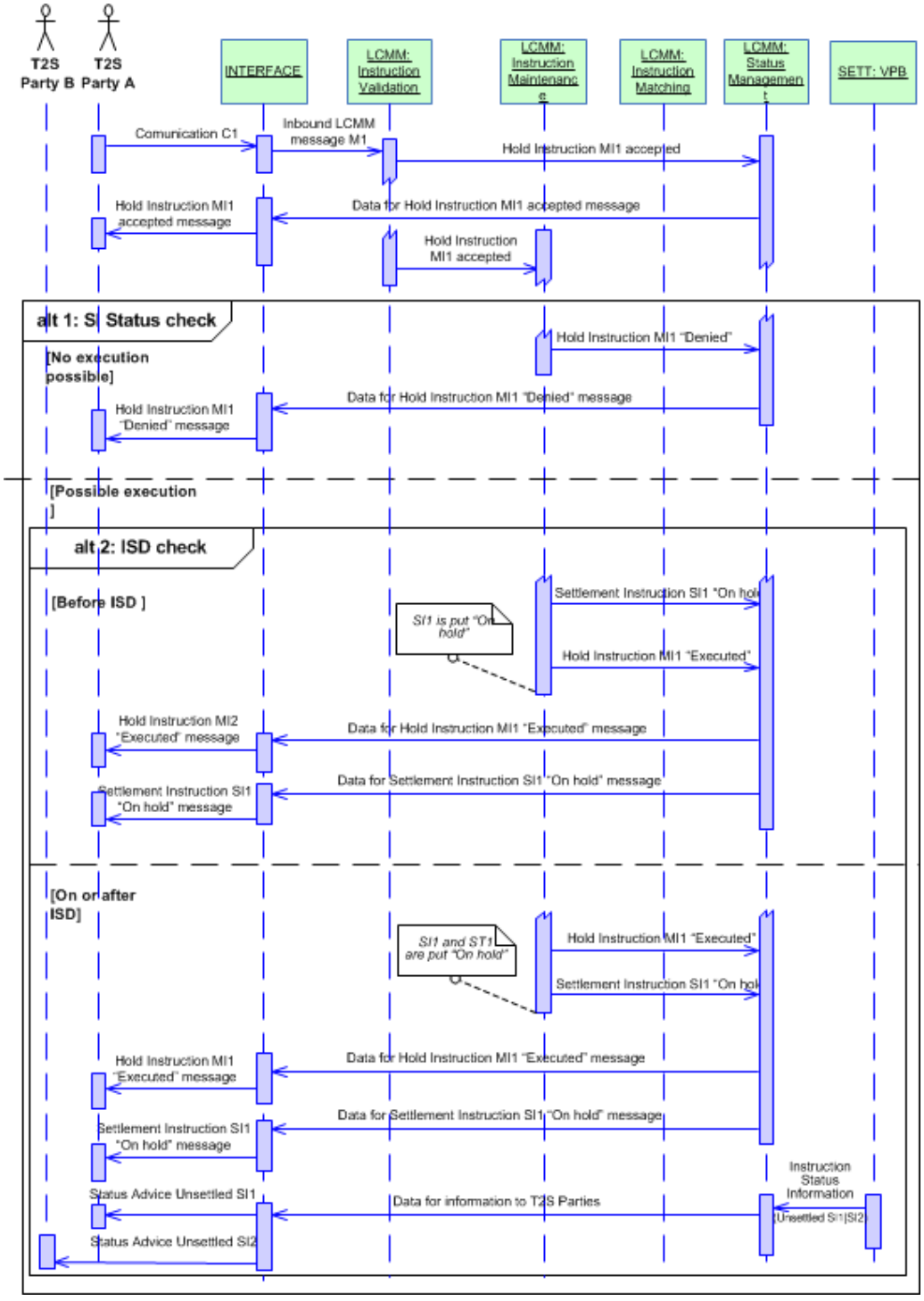
3.4.11.3 Processing of UC-MI-2: Hold / Release Instruction

a) Hold Instruction from a T2S Party

"Hold Instruction" refers to the entity *Hold/Release Instruction* of the data model, with Hold Indicator at "*Yes*".

All rights reserved.

DIAGRAM 21 - HOLD INSTRUCTION



Business assumption

This description relates to a *Communication* C1 which contains a Hold Instruction MI1 communicated by a T2S *Party* A , requesting to put on hold of a previously sent *Settlement Instruction* SI1 (Intra-CSD instruction).

Processing

All rights reserved.

Once the format and syntax validations for *Communication* C1 are successfully passed, the *Inbound LCMM Message* M1 enters the *Instruction Validation* module which validates it. When it is successfully validated, the Hold Instruction MI1 is created, and the *Status Management* module is informed on the acceptance of MI1 to be reported to the T2S *Party* A, via the *Interface* domain (depending on the T2S *Party*'s message subscription preferences*)*.

The Hold Instruction MI1 is then routed to the *Instruction Maintenance* module. This module performs the following checks in order to identify if the Hold Instruction can be executed:

- The Settlement Status of the referenced *Settlement Instruction* should not be *"Settled".*
- The Cancellation Status of the referenced *Settlement Instruction* should not be *"Cancelled"*.
- The CoSD Hold Status of the referenced *Settlement Instruction* should be *"No"*.
- The Party Hold Status of the referenced *Settlement Instruction* should be *"No"*.
- The Approval Status of the referenced *Settlement Instruction* should not be *"Awaiting Approval"*.

Alternative 1 – SI Status check: No execution possible (e.g. the Settlement Instruction is already put on hold by the Party)

If the Party Hold Status of the *Settlement Instruction* is *"Yes"*, then the Hold Instruction cannot be executed.

- The Cancellation Status of the Hold Instruction MI1 is set to *"Denied"* with Reason Code Reference *"Affected SI already on Hold"*;
- The Party Hold Status of the referenced *Settlement Instruction* SI1 is not modified;
- *Status Management* module is informed on the status of the *Hold Instruction* MI1 to be reported to T2S *Party* A, via the *Interface* domain, depending on its message subscription preferences.

Alternative 1 – SI Status check: Possible execution

The *Settlement Instruction* SI1 fulfils the conditions described above; then, the Hold Instruction MI1 can be executed, but depending whether the Intended Settlement Date has been reached or not, *Instruction Maintenance* module performs differently.

Alternative 2 - ISD Check: Before ISD

- The Execution Status of Hold Instruction MI1 is set to *"Executed"*;
- The Party Hold Status of the referenced *Settlement Instruction* SI1 is changed to *"Yes"*;
- *Status Management* module is informed on the status of the Hold Instruction MI1 and *Settlement Instruction* SI1 to be reported to the T2S *Party* A, via the *Interface* domain (depending on the *Party*'s message subscription preferences).

Alternative 2 – ISD Check: On or After ISD

- The Execution Status of Hold Instruction MI1 is set to "Executed";
- Both the *Settlement Transaction* ST1 and the *Settlement Instruction* SI1 are put on hold (*Settlement Instruction* SI1 Party Hold Status set to *"Yes"*) ;

All rights reserved.

Status Management module is informed on the status of the Hold Instruction MI1 and *Settlement Instruction* SI1 to be reported to the T2S *Party* A, via the *Interface* domain, depending on the *Party*'s message subscription preferences.

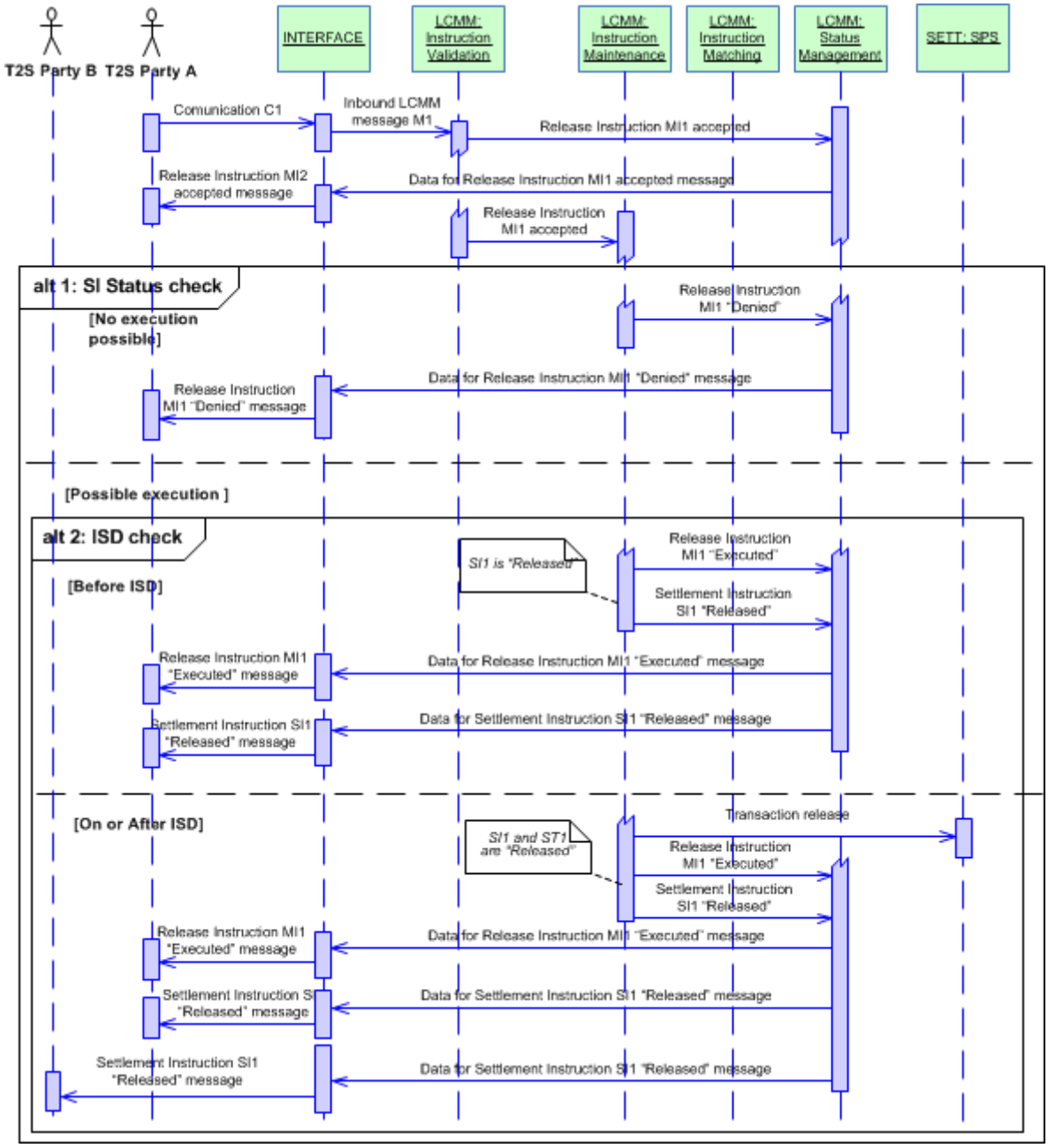
T2S does not inform the T2S *Party* B about the hold of SI1. Nevertheless, on the Intended Settlement Date of the *Settlement Instruction* SI2*,* T2S communicates T2S *Party* B that this *Settlement Instruction* is unsettled because the counterparty instruction is on hold. This is triggered when *Standardisation and Preparation to Settlement* module of the *Settlement* domain sends an *Instruction Status Information* to the *Status Management* module informing about the reason of the unsuccessful settlement. *Status Management* module reports to the T2S *Party* B, via the *Interface* domain, depending on the *Party*'s message subscription preferences.

b) Release Instruction from a T2S Party

"Release Instruction" refers to the entity *Hold/Release Instruction* of the data model, with Hold Indicator at *"No"*.

All rights reserved.

DIAGRAM 22 - RELEASE INSTRUCTION



Business assumption

This description relates to a *Communication* C1 which contains a Release Instruction MI1 communicated by a T2S *Party* A, requesting to release an already existing On Hold *Settlement Instruction* SI1 (Intra-CSD instruction). The instruction does not fulfil any CoSD Rule and it is not submitted under Four-Eyes principle.

Processing

All rights reserved.

Once the format and syntax validations for *Communication* C1 are successfully passed, the *Inbound LCMM Message* M1 enters the *Instruction Validation* module which validates it. When it is successfully validated, the *Release Instruction* MI1 is created, and the *Status Management* module is informed of the acceptance of Release Instruction MI1 to be reported to the T2S *Party* A, via the *Interface* domain (depending on the message subscription preferences of the *Party)*.

The *Release Instruction* MI1 is then routed to the *Instruction Maintenance* module. This module performs the following checks in order to identify if the Release Instruction can be processed:

- The Settlement Status of the referenced *Settlement Instruction* should not be *"Settled"*;
- The Cancellation Status of the referenced *Settlement Instruction* should not be *"Cancelled"*;
- The CoSD Hold Status of the referenced *Settlement Instruction* must be *"No"*;
- The Party Hold Status of the referenced *Settlement Instructions* should be *"Yes"*;
- The Approval Status of the referenced *Settlement Instruction* should not be *"Awaiting Approval"*;
- Before *Settlement* can process the referenced *Settlement Instruction*:
  - The CSD Hold Status of the referenced *Settlement Instruction* must be *"No"*,
  - The CSD Validation Hold Status of the referenced *Settlement Instruction* must be *"No"*.

Alternative 1 – SI Status check: No execution possible (i.e. the Settlement Instruction is already cancelled)

If the *Settlement Instruction* SI1 Cancellation Status value is *"Cancelled"*, then the *Release Instruction* cannot be executed and *Settlement Instruction* SI1 is not released.

- The Cancellation Status of the Release Instruction MI1 is set to "Denied" with Reason Code Reference "Affected Settlement Instruction already cancelled";
- The referenced Settlement Instruction SI1 is not modified;
- Status Management module is informed on the status of the Release Instruction MI1 to be reported to T2S Party A, via the *Interface* domain, depending on its message subscription preferences.

Alternative 1 – SI Status check: Possible execution

The *Settlement Instruction* SI1 fulfils the conditions described above then the Release Instruction can be executed, but depending whether the Intended Settlement Date has been reached or not, *Instruction Maintenance* module performs this execution differently.

Alternative 2 - ISD check: Before the ISD

- The Execution Status of the Release Instruction MI1 is set to *"Executed"*;
- The release is applied on the referenced *Settlement Instruction* SI1;
- *Status Management* module is informed on the status of the Release Instruction MI1 to be reported to the T2S *Party* A, via the *Interface* domain, depending on the *Party*'s message subscription preferences.

Alternative 2 – ISD check: On or After ISD

- The Execution Status of the Release Instruction MI1 is set to *"Executed"*;
- The referenced *Settlement Instruction* SI1 and the *Settlement Transaction* ST1 are released;

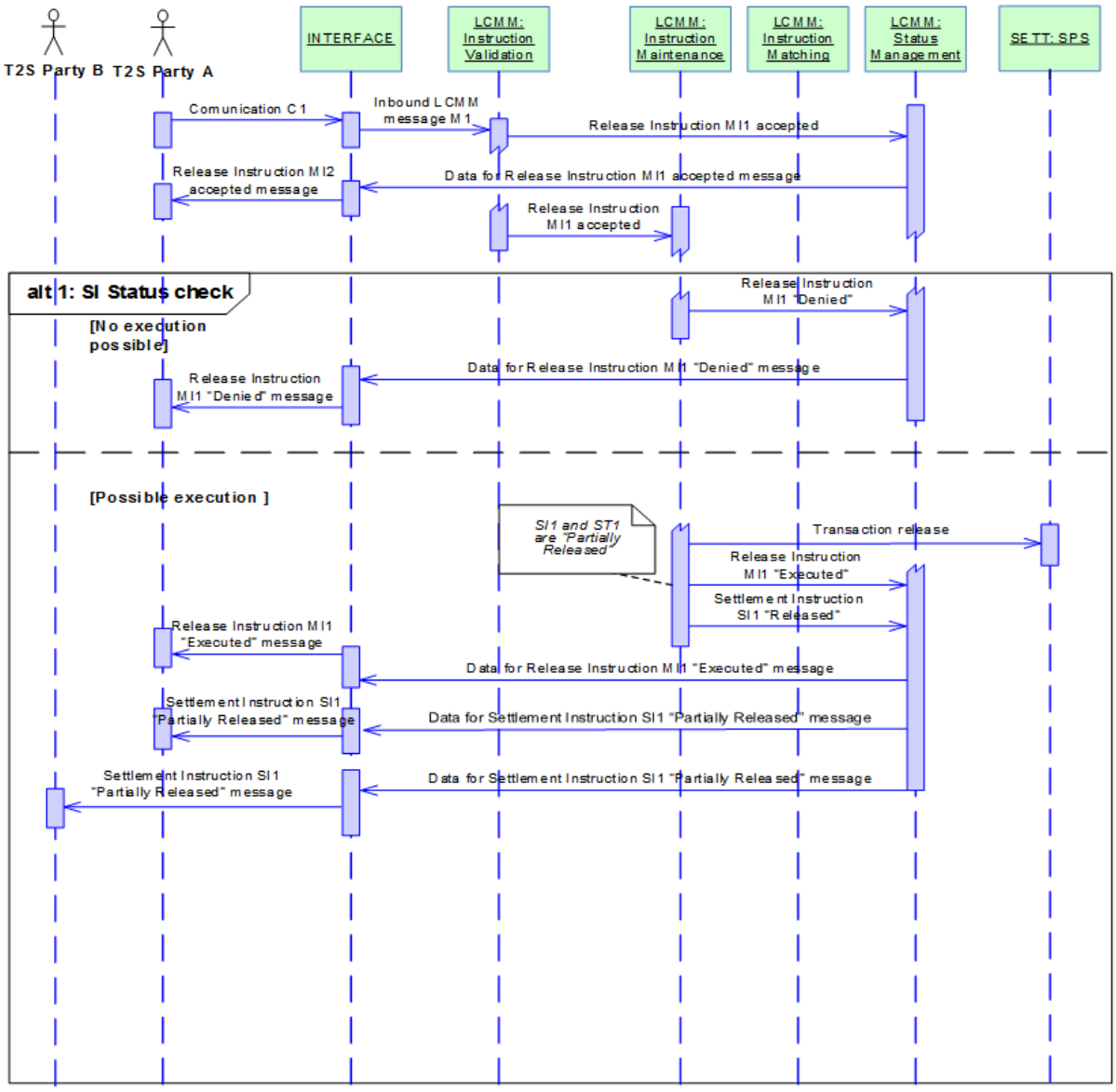
All rights reserved.

- A *Transaction Release* flow is sent to the *Standardisation and Preparation to Settlement* module of the *Settlement* domain;
- *Status Management* module is informed on the status of the Release Instruction MI1 and *Settlement Instruction* SI1 to be reported to the T2S *Party* A, via the *Interface* domain, depending on the *Party*'s message subscription preferences. T2S *Party* B is also informed, via the *Interface* domain, that SI1 has been released.

c) Release Instruction from a T2S Party to initiate a partial release process

"Release Instruction" refers to the entity Hold/Release Instruction of the data model, with Hold Indicator at "No".

DIAGRAM 23 - RELEASE INSTRUCTION TO INITIATE A PARTIAL RELEASE PROCESS



Business assumption

All rights reserved.

This description relates to a Communication C1 which contains a Release Instruction MI1 communicated by a T2S Party A, requesting to partially release an already existing On Hold Settlement Instruction SI1 (Intra-CSD instruction). The instruction is not submitted under Four-Eyes principle.

Processing

Once the format and syntax validations for Communication C1 are successfully passed, the Inbound LCMM Message M1 enters the Instruction Validation module which validates it. When it is successfully validated, the Release Instruction MI1 is created, and the Status Management module is informed of the acceptance of Release Instruction MI1 to be reported to the T2S Party A, via the Interface domain (depending on the message subscription preferences of the Party).

The Release Instruction MI1 is then routed to the Instruction Maintenance module. This module performs the following checks in order to identify if the Release Instruction can be processed:

- The Settlement Status of the referenced Settlement Instruction should not be "Settled";
- The Cancellation Status of the referenced Settlement Instruction should not be "Cancelled";
- The CoSD Hold Status of the referenced Settlement Instruction must be "No";
- The Party Hold Status of the referenced Settlement Instructions should be "Yes";
- The Approval Status of the referenced Settlement Instruction should not be "Awaiting Approval";
- The referenced Settlement Instruction is not already partially released or there is no pending Release Instruction of the same Hold type in the system.
- The referenced Settlement Instruction is matched
- No other holds apply either on the referenced Settlement Instruction or on the counterpart.
- None of the related realignment Settlement Instruction is on Hold
- Both the referenced Settlement Instruction and its counterpart allow partial settlement.
- No settlement link applies either on the referenced Settlement Instruction or on its counterpart.
- No non-reciprocal link applies either on the referenced Settlement Instruction or on its counterpart.
- The related cut-off is not reached
- The released quantity complies with partial settlement rules

Alternative 1 – SI Status check: No execution possible (e.g. the Settlement Instruction is already cancelled)

If the Settlement Instruction SI1 Cancellation Status value is "Cancelled", then the Partial Release Instruction cannot be executed and Settlement Instruction SI1 is not partially released.

- The Cancellation Status of the Release Instruction MI1 is set to "Denied" with Reason Code Reference "Affected Settlement Instruction already cancelled";
- The referenced Settlement Instruction SI1 is not modified;
- Status Management module is informed on the status of the Release Instruction MI1 to be reported to T2S Party A, via the Interface domain, depending on its message subscription preferences.

Alternative 1 – SI Status check: Possible execution

All rights reserved.

The Settlement Instruction SI1 fulfils the conditions described above then the Release Instruction can be executed,

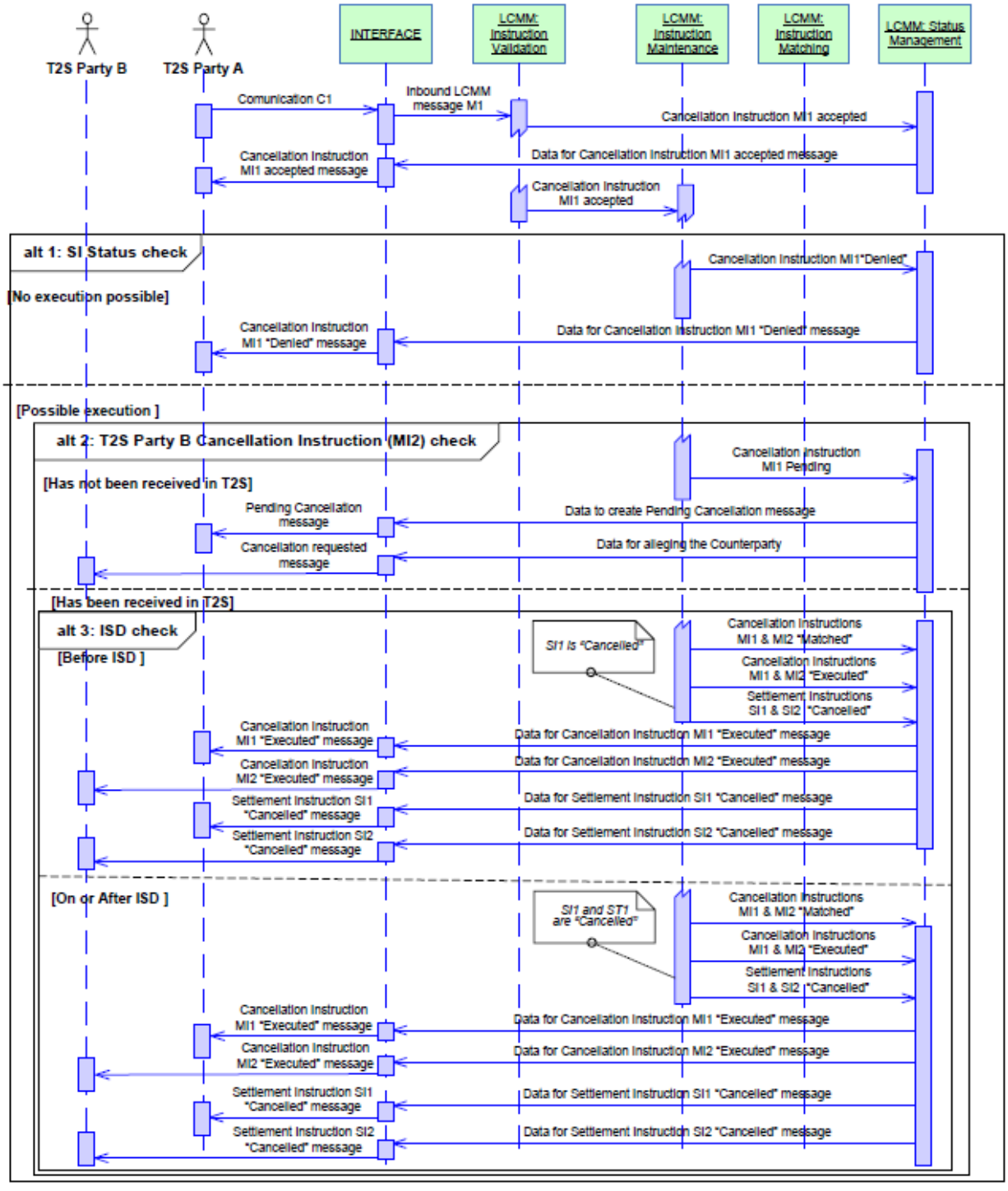
- The Execution Status of the Release Instruction MI1 is set to "Executed";
- The referenced Settlement Instruction SI1 and the Settlement Transaction ST1 remains on hold and only the quantity informed in the Release Instruction is released;
- A Transaction Release flow is sent to the Standardisation and Preparation to Settlement module of the Settlement domain;

Status Management module is informed on the status of the Release Instruction MI1 and Settlement Instruction SI1 to be reported to the T2S Party A, via the Interface domain, depending on the Party's message subscription preferences. T2S Party B is also informed, via the Interface domain, that SI1 has been partially released.

All rights reserved.

3.4.11.4 Processing of UC-MI-3: Cancellation Instruction

DIAGRAM 24 - CANCELLATION INSTRUCTION



Business assumption

- This description relates to the Communication C1 of a *Cancellation Instruction* MI1, communicated by a T2S *Party* A, requesting the cancellation of a previously sent *Settlement Instruction* SI1 (intra-CSD instruction).

All rights reserved.

- *Settlement Instruction* SI1 has been matched in T2S with *Settlement Instruction* SI2 (sent by T2S *Party* B). As both *Settlement Instructions* are matched, two *Cancellation Instructions* are required for their cancellation ("binding matching").

Processing

Once the format and syntax validations for Communication C1 are successfully passed, the *Inbound LCMM Message* M1 enters the *Instruction Validation* module which validates it. When it is successfully validated, the *Cancellation Instruction* MI1 is created, and the *Status Management* module is informed on the acceptance of *Cancellation Instruction* MI1 to be reported to the T2S *Party* A, via the *Interface* domain, depending on the message subscription preferences of the *Party*.

The *Cancellation Instruction* MI1 is then routed to the *Instruction Maintenance* module. This module performs the following checks in order to identify if the *Cancellation Instruction* can be executed:

- The Settlement Status of the referenced *Settlement Instruction* SI1 should not be *"Settled"*;
- The Cancellation Status of the referenced *Settlement Instruction* SI1 should not be "*Cancelled"* ;
- There is no *Cancellation Instruction* already sent by the same *Party* on the same referenced *Settlement Instruction*;
- The referenced *Settlement Instruction* should not be identified as "CoSD.;
- The Approval Status of the referenced *Settlement Instruction* SI1 should not be "*Awaiting Approval"*.

Alternative 1 – SI Status Check: No execution possible (e.g. the Settlement Instruction CoSD Hold Status set to "Yes")

If the CoSD Hold Status of *Settlement Instruction* SI1 is *"Yes"*, then the *Cancellation Instruction* cannot be executed:

- The Cancellation Status of the Cancellation Instruction MI1 is set to *"Denied"* with Reason Code Reference *"Referenced Settlement Instruction identified as CoSD on Hold"*;
- The referenced *Settlement Instruction* SI1 is not cancelled;
- The *Status Management* module is informed by the *Maintenance Instruction* module of the status of the *Cancellation Instruction* MI1 to be reported to *T2S Party* A via the *Interface* domain (depending on the T2S *Party*'s message subscription preferences).

Alternative 1 - SI Status check: Possible execution

The *Settlement Instruction* SI1 fulfils the conditions described above, but the cancellation can only take place if the counterparty *Cancellation Instruction* MI2 has been received.

Alternative 2 - T2S Party B Cancellation Instruction (MI2) check: has not been received in T2S

- The Instruction Maintenance module sets the Reason Code Reference of the *Cancellation Instruction MI1* to "Cancellation pending, awaiting cancellation from the counterparty";
- *The Status Management* module collects the relevant data and forwards it, via the *Interface* domain, to send a Pending Cancellation message to T2S *Party* A informing the latter that the counterparty's cancellation is missing, without informing of any status change;

All rights reserved.

- *The Status Management* module also collects the relevant data and forwards it, via the *Interface* domain, to allege to T2S *Party* B informing the latter that the counterparty has requested the cancellation of the *Settlement Instruction* SI1;

Note: If a Cancellation Instruction remains "Unmatched" (i.e. the Cancellation Instruction from the counterparty is not received) for 20 working days, this Cancellation Instruction is denied.

Alternative 2 – T2S Party B Cancellation Instruction (MI2) check: has been received in T2S

The *Settlement Instruction* SI1 fulfils the conditions described above, then the Cancellation Instruction can be executed but depending whether the Intended Settlement Date has been reached or not, *Instruction Maintenance* module performs this execution differently.

Alternative 3 - ISD check: Before the ISD

- The Match Status of both *Cancellation Instructions* MI1 and MI2 is set to *"Matched"*;
- The Execution Status of both *Cancellation Instructions* MI1 and MI2 is set to *"Executed"*;
- The Cancellation Status of the *Settlement Instructions* SI1 and SI2 and of the T2S generated Settlement Instruction(s) for realignment purposes (if any), is set to *"Cancelled"*;
- The *Status Management* module collects the relevant data and forwards them, via the *Interface* domain, to inform each T2S *Party* A and B about the status of its respective *Cancellation Instruction* MI1 or MI2;
- The *Status Management* module collects the relevant data and forwards them, via the *Interface* domain, to inform each T2S *Party* A and B about the status of its respective *Settlement Instruction* SI1 or SI2.

Alternative 3 - ISD check: After the ISD

- The Match Status of both *Cancellation Instructions* MI1 and MI2 is set to *"Matched"*;
- The Execution Status of both *Cancellation Instructions* MI1 and MI2 is set to *"Executed"*;
- The Cancellation Status of the *Settlement Instructions* SI1 and SI2 and of the T2S generated Settlement Instruction(s) for realignment purposes (if any), is set to *"Cancelled"*;
- The *Settlement Transaction* of the cancelled Settlement Instruction, and the *Settlement Transactions of the T2S Generated Realignment Instructions* (if any) are cancelled;
- The *Status Management* module collects the relevant data and forwards them, via the *Interface* domain, to inform each T2S *Party* A and B about the status of its respective *Cancellation Instruction* MI1 or MI2;
- The *Status Management* module collects the relevant data and forwards them, via the *Interface* domain, to inform each T2S *Party* A and B about the status of its respective *Settlement Instruction* SI1 or SI2.

All rights reserved.

3.5 Settlement

3.5.1 General Introduction

The modules belonging to the *Settlement* domain are in charge of the settlement both on a continuous basis during the daytime and in sequences [131] grouped into a predefined number of cycles during the night-time. Their objectives are to maximise the volume and value of settlement with the available securities and cash resources, and to minimise the time for settlement.

The *Settlement* domain comprises five modules:

- The *Standardisation and Preparation to Settlement* module (SPS);
- The *Daytime Validation, Provisioning and Booking* module (VPB);
- The *Daytime Recycling and Optimisation* module (R&O);
- The *Night-time Settlement module* (NTS);
- The *Auto-collateralisation* module (ACO).

The *Standardisation and Preparation to Settlement* module prepares the settlement of *Matching Objects* and the related *Settlement Instructions, Settlement Restrictions* or liquidity transfers. It creates *Settlement Transactions* in a standard format, taking notably into account Cross-CSD settlements and in/out T2S settlements [132].

In case of realignment deriving from matched *Settlement Instructions,* complementary T2S generated *Settlement Instructions* and related *Matching Objects* are created in this module.

If the intended settlement date is reached, the *Settlement Instructions* are analysed to check if CoSD rules previously defined in the T2S static data have to be applied. In this case, the *Settlement Instructions* are put on hold and the involved administering parties are informed per CoSD condition. Additional *Settlement Restrictions* are created to set up the related CoSD Blocking in order to block the resources needed.

Settlement transactions are then created deriving from *Settlement Instructions, Settlement Restrictions* or liquidity transfers.

When the created *Settlement Transactions* are eligible in respect to eligibility conditions such as links, hold settlement transaction, cut-off time, they are either submitted for immediate settlement during daytime or made available for settlement into sequences during the night-time.

In case of maintenance instructions (including hold, cancellation or updates of processing indicators on the settlement instructions) the *Settlement Transactions* are updated directly from *Lifecycle Management and Matching*. Where, following an instruction release, the *Standardisation and Preparation to Settlement* module releases the *Settlement Transactions* when triggered accordingly by the *Lifecycle Management and Matching* domain. Lastly, this module handles the cut-off, suspending the *Settlement Transactions* to prevent them from any further settlement attempt, as well as the EOD/SOD processing.

---

131 Gathering of *Settlement Transactions* according to their type (liquidity transaction, FOP, CB operation …).

132 As per URD definition Cross-CSD settlements are settlement where both the buyer and seller CSDs, as well as the Issuer CSD, are participating in T2S. In/out T2S settlements are settlements where at least one settling party or the Issuer CSD is not participating in T2S.

All rights reserved.

In the daytime settlement period, the *Settlement Transactions* are submitted to a settlement attempt through the *Daytime Validation, Provisioning and Booking* module and then submitted to the *Daytime Recycling and Optimisation* module. On successful settlement, the process comes to a logical conclusion. In case of failure to settle, the *Settlement Transactions* are submitted to the *Daytime Recycling and Optimisation* module.

In the night-time settlement period, the *Night-time Settlement* module handles the settlement of all created and remaining unsettled settlement transactions.

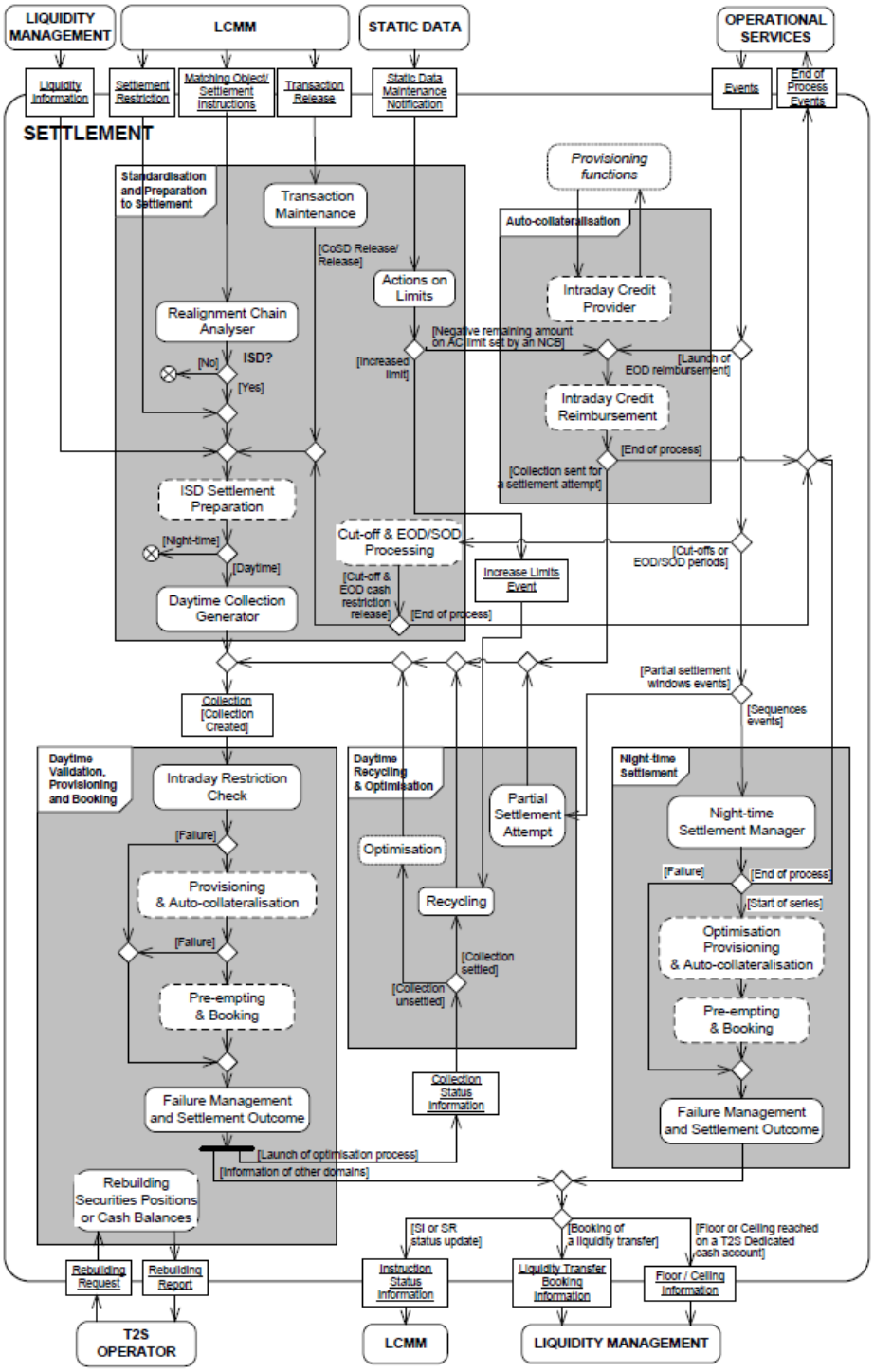
The *Settlement Transactions* are grouped into sequences, depending on the *Settlement Transaction* type (liquidity transaction, FOP, CB operation …). The selected *Settlement Transactions* are then submitted, per sequence, to optimisation and provisioning processes that identify the *Settlement Transactions* which can be booked with success. The remaining unsettled *Settlement Transactions* are then recycled into the next sequence.

Both for daytime and night-time:

- The provisioning process may call the *Auto-collateralisation* module for resolving:
  - the lacks of cash or securities through auto-collateralisation with a central bank (also referred as CB collateralisation in the description of the *Settlement* domain),
  - the insufficient external guarantee headroom and lack of securities (for a client of payment/settlement bank) through auto-collateralisation with a payment/settlement bank (also referred as client-collateralisation in the description of the *Settlement* domain);
- Partial settlement procedures are used if requested in appropriate conditions and windows;

All rights reserved.

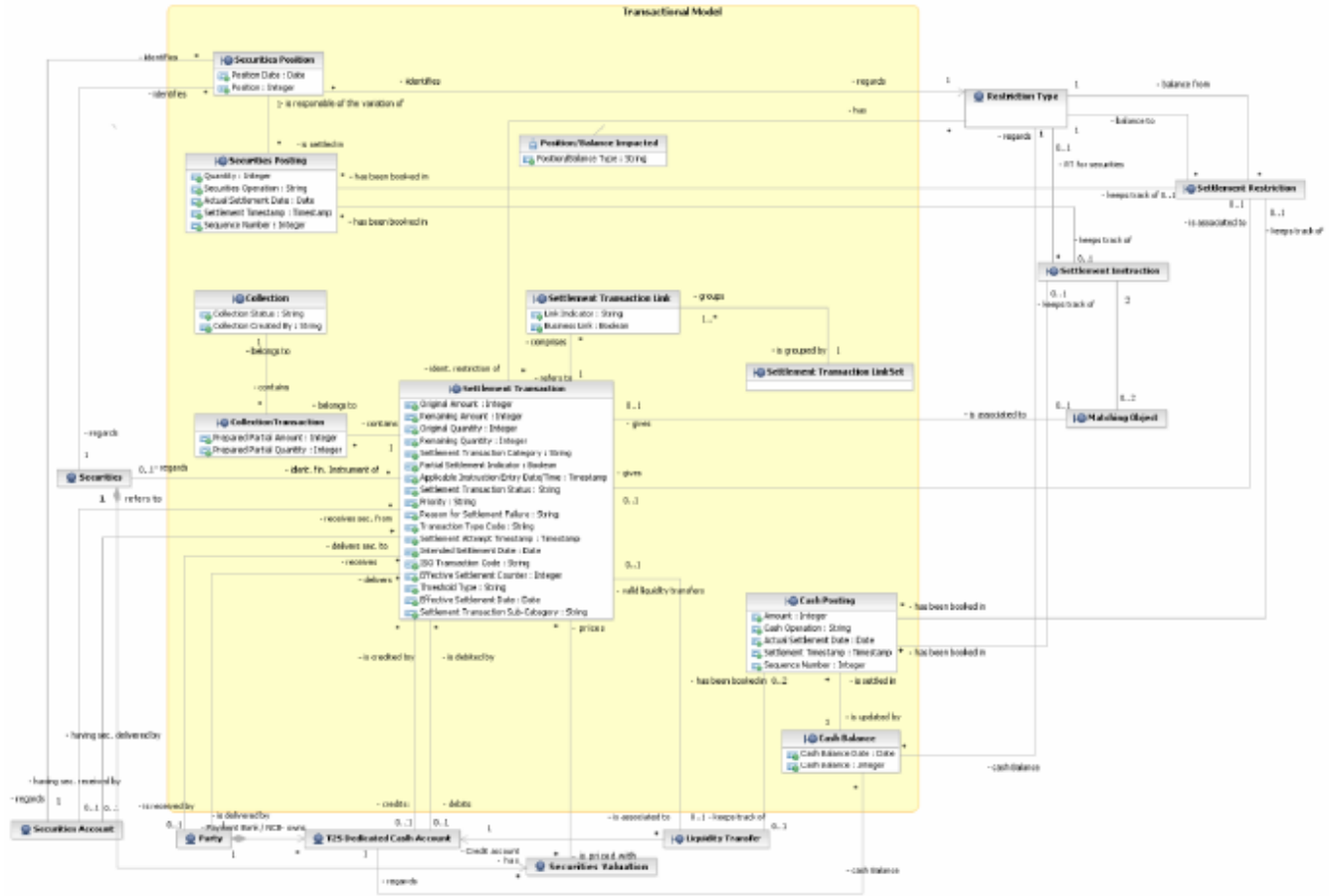
- The result of every settlement attempt is reported, either to the *Lifecycle Management and Matching* domain, when related to *Settlement Instructions* and *Settlement Restrictions,* or to the *Liquidity Management* domain, when related to liquidity transfers.



All rights reserved.

3.5.2 Dynamic data managed by the domain

3.5.2.1 Diagram of the dynamic data related to settlement



3.5.2.2 List of dynamic data managed by the settlement domain

Dynamic data are shared with the *Lifecycle Management and Matching* and the *Liquidity Management* domains.

The *Settlement* domain is managing data related to:

- *Settlement Transaction, Collection*;
- *Securities Position* and *Cash Balances*, including the management of restrictions (earmarking, blocking, reservation and CoSD blocking);
- *Limits Utilisation* and their journaling.

3.5.2.3 Description of the data related to Settlement Transactions

Settlement Transaction definition

The *Settlement Transaction* is designed in the Data Model as an entity optimised for the settlement process, i.e.:

- It contains the main data requested for the settlement (debited and credited accounts, original, partial and remaining quantities or amounts, priority, ISO transaction code, security, restriction type, threshold type, status…);
- It is the single and common entity that contains these data for their processing in the *Settlement* domain should they relate to *Settlement Instructions,* restriction set up or use, liquidity transfer orders or auto-collateralisation.

On the basis of *Settlement Instructions, Settlement Transactions* are created within the *Settlement* domain by the *Standardisation and Preparation to Settlement* module on the intended settlement date.

In addition, *Settlement Transactions* are also created in case of liquidity transfer orders, set up or use of restrictions and set up or reimbursement of CB collateralisation or client-collateralisation.

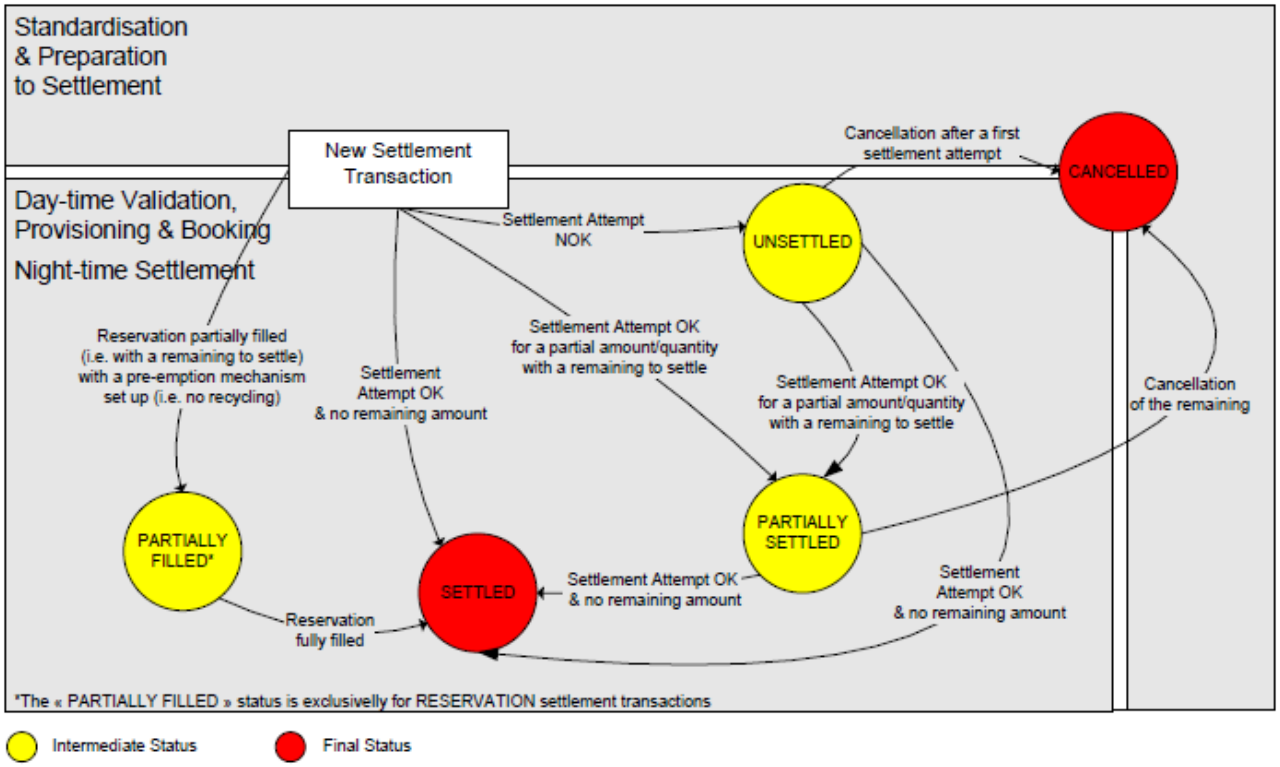
The *Settlement Transaction* entity contains:

- General information: Original Amount and/or Original Quantity, references of the delivering T2S Party and receiving T2S Party, debiting accounts and crediting accounts, *Matching Object* or *Settlement Restriction* or *Liquidity Transfer*, *Securities*, *Restriction Types*;
- Settlement information: Transaction Type Code, ISO Transaction Code, Settlement Transaction Category, Settlement Transaction Sub-Category, Settlement Transaction Status, Remaining Amount/Remaining Quantity and partial amount/quantity, Priority ("*Reserved Priority*", "*Top Priority*", "*High Priority*", "*Normal Priority*"), Intended Settlement Date, Settlement Attempt Timestamp, Threshold Type ("Cash value" or "Quantity");
- Optimisation information: Partial Settlement Indicator, Reason For Settlement Failure, Applicable Instruction Entry Date/Time, Effective Settlement Counter;
- Restriction references: restriction references given by the T2S parties for the execution of their settlement instructions.

All the accounts are linked to specific securities positions or cash balances of the accounts (see the description of the restrictions for details regarding the management of restricted positions within the accounts).

All rights reserved.

Settlement Transaction lifecycle



STATUS	DEFINITION	FROM MODULE	STATUS
Unsettled	Provisioning not successful, needs to be recycled by Day-time Recycling and Optimisation module [133]	VPB / NTS	R&O Status Management
Partially filled	Exclusively for reservation partially filled with remaining quantity/ amount to settle	VPB / NTS	Status Management
Settled	End status	VPB / NTS	Status Management
Partially Settled	Settlement OK with remaining amount/quantity to settle or new settlement attempt	VPB / NTS	Status Management
Cancelled	End status	SPS VPB / NTS	Status Management

In addition to these statuses, the *Settlement Transactions* can be set [134] as:

133 Transactions related to restrictions and *Liquidity Transfers* are not recycled.

134 The implementation phase shall identify the means to set *Settlement Transactions* as hold or suspended ones.

All rights reserved.

- Hold settlement transactions: which translates the fact that at least one of the "Hold" statuses (from T2S Party request, a CSD request or in case of detected CoSD) of the *Settlement Instruction* is set to "On hold";
- Suspended *Settlement Transactions* which are used in all other cases where the *Settlement Transaction* has to be excluded temporarily from the settlement processes (e.g. missing linked *Settlement Transaction* at the intended settlement date, cut-off reached).
- Partially Released Settlement Transactions: which translates the fact that the Settlement Transaction may be eligible for settlement up to its Released quantity.

These statuses are only internal statuses managed within the *Settlement* domain.

Relations between settlement instructions, settlement restrictions, liquidity transfers and settlement transactions

Settlement Instructions and Settlement Restrictions are created:

- Either inside the *Lifecycle Management and Matching* domain on the basis of the settlement messages sent by the T2S Actors;
- Or inside the *Settlement* domain, through the call of a *Lifecycle Management and Matching* domain service, as T2S generated *Settlement Instruction* for auto-collateralisation, CoSD or realignment purposes.

Liquidity Transfers are created in the *Liquidity Management* domain which sends liquidity information to the *Settlement* domain for their booking. *Liquidity Transfers* may be also created inside the *Settlement* domain, through the call of a *Liquidity Management* domain service, when transfers of liquidity between *T2S Dedicated Cash Accounts* are requested in the settlement process (e.g for corporate action or monetary policy operation rebalancing liquidity, rebalancing between *T2S Dedicated Cash Accounts* for EOD CB collateralisation reimbursement).

A Settlement transaction is created to process the settlement of *Settlement Instructions* related to a matching object, *Settlement Restrictions* and liquidity transfers.

Rules for updating the statuses of settlement instructions and Settlement Transactions

The update of the statuses and associated reason code of a *Settlement Instruction* within the *Settlement* domain translates:

- The ineligibility to settlement (due to missing linked settlement transaction, hold instruction or a cut-off reached) observed in the *Standardisation and Preparation to Settlement* module;
- Or the result of a settlement attempt performed in the *Daytime Validation, Provisioning and Booking* module or in the *Night-time Settlement* module.

The status value of the *Settlement Transaction* and the settlement status of the *Settlement Restrictions* are set simultaneously to "*Unsettled*", "*Partially Filled*" (Settlement Transaction) / *"Partially Settled"* (Settlement Restriction) or "*Settled*", in order to document the settlement attempt result.

This choice has been made to ensure that these statuses are managed through a single and consistent update by the domain in charge of the settlement actions leading to these statuses.

Links between settlement transactions

All rights reserved.

Settlement Transactions can be linked according to:

- The link between the underlying Settlement Instructions, set by T2S parties (external link: "With", "After", "Before");
- The T2S generated link in case of CoSD, cross-CSD, restriction and CB collateralisation /client-collateralisation.

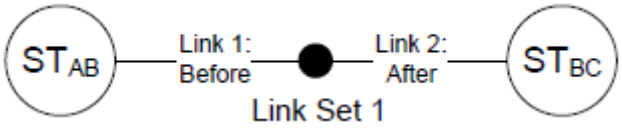
These links are taken into account when building the collection of *Settlement Transactions* for the *Daytime Validation, Provisioning and Booking* module or in the *Night-time Settlement* module that settles the collection on an "all or none" basis (or in due sequence for "After"/"Before" links).

The links between *Settlement Transactions* are managed with two entities:

- The *Settlement Transaction Link* entity which specifies the links associated to a Settlement Transaction;
- The *Settlement Transaction LinkSet* entity which stores a set of *Settlement Transaction Links* [135].

For example, if a *Settlement Transaction* AB (STAB) is linked to *Settlement Transaction* BC (STBC) through a before – after link: one link set (link Set 1) and two links (link 1: Before; Link 2: After) are created.

Link 1 and link 2 are associated to link set 1, as represented below:



Collection Definition

A *Collection* is a set of (linked) settlement transactions, as defined above, submitted together for a settlement attempt in the *Daytime Validation, Provisioning and Booking* module or in the *Night-time Settlement* module.

A *Collection* can contain from 0 [136] to many settlement transactions.

A *Settlement Transaction* may belong to 0 [137] to many collections.

The *Collection* entity contains the following attributes:

- The Collection Status;
- The Collection Created By indicating which function created the collection.

The *Collection* entity is linked to the *Securities Provisioning Net Flow, Cash Provisioning Net Flow* and *CMB Provisioning Net Flow* entities in order to store the sum, per *Securities Position* (identified by restriction type, securities account and securities) or *Cash Balance* (identified by restriction type and T2S Dedicated cash account), of credit and debit resulting from the *Settlement Transactions* that are part of a collection.

Collection lifecycle [138]

---

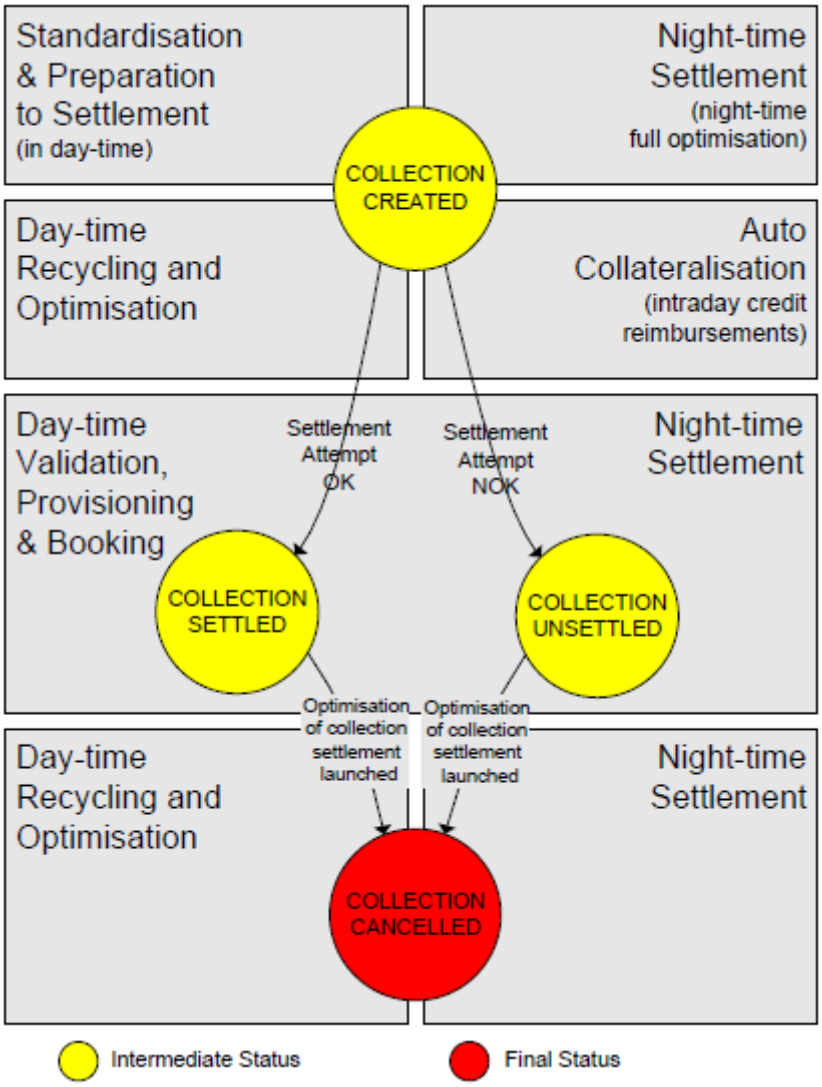
135 A set of *Settlement Transaction* links is a group of *Settlement Transaction* links.

136 During the optimisation process a collection can be empty as a result of a de-selection of settlement transactions.

137 When created, the *Settlement Transaction* does not belong to any collection

138 The status and associated lifecycle for *Collection* are used only to ease the functional description. The status shall not be implemented as an attribute of the entity at the implementation phase.

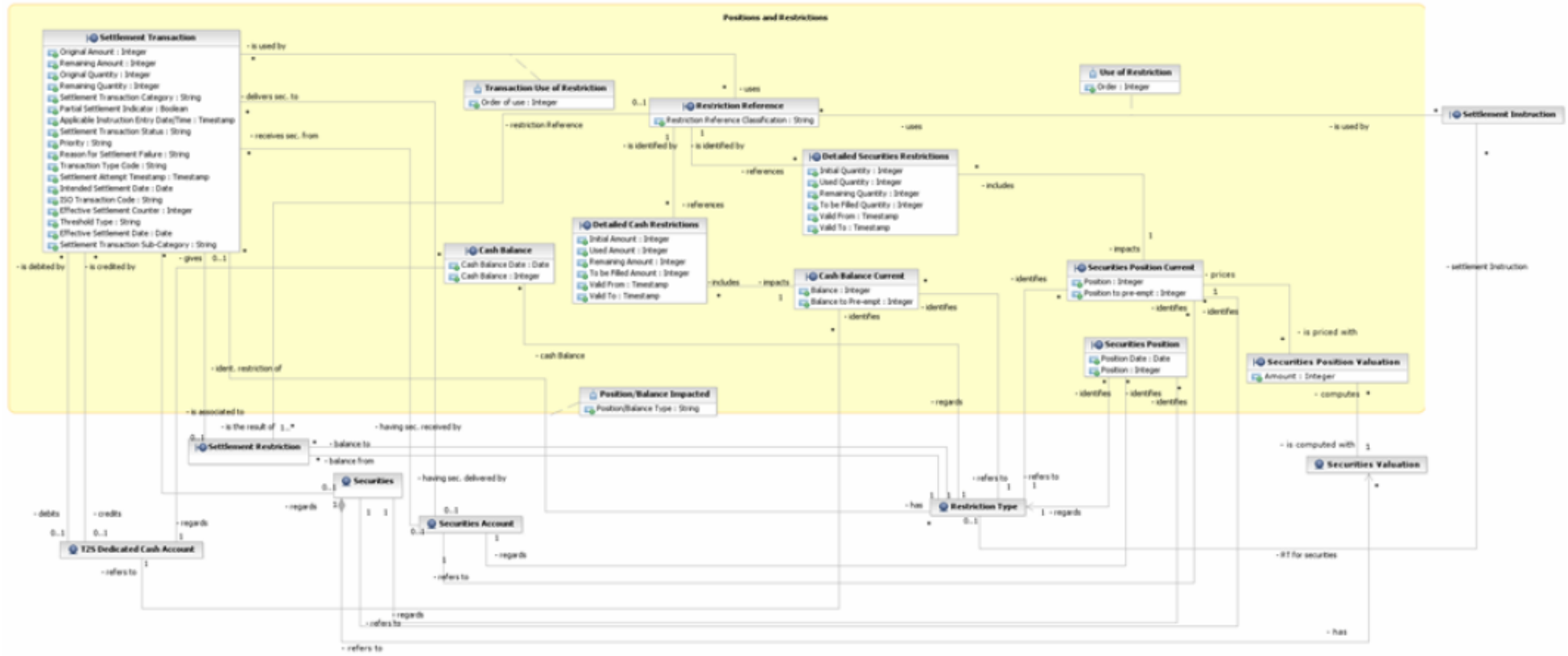
All rights reserved.

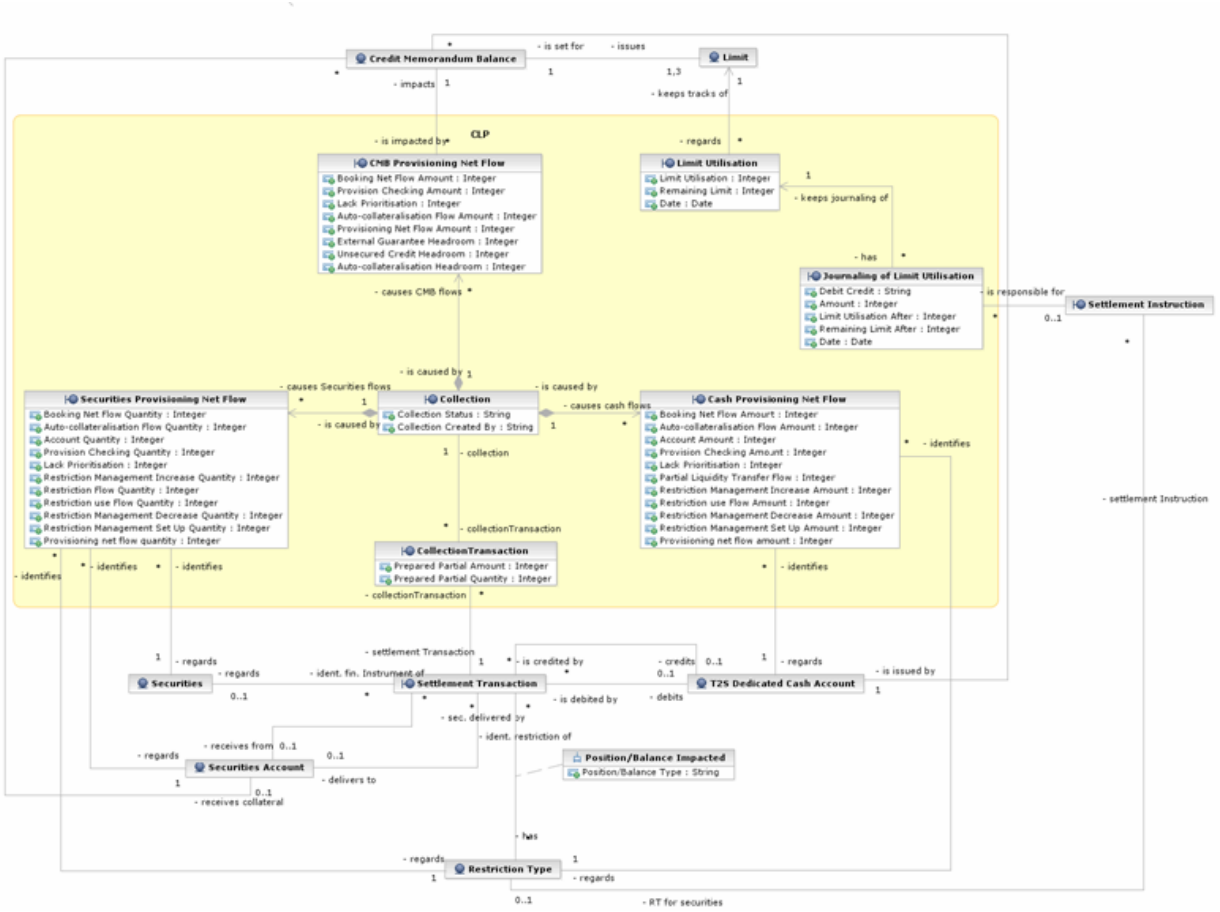


STATUS	DEFINITION	FROM MODULE (SET BY)	TO MODULE (USED BY)
Collection Created	First status	SPS	VPB
		Night-time Settlement	
		Auto-collateralisation	
		R&O	
Collection Settled	Settlement of the collection succeeded	VPB	R&O
		Night-time Settlement	Night-time Settlement
Collection Unsettled	Settlement of the collection failed	VPB	R&O
		Night-time Settlement	Night-time Settlement
Collection Cancelled	Final status	R&O	R&O
		Night-time Settlement	Night-time Settlement

All rights reserved.

3.5.2.4 Diagrams of the dynamic data related to limits and restrictions




Credit Memorandum Balance
- is set for
- issues
Limit
- impacts
1
1,3
- keeps tracks of
CLP
- is impacted by
- regards
CMB Provisioning Net Flow
Booking Net Flow Amount : Integer
Provision Checking Amount : Integer
Lack Prioritisation : Integer
Auto-collateralisation Flow Amount : Integer
Provisioning Net Flow Amount : Integer
External Guarantee Headroom : Integer
Unsecured Credit Headroom : Integer
Auto-collateralisation Headroom : Integer
Limit Utilisation
Limit Utilisation : Integer
Remaining Limit : Integer
Date : Date
- keeps journaling of
- causes CMB flows
- has
Journaling of Limit Utilisation
Debit Credit : String
Amount : Integer
Limit Utilisation After : Integer
Remaining Limit After : Integer
Date : Date
- is responsible for
Settlement Instruction
- is caused by
- causes Securities flows
- causes cash flows
Securities Provisioning Net Flow
Booking Net Flow Quantity : Integer
Auto-collateralisation Flow Quantity : Integer
Account Quantity : Integer
Provision Checking Quantity : Integer
Lack Prioritisation : Integer
Restriction Management Increase Quantity : Integer
Restriction Flow Quantity : Integer
Restriction use Flow Quantity : Integer
Restriction Management Decrease Quantity : Integer
Restriction Management Set Up Quantity : Integer
Provisioning net flow quantity : Integer
Collection
Collection Status : String
Collection Created By : String
Cash Provisioning Net Flow
Booking Net Flow Amount : Integer
Auto-collateralisation Flow Amount : Integer
Account Amount : Integer
Provision Checking Amount : Integer
Lack Prioritisation : Integer
Partial Liquidity Transfer Flow : Integer
Restriction Management Increase Amount : Integer
Restriction use Flow Amount : Integer
Restriction Management Decrease Amount : Integer
Restriction Management Set Up Amount : Integer
Provisioning net flow amount : Integer
- collection
- collectionTransaction
- identifies
CollectionTransaction
Prepared Partial Amount : Integer
Prepared Partial Quantity : Integer
- settlement Instruction
- settlement Transaction
- is credited by
- credits
- regards
- is issued by
Securities
- ident. fin. Instrument of
Settlement Transaction
T2S Dedicated Cash Account
- is debited by
- debits
- sec. delivered by
- ident. restriction of
- receives from
Securities Account
- delivers to
- receives collateral
Position/Balance Impacted
Position/Balance Type : String
- has
Restriction Type
- RT for securities

3.5.2.5 Description of the data related to limits and restrictions

Securities Position Definition

The *Securities Position* stores the position in a security for a securities account and a restriction type at a specific date. One instance is created for each day for which a movement has been settled (no instance of Securities Position is created, if no movement is booked on the considered *Securities Position* for a day). During the same settlement date, the existing record is updated. A *Securities Position* exists per Securities Account/ISIN/Restriction Type.

Securities Position Current Definition

The *Securities Position* Current corresponds to the current value of the *Securities Position* entity in a way to optimize the settlement process.

Cash Balance Definition

The entity *Cash Balance* stores the amount of cash held on the *T2S Dedicated Cash Account* for a given restriction type at a specific point in time. T2S updates the position each time a settlement occurs. T2S stores end-of-day balances for previous business days and the latest balance for the current business day.

Cash Balance Current Definition

The entity *Cash Balance Current* stores the last balance for a *T2S Dedicated Cash Account*.

Securities Position Valuation Definition

The entity *Securities Position Valuation* stores valuations of an earmarked securities position.

3.5.2.6 Description of the data related to net flows

Securities Provisioning Net Flows Definition

The entity *Securities Provisioning Net Flows* stores the net flows calculated by technical netting during the provisioning for each securities position involved in an underlying collection of Settlement Transactions

Cash Provisioning Net Flows Definition

The entity *Cash Provisioning Net Flows* stores the net flows calculated by technical netting during the provisioning for each cash balance involved in an underlying collection of Settlement Transactions.

CMB Provisioning Net Flows Definition

This entity *CMB Provisioning Net Flows* stores the net flows calculated by technical netting during the provision check of a collection of settlement transactions, for each credit memorandum balance involved.

3.5.2.7 Description of the data related to limits

Limits data can be divided in two categories of entities:

- *Limit* which are defined and maintained by CBs and payment/settlement banks via static data updates;
- *Limit Utilisation* and their journaling which are managed by the *Settlement* domain.

Limits management principle

Relations between *T2S Dedicated Cash Accounts* and securities accounts for securities settlement are set through the *Credit Memorandum Balance* entity.

All rights reserved.

To allow its clients to use its *T2S Dedicated Cash Accounts* for their securities settlement, the payment/settlement bank sets credit memorandum balance which applies to:

- One of its *T2S Dedicated Cash Accounts*;
- And one authorised BIC of the client of payment/settlement bank.

By setting limits to each credit memorandum balance, the payment/settlement bank manages the capacity of its clients to use its T2S Dedicated cash account.

Through this credit memorandum balance, T2S Parties related to the authorised BIC (i.e. the payment/settlement bank clients) can set a relation between one or several of their securities accounts and the considered T2S Dedicated cash account. Limits set on the considered credit memorandum balance then automatically apply to all securities settlement on the considered *T2S Dedicated Cash Account* and securities account. Those limits are then checked during the provisioning and the auto-collateralisation.

For securities settlements of the payment/settlement bank itself (i.e. when the T2S parties which own the involved *T2S Dedicated Cash Account* and securities account are related to the same BIC), the capacity to use the *T2S Dedicated Cash Account* cannot be limited by any limit. Only the auto-collateralisation limit set by the CB is associated to the initial credit memorandum balance automatically created at the creation of the *T2S Dedicated Cash Account*.

Since a *T2S Dedicated Cash Account* is related to a single currency, all limits related to one of its associated credit memorandum balances apply to the same currency than the related T2S Dedicated cash account.

Limit types for the relation between the payment/settlement bank and its client

For credit memorandum balance related to payment/settlement bank's clients (i.e. when the T2S Parties which own the *T2S Dedicated Cash Account* and Securities Accounts are related to different primary BICs), three limits are set:

- The external guarantee limit defines the capacity to use a *T2S Dedicated Cash Account* without guarantee in T2S before the performance of any client-collateralisation;
- The auto-collateralisation limit set by the payment bank/settlement bank defines the capacity to use a *T2S Dedicated Cash Account* once the external guarantee headroom is fully used. This capacity is submitted to the capacity of the client to provide the collateral corresponding to the used amount. It is done through a client-collateralisation operation automatically generated by T2S during the provisioning;
- The unsecured credit limit defines the capacity to use a *T2S Dedicated Cash Account* once both external guarantee headroom and the headroom of the auto-collateralisation set by the payment bank/ settlement bank are fully used. As for the external guarantee limit, this capacity is not submitted to the provision of a guarantee in T2S.

The external guarantee headroom is checked during the provisioning in the *Daytime Validation, Provisioning and Booking* module. If the headroom is not sufficient to fully settle the underlying settlement transaction(s), a client-collateralisation is attempted in the *Auto-collateralisation* module for the missing amount. Lastly, if the headroom of the auto-collateralisation limit set by the payment bank/settlement bank and the client's collateral capacity are not sufficient, the unsecured credit headroom could be used in complement.

All rights reserved.

All these limits can be set to zero (i.e. there is no capacity for the considered limit) or to infinite (i.e. the considered capacity is not limited).

The external guarantee headroom limit and unsecured credit headroom are decreased or increased at each securities settlement. The external guarantee headroom limit can be higher than the corresponding limit amount in case of credits higher than debits. The headroom of the auto-collateralisation limit set by the payment bank/settlement bank is decreased after the booking of the T2S generated collateral *Settlement Transaction* corresponding to the client-collateralisation.

Limit type set by Central Banks for auto-collateralisation

A unique credit memorandum balance related to BIC of the payment/settlement bank may be created after the creation of the T2S Dedicated cash account. The capacity to use the *T2S Dedicated Cash Account* for securities accounts associated to this credit memorandum balance is not limited by any limit. The provisioning then applies only on the available resources in the involved cash balance and securities position.

For this credit memorandum balance, only the Central Bank which opened the *T2S Dedicated Cash Account* has to set the amount of the auto-collateralisation limit set by a CB. This amount limits the capacity of the considered *T2S Dedicated Cash Account* to receive intraday credit in case of lack of cash. This limit is automatically created at the creation of the T2S Dedicated cash account. It is set by default to zero.

This limit is checked in the *Auto-collateralisation* module in case of lack of cash on the considered *T2S Dedicated Cash Account* and before any intraday credit provision. If the headroom is not sufficient, no intraday credit is provided at all. The lack of cash cannot be filled in and the provisioning is considered as failed.

The headroom of the auto-collateralisation limit set by a CB is decreased after the booking of the T2S generated collateral *Settlement Transaction* corresponding to the auto-collateralisation. It is increased after the booking of the corresponding reverse operation.

Limits Utilisation and journaling

The *Limit Utilisation* of a given limit allows checking that the remaining amount is sufficient at each settlement attempt being subject to this limit.

The *Limit Utilisation* entity contains:

- The Limit Utilisation which specifies the current amount of liquidity drawn down by the T2S Party and being subject to this limit;
- The Remaining Limit which specifies the current amount of liquidity available to the T2S Party and being subject to this limit;
- A Date which specifies the T2S settlement day to which the *Limit Utilisation* applies.

The *Journaling of Limit Utilisation* tracks, for audit trail purposes, each change in a T2S Party *Limit Utilisation* induced by a settlement instruction.

The *Journaling of Limit Utilisation* entity contains:

- An instruction reference of the *Settlement Instruction*, which updates, through the *Settlement Transaction*, the *Limit Utilisation*;
- A Debit Credit attribute which specifies the increasing or the decreasing of the Limit Utilisation;

All rights reserved.

- The Amount which specifies the amount of liquidity drawn down by the T2S Party;
- The Limit Utilisation After which specifies the amount of liquidity drawn down by the party for the settlement of securities transactions with credit provider;
- The Remaining Limit After which specifies the amount of liquidity available to the T2S Party;
- The Date to which T2S the *Limit Utilisation* applies.

3.5.2.8 Description of the data related to restrictions (reservation, blocking, CoSD blocking, earmarking)

Restriction types

In T2S several restrictions types can be set up:

- The blocking can be set up on a T2S Party, a security, a securities account, a T2S Dedicated cash account, a securities position or a cash balance;
- The reservation can be set up on a securities position or a cash balance;
- The earmarking can be set up at the securities account level or at securities positions level;
- The CoSD blocking is automatically set up on securities positions or cash balances in case of detection of a CoSD.

Except for the CoSD blocking which is internally generated by T2S in case of detection of a CoSD, the restrictions, such as blocking, reservation and earmarking can be set by the T2S Party, the CSD or the CB.

A restriction type is specified in T2S within the *Restriction Type* entity:

- By the T2S Operator for common purpose when it applies to every T2S Party whatever their CSD (e.g. deliverable holding, earmarking for auto-collateralisation services available in T2S or CoSD blocking);
- Or by a CSD for specific purpose when it applies only to the T2S Party of this CSD and their securities account and securities position;
- Or by a CB, when it applies to *T2S Dedicated Cash Accounts* and cash balances;
- Or by T2S Party, when it is reservation, blocking or earmarking of securities position and/or cash balances.

Intraday restriction

An *Intraday Restriction* is a blocking of a T2S Party/CSD/CB, a security, a currency, a securities account or a *T2S Dedicated Cash Account* which can be set up at any time of the settlement day. An *Intraday Restriction* may use some rules and parameters that modify their application scope.

It is handled within T2S as a static data update and is specified in the following entities: *Party Restriction*, *Security Restriction*, *Securities Account Restriction* or *T2S Dedicated Cash Account Restriction*.

Once an *Intraday Restriction* is set up, no booking is allowed on the *Settlement Transactions* impacted by this restriction (i.e. regarding one of the restricted T2S Party, security, currency or accounts). This is achieved with a check of the *Intraday Restriction* and its potential rules before any settlement attempt at the entry of the *Daytime Validation, Provisioning and Booking* module or at the selection of the *Settlement Transactions* in the *Night-time settlement* module.

All rights reserved.

Restrictions at the securities position or cash balance level

In T2S, securities positions are identified through the combination of three data:

- A *Securities Account* number;
- A *Securities* identifier;
- A *Restriction Type* identifier.

And cash balances are identified through the combination of two data [139]:

- A *T2S Dedicated Cash Account* number;
- A *Restriction Type* identifier.

By default, a *Securities Account* contains one (and only one) deliverable securities position and a *T2S Dedicated Cash Account* contains one (and only one) deliverable cash balance (in both cases having a Restriction Processing Type="*Deliverable*"). This deliverable securities position or cash balance contains the resource on which no restriction applies. This "*Deliverable*" Restriction Processing Type is used by default for *Settlement Instruction* with no indicated impacted securities position (except of earmarking at securities account level, see below).

In addition to the deliverable one, the *Securities Account* can be linked to a *Securities Position* for each *Restriction Type,* allowed by the CSD which manages it and a *T2S Dedicated Cash Account* can be linked to a *Cash Balance* for each *Restriction Type,* allowed by the CB which manages it [140].

These restricted securities positions or cash balances are automatically created during the booking of *Settlement Transactions* which credit them.

The *Settlement Transactions* which credit a restricted securities position or cash balance are related to:

- A *Settlement Restriction* for a blocked/reserved/earmarked securities position or for a blocked/reserved cash balance;
- A *Settlement Instruction* for earmarked securities position only;
- A T2S generated *Settlement Restriction* for a CoSD blocking on securities position or cash balance.

Detailed tracking of single blocking and reservation

A detailed tracking of each blocking and reservation step is ensured in addition to the split of securities position or cash balance per restriction type. An instance is created in the *Detailed Securities Restrictions* or *Detailed Cash Restrictions* entities at the booking of the *Settlement Transaction* related to the restriction. Each instance is referenced with a restriction reference.

This level of detail allows the individual tracking of the quantities/amounts already filled, to be filled with future additional pre-emption (only for reservation) or used.

---

139 The currency is not used in the combination to identify a *Cash Balance* since a *T2S Dedicated Cash Account* is only related to a single currency. In consequence, this information is not necessary at the *Cash Balance* level since all cash balances of a *T2S Dedicated Cash Account* are related to the same currency.

140 The check of the allowed restriction type for the considered account is done during the validation of the underlying *Settlement Instruction* or *Settlement Restriction*.

All rights reserved.

The total amount of blocked or reserved securities position or cash balance at a given point in time corresponds to the sum of the available quantity/amount in all the active *Detailed Securities Restrictions*/*Detailed Cash Restrictions* associated to the considered *Securities Position* or *Cash Balance* [141].

No detailed tracking is handled for the earmarking since it is identified with the restriction type, (see detailed description of earmarking management below).

Set up of a blocking or a reservation on the securities positions or on the cash balances

The set up of a blocking or a reservation is handled through a settlement restriction. It results in a movement between two securities positions of the same securities account and the same security (or two cash balances of the same T2S Dedicated cash account) but with a different restriction type.

The use of an existing blocking or an existing reservation to set up a new blocking or a new reservation is not possible. The securities position that is debited to set up a restriction can only be the deliverable one or an earmarked one. The cash balance that is debited to set up a restriction can only be the deliverable one.

Increase of an existing blocking or reservation on the securities positions or on the cash balances

A T2S Party can increase an existing blocking or reservation with the use of a *Settlement Restriction* that refers to a considered *Restriction Reference* previously generated by T2S and corresponding to the considered *Detailed Securities Restrictions*/*Detailed Cash Restrictions* to update.

The validity of a *Restriction Reference* is checked during the validation of the *Settlement Restriction* which contains this reference (cf. *Instruction Validation* module description in the *Lifecycle Management and Matching* domain). If no *Restriction Reference* corresponds, the *Settlement Restriction* is rejected by T2S.

If it exists, the referred *Detailed Securities Restriction*/*Detailed Cash Restriction* is updated at the settlement of this *Settlement Restriction* instead of creating a new one.

Use of blocking or reservation to settle a settlement instruction

A T2S Party can use an existing blocking or reservation for the settlement of a *Settlement Instruction* through the mention of the corresponding *Restriction Reference* in the settlement instruction. If several restriction references are indicated, the corresponding *Detailed Securities Restriction*/*Detailed Cash Restriction* are used in the order of their indication in the T2S Party's message.

The available quantity/amount in the considered *Detailed Securities Restriction*/*Detailed Cash Restriction* is used as much as necessary for the settlement of the related settlement transaction. The impacted *Detailed Securities Restriction*/*Detailed Cash Restriction* is decreased (with an historical track of its previous state) during the booking of the related settlement transaction.

Depending on the instruction of the T2S Party on the impacted securities position, the use of a blocking or a reservation results in:

- An additional *Settlement Transaction* per impacted restricted securities position or cash balance in order to transfer back the securities in the impacted securities position indicated in the *Settlement Instruction,* if this one is:
  - The deliverable securities position,

141 Through the entities optimised for the settlement processes as *Securities Position Current* and *Cash Balance Current*.

All rights reserved.

- An earmarked securities position,
- Not specified (in this case, the restricted securities is transferred back in the deliverable securities position);

- No additional *Settlement Transaction* if the blocked or reserved securities position is indicated as impacted securities position since the indicated one is directly debited to credit the securities position of the counterpart (according to its original settlement instruction);
- An additional *Settlement Transaction* per impacted restricted cash balance in order to transfer back the cash in the deliverable cash balance [142].

If the blocked or reserved securities position is indicated as the impacted securities position and if the quantity/amount available in the indicated *Detailed Securities Restriction*/*Detailed Cash Restriction* is not sufficient, the deliverable securities position is not used to complement the missing quantity/amount.

Restriction Reference consistency check for a use of blocking or reservation

If the received *Settlement Instruction* contains one of the following inconsistencies on the indicated *Restriction Reference* or impacted securities position, it will be rejected at its validation:

- At least one indicated restriction reference does not exist in the *Restriction Reference* entity;
- At least one indicated restriction reference is not related to the indicated blocked or reserved securities position as impacted securities position.

In those cases, the T2S Party has to send back an updated *Settlement Instruction* with consistent restriction references before any settlement attempt.

Automatic release and regeneration of unused restriction on cash balances

All unused blockings or reservations on cash balances are released during the EOD processing in order to be used for the intraday credit reimbursements and/or to be rebalanced to the related RTGS account. If related to a CoSD blocking, they are then regenerated at the beginning of the next settlement day using incoming cash on the considered T2S Dedicated cash account.

For the release, a *Settlement Transaction* is created (and the related settlement restriction) for each *Detailed Cash Restriction* (i.e. using the related Restriction Reference) with an available amount in order to transfer this amount back to the deliverable cash balance as a decrease of the existing detailed cash restriction.

Simultaneously, in case of CoSD blocking, the regeneration is prepared with the creation of opposite *Settlement Transactions* (and the related settlement restriction) for the same amount from the deliverable cash balance to the restricted one, as an increase of the existing detailed cash restrictions (i.e. using the Restriction Reference initially generated at the set-up).

At the beginning of the next settlement day, the regenerated CoSD blocking *Settlement Restriction* are sent for a settlement attempt, simultaneously to the first incoming *Liquidity Transfers* on the considered cash balance.

Decrease and cancellation of an existing blocking or reservation

142 This movement is systematic since the *Settlement Instruction* does not allow indicating directly the impacted *Cash Balance* contrary to *Securities Position*.

All rights reserved.

To decrease an existing blocking or reservation without using them in a *Settlement Instruction,* the T2S Party sends a *Settlement Restriction* which indicates the debited securities position or cash balance (through the corresponding restriction type) and the impacted detailed restriction (through its restriction reference).

For a decrease, an existing restriction reference is mandatory and only one restriction reference can be indicated.

The settlement of a *Settlement Restriction* for a decrease leads to:

- Debit the blocked or reserved securities position or cash balance associated to the indicated detailed restriction;
- Credit the deliverable or an earmarked securities position or the deliverable cash balance;
- Decrease the detailed restriction associated to the indicated restriction reference.

Set up and increase of an earmarked securities position

For the set up or the increase of an earmarked securities position, the T2S Party has to indicate in its *Settlement Restriction* the restriction type, identifier corresponding to the securities position to credit.

In addition, an earmarked securities position can be created or increased directly through a *Settlement Instruction* if its restriction type, is indicated as impacted securities position.

Use of an earmarked securities position to settle a settlement instruction

For the use of an earmarked securities position, the corresponding restriction type, has to be indicated as impacted securities position in a settlement instruction. No additional *Settlement Transaction* than the one related to the original *Settlement Instructions* is created since the earmarked securities position is directly debited to credit the securities position of the counterpart (according to its original settlement instruction) [143].

If no securities position corresponding to the indicated restriction type, exists at the settlement attempt point of time (or if its availability is insufficient), the *Settlement Transaction* is unsettled due to lack of securities [144].

If additional restriction references are indicated in the *Settlement Instruction,* their use is handled as a standard use of blocking or reservation associated to the deliverable securities position (i.e. with a transfer back of securities in the earmarked one).

Decrease of an existing earmarked securities position

An earmarked securities position can be decreased through a *Settlement Restriction* which debits it and credits the available securities position, since no detailed tracking is handled for earmarking.

Specific case of an earmarking at securities account level

An earmarking can be set up at the securities account level through a static data update.

In such a case, the securities positions associated to an earmarked securities account:

---

143 The underlying *Settlement Transaction* can credit directly another earmarked securities position if the counterpart has indicated it as an impacted securities position in its settlement instruction.

144 In this case, the underlying *Settlement Transaction* should be recycled only if is data is changed though an instruction maintenance (e.g. modification of impacted Securities Position) or the creation or the increase of the availability in the considered earmarked *S*ecurities *P*osition.

All rights reserved.

- Has one single earmarked securities position corresponding to the earmarking set at the securities account level;
- May have blocked or reserved securities position (following the standard management rules described above).

For the settlement of *Settlement Instructions* related to an earmarked securities account, the impacted securities position referred in the *Settlement Instruction,* except for reference to blocked and reserved securities position, is overridden by the unique earmarked securities position associated to the securities account level in T2S.

3.5.3 Optimisation Algorithms common features

Within the *Settlement* domain, optimisation algorithms can be used in two different modules and launched one after the other in groups called Series that will vary according to the optimisation strategy used in the *Daytime Recycling and Optimisation* module and in the *Night-time Settlement* module.

Even if they are not used in the same way during each settlement period, those optimisation algorithms use a common naming convention and have common features.

3.5.3.1 Optimisation Algorithms naming convention

The naming convention used for all optimisation algorithms is composed of two elements: the Family and the Name.

Family

Optimisation algorithms are categorised in three families:

- The "Build-up based" algorithms start with an unsettled *Settlement Transaction* (or a set of linked settlement transactions) and try to build a settlable set with other unsettled *Settlement Transactions* or incoming resources. This family is used during daytime;
- The "De-selection based" algorithms work on a given collection of unsettled settlement transactions. They progressively deselect the *Settlement Transactions* which would potentially causing failures until the settlement of remaining *Settlement Transactions* is possible. This family is used during the night-time settlement period.
- The "Mathematical optimisation" algorithms work on a given collection of unsettled *Settlement Transactions*. They use iterative methods to build feasible solutions and converge to a better one, driven by an objective function (based on the settlement efficiency in volume and value and taking into account the priority) that must be maximised. This family is used during the night-time settlement period.

In the name of the optimisation algorithms, the Family element can take the following values: "Deselect", "Mathematical" or "Build-up".

Name

In the name of the optimisation algorithms the Name element indicates the type of chain of *Settlement Transactions* searched by the considered optimisation algorithm (e.g. "Simple Circle", "Back to Back") or the main optimisation technical used (e.g. "Greedy").

All rights reserved.

3.5.3.2 Dedicated features per family of optimisation algorithms

"Build-up based" Optimisation Algorithms

This family of optimisation algorithms uses a selective strategy to resolve gridlocks situations by looking for chains (empty circles, back-to-back, more complex chains).

These optimisation algorithms select and sort other unsettled *Settlement Transactions* with which the collection can potentially settle. They iterate overall lacks and try to add other unsettled *Settlement Transactions* to the collection using the Provisioning functions.

If the optimisation algorithm succeeds in finding a settlable collection, it sends the settlable collection to a settlement attempt.

The "Build-up based" optimisation algorithms strategies are one of the following:

- Simple Circle strategy: for an unsettled settlement transaction, searches an opposite unsettled *Settlement Transaction* (same resource missing –ISIN or cash-, same participants, opposite direction) that can settle together with the original settlement transaction. It aims at covering the common market situation called "Simple circle";
- Back to Back strategy: for an unsettled settlement transaction, searches an unsettled *Settlement Transaction* sharing the same resources and one of the counterparties, which is able to settle when combined with the initial settlement transaction. It aims at covering the common market situation called "Back-to-back";
- Generic strategy: for an unsettled settlement transaction, searches among unsettled *Settlement Transactions* and multiple *Settlement Transactions* sharing the same resources and sharing one counterpart, that are able to settle when combined with the initial settlement transaction.

Dedicated features of "De-selection based" optimisation algorithms

This family of optimisation algorithms uses a removal strategy to resolve gridlocks situations by subtracting *Settlement Transactions* responsible for lacks in a collection. They iterate overall lacks in a collection using the Provisioning functions between each *Settlement Transaction* subtraction **{T2S.08.090} {T2S.08.100} {T2S.08.110}.**

If the optimisation algorithm succeeds in finding a settlable collection, it sends the settlable collection to the settlement functions depending on the running settlement period.

The "De-selection based" optimisation algorithms strategies are one of the following:

- Greedy strategy: deselects all *Settlement Transactions* responsible for the identified lack and then re-selects as many as possible whilst the lack remains resolved;
- One by One strategy: deselects any *Settlement Transaction* responsible for the identified lack, one at a time, until the lack is solved.

3.5.4 Multi-currency aspects

Settlement processes and currency

All rights reserved.

T2S ensures settlement in central bank money in any T2S settlement currency: the settlement functionalities described in the *Settlement* domain are available for all T2S settlement currencies. Notably, the same optimisation and auto-collateralisation algorithms apply to all *Settlement Instructions* independently from the T2S settlement currency in which their cash leg is denominated. Consequently, all *Settlement Instructions* are submitted to settlement through common procedures, allowing optimisation procedures to be run on collections of *Settlement Transactions* including *Settlement Transactions* denominated in different T2S settlement currencies.

In addition, whatever the currency of the cash part in a trade, the default securities position in case of *Settlement Instruction* without indicated impacted securities position is the deliverable one, or the earmarked one if specified at the securities account level.

T2S Dedicated cash account denomination currency

Each *T2S Dedicated Cash Account* can only be denominated in one single T2S settlement currency. Settlement in a given T2S settlement currency is possible only if the *T2S Dedicated Cash Account* where the *Settlement Instruction* settles is denominated in this currency. Consequently, all the settlement processes related to a T2S Dedicated cash account, such as the booking function, the management of limits, the management of lacks of cash applies to the currency of the T2S Dedicated cash account.

Under specific conditions [145], a CB can provide settlement in central bank money in several currencies. Hence, this allows a T2S Party holding several *T2S Dedicated Cash Accounts* in different T2S settlement currencies with the same CB, provided each *T2S Dedicated Cash Account* is denominated in one single T2S settlement currency. The fact that a CB provides one or more currencies has no impact on the settlement process (e.g. provision checking, auto-collateralisation...), since the liquidity available for settlement is appreciated at the level of each *T2S Dedicated Cash Account* (consequently, currency per currency).

Settlement transactions and denomination currency of the T2S Dedicated cash account

The currency of the cash leg for FOP *Settlement Instructions* is not checked in T2S since for these *Settlement Instructions* only the securities position is settled in T2S (the cash movement is settled outside T2S). Consequently, the related *Settlement Transactions* created in the *Settlement* domain contain only securities movements.

Similarly, no specific *Settlement Restriction* applies when securities issued in a given currency are settled against cash in a different T2S settlement currency **{T2S.02.070}**.

Link and collection of settlement transactions denominated in different T2S settlement currencies

Collections can contain *Settlement Transactions* denominated in different T2S settlement currencies. During the settlement attempt of such collections, the technical netting of debits and credits movements is performed per securities position and/or per cash balance. Consequently, the technical netting applies individually to the T2S settlement currency of each relevant T2S Dedicated cash account: no cross-currency netting is performed.

145 Limits might be introduced on euro denominated cash accounts managed by non-euro CBs.

All rights reserved.

Nevertheless, the processing of collections containing *Settlement Transactions* denominated in different T2S settlement currencies allows optimising securities resources. In particular, optimisation on the securities side of *Settlement Transactions* denominated in different currencies is possible when the same security purchased in one given currency and sold through a *Settlement Transaction* denominated in a different settlement currency (see example in the *Daytime Validation, Provisioning and Booking* module - § "Multi-currency aspects").

It is possible to link *Settlement Instructions* denominated in different T2S settlement currencies: this has no specific impact on the settlement process.

Auto-collateralisation in a cross-currency environment

The auto-collateralisation facility provided by a CB or a payment/settlement Bank is technically available for all T2S settlement currencies. Auto-collateralisation in a given currency is actually available for a given user if the relevant credit provider agrees to provide intraday credit through auto-collateralisation and takes the necessary steps for this purpose. Besides, the list and valuation of eligible collateral have to be provided for each T2S settlement currency, since the collateral eligibility and valuations based on the valuation price and own-use valuation price are managed for each T2S settlement currency with distinction of the collateralisation provided by Eurosystem CBs and non Eurosystem CBs.

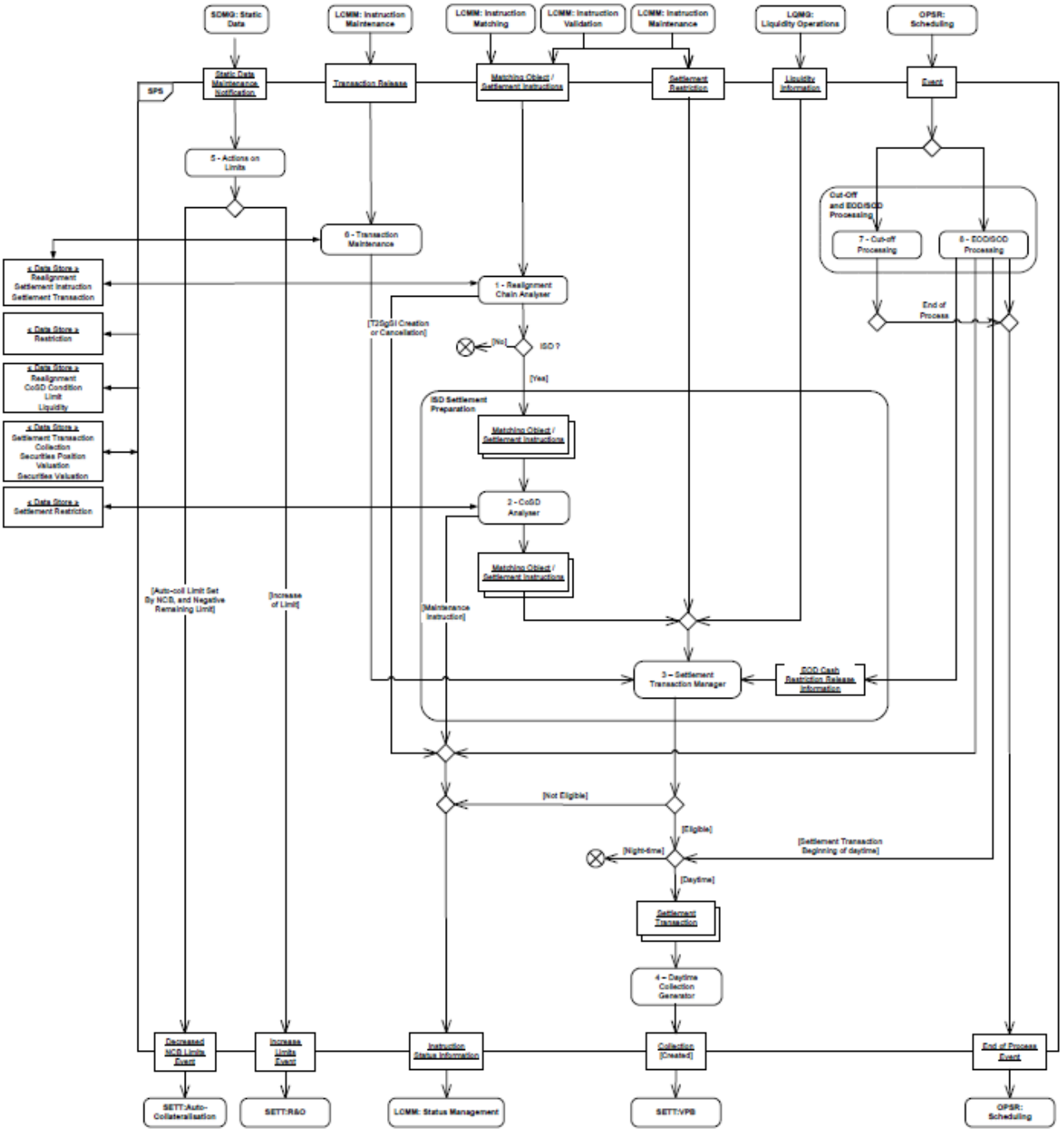
Specific dynamic reimbursement and automatic substitution features are available for any T2S settlement currency. These functionalities can also be triggered in a cross-currency situation, i.e. a situation where securities are collateralised in an intraday credit operation denominated in a given currency, but must be released for being delivered in a *Settlement Transaction* denominated in another T2S settlement currency. More precisely, in the case of a lack of securities detected during the provision-checking:

- The *Auto-collateralisation* module triggers a dynamic reimbursement or an automatic substitution of collateral even if the initial intraday credit granted is not in the same currency as the currency of the *Settlement Transaction* which needs the collateralised securities;
- A dynamic reimbursement is first attempted with the cash holdings available on the *T2S Dedicated Cash Account* which received the intraday credit. In a multi-currency situation, if this cash holding is insufficient, the cash proceeds of the underlying *Settlement Transaction* cannot be used to reimburse the initial intraday credit operation since it is in a different currency;
- An automated substitution is therefore triggered in order to replace the initial collateral with other securities eligible for the considered currency through a collateral settlement transaction. Thanks to this new collateral settlement transaction, the securities substitution is done without any cross-currency substitution (i.e. the amount used for the reimbursement is in the same currency as the original auto-collateralisation);
- As a result, the initial collateral can be released and can be delivered in a *Settlement Transaction* denominated in any (other) T2S settlement currency (Cf. detailed description and examples in the *Auto-collateralisation* module - § Multi-currency aspects).

All rights reserved.

3.5.5 Standardisation and Preparation to Settlement

3.5.5.1 Diagram of the module



3.5.5.2 Description of the module

The *Standardisation and Preparation to Settlement* (SPS) is the entry module in the *Settlement* domain for *Settlement Instructions, Settlement Restrictions,* matching objects, and liquidity information.

The *Standardisation and Preparation to Settlement* module comprises 8 functions grouped as follows.

Realignment Chain Analyser

All rights reserved.

The *Realignment Chain Analyser* function (function 1) receives the *Matching Objects* and related settlement instructions. It analyses the settlement chain, and creates T2S generated *Settlement Instructions* and their related matching objects, when required.

ISD Settlement preparation

Two functions compose the ISD settlement preparation:

- The *CoSD Analyser* function (function 2) processes the *Matching Objects* and related *Settlement Instructions* whose intended settlement date is reached. If CoSD is detected, the *CoSD Analyser* function triggers the blocking of the resources (cash and/or securities), according to the CoSD rules stored in the static data and linked to the settlement Instructions;
- Then the *Matching Objects* and related *Settlement Instructions* enter the *Settlement Transaction Manager* function (function 3) which creates and complements the settlement transactions. This function also receives the *Settlement Restrictions* and liquidity information flows, and creates the corresponding settlement transactions.

Daytime Collection Generator

The *Daytime Collection Generator* function (function 4) groups the *Settlement Transactions* into a collection according to their links to other *Settlement Transactions* and sends the collection to the *Daytime Validation, Provisioning and Booking* module.

Actions on Limit

The *Actions on Limits* function (function 5) is responsible for updating the remaining amount of limits at the reception of a notification sent by the *Static Data Management* domain following any modification of the limits.

Transaction Maintenance

The *Transaction Maintenance* function (function 6) releases the *Settlement Transactions* and handles the hold and the amendment of the settlement transactions.

Cut-off and EOD/SOD processing

Two functions are described under Cut-off and EOD/SOD processing:

- Upon receiving cut-off events from the *Scheduling* module, the *Cut-Off Processing* function (function 7) performs consecutively, among the *Settlement Transactions* concerned by the cut-off event, the following actions:
  - Submits to at least one settlement attempt the *Settlement Transactions* whose related *Settlement Instructions* have entered T2S before the concerned cut-off time (check done via the transaction's Applicable Instruction Entry Date/Time), and then, suspends all the remaining unsettled ones,
  - Suspends all created *Settlement Transactions* whose related *Settlement Instructions* have entered T2S after the concerned cut-off time, and stops their integration into collections;
- Upon receiving different events from the *Scheduling* module, the *EOD/SOD Processing* function (function 8) handles consecutively:
  - At the end of day:

All rights reserved.

- The cash restriction release for all remaining restrictions as well as the regeneration for the next settlement day, only of the CoSD blockings released,
- The stop of creation of collections,

– At the start of day: creates, if not already created in the continuous process, the Settlement Transactions corresponding to *Settlement Instructions, Settlement Restrictions* and liquidity information, received in the *Standardisation and Preparation to Settlement* module,

– At the beginning of daytime, the daytime settlement processing, with the triggering of the creation of collections.

3.5.5.3 Description of the functions of the module

1 – Realignment Chain Analyser

Introduction

Reference Id	***SETT.SPS.RCA.1.1***

The purpose of the realignment is to ensure the consistency of CSDs' books of accounts for securities movements involving several CSDs. The securities movements in the T2S parties' accounts and the securities movements resulting from the realignment in the accounts of each CSD have to be settled on an "all or none" basis.

The function receives two *Settlement Instructions* and the related *Matching Object* and **{T2S.05.174}** **{T2S.05.528}**:

- Analyses their realignment chain, based on the involved CSDs and securities accounts stored in the underlying *Settlement Instructions* and in the reference data;
- Checks if there are already linked T2S generated *Settlement Instructions,* revalidates them or cancels them when the realignment chain created above is inconsistent with the one associated to these existing T2S generated settlement instructions;
- Creates new T2S generated *Settlement Instructions* and their related *Matching Objects* when required;
- Links the T2S generated *Settlement Instructions* with the underlying *Settlement Instructions* for their settlement on an "all or none" basis.

In addition, the function notifies an *Instruction Status Information* to the *Status Management* module when creating or cancelling each T2S generated settlement instructions.

Once the realignment chain analysis is performed and if the intended settlement date of the underlying *Settlement Instructions* is reached **{T2S.05.610}**, the function sends them with their related *Matching Object* to the *CoSD Analyser* function. If the intended settlement date of the underlying *Settlement Instructions* is not reached, the function does no further action.

Analysis of the realignment chains

Introduction

Reference Id	***SETT.SPS.RCA.2.1***

All rights reserved.

Starting from the two matched *Settlement Instructions* which include the T2S (delivering and receiving) parties, their respective CSDs, the security and the securities accounts, the function performs its analysis in four steps:

- Building the two realignment chains:
  - Either from both delivering and receiving investors CSDs to the issuer CSDs of the security;
  - Or only the delivering chain from the delivering investor CSD up to the counterpart CSD (which is the receiving investor CSD in the matched settlement instructions) if the presence of an alternative link in the reference data allows to;
  - Or if no alternative link was found on the delivering chain, the receiving chain from the receiving investor CSD up to the counterpart (which is the receiving investor CSD in the matched settlement instructions) if the presence of an alternative link in the reference data allows to.

    In both cases, the alternative Security CSD Link might be stated by the investor CSD of the delivering or receiving party in the business settlement instructions (Direct Links), but it might also be stated by any subsequent Technical Issuer CSD towards the investor CSD of one counterpart (Relayed Link).

    This step is performed reading the reference data, storing the role of all the CSDs (Investor, Issuer, Technical Issuer), their participation in T2S (internal or external) and their links to other CSDs (either "default" or "alternative");
- Merging the two realignment chains previously analysed, by determining whether one intermediary CSD is common to both chains;
- Revalidating and cancelling, when required, existing T2S generated *Settlement Instructions* and their related matching objects;
- Generating all the realignment *Settlement Instructions,* using the securities accounts (mirror, omnibus and inter-CSD) stored in the static data, and linking them through a common matching reference to the underlying settlement instructions.

Building the two realignment chains from both investors CSDs to the issuer CSDs of the security

Reference Id	***SETT.SPS.RCA.2.2***

The function first identifies the T2S party delivering the securities and the Investor CSD to which it belongs, and looks within the reference data for its Technical Issuer CSD for the specific security. The analysis is performed iteratively (based on "default" security CSD links) up to identify all the Investor CSDs involved in the realignment chain. The process stops when:

- The CSD is the Issuer CSD;
- The Technical Issuer CSD is not participating to T2S (External CSD).

The function next performs the same analysis on the T2S party receiving the securities.

In the case where the issuer reached from the delivering investor CSD is not the same as the one reached from the receiving investor CSD (multi-issuance scenario) or if the delivering chain and/or the receiving chain reaches a CSD outside T2S (External CSD scenario), the analysis has to be resumed to check whether any alternative link pointing to the counterpart investor CSD exists. If such an "alternative" link is identified,

All rights reserved.

the preference should go first to the chain using the "alternative" link, more specifically the one pointing to the counterpart CSD. **If T2S cannot find an alternative link to the counterparty CSD or if such an alternative link is found, but unusable due to a missing static data (i.e. invalid or incomplete configuration of CSD Account Links), T2S reverts back to ~~Otherwise,~~** the **valid** chains built using the default links **~~should be considered~~**.**{CR-857}**

It is not allowed to define an "Alternative" security CSD link if the investor CSD or the counterpart CSD (playing the role of technical issuer CSD) in the matched settlement instructions is outside T2S.

NB: Only "alternative" link pointing to the counterpart investor CSD must be checked.

Merging the two chains

Reference Id	***SETT.SPS.RCA.2.3***

Three cases are possible:

- The two chains merge at the Issuer CSD;
- An intermediary Technical Issuer CSD is common to both chains (shortcut situation);
- No merge is possible, in case of CSDs not participating to T2S, or in case of multiple Issuer CSDs, except if an "alternative" links are eligible.

Revalidation and cancellation of possible existing T2S generated settlement instructions

Reference Id	***SETT.SPS.RCA.2.4***

This step is executed only at the start of day period. The function re-calculates the new realignment chain and compares it with the old one calculated the day before.

If the two realignment chains are identical, then the function revalidates the current T2S generated *Settlement Instructions* to check whether some restrictions apply on them. Otherwise, in case of discrepancies between these two realignment chains due to a static data update, the *Realignment Chain Analyser* function cancels the T2S generated *Settlement Instructions* and their related *Matching Objects* previously created, and generates new ones.

Generating the realignment settlement instructions

Reference Id	***SETT.SPS.RCA.2.5***

This step is performed only in the case of a first settlement attempt or when the existing T2S generated *Settlement Instructions* and their related *Matching Objects* have been cancelled at the previous step.

When the final chain affects one single CSD (scenarios intra or intra-like or in some cases of In/Out), the function does not create T2S generated settlement instructions.

In all the other cases, this step starts from both Investor CSDs:

- If they are relying on a Technical Issuer CSD in T2S (via either "default" or "alternative" security CSD links), a T2S generated *Settlement Instruction* is created affecting the mirror account of the Technical Issuer CSD against the T2S Party account;
- If they are relying on a Technical Issuer CSD outside of T2S, a T2S generated *Settlement Instruction* is created affecting the inter-CSD account of the Technical Issuer CSD against the T2S Party account.

All rights reserved.

The function iterates the same process at the level of the Technical Issuer CSD and, when required, creates at that level two T2S generated *Settlement Instruction* between the omnibus account of the previous Investor CSD and either the mirror or the inter-CSD account of the following Technical Issuer CSD.

When the CSD at which both parts of the chain merge is reached and if that CSD is different from the Issuer CSD, a complementary analysis is required:

- If that CSD is using the same omnibus account for both of its T2S Parties (or Investor CSD), two T2S generated *Settlement Instruction* affecting two T2S Parties accounts (or omnibus accounts of CSDs) are created;
- Otherwise, the function creates the T2S generated *Settlement Instructions* between the T2S Parties accounts (or omnibus account of CSDs) and either the mirror or the inter-CSD account of the following Technical Issuer CSD. In addition, the function creates the T2S generated *Settlement Instructions* between the omnibus accounts of the current Investor CSD in the following Technical Issuer CSD.

In the particular case where two Issuer CSDs in T2S are affected, the instructions created at the level of these CSDs are affecting their issuance accounts instead of a mirror or an intra-CSD account.

In such a scenario, the preference should be given to the "alternative" links over "default" links, if the configuration of the security CSD links in reference data allows it.

For each pair of T2S generated *Settlement Instructions* created, the function also creates the related *Matching Object* with a common matching reference.

List of scenarios

Reference Id	***SETT.SPS.RCA.2.6***

As an illustration, the present section describes the T2S generated *Settlement Instructions* and related *Matching Object* created for the main settlement scenarios identified in the URD.

SCENARIOS [146]	TITLE	UR REF	COMMENT
Intra			Intra
Cross 1	Two Investor CSDs and one Issuer CSD all in T2S	**{T2S.09.350}** **{T2S.09.360}**	Cross-CSD
Cross 2	Two Investor CSDs and two Issuer CSDs all in T2S		
Cross 3	Common Investor CSD in T2S		
External 1	both Investor CSDs are external to T2S and the Issuer CSD is in T2S	**{T2S.02.100}** **{T2S.09.390}**	Unconditional In/Out (Intra-CSD like)

146 Naming convention used in the next paragraph for the settlement scenarios.

All rights reserved.

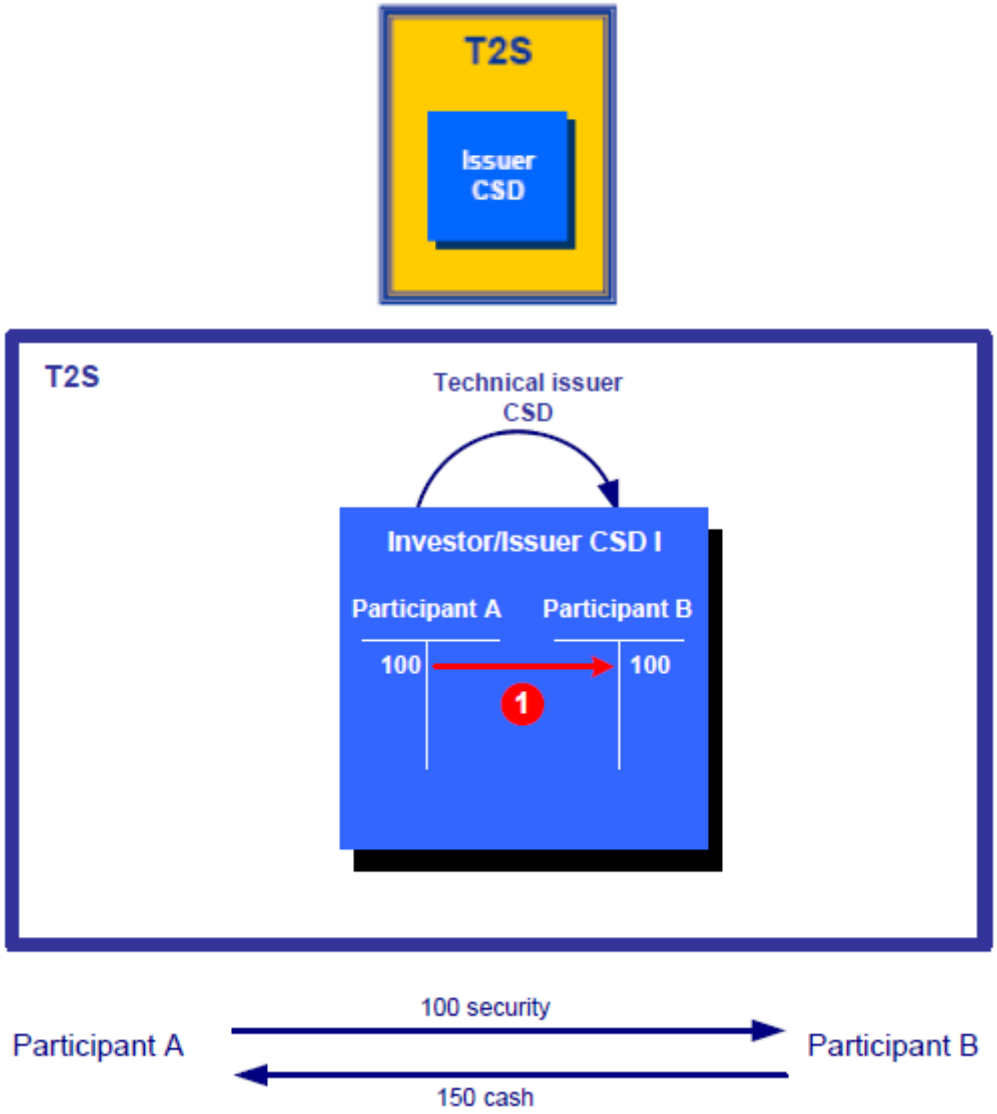
SCENARIOS [146]	TITLE	UR REF	COMMENT
External 2	One Investor CSD is external to T2S with one Investor CSD and the Issuer CSD in T2S	**{T2S.02.110}** **{T2S.09.400}**	Unconditional In/Out (Cross-CSD like)
External 3	One Investor CSD and the Issuer CSD are external to T2S with one Investor CSD in T2S	**{T2S.02.120}** **{T2S.09.410}**	Conditional In/Out (CoSD like)
External 4	The Issuer CSD is external to T2S and both Investor CSDs are in T2S	**{T2S.02.130}** **{T2S.09.420}**	Unconditional In/Out (Internal Cross-CSD like)

Scenario Intra

Reference Id	***SETT.SPS.RCA.3.1***

This scenario covers the Intra-CSD settlement defined as settlement between two participants belonging to the same CSD in T2S. In this case, the CSD cumulates both roles of Investor CSD of the two participants and Issuer CSD of the security.

All rights reserved.



Since both participants belong to the same CSD in T2S, there is no realignment mechanism. Consequently, the function does not create any T2S generated *Settlement Instruction* and related matching object.

Scenarios Cross

These scenarios cover the Cross-CSD settlement defined as a settlement between participants not belonging to the same CSD with all the involved CSDs (Investors and Issuer) being in T2S.

Scenario Cross 1: Two Investor CSDs and one Issuer CSD all in T2S

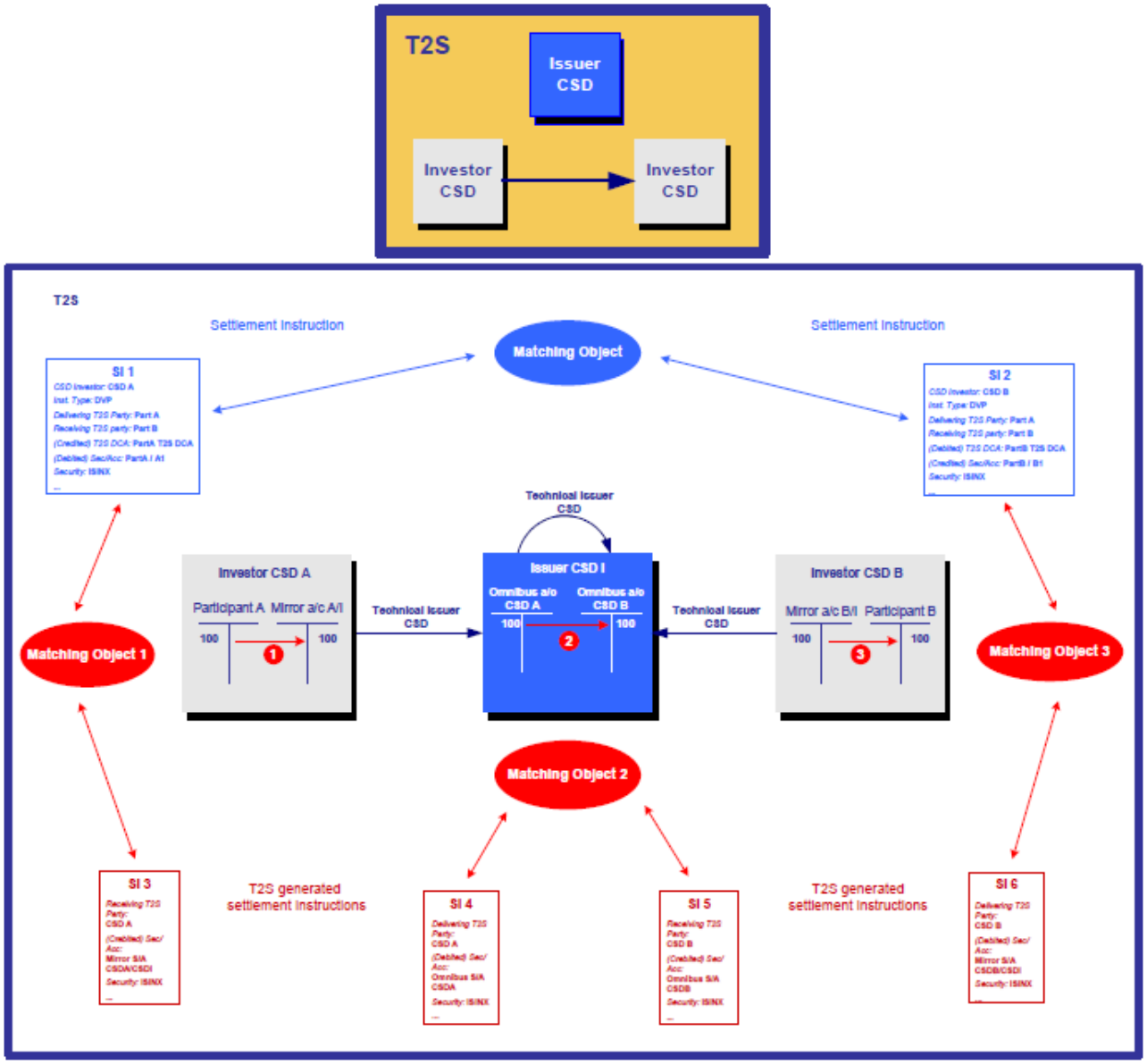
Reference Id	***SETT.SPS.RCA.4.1***

This scenario describes the settlement where participant A from Investor CSD A is selling securities to participant B from Investor CSD B. It implies:

- Two Investor CSDs (A and B) in T2S in relationship with the Issuer CSD as Technical Issuer;

All rights reserved.

- The Issuer CSD (I) in T2S.



Under such a simple configuration (no external CSD settlement, no multi-issuance), no "alternative" security CSD link should be checked by T2S.

The function creates three sets of T2S generated *Settlement Instructions* (and related matching objects) corresponding to the quantity indicated in the underlying settlement instructions:

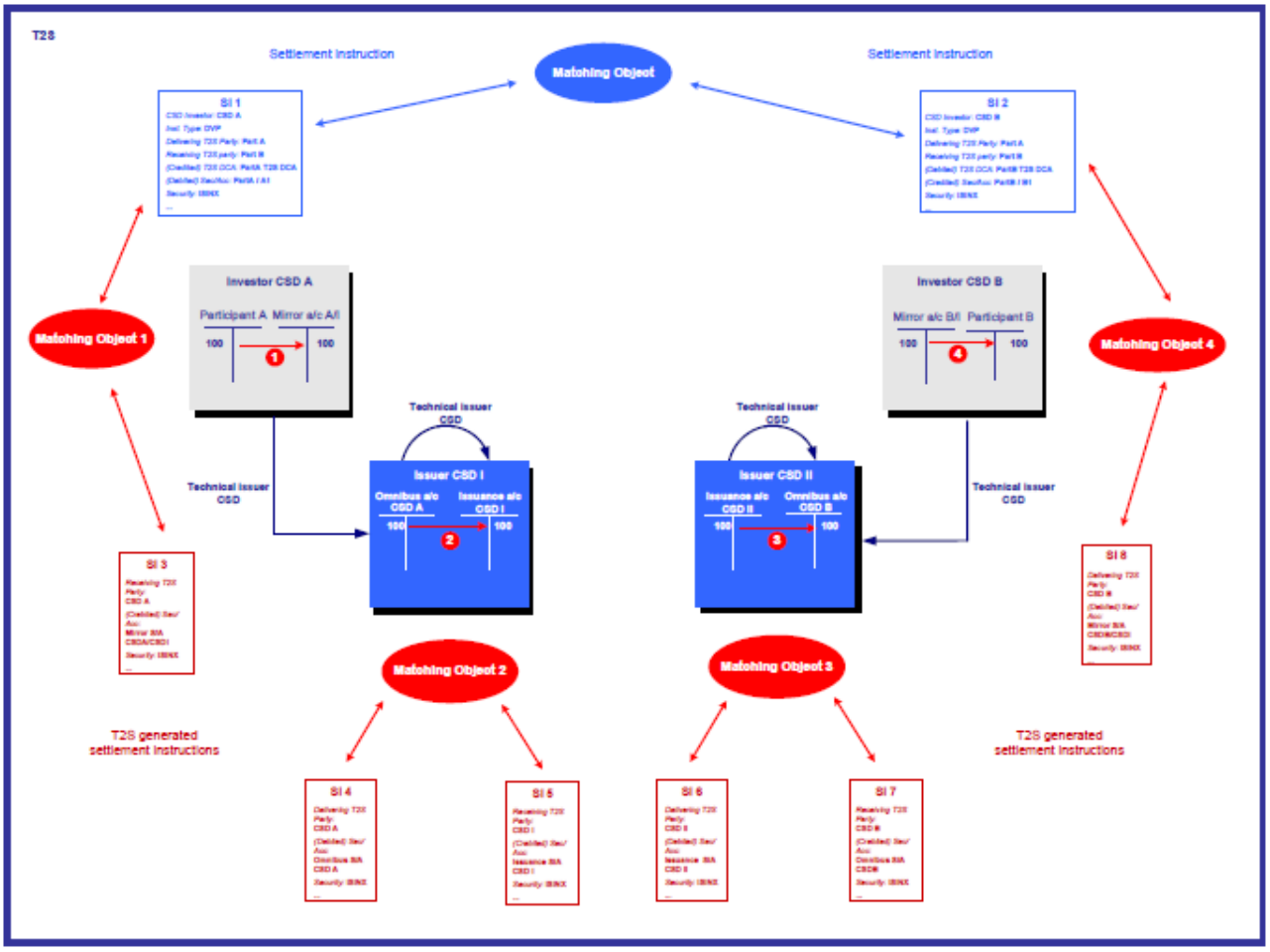
- A T2S generated *Settlement Instruction* (☐) which credits the mirror account A/I in the Investor CSD A and is associated with the *Settlement Instruction* debiting the participant A securities account;
- Two T2S generated *Settlement Instructions* (☐) between omnibus securities accounts in the Issuer CSD I:
  - A T2S generated *Settlement Instruction* which debits the CSD A omnibus account,
  - A T2S generated *Settlement Instruction* which credits the CSD B omnibus account;

All rights reserved.

- A T2S generated *Settlement Instruction* (☐) which debits the mirror account B/I in the Investor CSD B and is associated with the *Settlement Instruction* crediting the participant B securities account **{T2S.09.300} {T2S.09.310} {T2S.09.340} {T2S.09.350} {T2S.09.360} {T2S.09.370}**.

Scenario Cross 2: Two Investor CSDs and two Issuer CSDs all in T2S

Reference Id	***SETT.SPS.RCA.4.2***



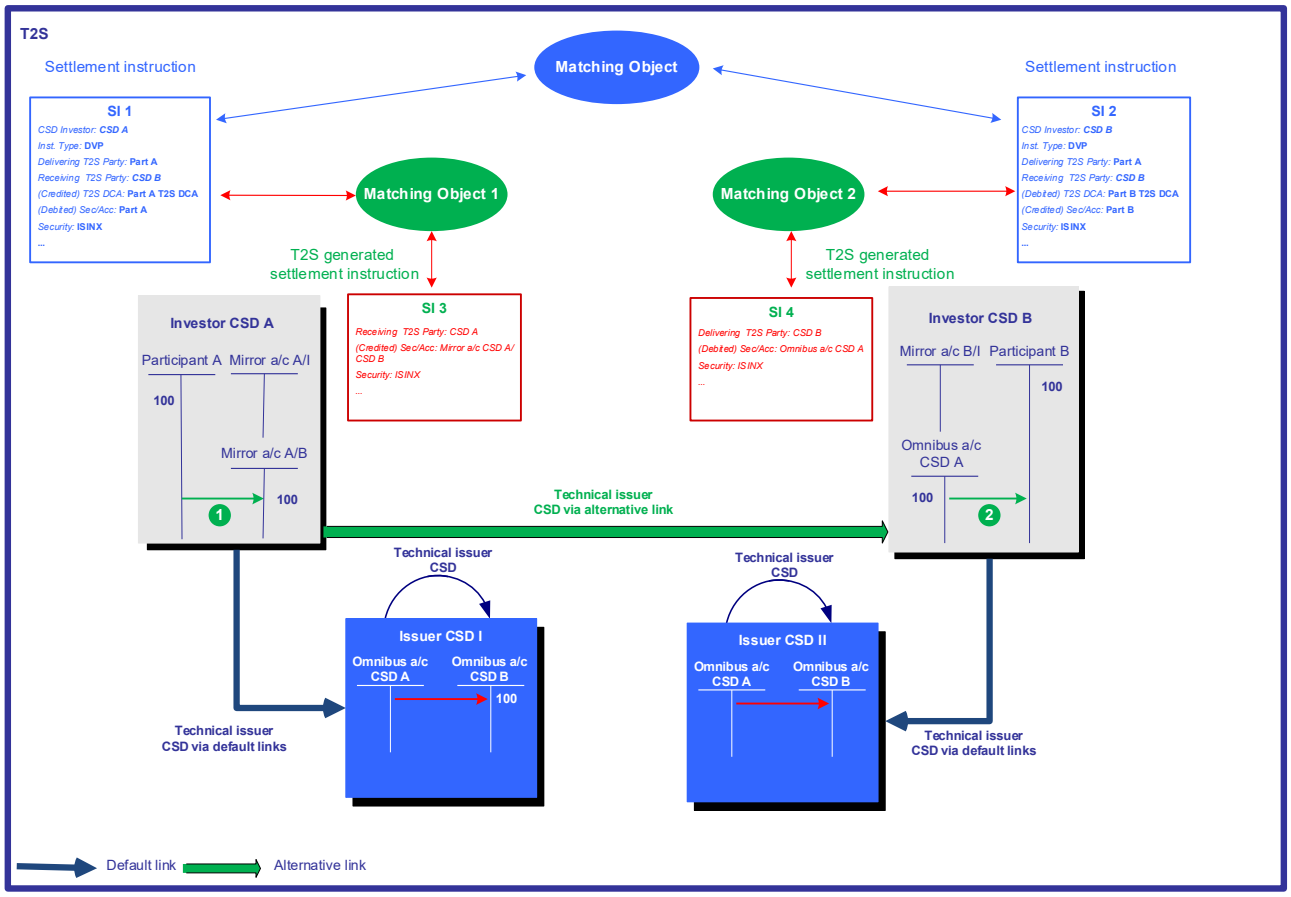
This scenario describes the settlement where participant A from Investor CSD A is selling securities to participant B from Investor CSD B. It implies:

- Two Investor CSDs (A and B) in T2S in relationship either with each other (via "alternative" security CSD link) or with different Issuer CSD as Technical Issuer CSD (via "default" security CSD links);
- Two Issuer CSDs (I and II) in T2S.

Such a scenario allows defining "alternative" security CSD links in the reference data, that can be used preferably over "default" links for the realignment purpose. According to whether "alternative" links are set up or not, the function builds the realignment chains following two different ways:

All rights reserved.

Delivering (or receiving) chain built up to the counterpart investor CSD using "alternative" security CSD link



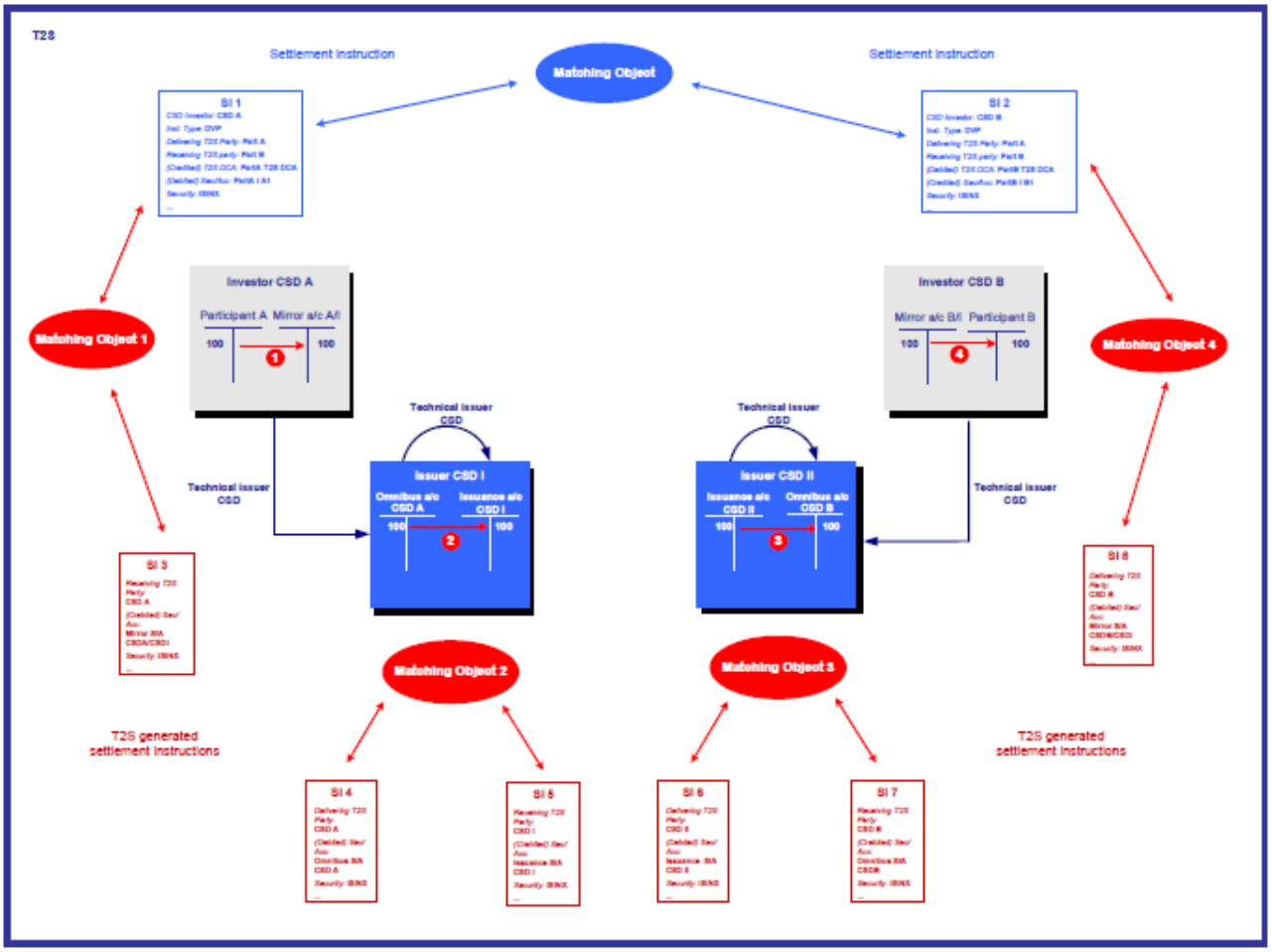
In this scenario, no T2S generated Settlement Instructions is created to update the issuance accounts.

The presence of the alternative link pointing to the investor CSD B (playing the role of technical issuer CSD for the investor CSD A on the security ISIN X) allows creating two sets of T2S generated Settlement Instructions (and related matching objects) corresponding to the quantity indicated in the underlying settlement instructions:

- A T2S generated Settlement Instruction (☐) which credits the mirror account A/B in the Investor CSD A and is associated with the Settlement Instruction debiting the participant A securities account;
- A T2S generated Settlement Instruction (☐) which debits the CSD A omnibus account in the Investor CSD B and is associated with the Settlement Instruction crediting the participant B securities account.

All rights reserved.

Delivering and receiving chains built up to the issuer CSDs using only "default" security CSD links



The scenario is equivalent to scenario Cross 1 with additional T2S generated *Settlement Instructions* (and related matching objects) to update the issuance accounts in each Issuer CSD. The issuance accounts have the nature of the position mark up and mark down indicating the decrease and the increase of the securities respectively depending on the direction of the transfer (receive or delivery).

The function creates four sets of T2S generated *Settlement Instructions* (and related matching objects) corresponding to the quantity indicated in the underlying settlement instructions:

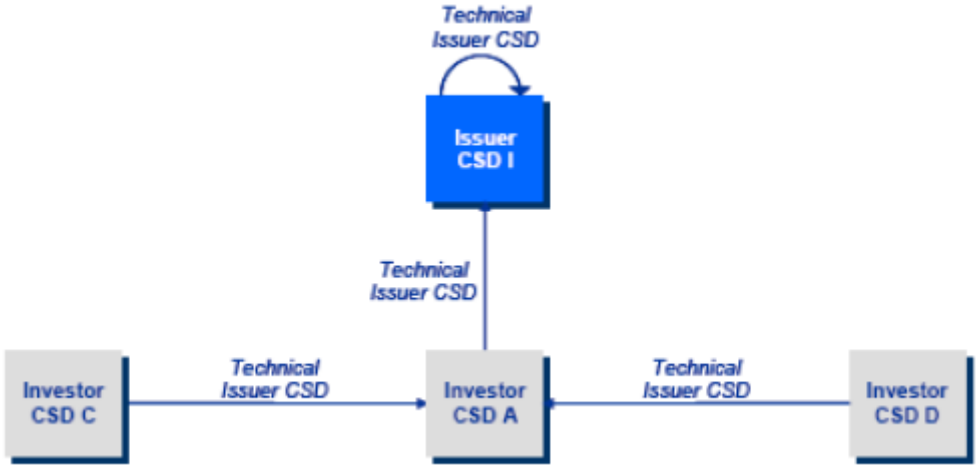
- A T2S generated *Settlement Instruction* (☐) which credits the mirror account A/I in the Investor CSD A and is associated with the *Settlement Instruction* debiting the participant A securities account;
- Two T2S generated *Settlement Instructions* (☐) between securities accounts in the Issuer CSD I:
  - A T2S generated *Settlement Instruction* which debits the CSD A omnibus account,
  - A T2S generated *Settlement Instruction* which credits the CSD I issuance account;
- Two T2S generated *Settlement Instructions* (☐) between securities accounts in the Issuer CSD II:
  - A T2S generated *Settlement Instruction* which debits the CSD II issuance account,
  - A T2S generated *Settlement Instruction* which credits the CSD B omnibus account;

All rights reserved.

- A T2S generated *Settlement Instruction* (☐) which debits the mirror account B/I in the Investor CSD B and is associated with the Settlement Instruction crediting the participant B securities account.

Scenario Cross 3: Common Technical Issuer CSD in T2S

Reference Id	***SETT.SPS.RCA.4.3***



This scenario describes the settlement where participant A from Investor CSD C is selling securities to participant B from Investor CSD D. It implies:

- Two Investor CSDs (C and D) in T2S in relationship with the same Investor CSD as Technical Issuer CSD;
- An Investor CSD (A) in T2S in relationship with the Issuer CSD as Technical Issuer CSD;
- The Issuer CSD (I) in T2S.

The scenario is equivalent to scenario Cross 1 with additional T2S generated *Settlement Instructions* (and related matching objects) within the Investor CSD A involved in the realignment.

Under such a simple configuration (no external CSD settlement, no multi-issuance), no "alternative" security CSD link should be checked by T2S.

In case CSD A uses two different omnibus accounts in the Issuer CSD I, two additional T2S generated *Settlement Instructions* (associated via a matching object) are created to ensure the realignment between these two omnibus accounts. If the CSD A uses the same omnibus account, no additional T2S generated *Settlement Instruction* is created **{T2S.09.380}**.

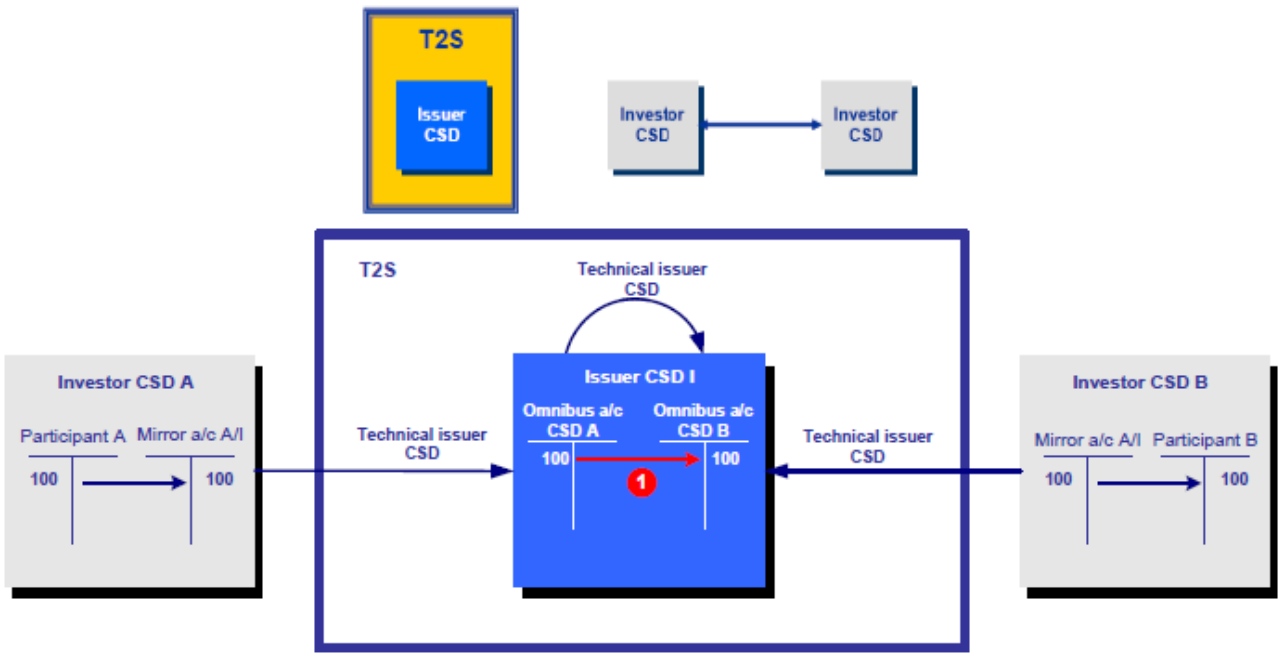
Scenarios External

The following scenarios cover the In/Out T2S settlement defined as a settlement between two participants not belonging to the same CSD with at least one involved CSD (Investors or Issuer) being external to T2S.

Scenario External 1: both Investor CSDs are external to T2S and the Issuer CSD is in T2S

Reference Id	***SETT.SPS.RCA.5.1***

All rights reserved.



This scenario describes the case of an external settlement where participant A of Investor CSD A is selling securities to participant B of Investor CSD B. It implies:

- Two Investor CSDs (A and B) external to T2S in relationship with the Issuer CSD (I) as Technical Issuer CSD;
- The Issuer CSD (I) in T2S.

From the perspective of T2S, this scenario appears as an Intra-CSD settlement in CSD I: both Investors CSD A and CSD B are participants of the Issuer CSD I in T2S where they own their omnibus account.

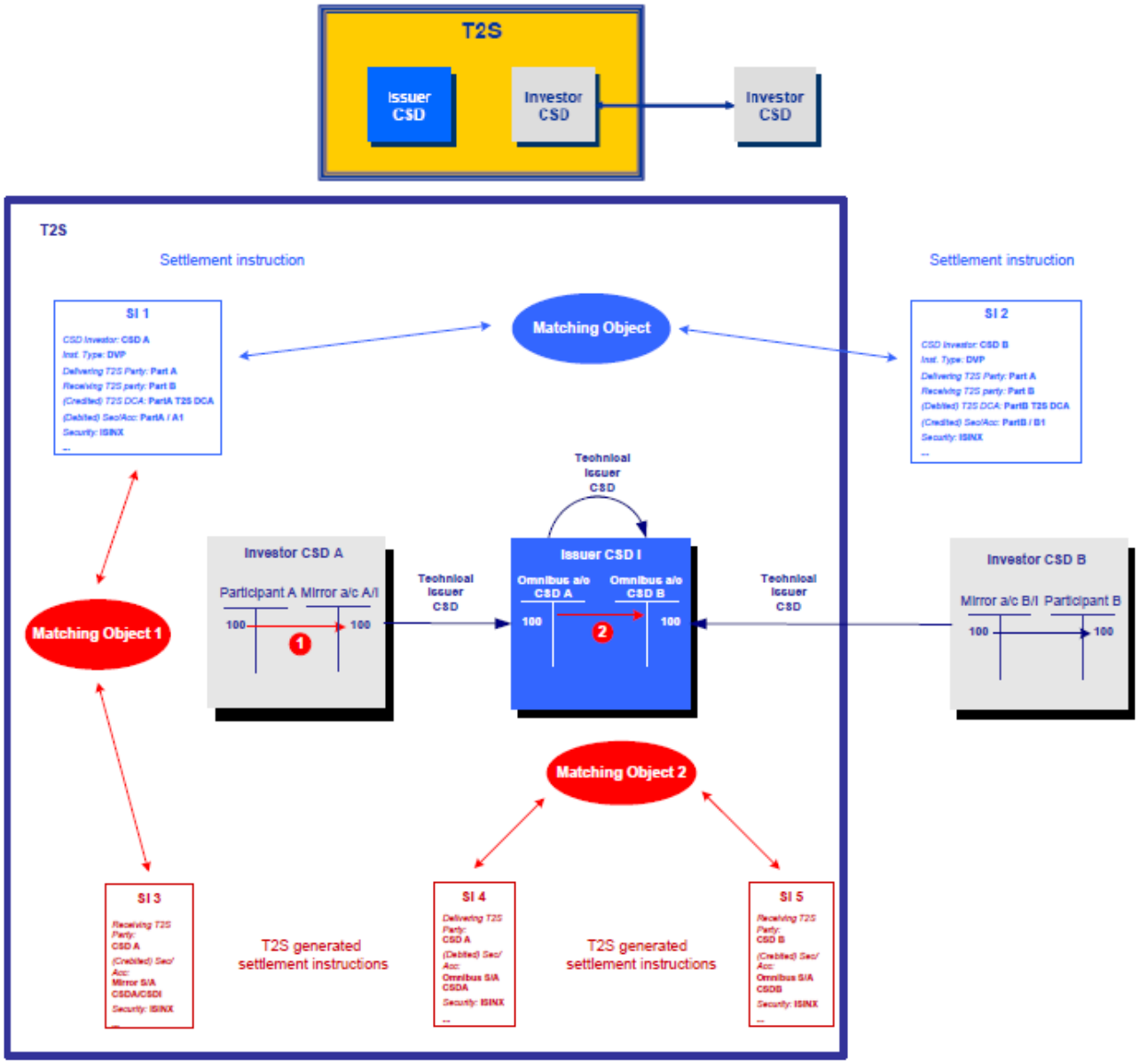
Since CSD A and CSD B as participants of the CSD I belong to T2S (and to the same CSD), there is no actual realignment mechanism. Consequently, the function does not create any T2S generated *Settlement Instruction* and related matching object **{T2S.02.100} {T2S.09.390}**.

In this scenario, no alternative security CSD link can be defined between the investor CSD A and the investor CSD B as such a link can only be set up between a T2S-in investor CSDs, pointing to a T2S-in technical issuer CSD.

Scenario External 2: One Investor CSD is external to T2S with one Investor CSD and the Issuer CSD in T2S

Reference Id	***SETT.SPS.RCA.5.2***

All rights reserved.



This scenario describes the case of a settlement where participant A from Investor CSD A in T2S is selling securities to participant B from Investor CSD B external to T2S. It implies:

- An Investor CSD (A) in T2S in relationship with the Issuer CSD (I) as Technical Issuer CSD;
- An Investor CSD (B) external to T2S in relationship with the same Issuer CSD (I) as Technical Issuer CSD;
- The Issuer CSD (I) in T2S.

In this scenario, no alternative security CSD link can be defined between the investor CSD A and the investor CSD B as such a link can only be set up between a T2S-in investor CSDs, pointing to a T2S-in technical issuer CSD.

From the perspective of T2S, this scenario appears as an unconditional In/Out settlement between:

- Participant A which belongs to Investor CSD A in T2S;

All rights reserved.

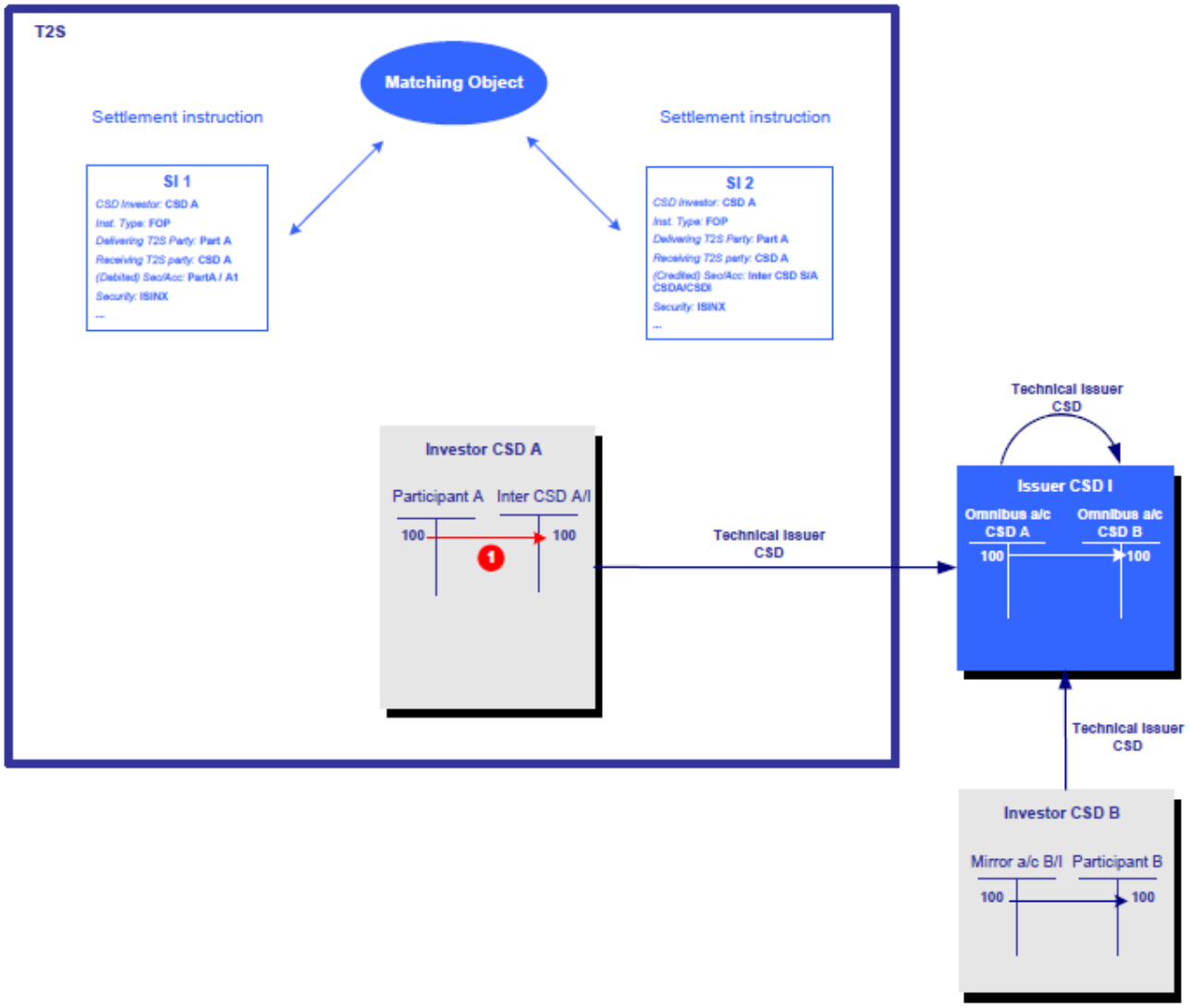
- The external CSD B as participant of the Issuer CSD I in T2S where it owns an omnibus account.

Consequently, the function creates two sets of T2S generated Settlement Instructions (and related matching objects) corresponding to the quantity indicated in the underlying settlement instructions:

- A T2S generated *Settlement Instruction* (☐) which credits the mirror account A/I in the Investor CSD A and is associated with the *Settlement Instruction* debiting the participant A securities account;
- Two T2S generated *Settlement Instructions* (☐) between omnibus securities accounts in the Issuer CSD I:
  - A T2S generated *Settlement Instruction* which debits the CSD A omnibus account,
  - A T2S generated *Settlement Instruction* which credits the CSD B omnibus account B **{T2S.02.110}** **{T2S.09.400}**.

Scenario External 3: One Investor CSD and the Issuer CSD are external to T2S with one Investor CSD in T2S

Reference Id	***SETT.SPS.RCA.5.3***



This scenario describes the case of a settlement where a participant A of the Investor CSD A in T2S is selling securities to a participant B of the Investor CSD B external to T2S. It implies:

All rights reserved.

- An Investor CSD (A) in T2S in relationship with the Issuer CSD (I);
- An Investor CSD (B) external to T2S in relationship with the same Issuer CSD (I);
- The Issuer CSD (I) external to T2S.

In this scenario, the cash delivery is handled as a CoSD since it is settled outside T2S. Securities are blocked until this settlement is fulfilled. The fulfilment of this cash settlement then allows the CoSD release to take place: the securities are released and delivered to the receiving Party.

From the perspective of T2S, this scenario appears as a conditional In/Out settlement between participant A of CSD A and the CSD A (as its own participant). This settlement is conditional since only one Inter-CSD account is affected (conditional with securities blocking as the Inter-CSD account is affected in credit).

Since there is no intermediary Technical Issuer CSD between the CSD A and the Issuer CSD I, no realignment mechanism occurs. Consequently, the function does not create any T2S generated *Settlement Instruction* and related matching object **{T2S.02.120} {T2S.09.410}**.

In this scenario, no alternative security CSD link can be defined between the investor CSD A and the investor CSD B as such a link can only be set up between a T2S-in investor CSDs, pointing to a T2S-in technical issuer CSD.

Scenario External 4: The Issuer CSD is external to T2S and both Investor CSDs are in T2S

Reference Id	***SETT.SPS.RCA.5.4***

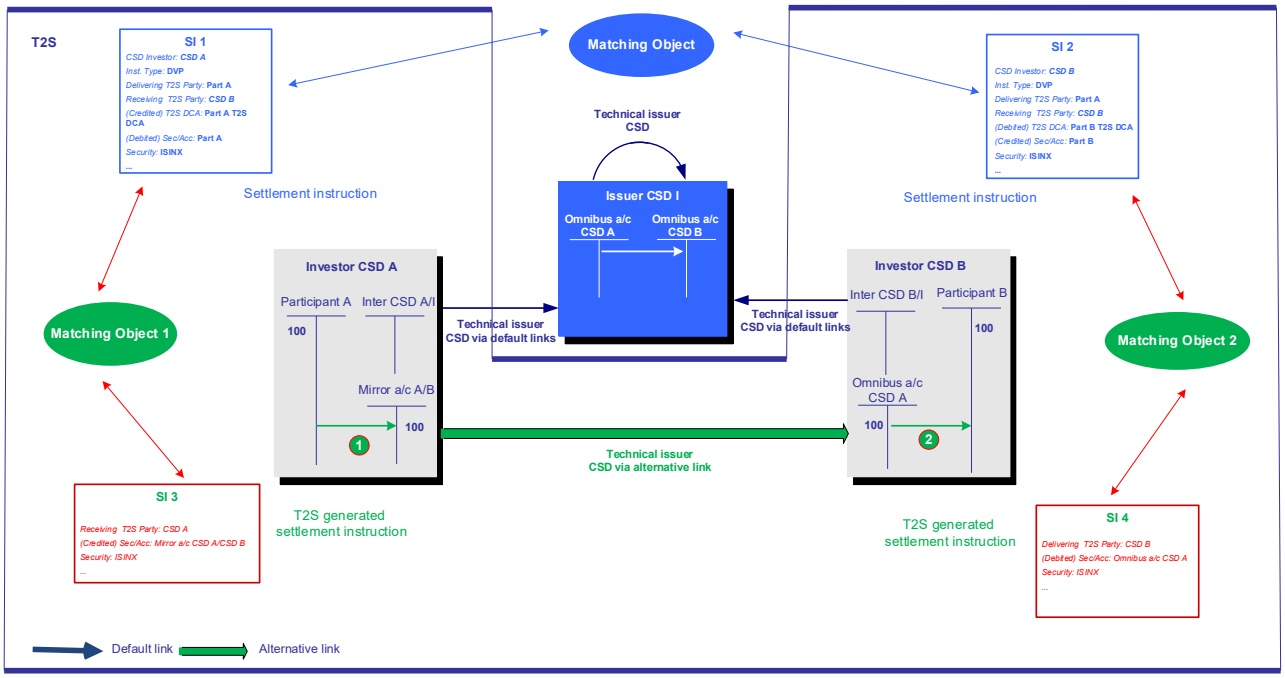
This scenario describes the case of a settlement where participant A from Investor CSD A is selling securities to participant B from Investor CSD B. It implies:

- Both Investor CSDs (A and B) in T2S in relationship with the same Issuer CSD (I);
- The Issuer CSD (I) external to T2S.

Such a scenario allows defining "alternative" security CSD links in the reference data that can be used preferably over "default" links for the realignment purpose. According to whether "alternative" links are set up or not, the function builds the realignment chains following two different ways:

All rights reserved.

Delivering (or receiving) chain built up to the counterpart investor CSD using "alternative" security CSD link



Thanks to the "alternative" security CSD link pointing to the investor CSD B (playing the role of technical issuer CSD for the investor CSD A on the security ISIN X), no T2S generated Settlement Instructions is created to update the inter-CSD accounts.

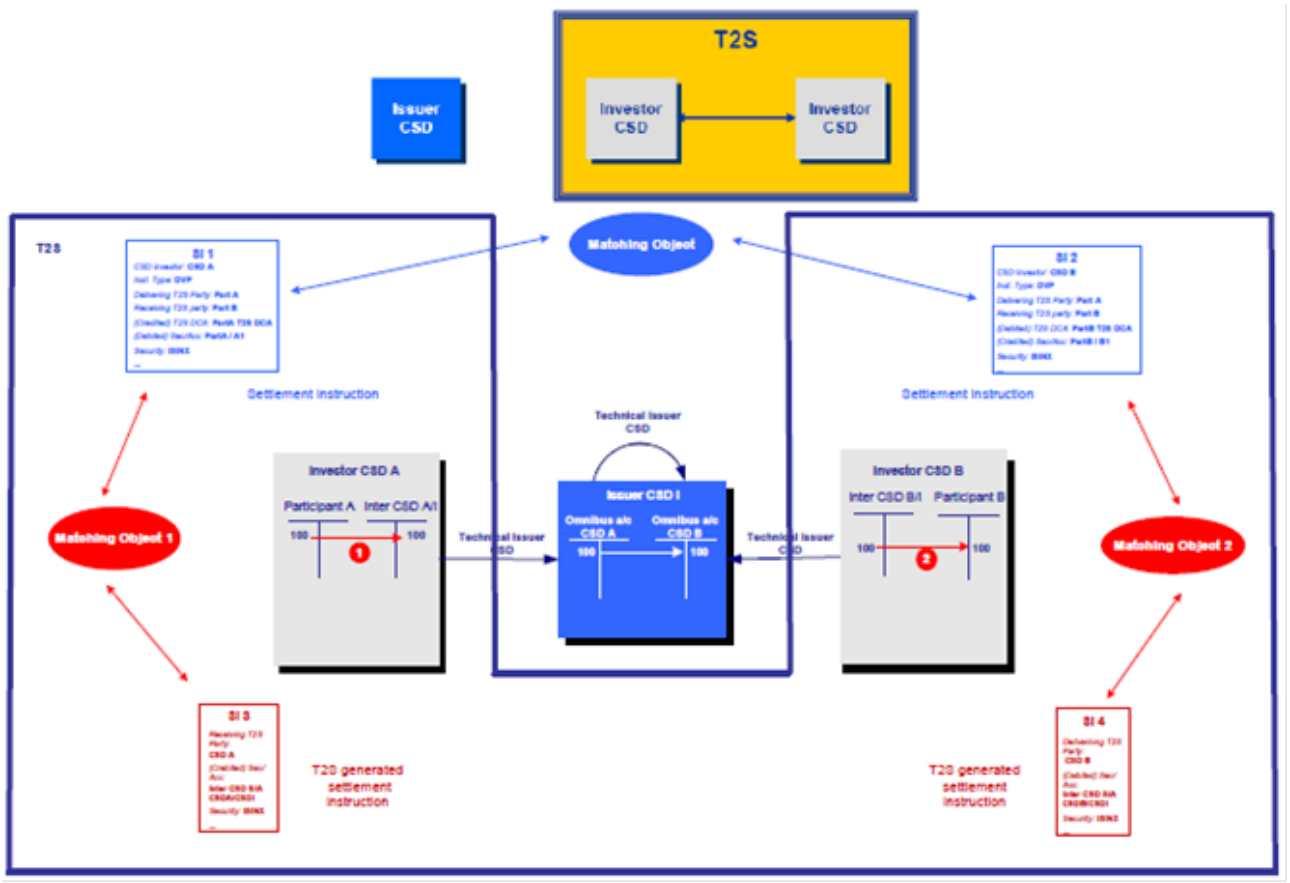
The presence of the alternative link pointing to the investor CSD B (playing the role of technical issuer CSD for the investor CSD A on the security ISIN X) allows creating two sets of T2S generated Settlement Instructions (and related matching objects) corresponding to the quantity indicated in the underlying settlement instructions:

Then, the function creates two T2S generated Settlement Instructions (and related matching objects) corresponding to the quantity indicated in the underlying settlement instructions:

- A T2S generated Settlement Instruction (☐) which credits the Mirror account A/B in the Investor CSD A and is associated with the Settlement Instruction debiting the participant A securities account;
- A T2S generated Settlement Instruction (☐) which debits the Omnibus CSD account A in the Investor CSD B and is associated with the Settlement Instruction crediting the participant B securities account.

All rights reserved.

Delivering and receiving chains built up to the external issuer CSD using only "default" security CSD links



Even if Issuer CSD I is external to T2S, the settlement of this scenario in T2S is unconditional In/Out an unsynchronised realignment is sent by Investor CSDs to the external Issuer CSD I for the realignment.

Consequently, the function creates two T2S generated *Settlement Instructions* (and related matching objects):

- A T2S generated *Settlement Instruction* (☐) which credits the inter CSD account A/I in the Investor CSD A and is associated with the *Settlement Instruction* debiting the participant A securities account;
- A T2S generated *Settlement Instruction* (☐) which debits the inter CSD account B/I in the Investor CSD B and is associated with the Settlement Instruction crediting the participant B securities account **{T2S.02.130} {T2S.09.420}**.

2 – CoSD Analyser

Introduction

Reference Id	***SETT.SPS.COA.1.1***

The *CoSD Analyser* function checks whether some CoSD rules apply to the *Settlement Instructions* **{T2S.05.625}**.

CoSD rules Analyser

Reference Id	***SETT.SPS.COA.2.1***

All rights reserved.

The function retrieves in the static data all the CoSD rules associated to the CSDs involved in the settlement, in order to check whether they apply on the *Settlement Instructions* (which can be either the *Settlement Instructions* or the T2S generated settlement instructions).

Each CoSD rule is a matrix of several occurrences of usable parameters among the following **{T2S.11.740}** that are used to condition a CoSD:

- ISIN;
- Settlement Currency;
- CSD;
- Securities Account;
- Country of Issuance;
- Place of Settlement;
- Transaction Type;
- Issuer CSD in T2S;
- Delivering CSD in T2S;
- Receiving CSD in T2S;
- BIC of Issuer CSD;
- BIC of Delivering CSD in T2S;
- BIC of Receiving CSD in T2S.

The function checks, for each CoSD rule retrieved in the static data, if its occurrences of parameters and their value match with the relevant attributes of the settlement instruction(s) and their value.

The *Settlement Instruction* is subject to conditional securities delivery, per CoSD rule for which the comparison matches on one occurrence of parameters, provided the boolean CoSD Processing value of this occurrence is set to "Yes".

The function then links each applicable CoSD rule to the relevant settlement instruction(s) **{T2S.09.210} {T2S.11.740}**.

Additionally conditional In/Out CoSD blocking, raised by the realignment analysis, may apply to the *Settlement Instruction*, through two specific CoSD rules defined by the T2S Operator and applying to all CSDs **{T2S.02.090} {T2S.09.410} {T2S.11.740}**:

- If only one Inter-CSD account is affected and affected in credit in the realignment chain, then the *Settlement Instructions* received from the T2S Parties are linked to the CoSD rule defined by the following set of parameters:
  - Issuer CSD in T2S: NO,
  - Delivering CSD in T2S: YES,
  - Receiving CSD in T2S: NO;

All rights reserved.

- If only one Inter-CSD account is affected and affected in debit in the realignment chain, then the *Settlement Instructions* received from the T2S Parties are linked to the CoSD rule defined by the following set of parameters:
  - Issuer CSD in T2S: NO,
  - Delivering CSD in T2S: NO,
  - Receiving CSD in T2S: YES.

CoSD Activation

Reference Id	***SETT.SPS.COA.3.1***

According to the CoSD rule(s) linked to the *Settlement Instructions* previously, the function **{T2S.09.210} {T2S.09.220} {T2S.09.230} {T2S.09.410}**:

- Creates maintenance instructions, one per CoSD rule and per T2S parties, to hold the settlement instructions;
- Updates the *Settlement Instructions* CoSD Hold Status attribute to "*Yes*" and associates a reason code referring to each CoSD rule;
- Updates the *Settlement Instruction* CoSD Flag to "*Yes*";
- Creates one *Settlement Restriction* per resource to be blocked (cash and/or the securities), possibly using the relevant Restriction References borne by the associated Settlement Instruction, as defined by the linked CoSD rules;
- Associates these *Settlement Restrictions* to the *Settlement Instructions* with the same value for CSD Hold, Party Hold, CSD Validation on Hold;
- Informs the *Status Management* module in the *Lifecycle Management and Matching* domain on the maintenance instructions and *Settlement Restrictions* created.

Next, the function sends the matching objects, their related *Settlement Instructions* and associated *Settlement Restrictions* to the *Settlement Transaction Manager* function.

3 – Settlement Transaction Manager

Introduction

Reference Id	***SETT.SPS.STM.1.1***

The *Settlement Transaction Manager* function creates the *Settlement Transactions* associated to:

- Settlement Restrictions (blocking, reservation, or earmarking) received on or after their intended settlement date;
- Matching Objects and their related Settlement Instructions received on or after their intended settlement date;
- Liquidity Information related to *Liquidity Transfer Order* received on their intended settlement date.

The *Settlement Transaction Manager* function also creates harmonised *Settlement Transactions* which aim at:

All rights reserved.

- Managing the securities attributes related to fund shares and coupons and dealing with fund shares decimals **{T2S.09.010} {T2S.09.020}**;
- Managing the coupon stripping and reattachment functionalities **{T2S.09.030} {T2S.09.040} {T2S.09.050} {T2S.09.260} {T2S.09.320} {T2S.09.330}**;
- Also, upon reception of a Transaction Release, the function releases the *Settlement Transaction* specified in the flow.

Existing Settlement Transaction

Reference Id	***SETT.SPS.STM.2.1***

In case the *Settlement Restriction, Matching Object* or *Liquidity Transfer Order* is already linked to a *Settlement Transaction*, the function does not create a new transaction and triggers directly the final checks process, including i-a the *Settlement Transactions* recycled from previous day. Otherwise, the *Settlement Transaction* is created as described below.

Settlement Transaction creation

Introduction

Reference Id	***SETT.SPS.STM.3.1***

The function analyses the flows, identifies those relating to a *Settlement Restriction* on a securities position or cash balance, creates the associated settlement transactions [147] and fills in the securities accounts and *T2S Dedicated Cash Accounts* as well as the securities positions and cash balance.

Earmarking at securities account level

Reference Id	***SETT.SPS.STM.3.2***

The function first checks whether the securities account is identified as "Earmarking". If so, the *Settlement Transaction* will refer to the *Securities Position* with that earmarking, even if the *Settlement Instruction* refers to the deliverable position or another earmarked securities position **{T2S.08.650}**.

Settlement Restriction

Reference Id	***SETT.SPS.STM.3.3***

For a *Settlement Restriction* which sets up or updates (increase or decrease) a reservation, a blocking or an earmarking, the Restriction Processing Type value is "*Reservation*", "*Blocking*" or "Balance Type/*Earmarking*" respectively. Then the function creates a *Settlement Transaction* and **{T2S.02.090} {T2S.09.180} {T2S.09.220} {T2S.09.290} {T2S.09.410}**:

- In case of restriction on securities, links the *Settlement Transaction* to the relevant restriction types, securities account and security provided by the settlement restriction;
- In case of restriction on cash, links the *Settlement Transaction* to the relevant restriction types and *T2S Dedicated Cash Account* provided by the settlement restriction.

147 The restriction at the account level is not managed by this function but by the *Static Data Management* domain. Consequently, the *Settlement Restrictions* which credit or debit an account with a restriction at the account level are handled by the *Standard Transaction Manager* function without specific processing.

All rights reserved.

Matching Object

Reference Id	***SETT.SPS.STM.3.4***

For each *Matching Object* present in the flow, the function creates a settlement transaction.

In the case of one single *Matching Object* in the flow, the function creates one single *Settlement Transaction* materialising both the securities and cash legs.

In the case of several *Matching Objects* in the flow associated to the *Settlement Instructions* and their linked T2S generated *Settlement Instructions,* the function creates:

- The *Settlement Transaction* materialising the cash leg and associates it to the *Matching Object* of the settlement instructions;
- The *Settlement Transactions* materialising the different securities leg and associates them to the *Matching Object* of the T2S generated settlement instructions.

When the *Settlement Instructions* are associated to a set up or an update of an earmarking, the function also associates the *Settlement Transaction* to the restriction types, securities account and security provided by the settlement instructions.

For *Settlement Instructions* corresponding to the use of a restriction, the associated restriction references are also associated to the *Settlement Transactions* **{T2S.09.190} {T2S.09.280}**.

Liquidity Information

Reference Id	***SETT.SPS.STM.3.5***

For each *Liquidity Information*, the function creates the *Settlement Transaction* to transfer the cash from a *T2S Dedicated Cash Account* to a transit account or from a transit account to a T2S dedicated cash account, or between two *T2S Dedicated Cash Accounts* of either the same payment bank or different payment banks but associated to the same RTGS account.

In case of Liquidity Information with the « All cash » information, the function does not fill any amount in the settlement transaction.

Priority attribute

Reference Id	***SETT.SPS.STM.3.6***

The priority of the created *Settlement Transactions* is based on the priority indicated in the Priority attribute of the *Settlement Instructions* and *Settlement Restrictions* **{T2S.05.145} {T2S.11.340}**.

The four different levels of priority are **{T2S.07.150}** (from the highest to the lowest):

- *"Reserved priority"* **{T2S.07.160}**;
- *"Top priority"* **{T2S.07.170}**;
- *"High priority"* **{T2S.07.180} {T2S.07.200}**;
- *"Normal priority"* **{T2S.07.190} {T2S.07.200}**.

In case of linked *Settlement Instructions,* the priority of the related linked *Settlement Transactions* is set with the highest level among the underlying *Settlement Instructions* **{T2S.07.130} {T2S.09.130}**.

All rights reserved.

The highest priority of the Settlement Instructions prevails for matched instructions with differing priorities. This is equally true when amendments to either instruction have occurred.

When the *Settlement Instruction* or *Settlement Restriction* has no priority, the *Settlement Transaction Manager* function assigns to the associated *Settlement Transactions* the level of priority according to the type of party in T2S **{T2S.11.340}**.

Settlement Transaction Category attribute

Reference Id	***SETT.SPS.STM.3.7***

This attribute is filled according to the value stored in the flows or following their analysis by this module [148]:

- In case of a *Settlement Transaction* corresponding to a blocking (including CoSD Blocking, and excluding blockings for cash regeneration) or unblocking (including EOD Cash restriction release), to a reservation or unreservation, to an earmarking or unearmarking, the Settlement Transaction Category is set to *"Restriction management"*;
- In case of a *Settlement Transaction* corresponding to the regeneration of a cash restriction CoSD Blocking for the next settlement day, the Settlement Transaction Category is set to "*Cash restriction regeneration*";
- In case of a *Settlement Transaction* associated to the original matching object, the Settlement Transaction Category is set to "*Standard*";
- In case of a *Settlement Transaction* associated to a realignment matching object, the Settlement Transaction Category is set to "Realignment";
- In case of a *Settlement Transaction* associated to a *Liquidity Transfer Order*, the Settlement Transaction Category is set to:
  - "*Liquidity Transfer All Cash*" if the "All cash" information is present in the Liquidity Information,
  - "*Liquidity Transfer*" in all other cases.

Other attributes and links [149]

Reference Id	***SETT.SPS.STM.3.8***

The following attributes and links are also filled in or created at the creation of the settlement transaction, according to the corresponding attributes of its associated *Settlement Instructions* or settlement restriction:

- Transaction Type Code: *"DVP", "FOP", "DWP", "PFOD" or "LQT"* according to the information present in the associated *Settlement Instructions, Settlement Restriction* or liquidity transfer;
- Settlement Transaction Sub-Category: set according to the information present in the associated *Settlement Instructions, Settlement Restriction* or Liquidity Transfer and to the Settlement Transaction Category calculated above;
- Is Delivered By: contains the reference to the Delivering T2S Party;

148 The following list of possible values is not exhaustive and may be updated.

149 This list of attributes is not exhaustive and may be updated at the IDFS level.

All rights reserved.

- Is Received By: contains the reference to the Receiving T2S Party;
- ISO Transaction Code: corresponds to the ISO Transaction Code of the associated settlement instruction **{T2S.05.200}**;
- Creation date: is set with the settlement day and time when the *Settlement Transaction* is created;
- Applicable Instruction Entry Date/Time: is filled with the latest entry date/time of the underlying *Settlement Instructions, Settlement Restriction* or liquidity transfer;
- When there is no *T2S Dedicated Cash Account* specified in the *Settlement Instruction,* the default cash account is retrieved from the static data and linked to the settlement transaction.

In addition the *Settlement Transaction* associated is hold, when the corresponding *Settlement Instruction* or *Settlement Restriction* is on hold.

Settlement Transaction Completion

Introduction

Reference Id	***SETT.SPS.STM.4.1***

The function complements each *Settlement Transaction* created upfront with the partial settlement and links information. These data are retrieved from the underlying *Settlement Instruction, Settlement Restriction* or liquidity information, and from the static data **{T2S.09.060}**.

Partial settlement

Reference Id	***SETT.SPS.STM.4.2***

In order to allow the partial settlement in the *Settlement* domain, the *Settlement Transaction Manager* function fills in the Partial Settlement Indicator attribute of the created settlement transactions.

This Partial Settlement Indicator attribute is set to "*Yes*" in the following cases:

- For a *Settlement Transaction* associated to a pair of settlement instructions:
  - The Transaction Type Code is set to "*FOP*" or "*DVP*" or "*DWP*" **{T2S.08.230}**,
  - The Partial Flag value of both *Settlement Instructions* is set to "*Yes*" or not filled in **{T2S.05.141}** **{T2S.08.270}**,
  - The *Settlement Instructions* is not linked to any other *Settlement Instruction* (except to realignment settlement instruction) or *Settlement Restriction* **{T2S.08.400}** **{T2S.09.120}** **{T2S.09.170}**;
- For a Settlement Transaction associated to a Settlement Restriction:
  - The *Settlement Restriction* is not a securities blocking or a cash blocking,
  - The *Settlement Restriction* is not linked to any *Settlement Instruction*;
- For a *Settlement Transaction* associated to a liquidity transfer, the Partial Execution Flag value of the associated *Liquidity Transfer* is set to "*Yes*" **{T2S.05.141}**.

For each *Settlement Transaction* created for which the Partial Settlement Indicator attribute is set to "*Yes*", the *Settlement Transaction Manager* function fills in the Threshold Type attribute.

This Threshold Type attribute is determined according to the following rules:

All rights reserved.

- For a *Settlement Transaction* for which the Transaction Type Code is set to *"FOP",* the Threshold Type is set to *"Quantity"* **{T2S.08.310}**;
- For a *Settlement Transaction* for which the Transaction Type Code is set to *"DVP"*or *"DWP"* **{T2S.08.315}**:
  - If both T2S parties have indicated in their instructions a threshold in quantity, the Threshold Type is set to *"Quantity"*,
  - Otherwise, in all other cases, the Threshold Type is set to *"Cash value"*.

Links

Reference Id	***SETT.SPS.STM.4.3***

The function also creates the settlement transaction's links. The Link Indicator attribute is stored with the value (*"After", "Before" or "With"*) in the *Settlement Transaction Link* entity **{T2S.05.147} {T2S.09.060} {T2S.09.070} {T2S.09.080} {T2S.09.090} {T2S.09.100} {T2S.09.110} {T2S.09.140}** {**T2S.09.150} {T2S.09.160} {T2S.09.290}**.

When two matched *Settlement Instructions* have no link, no link is created either to the associated *Settlement Transaction*.

For any linked *Settlement Transactions*, the function also:

- Creates the *Settlement Transaction LinkSet* which contains the Settlement Transaction Link references of *Settlement Transactions* to be settled together;
- Verifies that all the *Settlement Transactions* linked by a link *"With"* to the underlying *Settlement Instruction* are created. Otherwise, the function updates the *Settlement Instruction* reason code to "*Missing Linked Instruction*";
- Checks the following link consistency: a *Settlement Transaction* can be linked to a *Settlement Transaction* previously received unlinked. This can be done only if creating this link doesn't put the settlement engine in an inconsistent state. If possible, the existing *Settlement Transaction* is linked to the one just received with the reciprocal link type. Otherwise, the function rejects the underlying *Settlement Transaction* and updates the settlement reason code accordingly (e.g. "*Linked instruction already settled")*. Additionally, when the link creation was possible, any ongoing partial release is cancelled.

Final Checks

Introduction

Reference Id	***SETT.SPS.STM.5.1***

During the daytime, the function sends all the *Settlement Transactions* which are eligible to settlement to the *Daytime Collection Generator* function.

The Settlement Transactions not eligible for settlement are the following ones:

- Transactions having a "Missing Linked Instruction"
- Suspended transactions,

All rights reserved.

- On hold transactions not under ongoing partial release process i.e. for which no partial release was requested or for which a partial release was either cancelled or successfully settled,
- ~~**On hold transactions under ongoing partial release process when no partial settlement window is open (including sequence X),** **{CR-827}**~~
- PFOD Transactions involving a matured Security whose instructing parties are not CSDs nor CCPs
- DVP, FOP or DWP Transactions involving a matured Security whose instructing parties are not CSDs.
- Settlement Restrictions involving a matured Security whose instructing party (or instructing party of the underlying settlement instruction in case the Settlement Restriction relates to a CosD) is not a CSD.

The *Settlement Transactions* linked by a link *"With"* are sent all together whereas the *Settlement Transactions* with no link or having a link *"Before"* or *"After"* are sent separately.

Then the function also sends an *Instruction Status Information* to the *Status Management* module with the corresponding reason code for each pair of *Settlement Instructions* or *Settlement Restriction* whose associated *Settlement Transaction* is not sent to the *Daytime Collection Generator* function **{T2S.09.100} {T2S.09.110} {T2S.05.600} {T2S.05.620} {T2S.05.625}**.

Collateral valuation of a settlement transaction

Reference Id	***SETT.SPS.STM.5.2***

This step is performed after the Final Checks step as follows:

- During the daytime, the function creates the new settlement transaction, calculates its collateral valuations based on the valuation price and the own-use valuation price and sends it to the *Daytime Collection Generator* function;
- During the start of day, upon reception of the revalidated and recycled *Settlement Instructions* (and related matching object), the function does not create a new *Settlement Transaction* (since it already exists), but revaluates the securities on the existing settlement transaction;
- During the night-time, the function creates the new settlement transaction, calculates its collateral valuations, and makes it available for selection by the *Night-time Settlement* module.

In order to calculate the collateral valuations of the *Settlement Transaction* subject to this valuation, the function checks if the two involved securities positions are earmarked for a collateral purpose. In such a case, the function identifies the credit providers (Eurosystem central bank, non Eurosystem central bank or payment/settlement bank) which can use this earmarked securities position as collateral by using the related securities account's links for collateral supply.

Then, for each identified potential credit provider, the close links are checked and the function calculates the collateral valuations, based on the valuation price and the own-use valuation price for each credit provider and for the currency of the related *T2S Dedicated Cash Account* **{T2S.03.060}**.

Settlement Transaction Release

Reference Id	***SETT.SPS.STM.6.1***

The function analyses the Transaction Release received, identifies the *Settlement Transaction* specified in the flow, and then releases it.

All rights reserved.

4 – Daytime Collection Generator

Introduction

Reference Id	***SETT.SPS.DCG.1.1***

During the daytime, the *Daytime Collection Generator* function receives in each flow one or several *Settlement Transactions* (depending on their *Settlement Transaction* links), and groups them into a collection which is sent to the *Daytime Validation, Provisioning and Booking* module for their settlement attempt **{T2S.03.040} {T2S.07.090} {T2S.08.080} {T2S.07.120}**.

Collection definition/rules

Reference Id	***SETT.SPS.DCG.2.1***

A collection is composed of settlement transactions. On the basis of the links stored, the rules to create a collection are the following **{T2S.05.147} {T2S.09.110} {T2S.09.070}**:

- Settlement transactions linked by a link *"With"* are grouped all together in the same collection;
- A *Settlement Transaction* with no link or with a link *"Before"* or *"After"* is kept alone in the collection.
- ~~**A Settlement Transaction under partial release can be included into a collection only if a Partial Settlement Window is open.** {CR-827}~~

5 – Actions on Limits

Introduction

Reference Id	***SETT.SPS.AOL.1.1***

This function is in charge of updating the headroom after the update of any limit for a credit memorandum balance.

In case of decrease of auto-collateralisation limit set by a Central Bank under the already provided intraday credit, the function also triggers a forced auto-collateralisation reimbursement.

In case of an increase of any type of limit, the function informs the *Daytime Recycling and Optimisation* module of this limit increase.

A Static Data Maintenance Notification received from the *Static Data Management* domain triggers the function following a Central Bank or payment/settlement bank request to update any kind of limit defined in T2S.

Request from Static Data

Reference Id	***SETT.SPS.AOL.2.1***

A *Static Data Maintenance Notification* received from the *Static Data Management* domain **{T2S.08.800} {T2S.08.810}**, triggers the function which:

- Updates the Remaining Limit amount and the Remaining Cash To Be Reimbursed amount of the Limit Utilisation for all limit types (external guarantee, auto-collateralisation, unsecured credit) and according to the information stored in the notification;
- Sends a *Decreased CB Limits* to the *Auto-collateralisation* module if the following conditions are met **{T2S.08.800}**:

All rights reserved.

- The updated limit is an auto-collateralisation limit set by a CB,
- The update results in a negative Remaining Limit amount;

- Sends an Increase Limits Event to the *Daytime Recycling and Optimisation* module if the notification concerns an increase of a limit.

6 – Transaction Maintenance

Introduction

Reference Id	***SETT.SPS.IMA.1.1***

This function handles the maintenance of settlement transactions. It allows amending, cancelling, holding, partially releasing and releasing settlement transactions.

In the case of a release, the function is triggered by the *Transaction Release*. Upon reception, the function analyses the request and applies the maintenance on the associated settlement transaction.

In the case of a partial release, the function is triggered by the Transaction Partial Release. Upon reception, the function analyses the request and either denies the request or applies the maintenance on the associated settlement transaction.

The cases of hold, amendment and cancellation are triggered within the *Lifecycle Management and Matching* domain only.

Management of Release

Reference Id	***SETT.SPS.IMA.2.1***

In the case of a release instruction (including CoSD release instructions) which should result in the release of the associated settlement transaction, the function:

- Receives the *Transaction Release* including the reference number of the *Settlement Transaction* to be released;
- In case of a release applying on a CoSD on Hold, identifies all related *Settlement Transactions* that are to be released together;
- In case of a release applying on a Settlement Transaction under ongoing Partial Release, cancels the ongoing Partial Release;
- Sends the information to the *Settlement Transaction Manager* function.

Management of Hold, Amendment and Cancellation

Reference Id	***SETT.SPS.IMA.3.1***

The function allows either holding, amending or cancelling the settlement transaction, updating the Settlement Transaction Status and cancelling the ongoing partial release of the settlement transaction if relevant:

- In the case of a hold, the function holds the settlement transaction; the Settlement Transaction Status is not updated and any ongoing partial release of the settlement transaction is cancelled;

All rights reserved.

- In the case of amendment or cancellation, the function updates the relevant attributes of the settlement transactions, concerning either the settlement transaction's status, priority, partial settlement and/or links information;
- In the case of amendment concerning links, any ongoing partial release of the settlement transaction is cancelled;
- In the case of amendment concerning partial, rendering the settlement transaction not allowed to partial settlement anymore, any ongoing partial release of the settlement transaction is cancelled;
- In the case of amendment concerning partial, rendering the settlement transaction submitted to cash threshold, if the remaining released quantity is not sufficient regarding the cash threshold, any ongoing partial release of the settlement transaction is cancelled.

In those two cases, this function is triggered only within the *Lifecycle Management and Matching* domain in a synchronous manner.

Management of Partial Release

Reference Id	SETT.SPS.IMA.4.1

In the case of a partial release instruction which should result in the release of part of the quantity of the associated settlement transaction, the function:

- Receives the Transaction Partial Release including the reference number of the Settlement Transaction to be partially released and the quantity to be released;
- Checks if the partial release is applicable;

If partial release is not applicable, the partial release is denied.

In any case, Lifecycle Management and Matching domain is informed of the result of the checks.

7 – Cut-Off Processing

Introduction

Reference Id	***SETT.SPS.COP.1.1***

The *Cut-off Processing* function aims at preventing the settlement transactions' recycling within the *Settlement* domain.

It is triggered at the reception of the following events from the *Scheduling* module:

- The *Events (Intraday DVP Cut-off);*
- The *Events (Intraday FOP Cut-off);*
- The *Events (Intraday BATM Cut-off);*
- The *Events (Intraday Cash Settlement Restrictions Cut-off);*
- The *Events (Intraday Securities Settlement Restrictions Cut-off);*
- The *Events (Intraday CBO Cut-off);*
- The *Events (Inbound Liquidity Transfer Cut-off).*

All rights reserved.

Some of these events can be received many times, if they are currency-dependent: in this case, each event will be received once per currency open during the business day.

After receiving each of these events, the function carries out several actions in a coordinated and specific order.

Checking the settlement attempt in Settlement domain

Reference Id	***SETT.SPS.COP.2.1***

On the basis of the Applicable Instruction Entry Date/Time of all settlement transactions, i.e. the entry date/time in T2S of the *Settlement Instructions,* the function checks that the *Settlement Transactions* whose associated *Settlement Instructions* have entered T2S before the event have been submitted at least once to a settlement attempt during the settlement day.

- The *Settlement Instructions* concerned are **{T2S.05.610} {T2S.05.625}**:
- At the reception of an *Event (Intraday DVP Cut-off)*, the "DVP", "DWP" or "PFOD" *Settlement Instructions* **{T2S.03.250} {T2S.07.100}**;
- At the reception of an *Event (Intraday FOP Cut-off),* the Free Of Payment *Settlement Instructions* **{T2S.03.280} {T2S.07.110}**;
- At the reception of an *Event (Intraday BATM Cut-off),* the Bilaterally Agreed Treasury Management *Settlement Instructions* **{T2S.03.270} {T2S.07.110}**;
- At the reception of an *Event (Intraday Cash Settlement Restrictions Cut-off)*, the *Settlement Restrictions* on cash;
- At the reception of an *Event (Intraday Securities Settlement Restriction Cut-off)*, the *Settlement Restrictions* on securities;
- At the reception of an *Event (Intraday CBO Cut-off), t*he Central Bank Operations **{T2S.03.290} {T2S.07.110}**.
- At the reception of an *Event (Inbound Liquidity Transfer Cut-off)*, the *Liquidity Transfers*.

Stops integrating the selected settlement transactions into a collection

Reference Id	***SETT.SPS.COP.3.1***

The *Cut-Off Processing* function stops the process of *Settlement Transactions* being sent to the *Daytime Collection Generator* function which integrates the *Settlement Transactions* in a collection.

Suspending settlement transactions "Unsettled" or "Partially settled"

Reference Id	***SETT.SPS.COP.4.1***

The function suspends from settlement the "*Unsettled*" or "*Partially Settled*" *Settlement Transactions* of the current settlement day, depending on the cut-off event received:

All rights reserved.

- At the reception of an *Event (Intraday DVP Cut-off)*, the function first proceeds to the credit propagation process [150] and then suspends the *"Unsettled"* or *"Partially Settled" Settlement Transactions* with a Transaction Type Code set to *"DVP"*, "DWP" or "PFOD". **{T2S.07.100}**;
- At the reception of an *Event (Intraday FOP Cut-off)*, the function suspends the *"Unsettled"* or *"Partially Settled" Settlement Transactions* with a Transaction Type Code set to "FOP" **{T2S.07.110}**;
- At the reception of an *Event (Intraday BATM Cut-off)*, the function suspends the *"Unsettled"* or *"Partially Settled"* BATM *Settlement Transactions* (except if Transaction Type Code is set to "FOP") **{T2S.07.110}**;
- At the reception of an Event (Intraday Cash Settlement Restrictions Cut-off), the function suspends the *Settlement Transactions* for cash restrictions;
- At the reception of an Event (Intraday Securities Settlement Restrictions Cut-off), the function suspends the *Settlement Transactions* for securities restrictions;
- At the reception of an *Event (Intraday CBO Cut-off)*, the function suspends the remaining *"Unsettled"* or *"Partially Settled" Settlement Transactions* with an ISO Transaction Code set to "CNCB"(except if Transaction Type Code is set to "FOP") [151] **{T2S.07.110}**;
- At the reception of an *Event (Inbound Liquidity Transfer Cut-off)*, the function suspends the *Settlement Transactions* for liquidity transfers

The suspension of a Settlement Transaction related to matched Settlement Instructions or to Settlement Restrictions leads to inform Lifecycle Management and Matching domain in order to report the "Failing" ISO settlement status.

The *Cut-Off Processing* function also suspends the *Settlement Transactions* whose associated *Settlement Instructions* have entered T2S after the reception of the event.

The suspension of the *Settlement Transactions* ensures that these *Settlement Transactions* are not recycled in the *Daytime Recycling and Optimisation* module **{T2S.03.250} {T2S.03.270} {T2S.03.280} {T2S.03.290}**.

End of auto-collateralisation availability

Reference Id	***SETT.SPS.COP.5.1***

At the closure of any *Event (Intraday DVP Cut-off)* whichever its currency, the function suspends the availability of the auto-collateralisation function for the currency of the Intraday DVP cut-off received.

End of process event

Reference Id	***SETT.SPS.COP.6.1***

After the processing of each cut-off event, the function sends an end of process event to the *Scheduling* module.

Specific events

150 The credit propagation process consists of ensuring that all credits stemming from transactions submitted to IDVP cut-off are propagated through recycling until the end of the propagation or until a scheduled time

151 The complete list of codes eligible for the CBO identification will be defined at the detailed specification phase.

All rights reserved.

Reference Id	***SETT.SPS.COP.7.1***

In case of exceptional circumstances, the *Cut-Off Processing* function handles dedicated cut-off events per currency, when CBs require the possibility to change the daily event scheduling deadline **{T2S.03.303}**.

Cancel uncompleted partial releases

Reference Id	SETT.SPS.COP.8.1

When a cut-off is reached, the Cut-Off Processing function cancels all the ongoing partial releases regarding the transactions submitted to this cut-off.

8 – EOD/SOD Processing

During the End of Day and Start of Day periods, this function receives different events from the *Scheduling* module and starts actions according to each of them.

End of day cash restriction release information

Reference Id	***SETT.SPS.ESP.1.1***

At the reception of an *Event (EOD Cash Restrictions Release)* from the *Scheduling* module, this function handles the release of any reserved/blocked *Cash Balance* that has not been used during the settlement day and related to the currency of the EOD Cash Restrictions Release received **{T2S.07.380}**.

For each cash restriction release, the function:

- Creates the *Settlement Restrictions* by using the original *Settlement Restriction* reference generated at the set-up processing:
  - For releasing the detailed cash restrictions not used during the current settlement day, which corresponds to a decrease of the cash restrictions,
  - For regenerating the detailed cash restrictions released (only for CoSD related cash blocking), with an intended settlement date set to the next settlement, which corresponds to an increase of the CoSD blocking restrictions on cash **{T2S.09.240}**,
  - In each case, a notification is sent to the Status Management module to inform about the creation of the settlement restriction;
- Sends the created *Settlement Restriction* which releases the detailed cash restriction to the Settlement Transaction Manager function;
- Once all *Settlement Transactions* which released the detailed cash restrictions are settled in the *Daytime Validation, Provisioning and Booking* module, the function sends an *End of Process Event (EOD cash restrictions release)* to the *Scheduling* module.

Stop the creation of the collections

Reference Id	***SETT.SPS.ESP.2.1***

The function receives an *Event (Stop Collection Creation)* from the *Scheduling* module and triggers the stop of *Settlement Transactions* being sent by the *Settlement Transaction Manager* function to the *Daytime Collection Generator* function.

All rights reserved.

Once the process of stopping the creation of collections is finished, the function sends an *End of Process Event (Stop Collection Creation)* to the *Scheduling* module.

Start of day processing

Reference Id	***SETT.SPS.ESP.3.1***

Following the reception of an *Event (SOD)* from the *Scheduling* module at the start of a new settlement day, the function verifies that all *Matching Objects* and their related *Settlement Instructions* as well as all *Settlement Restrictions* and liquidity information received in the *Standardisation and Preparation to Settlement* module are processed, i.e. the corresponding *Settlement Transactions* are created, or in case of recycling, the *Settlement Transactions* are updated with the new collateral valuation.

Once these actions are completed, the function sends an End of Process *Event (SOD)* to the *Scheduling* module.

Valuation of earmarked securities positions for collateral purposes

Reference Id	***SETT.SPS.ESP.4.1***

Following the reception of an *Event (SOD)* from the *Scheduling* module, the function identifies the credit providers (Central Bank or payment/settlement bank) which can use each earmarked securities position for collateral purpose. This identification is done according to the related securities account's links for collateral supply.

For each *Securities Position Valuation* regarding each earmarked securities position, the function verifies that the collateral instructions shall be valid, notably regarding auto-collateralisation eligibility criteria (e.g. *Close links* without an applicable valuation).

For each identified potential credit provider, the function calculates the new valuations:

- Standard valuation based on the valuation price for the securities where there is no close link between the credit consumer and the security provided as collateral;
- Own-used valuation based the own-use valuation price for the securities where there is a close link between the credit consumer and the security provided as collateral.

Both valuations are calculated for each earmarked securities position by multiplying the quantity available at the start of day on the securities account by the collateral price retrieved from the credit provider **{T2S.03.060}**.

Start the creation of the collections

Reference Id	***SETT.SPS.ESP.5.1***

The function receives an *Event (Beginning of Daytime)* from the *Scheduling* module at the beginning of the daytime settlement period **{T2S.03.160}** **{T2S.08.030}** and then, for *Settlement Transactions* which were created after the beginning of the last sequence of the night-time settlement period:

- Retrieves them and selects them all;

All rights reserved.

- Sends them, according to the entry order in T2S of the associated *Settlement Instructions* and their related matching objects, in collections to the *Daytime Collection Generator* function **{T2S.03.090}** **{T2S.03.110}**.

Once these actions are completed, the function sends an End of Process Event (Beginning of Daytime) to the *Scheduling* module.

Replication

Reference Id	***SETT.SPS.ESP.6.1***

Following the reception of an Event (SOD) from the Scheduling module, the function identifies all the Static Data that need to be used in Settlement.

For each Static Data, the function checks its validity against the current business date.

For each valid Static Data, the function propagates them into the Settlement domain.

The following objects are replicated:

- Relevant Parties and their related System Entity
- All Settlement Priority Default rules from *Attributes Domain*
- All CosD rules and rule sets
- All Restriction Types related to *Cash Balances*, *Securities Positions*, *Intraday Restrictions* and *Earmarking* at Securities Account Level
- All Securities and Securities accounts for which Securities Positions exist
- All T2S Dedicated Cash Accounts
- All CMBs with their auto-collateralisation configuration (Receiving Securities account, Regular Securities account, ACO rule, NCB T2S Dedicated Cash Account)
- All information relative to Securities Auto-collateralisation Eligibility (Close Link without an applicable valuation, Eligible Counterpart CSD Securities, Securities Valuations, Securities ACO eligibility)

3.5.5.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Transaction Release	In	Transaction release including the reference of the settlement transaction to be released	LCMM: Instruction Maintenance	
Matching Object/ Settlement Instructions	In	Matching Object and additional information needed to analyse the realignment and to	LCMM: Instruction Validation	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
		create/revalidate the settlement transactions		
Settlement Restriction	In	Settlement restriction information needed to create/revalidate the settlement transaction	LCMM: Instruction Validation	
Settlement Restriction	In	Settlement restriction information needed to unblock a COSD blocking for a CoSD Cancellation	LCMM: Instruction Maintenance	
Matching Object/ Settlement Instructions	In	Matching Object and additional information needed to analyse the realignment and to create the settlement transactions	LCMM: Instruction Matching	
Matching Object/ Settlement Instructions	In	Matching Object and updated fields of Settlement Instruction affected by a maintenance	LCMM: Instruction Maintenance	
Instruction Status Information	Out	Notify the creation, hold, amendment, cancellation or the non-presentation of a settlement transaction in a collection due to the ineligibility of Settlement Instructions		LCMM: Status Management
Collection	Out	Object in which the settlement transactions created are put together according to their links		SETT: Daytime Validation, Provisioning and Booking

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Static Data Maintenance Notification	In	Request received for applying a limit update once it is done in static data	SDMG: Static Data	
Decreased CB Limits Event	Out	Request sent in case of a negative remaining limit amount		SETT: Auto-collateralisation
Increase Limits Event	Out	Notification of a static data update for an increase of a limit		SETT:R&O
Liquidity Information	In	Coming from Liquidity Operations module, and related to liquidity transfers	LQMG: Liquidity Operations	
Event (Intraday DVP Cut-off)	In	On reception of the event, the SPS module stops sending the settlement transactions related to Intraday DVP	OPSR: Scheduling	
End of Process Event (Intraday DVP Cut-off)	Out	Event sent after the processing of the DVP cut-off event in SPS		OPSR: Scheduling
Event (Intraday FOP Cut-off)	In	On reception of the event, the SPS module stops sending the settlement transactions related to FOP	OPSR: Scheduling	
End of Process Event (Intraday FOP Cut-off)	Out	Event sent after the processing of the FOP cut-off event in SPS		OPSR: Scheduling
Event (Intraday BATM Cut-off)	In	On reception of the event, the SPS module stops sending the settlement transactions related to BATM	OPSR: Scheduling	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
End of Process Event (Intraday BATM Cut-off)	Out	Event sent after the processing of the BATM cut-off event in SPS		OPSR: Scheduling
Event (Intraday CBO Cut-off)	In	On reception of the event, the SPS module stops sending the settlement transactions related to CBO	OPSR: Scheduling	
End of Process Event (Intraday CBO Cut-off)	Out	Event sent after the processing of the CBO cut-off event in SPS		OPSR: Scheduling
Event (Intraday Cash Settlement Restrictions Cut-off)	In	On reception of the event, the SPS module stops sending the settlement transactions related to settlement restrictions on cash	OPSR: Scheduling	
End of Process Event (Intraday Cash Settlement Restrictions Cut-off)	Out	Event sent after the processing of the Intraday Cash Settlement Restrictions Cut-off event in SPS		OPSR: Scheduling
Event (Intraday Securities Settlement Restrictions Cut-off)	In	On reception of the event, the SPS module stops sending the settlement transactions related to settlement restrictions on securities	OPSR: Scheduling	
End of Process Event (Intraday Securities Settlement Restrictions Cut-off)	Out	Event sent after the processing of the Intraday Securities Settlement Restrictions Cut-off event in SPS		OPSR: Scheduling
Event (EOD Cash Restrictions Release)	In	On reception of the event, the SPS module releases the unused cash restrictions	OPSR: Scheduling	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
End of Process Event (EOD Cash Restrictions Release)	Out	Event sent once all "EOD cash restrictions release" settlement transactions have been settled		OPSR: Scheduling
Event (Stop Collection Creation)	In	On reception of this event, the SPS module stops the creation of the collections	OPSR: Scheduling	
End of Process Event (Stop Collection Creation)	Out	Event sent once the process of creating collections is stopped		OPSR: Scheduling
Event (SOD)	In	On reception of this event, the SPS module ensures that all the settlement preparation is done before starting the Night-time period	OPSR: Scheduling	
End of Process Event (SOD)	Out	Event sent once the whole preparation process is done		OPSR: Scheduling
Event (Beginning of Daytime)	In	On reception of this event, the SPS module restarts the creation of collections	OPSR: Scheduling	
End of Process Event (Beginning of Daytime)	Out	Event sent after the processing of the Beginning of Daytime in SPS is done		OPSR: Scheduling
Event (Liquidity Transfer Cut-off)	In	On reception of the event, the SPS module stops sending the settlement transactions related to Liquidity Transfers	OPSR: Scheduling	

All rights reserved.

3.5.5.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
STATIC DATA			
Realignment	Party	Read	(Participant)
	Securities	Read	
	Security CSD Link	Read	
	T2S Dedicated Cash Account	Read	
	Securities Account	Read	Identification of the Investor and Technical Issuer CSDs
	CSD Account Link	Read	Identification of the Mirror, Omnibus, Issuance and Inter CSD accounts
	Party Securities Account Relationship	Read	
CoSD Condition	Securities Code	Read	ISIN
	Currency	Read	
	Country	Read	Country of Issuance, Deliverer Location, Receiver Location, Issuer CSD Location
	Securities Account	Read	
	CoSD Rule Set	Read	
	CoSD Rule	Read	
	CoSD Rule Parameter	Read	
	CoSD Processing Condition	Read	
Restriction	Restriction Type	Read	
Dynamic DATA			
Liquidity	Liquidity Transfer	Read	
Settlement Restriction	Restriction Reference	Read	
	Settlement Restriction	Read/Write	
	Securities Position	Read	
	Cash Balance	Read	
Settlement Instruction	LCMM Instruction	Read	
	Settlement Instruction	Read/Write	
	Matching Object	Read	

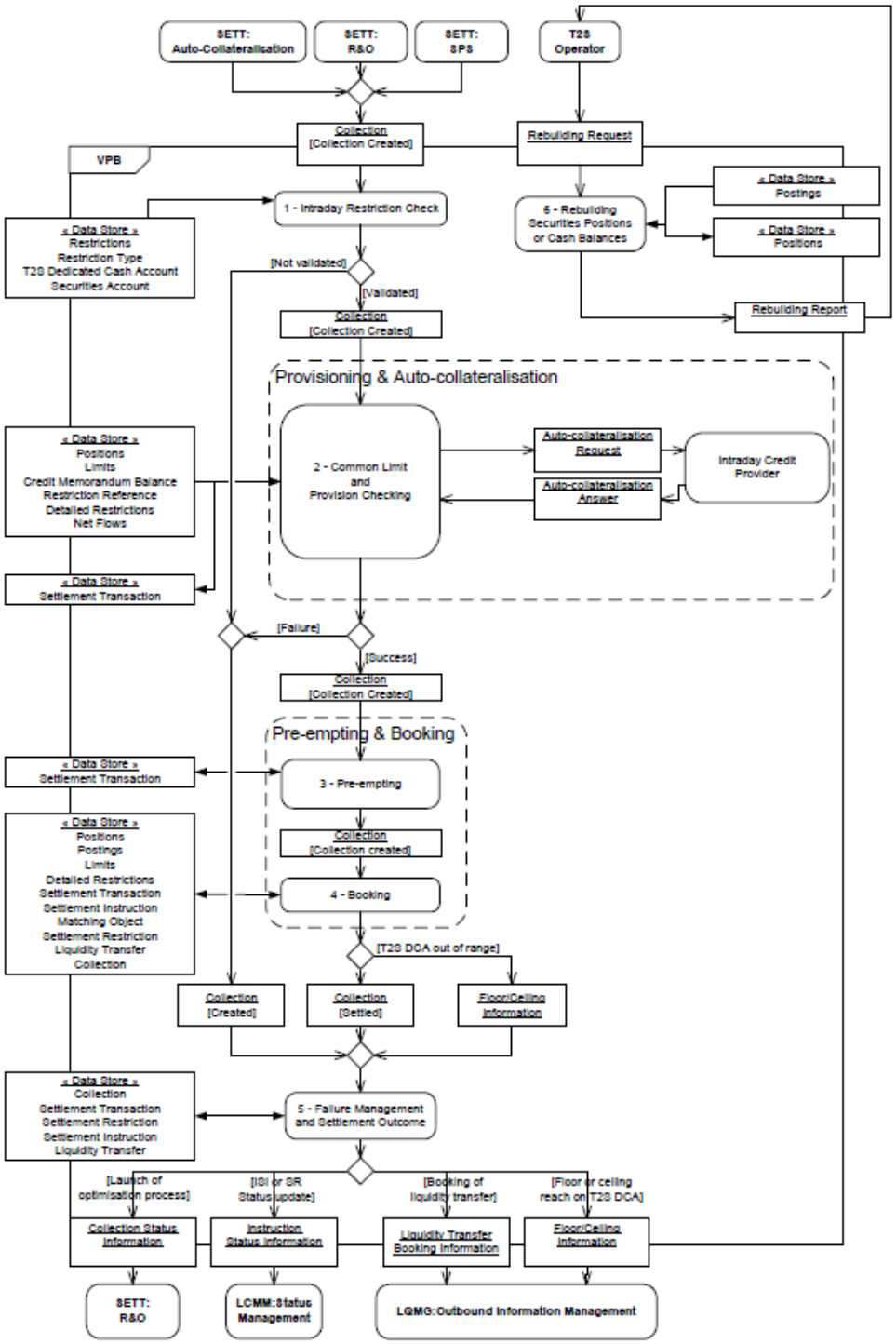
All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
	Hold/Release Instruction	Read	
	LCMM Instruction Link	Read	
Settlement Transaction	Settlement Transaction	Read/Write	Settlement transaction associated to the matching object and Settlement Instructions received from LCMM Settlement transaction associated to the settlement restriction received from LCMM Cash restriction release settlement transaction Regenerating of cash restriction settlement transaction Settlement transaction associated to CoSD blocking/unblocking Settlement transaction associated to liquidity transfers.
	Settlement Transaction Link	Read/Write	Copy of the settlement instruction link data
	Settlement Transaction Link-Set	Read/Write	Determined from Settlement Instruction Link data and Settlement Transaction Link
Collection	Collection	Read/Write	Group of (linked) settlement transactions
Limit	Limits Utilisation	Read/Write	Remaining Limit Remaining Cash to be Reimbursed
Securities Position Valuation	Securities Position Valuation	Read/Write	
Securites Valuation	Securites Valuation	Read	

All rights reserved.

3.5.6 Daytime Validation, Provisioning and Booking

3.5.6.1 Diagram of the module



3.5.6.2 Description of the module

The core purpose of the *Daytime Validation, Provisioning and Booking* module is to update in a centralised process the securities positions and cash balances with the cash and securities movements of the incoming collection of *Settlement Transactions* on the basis of available resources **{T2S.07.240}**.

All rights reserved.

This module describes the general behaviour for the daytime period only. The specificities of the night-time period are described in the *Night-time Settlement* module.

Whatever their origin (*Settlement Instruction, Settlement Restriction* or liquidity transfer), the *Settlement Transactions* coming into the *Daytime Validation, Provisioning and Booking* module through a collection, are processed in the same way.

To that purpose, the module receives collections (see definition into the introduction of the domain) containing a set of settlement transactions, either for their first settlement attempt or following optimisation process.

The module settles all the received *Settlement Transactions* of a collection in an "all or none" basis **{T2S.09.080}** and does not attempt any selection or de-selection inside a collection.

It performs on each collection:

- Validation, with the *Intraday Restriction Check* function (function 1) which checks that no intraday restriction is applicable on any of the traded ISIN, accounts or T2S Party involved in the collection and checks the consistency of the *Settlement Transactions* in the collection;
- Provisioning, with the *Common Limit and Provision Checking* function (function 2) which:
  - Adds to the collection the *Settlement Transactions* representing the use of restricted resources requested by the T2S Parties,
  - Checks if the net flows (i.e. the sum using technical netting of debits and credits movements on a considered securities position or cash balance) can be settled with the resources available on the securities positions, or cash balances,
  - In case of *Settlement Transaction* regarding a client of a payment/settlement bank [152], checks the limits for the credit memorandum balance, i.e. the external guarantee limit, the auto-collateralisation limit set by a payment/settlement bank and the unsecured credit limit set by its payment/settlement bank,
  - In case of lack of cash, checks the auto-collateralisation limit set by the CB,
  - Calls the *Intraday Credit Provider function* of the *Auto-collateralisation* module in case of insufficient external guarantee headroom or lacks of cash or securities [153];
- Pre-emption, with the *Pre-empting* function (function 3) which pre-empts the incoming resources in case of reservation partially filled;
- Booking, with the *Booking* function (function 4) which updates the securities positions, (with their valuation in case of earmarked for auto-collateralisation securities position) and cash balances with the cash and securities movements involved in the settlement transactions, and which updates the journaling limits.

152 i.e. the owner of the T2S Dedicated cash account and the owner of Securities Account have a different BIC.

153 In case of securities already collateralised.

All rights reserved.

Lastly, the *Failure Management and Settlement Outcome* function (function 5) provides the other modules and domains with the result of the settlement process. The status of each *Settlement Transaction* has one of the following values:

- "*Settled*", in case of successful settlement;
- "*Partially Settled*", in case of partial quantity or amount settled;
- "*Partially Filled*", in case of a *Settlement Transaction* for a reservation is not totally satisfied;
- "*Unsettled*", in case of settlement failure.

Meanwhile, the collection is unsettled if at least one *Settlement Transaction* fails or settled otherwise.

The status of each related *Settlement Instruction, Settlement Restriction* and *Liquidity Transfer* is updated with the same status as its corresponding *Settlement Transaction* in case of successful or partial settlement. In case of failure, the status value remains as "*Unsettled*".

Irrespective of the result of the settlement attempt, the result of settlement of a collection (*Collection Status Information*) is sent to the *Daytime Recycling and Optimisation* module.

In addition to its standard functions, the *Daytime Validation, Provisioning and Booking* module performs the *Rebuilding Securities Positions or Rebuilding Cash Balances* function (function 6) when receiving the request from T2S Operator.

Management of the consistency of the resources

In order to ensure consistency from the first validation to the final booking, the *Daytime Validation, Provisioning and Booking* module locks the resources used for the current settlement attempt (i.e. settlement transactions, limit utilisations, securities positions and cash balances) in order to avoid any update from another settlement process.

Management of the restricted securities positions and cash balances (reservation, blocking and earmarking)

The settlement process may involve *Settlement Transactions* that aim at setting up new restriction, updating existing restriction [154] and using these restricted resources.

Management of a reservation

The set up of a new reservation (securities or cash) is achieved with the booking of the relevant settlement transaction. The module credits the restricted securities position or cash balance, creates the detailed securities/cash restriction with the corresponding quantity or amount and generates a restriction reference [155] (see the *Booking* function and the *Pre-empting* function). If the quantity/amount available into the securities position or cash balance is not sufficient to cover the quantity of securities and/or the amount of cash specified in the reservation *Settlement Restriction,* the reservation is filled in with quantity/amount available. The remaining unsettled part is filled in with the incoming securities and/or cash **{T2S.07.351}**.

154 An existing restriction can be either increased or decreased (including cancelled).

155 This internal reference created by T2S and sent back to the T2S Party will be the only one to use by this party to identify the restriction for a use or an increase/decrease

All rights reserved.

The update (increase or decrease) of an existing reservation (securities or cash) is achieved with the booking of the relevant *Settlement Transaction* with the restriction reference indicated.

The use of the reserved securities positions/cash balance **{T2S.06.120}** is detected when the *Settlement Transaction* contains one or several restriction references. In this case, the quantity or amount necessary to settle the trade is checked (refer to the *Common Limit and Provision Checking* function and notably the use of restricted position).

Management of a blocking

The set up of a new blocking and the update (increase or decrease) of an existing blocking (securities or cash) is achieved with the same process as the one described for the reservation. The only difference is that in case of insufficient securities and/or cash, the associated *Settlement Transaction* remains unsettled. The *Settlement Transactions* associated to a blocking may be recycled or optimised **{T2S.07.352}**.

The use of blocked resources is handled as the use of reservation.

Management of earmarking

The set up and the use of an earmarking are managed by identifying the restriction type referred in the settlement instruction.

The set up of a new earmarked securities position or the update (increase or decrease) of an existing earmarked securities position is achieved:

- Either by sending a settlement restriction. The module behaves as for blocking, but in case of insufficient securities the restriction is partially executed with the resources available and the remaining part is not filled afterwards. Contrary to the blocking or reservation processes, no restriction reference is managed;
- Or by sending a *Settlement Instruction* **{T2S.08.651}**.

When a *Settlement Instruction* requires delivering the securities from an earmarked position, the quantity necessary to settle the trade is checked and if the provision-checking is negative, the settlement fails.

Management of the partial settlement (settlement instructions)

For collection coming from the *Daytime Recycling and Optimisation* module and containing only a single *Settlement Transaction* and the relevant potential [156] realignment settlement transactions, if the collection cannot be settled for the total quantity of securities, the *Daytime Validation, Provisioning and Booking* module tries to settle partially, on the basis of the available resources, the *Settlement Transaction* of this collection.

This process of partial settlement applies only during a partial settlement window and after a negative answer from Intraday Credit Provider functions. It is performed according to specific criteria such as the partial settlement threshold as defined in the *Daytime Recycling and Optimisation* module.

156 The realignment *Settlement Transactions* exist only in case of Cross-CSD operations.

All rights reserved.

For all other collections containing more than one settlement transaction, the partial settlement of a *Settlement Transaction* is handled (and described) in the *Daytime Recycling and Optimisation* module since it applies to unsettled *Settlement Transactions* **{T2S.08.210}** and depends on specific windows and several other criteria. In such partial settlement, the *Daytime Validation, Provisioning and Booking* module attempts to settle with the partial quantities calculated by the *Daytime Recycling and Optimisation* module.

In case of failure, the calculated partial quantity is not taken into account for the next settlement attempt. If the *Settlement Transaction* can be settled for the partial quantity, the *Booking* function updates the securities positions and cash balances with this partial quantity and not with the initial quantity.

The optimisation for the settlement of the remaining quantity or amount is carried out by the *Daytime Recycling and Optimisation* module processes **{T2S.08.240}**.

Management of the Partial settlement of Partially Released Settlement Instructions

A Settlement Instruction on Party Hold may be partially released to allow the partial settlement of a specified quantity. This Partially Released Settlement Instruction must conform to all the conditions of partial settlement as for any other Settlement Instruction. The partial settlement of Partially Released Settlement Instructions ~~**will only**~~ **can**{CR-827} occur **outside a partial settlement window for the total quantity of the released instruction, as well as** during a partial settlement window **for the total or partial quantity of the released instruction**{CR-827} and until the relevant cut-off time (partial release is only valid for the current business day) at which point the partial release will be cancelled and the underlying Settlement Instruction set back on Party Hold for the full unsettled quantity.

Management of the partial execution during the settlement process (settlement restrictions or liquidity transfers)

Complementary to the partial settlement process described above, a few *Settlement Transactions* are subject to a partial execution during their settlement (*Liquidity Transfer* with the partial indicator set to "*Yes*", *earmarking*, *reservation*). These partial executions are not submitted to a specific window, threshold or T2S Party agreement.

When applying to *Settlement Transaction* subject to partial execution, the *Common Limit and Provision Checking* function can fill in the partial quantity or amount of the considered *Settlement Transaction* with the available quantity or amount.

The *Booking* function updates the securities positions and cash balances with this partial quantity or amount instead of the initial quantity or amount.

Management of the collection composed with both settlement restriction and settlement instruction

A collection with only a *Settlement Transaction* related to a *Settlement Restriction* is always settled and the *Settlement Transaction* never recycled.

But it is possible that *Settlement Instruction* and *Settlement Restriction* are linked by a "with" link by the T2S Party. When settling such a collection, several cases can occur, but whatever the case, the *Daytime Validation, Provisioning and Booking* module behaves in the same way as for a standard collection.

All rights reserved.

According to the "all or none" basis settlement rules and taking into account the possible partial execution of the settlement restriction:

- If the resources available are sufficient to settle the *Settlement Transaction* related to the *Settlement Instruction* and not to settle the *Settlement Restriction,* the module achieves the settlement in either of the following ways :
  - The *Settlement Transaction* related to the *Settlement Instruction* is settled and the *Settlement Transaction* related to the *Settlement Restriction* is partially filled in case of a set-up or increase of reservation and settled in case of restriction regarding earmarking, or in case of decrease of reservation,
  - In case of a *Settlement Restriction* regarding blocking, the module fails to settle both *Settlement Instruction* and *Settlement Restriction*;
- If the resources available are not sufficient to settle the *Settlement Transaction* related to the *Settlement Instruction,* whatever the state of the *Settlement Restriction,* the module fails to settle both *Settlement Instruction* and settlement restriction.

As for any other *Settlement Transaction* allowed to, intraday credit is provided only for lack (of cash or of securities) concerning the *Settlement Transaction* related to a settlement instruction.

Management of the Rebalancing Liquidity following Corporate Action or Monetary Policy Operation

A collection can contain one or several *Settlement Transactions* related to corporate actions **{T2S.06.111}** **{T2S.06.112}** or central bank monetary policy operations (in repo countries) **{T2S.06.630}**. They are identified by their ISO transaction code.

In case of monetary policy operations, the rebalancing of the credited cash to the RTGS is mandatory.

In case of corporate action, the rebalancing of the credited cash to the RTGS is performed if the credited party has subscripted to the Corporate Action Rebalancing Liquidity (CARL) and if the amount to be credited is equal to or above the minimum threshold configured on the standing order for rebalancing.

In both cases, the module generates an additional *Settlement Transaction* in order to automatically transfer the received cash into the relevant RTGS Dedicated transit account **{T2S.06.117}**. This *Settlement Transaction* is added to the collection to be settled together with the *Settlement Transactions* related to corporate actions **{T2S.06.118}** or monetary policy operations **{T2S.06.630}**.

As a result an immediate *Liquidity Transfer* is created with the Settlement Status set to "*Settled*".

Any liquidity generated by secured money market trades, bilaterally agreed treasury management instructions or CB operations will not be re-used for other settlement purposes in T2S (i.e. recycling of DVP failures) **{T2S.03.270}** **{T2S.07.110}**.

3.5.6.3 Description of the functions of the module

1 – Intraday Restriction Check

Introduction

Reference Id	***SETT.VPB.IRC.1.1***

All rights reserved.

The *Intraday Restriction Check* function makes the last consistency and validity checks before the booking.

More specifically, it:

- Checks if no intraday restriction is posted on any *Settlement Transaction* of the collection;
- Checks the consistency of the *Settlement Transactions*.

The function reads the first collection in the input queue in a FIFO mode and retrieves all *Settlement Transactions* that are contained in this collection in order to attempt to settle them in an "all or none" basis.

If any *Settlement Transaction* in the collection has the status "*Cancelled*" or "*Settled*", or if it is hold or suspended, the function considers the collection as unsettled and reads the following one in the queue.

Intraday restrictions check

Reference Id	***SETT.VPB.IRC.2.1***

The function checks on the *Settlement Transactions* of the collection that no *Intraday Restriction* considering their potential rules applies on the following entities **{T2S.07.260}**:

- *Securities* **{T2S.16.510}**;
- *Securities Account* **{T2S.16.680}**;
- *T2S Dedicated Cash Account* **{T2S.16.658}** **{T2S.16.680}**;
- T2S Party set by a CSD regarding the owners of the securities accounts **{T2S.16.680}**;
- T2S Party set by a CB regarding the owners of the *T2S Dedicated Cash Accounts* used **{T2S.16.680}**.

If at least one intraday *Settlement Restriction* is detected, the collection is sent to the *Failure Management and Settlement Outcome* function with the reason [157] for failure for each settlement transaction.

Consistency checking

Reference Id	***SETT.VPB.IRC.3.1***

The function performs the following checks on the consistency of the *Settlement Transactions* contained in the collection.

- If a *Settlement Transaction* in the collection is linked with an "after" link to another one, the function checks that the linked *Settlement Transaction* is either in a "*Settled*" status or contained in the same collection [158] **{T2S.05.186}**;
- ~~**If a Settlement Transaction in the collection is under partial release, the function checks that a partial settlement window is currently opened.**~~**{CR-827}**

157 Since their result strongly depends on the context of the settlement attempt (i.e. securities positions, cash balances and composition of the collection) and not only on the content of the *Settlement Instructions,* the checks in the *Daytime Validation, Provisioning and Booking* module are performed successively till a failure occurs. In this case, for a given collection, a single reason code is communicated for each settlement transaction. This reason code corresponds to the reason code applicable for the related *Settlement Transaction* due to this failure.

158 A *Settlement Transaction* linked with an "after" link to another one can never be settled before the linked settlement transaction. If this one is not submitted in the same collection or not already settled, the *Settlement Transaction* with the "after" link should not to be submitted to a settlement attempt.

All rights reserved.

If at least one *Settlement Transaction* in the collection does not respect any of the aforementioned conditions, the collection is sent to the *Failure Management and Settlement Outcome* function with the reason for failure for each settlement transaction.

Once the checks performed with success on all transactions, the function sends the collection to the *Common Limit and Provision Checking* function.

2 – Common Limit and Provision Checking

Introduction

Reference Id	***SETT.VPB.CLP.1.1***

The *Common Limit and Provision Checking* function checks whether a collection is settable or not by calculating the provision net flows regarding the impacted:

- Securities positions;
- Cash balances;
- Credit memorandum balances;

that, are then checked against the available securities, cash, and external guarantee headroom **{T2S.16.616} {T2S.16.650}**.

The objective is to submit a set of *Settlement Transactions* contained into a collection to a settlement attempt by executing chronologically the following steps:

- Preparing the *Settlement Transactions* corresponding to immediate *Liquidity Transfers* related to corporate action or monetary policy settlement transactions;
- Calculating the provisioning net flows (sum of debits and credits for each involved securities position, cash balance and credit memorandum balances);
- Preparing the use of restricted resources (restricted securities positions and/or restricted cash balances) **{T2S.10.030} {T2S.10.040}**;
- Executing the provision-checking from the provisioning net flows **{T2S.07.274} {T2S.07.260}** including, in case of *Settlement Transaction* related to a client of a payment/settlement bank, the checks of the applicable external guarantee headroom;
- In case of lacks of securities or cash, or in case of insufficient external guarantee headroom, detected following the provision-checking:
  - Sending a request to the *Intraday Credit Provider* function for auto-collateralisation if the relevant conditions are fulfilled,
  - And/or partialising the *Settlement Transactions* which can be either partially executed or partially settled during their settlement.

Preparation of Corporate Action and Monetary Policy Rebalancing Liquidity transfers

Reference Id	***SETT.VPB.CLP.2.1***

All rights reserved.

When receiving the collection and before performing any other treatment, the function prepares the *Settlement Transactions* corresponding to immediate *Liquidity Transfers* related to corporate action or monetary policy settlement transactions. These prepared *Settlement Transactions* are to be taken into account by the provisioning net flow preparation.

More specifically, the function:

- Selects the *Settlement Transactions* which have an ISO transaction code related to a corporate action or monetary policy operation;
- For each *Settlement Transaction* related to a corporate action, checks in the static data if an option of automatic transfer to RTGS account is defined for the credited T2S Dedicated cash account and if the amount to settle on the settlement transaction is equal to or above the optional minimum threshold configured for the rebalancing standing order;
- In case of a *Settlement Transaction* related to monetary policy operation, or in case of a corporate action with the rebalancing option, prepares the associated immediate *Liquidity Transfer* settlement transaction, moving the received cash into the RTGS Dedicated transit account corresponding to the currency. No immediate Liquidity Transfer for rebalancing is generated in case of a corporate action with the rebalancing option if the amount to be settled move is below the optional minimum threshold.

Preparation of Liquidity transfers for all available cash

Reference Id	***SETT.VPB.CLP.3.1***

When receiving a collection with a *Settlement Transaction* with the "Liquidity Transfer All Cash" category, the function fills the Original Amount of the *Settlement Transaction* with the full amount available in the considered deliverable cash balance.

Provisioning net flows preparation

Reference Id	***SETT.VPB.CLP.4.1***

In order to perform the provision-checking on the involved securities positions, on the cash balances and on the headrooms of the associated credit memorandum balances, the function calculates a provisioning net flow per:

- Securities position (i.e. combination security/securities account/restriction type);
- Cash balance (i.e. combination T2S dedicated cash account/restriction type);
- Credit memorandum balance in case of settlement related to clients of payment/settlement banks (determined by the combinations T2S dedicated cash account/ securities account [159]) **{T2S.07.270}** **{T2S.08.090}**.

To that purpose, for each securities position or cash balance or external guarantee headroom (of the associated credit memorandum balance) **{T2S.16.582}**, **{T2S.16.650}**, the function:

- Selects all the *Settlement Transactions* which debit and credit the deliverable or earmarked securities position or/and cash balance;

159 Different combinations T2S dedicated cash account/ securities account can lead to the same credit memorandum balance

All rights reserved.

- Identifies the credit memorandum balances impacted by the selected *Settlement Transactions* ;
- Sums all debits and credits using:
  - Their partial quantity or amount calculated by the *Daytime Recycling & Optimisation* module if filled for collections containing more than one settlement transactions,
  - Their remaining quantity or amount in other case.

Management of restricted resources

Reference Id	***SETT.VPB.CLP.5.1***

This function prepares the use of restricted resources by checking if the T2S party has required to use one or several restricted positions and/or restricted cash balance (when indicated in the *Settlement Instruction* either through the restriction reference(s) or the associated restriction type) **{T2S.07.360} {T2S.07.370} {T2S.09.190} {T2S.09.220} {T2S.10.030} {T2S.10.040} {T2S.10.050} {T2S.10.130}**.

The process differs depending on whether the *Settlement Transaction* involves the deliverable (or earmarked) securities position/cash balance or a reserved or blocked one:

- In case the *Settlement Transaction* involves the deliverable cash balance or deliverable or earmarked securities position the restricted resources identified by the restriction reference(s) are moved into the deliverable (earmarked) securities position/cash balance. The provision-checking is then performed taking into account both the restricted resources and the deliverable (earmarked) securities position/cash balance if necessary. In that way, for each settlement transaction, the function prepares the use of restricted resources (blocking or reservation) by:
  - Searching the related restricted securities position and/or cash balance,
  - Searching the detailed securities/cash restriction associated with each restriction reference indicated till the quantity/amount in the *Settlement Transaction* is covered or till the resources available via the restriction reference(s) have been totally used [160],
  - For the available quantity or amount, limited by the quantity/amount necessary to settle the settlement transaction, creating a new *Settlement Transaction* which:
    - Debits the restricted securities position or cash balance,
    - Credits the deliverable securities position or cash balance;
  - Linking it with the corresponding original *Settlement Transaction* (the link type is "Use of Restriction");
  - Adding this *Settlement Transaction* to the collection;
  - Re-calculating the provisioning net flows taking into account the resources moved into the deliverable (earmarked) securities position and/or cash balance;
- In case the *Settlement Transaction* involves a reserved or blocked securities position, the provision-checking is performed taking into account only the resources identified by the restriction reference(s) indicated.

160 Once the quantity/amount in the trade is covered, the function stops searching restricted resources even if other restrictions references are indicated in the settlement transaction. Otherwise, if the amount or quantity is not covered, the function seeks for additional restricted resource if indicated in the settlement transaction.

All rights reserved.

Lastly, the function calculates the provisioning net flows for the *Settlement Transactions* related to set-up, the increase or the decrease of restrictions.

Execution of the provision-checking

Reference Id	***SETT.VPB.CLP.6.1***

The function checks in a first time:

- For each involved securities positions, if the related provisioning net flow can be settled with the available quantity **{T2S.07.272}**;

Then in a second time and only if the securities provisioning net flow is successful:

- For each involved cash balance if the related provisioning net flow can be settled with the available amount **{T2S.07.274}**;
- For each involved credit memorandum balance regarding a client of a payment/settlement bank if the related provisioning net flow can be settled with the available external guarantee headroom **{T2S.07.278}**.

This provision-checking does not apply to accounts which are allowed to have a negative balance (i.e. T2S CB cash account, RTGS transit accounts and Issuer CSD balance accounts **{T2S.07.274}**).

When the provision-checking is executed:

- If the provision-checking is successful, the function sends the collection to the *Pre-empting* function **{T2S.07.220}**;
- Else, if the provision-checking detects a lack of cash, a lack of securities or insufficient external guarantee headroom the function sends the collection to the lack and insufficient headroom management or partial settlement management sub-function.

Lack and insufficient headroom management or partial settlement management

Introduction

Reference Id	***SETT.VPB.CLP.7.1***

This function is performed after the provision-checking execution and consists in:

- A resolution of either lack of cash, lack of securities or insufficient external guarantee headroom concerning *Settlement Transactions* related to settlement instructions;
- A partial execution of *Settlement Transactions* related to *Settlement Restrictions* and liquidity transfer, if allowed for partial execution;
- A partial settlement, if need be, of *Settlement Transactions* related to *Settlement Instructions,* during a partial settlement window.

It checks the collection in order to determine the way to process it:

All rights reserved.

- If the collection contains only *Settlement Transactions* not eligible to auto-collateralisation (e.g. *Settlement Restriction* and liquidity transfer), the auto-collateralisation request for lacks of cash or securities or insufficient external guarantee headroom is bypassed. The *Settlement Transactions* (*Settlement Restriction* and liquidity transfer), if allowed to partial execution, are settled according to the resources available;
- Otherwise (i.e. the collection contains at least one *Settlement Transaction* eligible to auto-collateralisation **{T2S.07.286}**), the function checks that the conditions for requesting *Auto-collateralisation* module are met, i.e.:
  - In case of lack of securities, it checks if the securities position is earmarked for a collateral purpose,
  - In case of lack of cash, it checks:
    - If the DVP cut-off of the currency related to the T2S Dedicated Cash Account has not already been reached;
    - If the payment/settlement bank which owns the *T2S Dedicated Cash Account* is eligible for CB collateralisation,
    - If the headroom of the relevant auto-collateralisation limit set by the CB is sufficient to fill the lack **{T2S.07.274} {T2S.10.061}**;
  - In case of insufficient external guarantee headroom, it checks:
    - If the client of the payment/settlement bank is eligible for client-collateralisation with this payment/settlement bank;
    - If the total amount of headroom for the credit memorandum balance (all types of limits included), is sufficient to cover the need of the settlement transaction(s) related to the client of the payment/settlement bank **{T2S.07.286} {T2S.10.063} {T2S.10.064} {T2S.07.270}**;
- If the conditions are fulfilled, the function sends an auto-collateralisation request to the Intraday Credit Provider functions:
  - If the answer is positive (including the potential use of unsecured credit limit **{T2S.07.286} {T2S.07.288} {T2S.10.060}**), the function goes on, including the partial execution of *Settlement Restrictions* and *Liquidity Transfer* if necessary,
  - If the answer is negative, and in case of lack of securities only **{T2S.08.210}**, the function performs the partialisation of the settlement, based on the remaining amount of the settlement transaction, provided the following conditions are met:
    - A partial settlement window is currently running,
    - The collection comes from *Daytime Recycling and Optimisation* module and contains only a single *Settlement Transaction* and its realignment *Settlement Transactions* if any (i.e. no partial settlement attempt during the first settlement attempt),
    - All partial settlement application rules (i.e. agreements, threshold) are fulfilled;
  - If the answer is negative, and the partial settlement is not applicable or fails, the collection fails and is sent to the *Failure Management and Settlement Outcome* function with the reason for settlement failure for each settlement transaction.

All rights reserved.

Intraday Credit Provider request for lacks or for insufficient external guarantee headroom on credit memorandum balance

Reference Id	***SETT.VPB.CLP.7.2***

Except for the *Liquidity Transfers* and for the restriction set up, if at least one failed provision-checking remains on cash **{T2S.08.140}** **{T2S.08.480}** or on securities (can be due to a previous collateralisation) [161] **{T2S.08.910}** or one insufficient external guarantee headroom for *Settlement Transactions* related to a client of a payment/settlement bank, the function sends an *Auto-collateralisation Request* to the Intraday Credit Provider functions.

This request contains:

- The collection;
- The calculated provisioning net flows;
- The corresponding provision checking result [162];
- The request to create and/or release *Settlement Transaction* relating to auto-collateralisation.

If the reply from the Intraday Credit Provider functions is:

- Positive (the collection has been enriched with the auto-collateralisation *Settlement Transactions* by the Intraday Credit Provider functions), the *Common Limit and Provision Checking* function:
  - Re-calculates the provisioning net flows taking into account the auto-collateralisation *Settlement Transactions* added,
  - If lacks are still remaining related to *Settlement Transactions* not eligible to auto-collateralisation, triggers the partial execution,
  - Sends the collection to the *Pre-empting* function **{T2S.07.220}**;
- Negative (when at least one lack related to *Settlement Instruction* cannot be solved) and the lacks only regards the securities **{T2S.08.210}**, the *Common Limit and Provision Checking* function triggers the partial settlement.

Partial execution of settlement transaction related to liquidity transfer or settlement restriction

Reference Id	***SETT.VPB.CLP.7.3***

The function checks if the calculation of the relevant provisioning net flows involves *Settlement Transactions* that can be partially executed in the *Daytime Validation, Provisioning and Booking* module. This covers *Settlement Transactions* relating to:

- Settlement restrictions: reservation, earmarking **{T2S.07.350}** **{T2S.07.351}** **{T2S.10.030}** **{T2S.10.130};**
- *Liquidity Transfer* with the Partial Execution set to *"Yes".*

161 If the debited position in lack is an earmarked securities position for auto-collateralisation purpose.

162 The negative values correspond to the lacks.

All rights reserved.

In such a case, the function checks if a partial execution of *Settlement Transactions* relating to *Settlement Restrictions* and immediate liquidity transfers, allows a successful provisioning check. To that end the function:

- Fills in the partial quantity/amount of the relevant *Settlement Transactions* with the calculated quantity or amount;
- Re-calculates the provisioning net flows taking into account the partial execution of settlement transactions;
- Sends the collection to the *Pre-empting* function **{T2S.07.220}.**

Partial settlement of settlement transaction related to settlement instruction

Reference Id	***SETT.VPB.CLP.7.4***

If the collection comes from the Daytime Recycling & Optimisation and contains one single *Settlement Transaction* (and its optional realignment settlement transactions) in lack of securities, and only during the partial settlement window, the function checks if the *Settlement Transaction* has its Partial Settlement Indicator set to *"Yes"*.

In such a case, the function calculates the partial quantities and the partial amounts of the *Settlement Transactions* to be partialised, according to the available resources and the rules applicable to the partial settlement.

If a solution is found, the function sends the collection to the *Pre-empting* function. Otherwise, the function sends the collection to the *Failure Management and Settlement Outcome* function with the reason for settlement failure for the settlement transaction **{T2S.07.260}**.

Illustrations of the provision-checking concepts and processes

Reference Id	***SETT.VPB.CLP.8.1***

The following illustrations are independent of each other and have increasing complexity.

They describe the different cases of provision-checking:

- Provision-checking limited to securities side;
- Provision-checking involving credit memorandum balance, client-collateralisation and use of unsecured credit;
- Provision-checking involving use of securities restriction and CB collateralisation;
- Provision-checking involving cash restriction, CB collateralisation and client-collateralisation.

Each case contains one or two tables that illustrate the provision-checking concepts and processes according to the possible contents of collection and provision-checking result.

In each table, for the securities side the provisioning net flows are calculated per securities account involved in collection, and following the case, either per securities position or per detailed restriction.

For the cash side the provisioning net flows are calculated per *T2S Dedicated Cash Account* involved in collection, following the case, either per cash balance, credit memorandum balance or detailed restriction.

All rights reserved.

The first table represents the execution of the provision checking taking into account the available resources or/and external guarantee headroom according the case.

In case of lack(s) of cash, the headroom related to auto-collateralisation set by a central bank is checked to determine if a call to *Auto-collateralisation* module is possible or not.

In case of insufficiency(ies) of external guarantee headroom, the headroom related to client-collateralisation and the unsecured credit headroom are checked to determine if a call to *Auto-collateralisation* module is possible or not.

The second table represents the execution of provision checking after a positive answer of the Intraday Credit Provider functions. At this step, collateral *Settlement Transactions* have either brought some cash on the *T2S Dedicated Cash Account* having the lack or increased the external guarantee headroom. For the involved credit memorandum balances, the provision-checking takes into account the external guarantee headroom and the unsecured credit headroom.

Illustration 1 : Provision-checking limited to the securities side

Reference Id	***SETT.VPB.CLP.8.2***

The following table presents an illustration of the provision-checking concepts and processing:

COLLECTION	**SECURITIES SIDE**					**CASH SIDE**		
	SA1	**SA2**	**SA3**	**SA4**		**DCA1**	**DCA2**	**DCA3**
	SP1	**SP2**	**SP3**	**SP4**		**CB1**	**CB2**	**CB3**
ST1	+10		-10			-22	+22	
ST2		+14	-14				-30	+30
ST3	-30			+30		+43		-43
ST4	-18		+18			+30	-30	
ST5			+50	-50			+64	-64
ST6	+8			-8				
NET FLOWS	-30	+14	+44	-28		+51	+26	-77
Available	+12	+2	0	+50		+21	0	0
External Guarantee Headroom								
PROVISION CHECK RESULT	-18 **LACK**	+16 **OK**	+44 **OK**	+22 **OK**		+72 N/A [163]	+26 N/A [163]	-77 N/A [163]

163 Cash Side provided for completeness. In the case of a lack of securities the cash provision check is not performed.

All rights reserved.

COLLEC-TION	SECURITIES SIDE					CASH SIDE		
	SA1	SA2	SA3	SA4		DCA1	DCA2	DCA3
	SP1	SP2	SP3	SP4		CB1	CB2	CB3
Unsecured Credit Head-room								
Collateralisa-tion Head-room								
CALL TO COLLATER-ALISATION								

Based on a collection C1 of six settlement transactions, for which the parties are exclusively payment/settlement banks, the *Common Limit and Provision Checking* function calculates the provisioning net flows of each deliverable securities positions (SP1 to SP4) and cash balances (CB1 to CB3) involved in the collection.

For each securities position, the function executes the provision-checking by subtracting from the available quantity in the relevant securities positions the provisioning net flows.

If the provision-checking is positive (here SP2, SP3 and SP4), the provisioning net flows can be settled during the booking.

If the provision-checking is negative (here SP1), the provisioning net flows cannot be settled during the booking. Then the function tries to fill in the lacks by:

- Sending a request to Lack Security Manager function for Settlement Transactions associated to Settlement Instructions for the lack of securities;
- During the partial window, calling the *Partial Settlement Preparation* function if necessary.

Illustration 2: Provision-checking involving credit memorandum balances, client-collateralisation and unsecured credit

Reference Id	***SETT.VPB.CLP.8.3***

The following table illustrates the first step of the provision-checking taking into account the credit memorandum balances related to the client of the settlement bank:

COLLECTION	SECURITIES SIDE				CASH SIDE			
	SA1	SA2	SA3		DCA1		DCA2	DCA3
	SP1	SP2	SP3		CB1	CMB11	CB2	CB3
ST1	+15	-15			-30	-30	+30	
ST2	+22		-22		-45	-45		+45

All rights reserved.

COLLECTION	SECURITIES SIDE				CASH SIDE			
	SA1	SA2	SA3		DCA1		DCA2	DCA3
	SP1	SP2	SP3		CB1	CMB11	CB2	CB3
NET FLOWS	+37	-15	-22		-75	-75	+30	+45
Available	+13	+30	+50		+200		+50	+15
External Guarantee Headroom						+5		
PROVISION CHECK RESULT	+50 **OK**	+15 **OK**	+28 **OK**		+125 **OK**	-70 **KO**	+80 **OK**	+60 **OK**
Unsecured Credit Headroom						+20		
Collateralisation Headroom						+80		
CALL TO COLLATERALISATION						+30 **OK**		

A collection C2 having two *Settlement Transactions* ST1 and ST2 is submitted for a settlement attempt to the *Daytime Validation, Provisioning and Booking* module. The available securities and cash are sufficient to settle the collection.

The credit memorandum balance CMB11 is defined for the joint use of the T2S dedicated cash account DCA1 (cash balance CB1) and the securities account SA1 (securities position SP1) of the client of the payment/settlement bank.

The external guarantee headroom regarding the credit memorandum balance CMB11 is not sufficient (need of 70) but by addition with the auto-collateralisation headroom and with the unsecured credit headroom the settlement would be possible. So the function sends the collection to the *Auto-collateralisation* module.

After the positive answer from the *Auto-collateralisation* module, the collection contains a collateral *Settlement Transaction* in addition which has increased the external guarantee headroom by 65 and in the same time has decreased the auto-collateralisation headroom by the same amount.

COLLECTION		SECURITIES SIDE						CASH SIDE			
		SA1	SA2	SA3	SA4	SA5		DCA1		DCA2	DCA3
		SP1	SP2	SP3	SP4	SP5		CB1	CMB11	CB2	CB3
ST1		+15	-15					-30	-30	+30	
ST2		+22		-22				-45	-45		+45
ST3 (client collateralisation)	Client				-80			+65	+65		

All rights reserved.

COLLECTION	SECURITIES SIDE						CASH SIDE			
	SA1	SA2	SA3	SA4	SA5		DCA1		DCA2	DCA3
	SP1	SP2	SP3	SP4	SP5		CB1	CMB11	CB2	CB3
Settl. Bank					+80		-65			
PROVISIONING NET FLOWS	+37	-15	-22	-80	+80		-75	-10	+30	+45
Available	+13	+30	+50	+105	+10		+200		+50	+15
External Guarantee Headroom								+5		
Unsecured Credit Headroom								+20		
PROVISION CHECK RESULT	+50 **OK**	+15 **OK**	+28 **OK**	+25 **OK**	+90 **OK**		+125 **OK**	+15 **OK**	+80 **OK**	+60 **OK**

The collection after the auto-collateralisation can be settled. Indeed, the external guarantee headroom and the unsecured credit headroom related to the credit memorandum balance CMB11 are sufficient. In this case, the unsecured credit headroom is used (for an amount of 5) to complement the external guarantee headroom.

Illustration 3: Provision-checking involving use of securities restrictions and CB collateralisation

Reference Id	***SETT.VPB.CLP.8.4***

The following table illustrates the first step of the provision-checking taking into account the use of restrictions:

COLLECTION	SECURITIES SIDE						CASH SIDE		
	SA1			SA2	SA3		DCA1	DCA2	DCA3
	SP1	SP2		SP3	SP4		CB1	CB2	CB3
		R1	R2						
ST1 (initial)	-100			+100			+700	-700	
ST2 (initial)				-75	+75			+550	-550
ST3 (Restriction)	+70	-70							
ST4 (Restriction)	+30		-30						
PROVISIONING NET FLOWS	0	-70	-30	+25	+75		+700	-150	-550
Available	+20	+70	+50	0	+15		0	+100	+900

All rights reserved.

COLLECTION	SECURITIES SIDE						CASH SIDE		
	SA1			SA2	SA3		DCA1	DCA2	DCA3
	SP1	SP2		SP3	SP4		CB1	CB2	CB3
		R1	R2						
External Guarantee Headroom									
PROVISION CHECK RESULT	+20 **OK**	0 **OK**	+20 **OK**	+25 **OK**	+90 **OK**		+700 **OK**	-50 **KO**	+350 **OK**
Unsecured Credit Headroom									
Auto-Collateralisa-tion Headroom								+70	
CALL TO AUTO-COLLATERALISA-TION	+20	0	+20	+25	+90		+700	+20 **OK**	+350

A collection C3 having two *Settlement Transactions* ST1 and ST2 is submitted for a settlement attempt to the *Daytime Validation, Provisioning and Booking* module. ST2 refers to the use of the detailed securities restrictions SP2/R1 and SP2/R2. The use of restricted resources leads to the creation of the *Settlement Transactions* ST3 and ST4 in order to update the deliverable securities position SP1, and therefore the provision checking result.

As a result of this first step, the collection lacks for cash for the cash balance CB2. Therefore the collection containing the four *Settlement Transactions* is sent to the Intraday Credit Provider functions in order to fill in the lack.

Should the auto-collateralisation answer be positive, the collection is complemented with the collateral *Settlement Transaction* ST5 by the Intraday Credit Provider functions and submitted to a provision checking.

The following table illustrates the second step of provision-checking including the collateral settlement transaction:

COLLECTION		SECURITIES SIDE						CASH SIDE			
	SA1			SA2	SA3	SAC		DCA1	DCA2	DCA3	DCAC
	SP1	SP2		SP3	SP4	SPC		CB1	CB2	CB3	CBC
		R1	R2								
ST1 (initial)	-100			+100				+700	-700		
ST2 (initial)				-75	+75				+550	-550	
ST3 (Restriction)	+70	-70									

All rights reserved.

COLLECTION		SECURITIES SIDE						CASH SIDE			
	SA1			SA2	SA3	SAC		DCA1	DCA2	DCA3	DCAC
	SP1	SP2		SP3	SP4	SPC		CB1	CB2	CB3	CBC
		R1	R2								
ST4 (Restriction)	+30		-30								
ST5 (auto–coll.)				-10		+10			+50		-50
PROVISIONING NET FLOWS	0	-70	-30	+15	+75	+10		+700	-100	-550	-50
Available	+20	+70	+50	0	+15	0		0	+100	+900	+1000
External Guarantee Headroom											
Unsecured Credit Headroom											
PROVISION CHECK RESULT	+20 OK	0 OK	+20 OK	+25 OK	+90 OK	+10 OK		+700 OK	0 OK	+350 OK	+950 OK

The lacks encountered in step one are addressed with ST5 resulting from the auto-collateralisation and the collection can be directed to the next step.

Illustration 4: Provision-checking involving use of cash restriction, client-collateralisation and CB collateralisation

Reference Id	***SETT.VPB.CLP.8.5***

The following table illustrates the first step of the provision-checking taking into account the use of cash restrictions by a client of a payment/settlement bank:

COLLECTION	SECURITIES SIDE						CASH SIDE				
	SA1	SA2	SA3	SA4	SA5	SAC	DCA1			DCA2	DCAC
	SP1	SP2	SP3	SP4	SP5	SPC	CB10	CMB11	CB11	CB2	CBC
									R11		
ST1 (Initial)	+50	-50					-100	-100		+100	
ST2 (restriction use)							+70	0	-70		
NET FLOWS	+50	-50					-30	-100	-70	+100	
Available	0	+70					+20		+70	0	

All rights reserved.

COLLECTION	SECURITIES SIDE						CASH SIDE				
	SA1	SA2	SA3	SA4	SA5	SAC	DCA1			DCA2	DCAC
	SP1	SP2	SP3	SP4	SP5	SPC	CB10	CMB11	CB11	CB2	CBC
									R11		
External Guarantee Headroom								+65			
PROVISION CHECKING RESULT	+50 **OK**	+20 **OK**					-10 **LACK**	-35 **LACK**	0 **OK**	+100 **OK**	
Unsecured Credit Headroom								+70			
Collateralisation Headroom							+90	+60			
CALL TO COLLATERALISATION							+80 OK	+95 OK			

A collection C4 having one *Settlement Transaction* ST1, referring to the use of the detailed cash restrictions R1 associated the cash balance CB11, submitted for a settlement attempt to the *Daytime Validation, Provisioning and Booking* module.

The credit memorandum balance CMB11 is associated to the joint use of the TS2 dedicated cash account DCA1 (cash balance CB10) and the securities account SA1 (securities position SP1) of the client of the payment/settlement bank.

The use of restricted resources leads to the creation of the *Settlement Transactions* ST2 in order to update the deliverable cash balance CB10, and therefore the provision checking result.

As a result of this first step, the collection lacks for cash for the cash balance CB10 and has insufficient external guarantee headroom for the credit memorandum balance CMB11. Since the headroom associated to the T2S dedicated cash account DCA1 (via the relevant credit memorandum balance allows the CB collateralisation and the credit memorandum balance CMB11 allows the client-collateralisation, the collection is sent to the Intraday Credit Provider functions in order to fill in the lacks.

Should the auto-collateralisation answer be positive, the collection is complemented with the collateral *Settlement Transactions* ST3 and ST4 corresponding respectively to a client-collateralisation operation and to a CB collateralisation operation by the Intraday Credit Provider functions and submitted to a provision-checking.

The following table illustrates the second step of provision-checking including the auto-collateralisation settlement transactions:

All rights reserved.

COLLECTION		SECURITIES SIDE						CASH SIDE				
		SA1	SA2	SA3	SA4	SA5	SAN	DCA1			DCA 2	DCAC
		SP1	SP2	SP3	SP4	SP5	SPN	CB10	CMB 11	CB11	CB2	CBC
ST1 (Initial)		+50	-50					-100	-100		+100	
ST2 (restriction use)								+70	0 [164]	-70		
ST3 (client collat-eralisation)	Client			-40				+35	+35			
	Settl. Bank				+40			-35				
ST4 (CB collateralisation)						-9	+9	+11				-11
NET FLOWS		+50	-50	-40	+40	-9	+9	-19	-65	-70	+100	-11
Available		0	+70					+20		+70	0	0
External Guarantee Headroom									+65			
Unsecured Credit Headroom									+70			
PROVISION CHECKING RESULT		+50 OK	+20 OK	0 OK	+40 OK	+41 OK	+9 OK	+1 OK	+70 OK	0 OK	+100 OK	-11 OK

The collection after the auto-collateralisation can be settled. Indeed, the external guarantee headroom related to the credit memorandum balance CMB11 is sufficient and the available resource are sufficient to support the net flow regarding the cash balance CB11.

3 – Pre-empting

Introduction

Reference Id	***SETT.VPB.PRE.1.1***

When the function receives a collection, it checks whether an incoming securities or cash resource should be pre-empted in case of partially filled reservations on involved cash balances or securities positions.

Additional filling of partially filled reservations

Reference Id	***SETT.VPB.PRE.2.1***

This function handles the pre-emption of incoming resource (securities or cash) on a securities position or cash balance with associated reservations partially filled. This case occurs when a *Settlement Transaction* reserving a cash or securities is partially executed. An incoming cash or security resource is pre-empted in order to complement the previous reservation **{T2S.07.351}**.

In order to handle that pre-emption, the function respects the following principles:

164 It sum of the movements on the T2S dedicated cash account, i.e. on CB_{10} and CB_{11}.

All rights reserved.

- Only the incoming resources in a deliverable securities positions (or the earmarked positions for the earmarked securities accounts) and cash balance can be pre-empted (i.e. the incoming resources in restricted securities positions and restricted cash balances are never pre-empted to complement partially filled reservation);
- The provisioning net flows related to those securities positions or cash balances must be positive (i.e. the deliverable securities positions or cash balances receive more resources than delivered);
- The provisioning net flows quantity or amount is the maximum quantity or amount which can be pre-empted.

For each pre-empting restricted position, the function:

- Selects the relevant reservation *Settlement Transaction* which has been previously partially filled in;
- Completes the necessary attributes for the pre-emption of the incoming resources (i.e. transfer the incoming resources from the deliverable securities position or cash balance to the related restricted one and updates the corresponding detailed securities/cash restriction);
- Adds it to the collection;
- Updates the provisioning net flows.

Once all possible resources have been pre-empted, the function sends the collection to the *Booking* function **{T2S.07.220}**.

4 – Booking

Introduction

Reference Id	***SETT.VPB.BKG.1.1***

This function aims at **{T2S.07.240}:**

- Updating the securities positions and the cash balances **{T2S.06.150}** and creating the relevant securities and cash postings;
- Updating the limit utilisations and the journaling of limit utilisations;
- Checking the floor and ceiling;
- Updating the statuses (for the *Settlement Transactions* and for the *Settlement Instructions* the *Settlement Restrictions* or the *Liquidity Transfer* related to the settlement transactions);
- Creating:
  - The T2S generated *Settlement Instructions* and their related *Matching Objects* related to the collateral settlement transactions;
  - The T2S generated *Settlement Instructions* and their related *Matching Objects* related to the reverse collateral settlement transactions;
  - The *Settlement Restrictions* related to the generated use of restrictions;
  - The immediate *Liquidity Transfers* related to the generated rebalancing settlement transactions.

Securities positions and cash balances update and postings creation

All rights reserved.

Reference Id	***SETT.VPB.BKG.2.1***

The function updates the securities positions and cash balances credited and debited by all *Settlement Transactions* of the collection **{T2S.07.230}** and creates the relevant securities and cash postings.

In addition, if a *Settlement Transaction* debits blocked or reserved securities/cash identified by restriction references, the function decreases each relevant detailed securities/cash restriction by the calculated amount cash or securities quantity to be used.

During this process, the *Settlement Transactions* contained in the collection are booked one by one, without any particular order. All the accounts will be in credit when all *Settlement Transactions* of the collection are handled, because the collection has previously passed the provision-checking successfully.

As a result:

- The securities positions are updated from the quantity or partial quantity (if it is filled in **{T2S.08.210}**) for each *Settlement Transaction*;
- If the updated or created securities position is earmarked for a collateral purpose, the already calculated potential collateral valuations are updated accordingly **{T2S.03.060}**;
- If the updated securities positions are associated with a blocking or reservation restriction type, the impacted detailed securities restriction(s) are updated accordingly;
- If a detailed securities restriction corresponds to a new restriction, the restriction reference associated is created;
- The securities postings are created in order to justify the securities positions updates;
- The cash balance and, if any, the detailed cash restrictions are updated from the amount or partial amount (if it is filled **{T2S.08.210}**) of each *Settlement Transaction* **{T2S.06.116}**;
- If the updated cash balances are associated with a blocking or reservation restriction type, the impacted detailed cash restriction(s) are updated accordingly**{T2S.09.200}**;
- If a detailed cash restriction corresponds to a new restriction, the restriction reference associated is created;
- The cash postings are created in order to justify the cash balances updates.

This function then updates the data related to reservation as follows:

- In case of partial execution of a *Settlement Transaction* with the *Settlement Transaction* related to a reservation, the function fills in the missing quantity or amount on the relevant restricted securities position or restricted cash balance **{T2S.07.351}**;
- In case of pre-empting to fill in a restriction, the function decreases the missing quantity or amount on the relevant restricted securities position or restricted cash balance.

Once the securities positions and the cash balances are updated, the *Settlement Transactions* are irrevocable **{T2S.07.250} {T2S.10.010} {T2S.10.110}**.

Journaling of Limit Utilisations

Reference Id	***SETT.VPB.BKG.3.1***

All rights reserved.

The function calculates the new headroom resulting from the booking of the *Settlement Transactions* for each of the following limit types:

- The external guarantee limits (case of the clients of the payment/settlement banks);
- The auto-collateralisation limit set by the CB (case of the settlement banks);
- The auto-collateralisation limit set by a payment/settlement bank (case of the clients of the payment/ settlement banks);
- The unsecured credit limits (case of the clients of the payment/settlement banks).

Each debit or credit of the settled *Settlement Transactions* relevant for the limit can increase or decrease one or several limit headroom amounts.

In case the cash debit of a *Settlement Transaction* regards a client of a payment/settlement bank, the order of the limit utilisation is the following:

- External guarantee limit;
- Auto-collateralisation limit set by the payment/settlement bank for one of its clients when there is no external guarantee headroom anymore and according to the relevant auto-collateralisation settlement transactions;
- Unsecured credit limit to complement.

In case the cash credit of a *Settlement Transaction* regards a client of a payment/settlement bank, the order of increase of the limits is the following:

- Unsecured credit limit until the headroom is complete (this headroom cannot exceed the limit);
- External guarantee limit to complement the unsecured credit headroom. The external guarantee headroom can exceed the external guarantee limit.

In case the cash debit of a *Settlement Transaction* regards a payment/settlement bank, the relevant auto-collateralisation limit set by the CB is impacted by the relevant auto-collateralisation settlement transaction(s).

In case of reverse collateral settlement transaction, the limit impacts are the opposite of the impacts of the initial collateral settlement transactions.

For each defined limit, the function creates a new occurrence in *Journaling of Limit Utilisation* entity for each involved *Settlement Transaction* and updates the attributes of *Limit Utilisation* entity from the total amount of the involved settlement transactions.

To do this, for each account of each settlement transaction:

- The function identifies the limits linked to each *T2S Dedicated Cash Account* or to each credit memorandum balance;
- For each limit to be respected, the function;
  - Searches for the last *Journaling of Limit Utilisation.* If the limit has not been used since the start of the settlement day, the Remaining Limit After is initialised with the limit and the Limit Utilisation After is set to zero **{T2S.10.087}**,

All rights reserved.

- Calculates the new Remaining Limit After, the remaining cash to be reimbursed (for the auto-collateralisation limits set by the CB only) and the Limit Utilisation After of *Journaling of Limit Utilisation* using the cash amount debited or credited regarding the related account **{T2S.10.100}**;
- Creates a new *Journaling of Limit Utilisation* with this data.

When all the *Journaling of Limit Utilisation* is created, for each limit, the function updates the relevant *Limit Utilisation* from the last *Journaling of Limit Utilisation*. In the case of the first utilisation of the limit since the start of the settlement day, the limit utilisation is only created if the limit utilisation is not equal to zero (sum of the debits equal to the sum of the credits **{T2S.10.090}**).

Checking the floor and ceiling

Reference Id	***SETT.VPB.BKG.4.1***

For each updated cash balance:

- If a floor is defined on the corresponding *T2S Dedicated Cash Account* (static data) and if the cash balance is less than the floor amount, the function sends floor information to the *Failure Management and Settlement Outcome* function **{T2S.06.233}**; [165]
- If a ceiling is defined on the corresponding *T2S Dedicated Cash Account* (static data) and if the cash balance is greater than the ceiling amount, the function sends ceiling information to the *Failure Management and Settlement Outcome* function **{T2S.06.234}**. [166]

Updating the settlement transactions and all associated statuses

Reference Id	***SETT.VPB.BKG.5.1***

The function updates simultaneously the statuses of:

- The *Settlement Transactions* contained in this collection;
- The underlying *Settlement Instructions* or *Settlement Restriction* or liquidity transfer.

All those statuses are updated as follows:

- The Settlement Transaction Status of every *Settlement Transaction* corresponding to a *Matching Object* in the collection is set to:
  - *"Settled"*,
  - Or *"Partially Settled"* (in case of partial settlement) [167] **{T2S.07.210}**;
- The Settlement Transaction Status of every *Settlement Transaction* corresponding to a securities or cash reservation in the collection *Settlement Restriction* is set to:
  - *"Settled"* in case of blocking and in case of earmarking or decrease of reservation (even if in case of partial execution for the two latter cases) **{T2S.07.210}**,

165 Floor information is only sent the first time a breach happens, not for additional bookings where the balance is less than the floor.

166 Ceiling information is only sent the first time a breach happens, not for additional bookings where the balance is greater than the ceiling.

167 A *Settlement Transaction* related to a blocking restriction is settled even if it is partially satisfied. There is non pre-empting mechanism for this type of restriction. *"Partially Filled"* status is only used for *Settlement Transactions* related to a reservation when it is partially satisfied during this settlement attempt.

All rights reserved.

- Or *"Partially Filled"* (in case of partial execution for set-up or increase of reservation) **{T2S.07.210}**;

- The Settlement Transaction Status of every *Settlement Transaction* corresponding to a *Liquidity Transfer* is set to:
  - *"Partially Settled"* (in case of partial execution),
  - Or *"Settled"* otherwise [168];
- The Settlement Status of the underlying *Settlement Instructions* or *Settlement Restrictions* or *Liquidity Transfers* is set to the same value as that of the relevant settlement transaction:
  - *"Partially Settled"*,
  - Or *"Settled"* otherwise [169];
- The Partially released flag of the underlying Settlement Instructions is reset in case the quantity to be released was fully settled.

Synchronously to the update to statuses above, the function fills in:

- The settlement timestamp with the system date time;
- The settlement date with the current date of the settlement date;
- In case of partial settlement or partial execution, the remaining quantity (and On Hold quantity in case of partial release) and/or amount to be settled.

The collection is considered as settled.

Additional processing for auto-collateralisation operation

Reference Id	***SETT.VPB.BKG.6.1***

For each new collateral *Settlement Transaction* contained in the collection, the function:

- Creates the reverse collateral *Settlement Transaction* **{T2S.08.815}** **{T2S.08.820}** and holds the related *Settlement Instruction* concerning the payment/settlement bank party whatever the type of auto-collateralisation (credit provider side for a client-collateralisation or credit consumer side for a CB collateralisation);
- Creates also:
  - The *Matching Object* and the T2S generated *Settlement Instructions* (or the T2S generated *Settlement Restriction* in case of pledge sub auto-collateralisation procedure) corresponding to the settled collateral *Settlement Transaction* (with their Settlement Status set to *"Settled"*),
  - the necessary *Settlement Transactions* and the associated T2S generated *Settlement Instructions* if the booking process detects a cross-CSD Repo or Pledge, through a call to the *Realignment Chain Analyser* function of *Standardisation and Preparation to Settlement* module,

168 The status of the *Settlement Transactions* for which a check has previously failed is updated by the *Failure Management and Settlement Outcome* function.

169 The *Booking* function only handles the collection which has passed all checks successfully. The status of the other *Settlement Transactions* is updated by the *Failure Management and Settlement Outcome* function.

All rights reserved.

– The *Matching Object* and the T2S generated *Settlement Instructions* (or the T2S generated *Settlement Restriction* in case of pledge sub auto-collateralisation procedure) corresponding to the reverse collateral settlement transaction [170, 171]

Additional processing for rebalancing liquidity

Reference Id	***SETT.VPB.BKG.7.1***

For each rebalancing *Settlement Transaction* related to a corporate action, the function creates the corresponding immediate *Liquidity Transfer* with the settlement status set to *"Settled"*.

5 – Failure Management and Settlement Outcome

Introduction

Reference Id	***SETT.VPB.FMS.1.1***

The function manages the failure in order to prepare the collection and its related *Settlement Transactions* to further processing. It informs the following modules and domains with the settlement results:

- The *Daytime Recycling and Optimisation* module of the Settlement domain;
- The *Status Management* module of the Lifecycle Management and Matching domain;
- The *Outbound Information Management* module of the Liquidity Management domain.

Failure Management

Reference Id	***SETT.VPB.FMS.2.1***

This function provides the domains with the result of the settlement process.

In case of failure, the function considers the collection as unsettled and:

- Deletes the possible value in the partial quantity or amount in settlement transactions;
- Updates [172]:
  – The status of other Settlement Transactions Status to *"Unsettled"* with the reason for failure received from the function,
  – The reason code associated with the unsettled settlement status of the *Settlement Instruction* or settlement restriction.

Also for the unsettled or partially settled settlement transactions, the function also checks if the relevant cut-off is reached. In this case, the *Settlement Transactions* are suspended in order to avoid a new settlement attempt for the current settlement day.

Synchronously to the statuses update, the function fills in the Settlement Attempt Timestamp of the *Settlement Transaction* entity with the system date time.

---

170 The Party Hold Status of the T2S generated *Settlement Instruction* related to the T2S Party is set to *"Yes"*.

171 If a realignment has been detected for the collateral settlement transaction, the same realignment chain will be applied for its reverse collateral settlement transaction

172 A *Settlement Transaction* submitted to a settlement attempt with a Settlement Status *"Partially Settled"* keeps this value of its Settlement Status following a settlement failure (i.e. not updated to *"Unsettled"*).

All rights reserved.

Informing the domains

Reference Id	***SETT.VPB.FMS.3.1***

In every case, settlement (or partial settlement) or failure, the function provides the concerned domains with information allowing them to launch other processes or to inform the involved T2S Party **{T2S.06.233} {T2S.06.234} {T2S.06.730} {T2S.06.740} {T2S.13.090} {T2S.13.130} {T2S.13.133}**.

Depending on the *Settlement Transaction* category:

- *Collection Status Information* is sent to the *Daytime Recycling and Optimisation* module;
- *Instruction Status Information* (containing notably the generated restriction reference in case of settlement of a restriction or the quantity/amount actually settled in case of partial settlement or partial execution) is sent to the Lifecycle Management and Matching domain;
- *Floor/Ceiling Information* is sent to the Liquidity Management domain;
- *Liquidity Transfer Booking Information* is sent to the Liquidity Management domain.

The description of each of these flows is provided in the table "Input/Output of the module" below.

6 – Rebuilding Securities Positions or Cash Balances

Reference Id	***SETT.VPB.RSP.1.1***

The function is triggered by the reception of a *Rebuilding Request* from the T2S Operator **{T2S.10.020} {T2S.10.120}**.

The function aims at rebuilding securities positions or cash balances from the parameters filled in the event in real time.

A *Rebuilding Request* for the rebuilding of securities positions contains the following parameters:

- A starting date (mandatory);
- Parameters related to securities accounts:
  - No parameter: the rebuilding covers all the securities accounts in T2S,
  - CSD: the rebuilding covers all the securities accounts hold by the related CSD,
  - Securities account: the rebuilding covers the specified securities account;
- Parameters related to securities:
  - No parameter: the rebuilding covers all the securities,
  - Securities code: the rebuilding covers the specified security.

A *Rebuilding Request* for the rebuilding of cash balances event contains the following parameters related to cash balances:

- No parameter: the rebuilding covers all the cash accounts in T2S;
- CB: the rebuilding covers all the cash accounts of the related CB;
- Cash account: the rebuilding covers the specified cash account.

Rebuilding a cash balance can be only done for the current settlement day.

All rights reserved.

From the parameters sent, the function identifies the relevant securities or T2S Dedicated cash accounts, then for each one, it:

- Selects the securities positions or the T2S Dedicated cash balances for the relevant period, and the postings related to it;
- From the selected securities or cash postings, recalculates the securities positions or cash balances following the order of their settlement date/time.

At the end, the function performs a validation to ensure that any restrictions pertaining to the rebuilt positions are consistent **{T2S.10.022}**. The function sends the *Rebuilding Report* to the T2S Operator.

3.5.6.4 Multi-currency aspects

Settlement of a collection involving several currencies

The collection submitted for settlement attempt to the *Daytime Validation, Provisioning and Booking* module may be multi-currency, i.e. with *Settlement Transactions* using different currencies.

In such case, each *Settlement Transaction* of the collection impacts a cash balance which relates to a single currency, since a *T2S Dedicated Cash Accounts* relates to one single currency.

The provisioning net flows, calculated per cash balance, are therefore computed implicitly per currency with no mix across different currencies.

The following table illustrates the calculation of net flows for a collection containing *Settlement Transactions* using different currencies:

COLLECTION	SECURITIES SIDE			CASH SIDE			
	SA1	SA2	SA3	DCA1 (GBP)	DCA2 (EUR)	DCA3 (GBP)	DCA4 (EUR)
	SP1	SP2	SP3	CB1	CB2	CB3	CB4
ST1	+100	-100		-9795		+9795	
ST2	-100		+100		+9905		-9905
NET FLOWS	0	-100	+100	-9795	+9905	+9795	-9905
Available	0	+220	+50	+9000	0	+105	+10000
External Guarantee Headroom							
PROVISION CHECK RESULT	0 **OK**	+120 **OK**	+150 **OK**	-795 **LACK**	+9905 **OK**	+9900 **OK**	+95 **OK**
Unsecured Credit Headroom							

All rights reserved.

COLLECTION	SECURITIES SIDE				CASH SIDE			
	SA1	SA2	SA3		DCA1 (GBP)	DCA2 (EUR)	DCA3 (GBP)	DCA4 (EUR)
	SP1	SP2	SP3		CB1	CB2	CB3	CB4
Collateralisation Headroom								
Call to collater-alisation								

Based on a collection of two linked *Settlement Transactions* on the same security and two different currencies, the function calculates the provisioning net flows of each securities position (SP1 to SP3) and cash balance (CB1 and CB3 in GBP and CB3 and CB4 in EUR) involved in the collection.

The cash balances CB1 and CB2 are respectively related to the *T2S Dedicated Cash Accounts* DCA1 in GBP and DCA2 in EUR which are owned by the same party.

3.5.6.5 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Collection [Collection Created]	In	Created collections	SETT:SPS SETT:R&O SETT:Auto-collateralisation	
Rebuilding Request	In	Rebuilding of securities positions asked by a system administrator of the T2S Operator	T2S Operator	
Rebuilding Report	Out	Reporting of the rebuilding of securities positions or cash balances		T2S Operator
Auto-collateralisation Request	Out	Provisioning net flows (with the missing quantity or amount) and the collection of settlement transactions		SETT:Auto-collateralisation
Auto-collateralisation Answer	In	Collection of settlement transactions after auto-collateralisation	SETT:Auto-collateralisation	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Collection Status Information	Out	In case of settlement (new resources available) or failed (new unsettled settlement transaction		SETT:R&O
Instruction Status Information	Out	In case of failure, settlement or partial settlement (with the partial quantity or amount corresponding to the quantity or amount actually settled during this partial settlement) Concerning settlement transaction category related to settlement restriction, contains also the restriction reference		LCMM:Status Management
Liquidity Transfer Booking Information	Out	In case of Settlement Transaction Category set to "Liquidity Transfer"		LQMG:Outbound Information Management
Floor/Ceiling Information	Out	Floor or Ceiling reached		LQMG:Outbound Information Management

3.5.6.6 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
STATIC DATA			
Restriction type	Restriction Type	Read	For the intraday settlement restrictions
Securities Account	Securities Account	Read	For restriction on acct, party identifier...
T2S Dedicated Cash Account	T2S Dedicated Cash Account	Read	For acct status, closing date etc...
Headroom	Limit	Read	For external guarantee headroom, for unsecured credit headroom and for auto–collateralisation headroom
Credit Memorandum Balance	Credit Memorandum Balance	Read	For access to the limits

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
Restrictions	Party Restriction	Read	For the intraday settlement restrictions
	Security Restriction	Read	For the intraday settlement restrictions
	T2S Dedicated Cash Account Restriction	Read	For the intraday settlement restrictions
	Securities Account Restriction	Read	For the intraday settlement restrictions
DYNAMIC DATA			
Collection	Collection	Write	For the result of the process of the collection
	Collection Transaction	Read	For the potential prepared partial amount quantity
Settlement Transaction	Settlement Transaction	Read/Write	For transaction type, status...
Settlement Instruction	Settlement Instruction	Write	For auto-collateralisation and the status
Matching Object	Matching Object	Write	For auto-collateralisation
Settlement Restriction	Settlement Restriction	Write	For the status
Liquidity Transfer	Liquidity Transfer	Write	For the status
Positions	Securities Position	Write	Securities position to be updated
	Securities Position Current	Write	Current securities position to be updated
	Cash Balance	Write	Cash balance to be updated
	Cash Balance Current	Write	Current cash balance to be updated
Postings	Securities Posting	Write	Justifying the securities positions changes
	Cash Posting	Write	Justifying the cash balances changes
Limits	Limit Utilisation	Write	Limit Utilisation to be updated
	Journaling of Limit Utilisation	Write	Journaling of Limit Utilisation to be updated

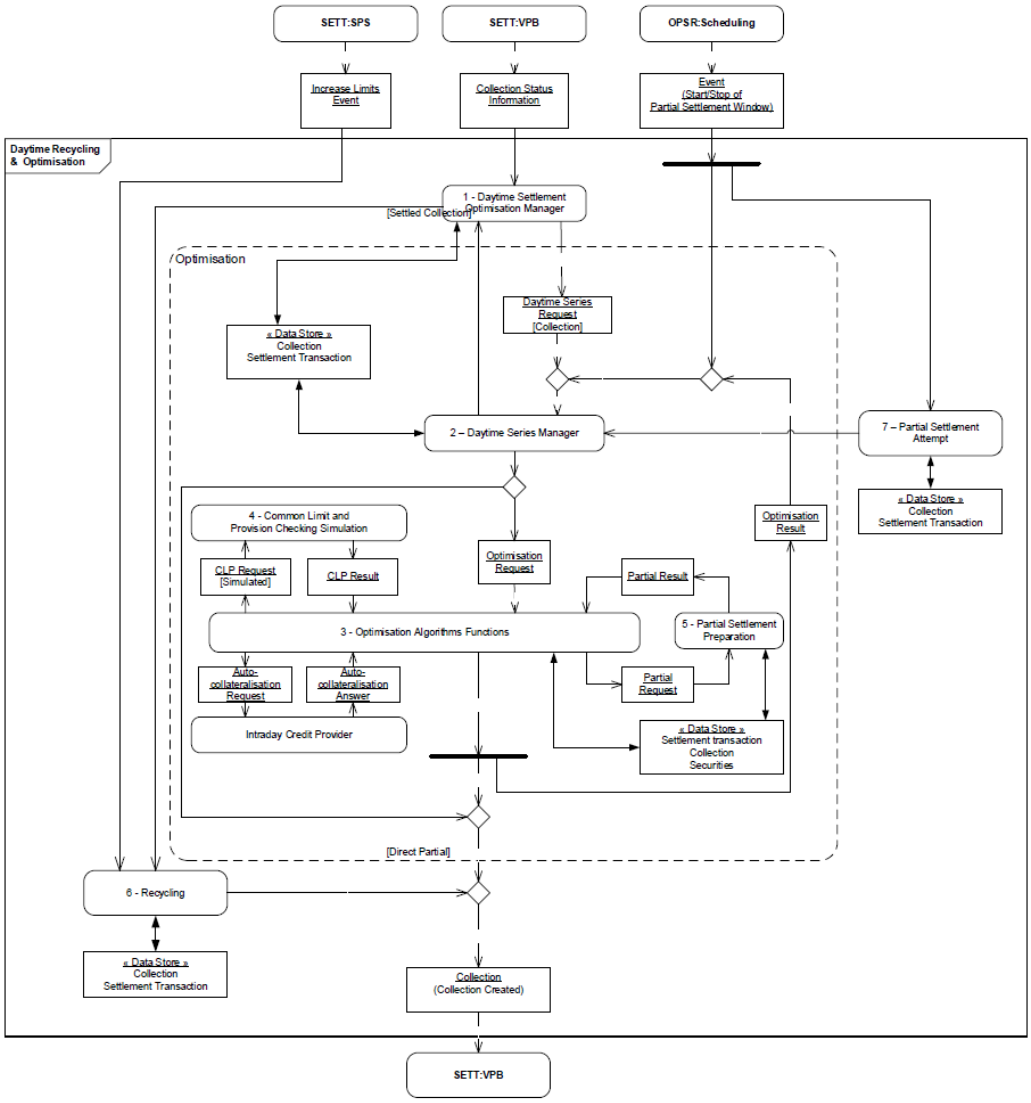
All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
Detailed Restrictions	Detailed Securities Restriction	Read/Write	Meter of securities blocked or reserved to be updated
	Detailed Cash Restriction	Read/Write	Meter of cash blocked or reserved to be updated
Restriction Reference	Restriction Reference	Read	Reference of the restriction to be used
Net Flows	Cash Provisioning Net Flows	Read/Write	For identifying a possible lack of cash
	Securities Provisioning Net Flows	Read/Write	For identifying a possible lack of securities
	CMB Provisioning Net Flows	Read/Write	For identifying an insufficient headroom of external guarantee

All rights reserved.

3.5.7 Daytime Recycling and Optimisation

3.5.7.1 Diagram of the module



3.5.7.2 Description of the module

This module runs during the daytime period only. It receives the result of a settlement attempt as a *Collection Status Information.*

When the result consists in a failure, the module triggers the optimisation functions. They combine this new fail with previous ones in order to build a new settlable collection. In such a case, they propose the collection for a new settlement attempt in the *Daytime Validation, Provisioning and Booking module.*

When the result consists in a success, the module triggers the Recycling function, except for settled collections with partially settled transactions which are sent to optimisation functions before recycling. The recycling function identifies and proposes for a new settlement attempt, in the Daytime Validation, Provisioning

All rights reserved.

and Booking module, Settlement Transactions that have failed during a previous settlement attempt and could profit from the new resources (cash, securities for delivery, securities in positions earmarked available for collateral) **{T2S.07.090} {T2S.08.080}**.

The *Daytime Settlement Optimisation Manager* function (function 1):

- Sends the settled collections without any partial settlement to Recycling function.
- Separates the *Settlement Transactions*, that have previously failed to settle including partially settled transactions into as many collections as needed. For instance, if the collection contains two independent - from a *Settlement Transaction LinkSet* perspective - settlement transactions, the output is two collections; if the incoming collection contains two *Settlement Transactions* that are in a "with" relation, the output is one collection. The *Daytime Settlement Optimisation Manager* function then sends *Daytime Series Request*, consisting in each of those collections, to the *Daytime Series Manager* function (function 2) for optimisation.

Optimisation

The Optimisation functions aim at identifying and submitting to a new settlement attempt, with an expected success, new collections of Settlement Transactions that have previously failed to settle.

The *Daytime Series Manager* function (function 2) launches, for each *Daytime Series Request*, a succession of *Optimisation Algorithm Functions* - referred to as a series - on the collection received.

During the daytime, the optimisation relies on "build up" based algorithms. Such algorithms identify settlable collection by looking for chains of transactions such as simple circles, back-to-back or more complex chains.

The *Optimisation Algorithms Functions* (function 3) rely on the *Common Limit and Provision Checking Simulation* function (function 4) to compute the simulated provision checking, at the optimisation algorithm's decision. In addition, the *Optimisation Algorithms Functions* may use the *Intraday Credit Provider* functions from the *Auto-collateralisation* module in order to solve a lack of cash, or insufficient external guarantee headroom, by providing intraday credit through auto-collateralisation, or to solve a lack of securities by releasing the relevant reverse collateral transactions on hold and ensuring the availability of resources to settle them.

Eventually, the *Optimisation Algorithms Functions* may use the *Partial Settlement Preparation* function (function 5), for partial settlement only in case of lack of securities as a last resort during partial settlement windows.

When the *Optimisation Algorithm Functions* identify a settlable collection of settlement transactions, they send it to the *Daytime Validation, Provisioning and Booking* module.

If no optimisation is found, during the partial settlement window, and the series is a first attempt on the settlement transaction, the *Daytime Series Manager* function sends it to the *Daytime Validation, Provisioning and Booking* module in a collection.

Recycling

The Recycling function (function 6) aims at identifying and sending to the Daytime Validation, Provisioning and Booking module a collection of Settlement Transactions that are impacted by any newly settled Settlement Transaction (providing cash, securities for delivery or securities in positions earmarked available for

All rights reserved.

collateral) or by any limit increase. This impacts both new resources provided, with limit increase being considered as a new resource, and the possibility to settle previously constrained Settlement Transactions (e.g. a Settlement Transaction with a link "after" waiting for the settlement of a Settlement Transaction with a link "before").

Additionally, the Recycling function may be triggered in order to recycle all the pending *Settlement Transactions* (Massive recycling).

Partial Settlement Attempt

At the reception of an *Event (Start of Partial Settlement Window),* the *Partial Settlement Attempt* function (function 7) aims at submitting to partial settlement all *Settlement Transactions* eligible to partial settlement (as identified upfront by the *Standardisation and Preparation to Settlement* module) that could not settle in an earlier attempt due to a lack of securities and partially released Settlement Transactions. It sends them to the Daytime Series Manager function complying with the priorities defined in SETT.R&O.OAF.2.1 **{T2S.08.050} {T2S.08.190}**.

This module handles collections of *Settlement Transactions* in accordance with the links defined by the T2S parties **{T2S.05.147} {T2S.09.110}**.

3.5.7.3 Description of the functions of the module

1 – Daytime Settlement Optimisation Manager

Reference Id	***SETT.R&O.DSO.1.1***

The *Daytime Settlement Optimisation Manager* function receives a *Collection Status Information* related to an unsettled collection, and:

- Cancels the collection received;
- Ignores all *Settlement Transactions* that require the settlement of at least one *Settlement Transactions* before it can be settled (for instance, a *Settlement Transaction* that is "after" in a *Settlement Transaction LinkSet* for which one of its "before" *Settlement Transaction* is unsettled) **{T2S.09.110} {T2S.09.100};**
- Regroups the others into as many collections as necessary with respect of their "with" link;
- Sends, for each of those newly created collections, a *Daytime Series Request* to the *Daytime Series Manager* function for optimisation with the collection as a parameter **{T2S.08.050}**.

Reference Id	SETT.R&O.DSO.1.2

The *Daytime Settlement Optimisation Manager* function receives a *Collection Status Information* related to a settled collection, and:

- Sends the settled collections without any partial settlement to Recycling function;
- Includes each partially settled transaction into as many collections as necessary;
- Sends, for each of those newly created collections, a *Daytime Series Request* to the *Daytime Series Manager* function for optimisation with the collection as a parameter {T2S.08.050};

All rights reserved.

- Waits for the result of all the optimisation attempts before sending the initial settled collection to the Recycling function.

2 – Daytime Series Manager

Introduction

Reference Id	***SETT.R&O.SMR.1.1***

The following table summarises the series of Optimisation Algorithms launched by the *Daytime Series Manager* and the related parameters defined below:

TRIGGERING EVENT	PARAMETERS	LAUNCHED OPTIMISATION ALGORITHMS
DAYTIME UNSETTLEMENT EVENT SERIES		
Daytime Series Request sent by the *Daytime Settlement Optimisation Manager* function	- Duration - Maximum size of a build-up chain - Partial settlement	Build-up Simple Circle Build-up Back to Back Build-up Generic

The main features of the algorithms listed above are described in the *Settlement* domain introduction.

Parameters of the series

Reference Id	***SETT.R&O.SMR.2.1***

Series of Optimisation Algorithms use the following parameters to adapt and/or fine tune the optimisation process:

- The "Duration" parameter is the total time given for the execution of the successive calls of *Optimisation Algorithms Functions* by the *Daytime Series Manager* function. If the series exceeds its allocated time, the *Daytime Series Manager* stops it;
- The "Maximum size of a build-up chain" parameter is used by "Build up based" *Optimisation Algorithms Functions* as a cap on the maximum number of lacks, either on cash balances and securities positions, or insufficient external guarantee headrooms on credit memorandum balances, an algorithm can solve;
- The "Partial settlement" parameter allows an *Optimisation Algorithms Functions* to use partial settlement, as a mean of a last resort to solve a lack of securities **{T2S.08.210}.** The *Daytime Series Manager* function maintains this parameter based on the reception of start/stop of partial settlement window events.

Daytime Unsettlement Event Series

Reference Id	***SETT.R&O.SMR.3.1***

Based on the unsettled *Settlement Transactions* of the collection underlying the *Daytime Series Request,* the *Daytime Series Manager* function **{T2S.08.150} {T2S.08.180}:**

- Launches the "Build-up Simple Circle" optimisation algorithm;
- If no settlable collection of *Settlement Transactions* is found, launches the "Build-up Back to Back" optimisation algorithm;

All rights reserved.

- If no settlable collection of *Settlement Transactions* is found, launches the "Build-up Generic" optimisation algorithm.

Partial settlement window

Reference Id	***SETT.R&O.SMR.4.1***

At the reception of an *Event (Start of Partial Settlement Window)* and until it receives an *Event (Stop of Partial Settlement Window),* the *Daytime Series Manager* function allows all optimisation algorithms to use partial settlement, as a mean of a last resort to solve a lack of securities **{T2S.08.210}.**

Partial settlement in case of optimisation failure

Reference Id	***SETT.R&O.SMR.4.2***

When a series ends with no solution, the *Daytime Series Manager* function checks:

- If the *Settlement Transaction* was received during a partial settlement window;
- If the series was the first optimisation attempt for this settlement transaction;
- If the *Settlement Transaction* may resort to partial settlement.

If all the three conditions are met (condition name "Direct Partial" in the diagram), the *Daytime Series Manager* function puts the *Settlement Transaction* in a new collection (possibly grouped with others with respect of their T2S generated links if relevant) **and** sends this collection for settlement to the *Daytime Validation, Provisioning and Booking* module. This function ensures that all the *Settlement Transactions* eligible for partial settlement and received during the partial settlement window, are attempted at least once for partial settlement.

3 – Optimisation Algorithms Functions

The *Optimisation Algorithms Functions* listed above aim at identifying a settlable collection. These algorithms share a common framework.

Framework

Reference Id	***SETT.R&O.OAF.1.1***

All the optimisation algorithms begin with a collection containing at least one *Settlement Transaction* and try to achieve a settlable collection. Every time the optimisation algorithm tries to add a *Settlement Transaction* to the collection, it calls the *Common Limit and Provision Checking Simulation* function to update all relevant net flows.

Optimisation algorithms iterate the following steps:

- Determination of the candidates *Settlement Transactions* among the whole stock to fill in the main lack or insufficient external guarantee headroom based on the algorithm's strategy;
- Sorting of the candidates based on *Settlement Transaction* priority;
- Selection of the group of *Settlement Transactions* to add:
  - If the latter selection does not find *Settlement Transactions* that can fill in the lack or insufficient external guarantee headroom, the algorithm may resort to auto-collateralisation **{T2S.08.050},**

All rights reserved.

- If auto-collateralisation fails and T2S is in a partial settlement window **{T2S.08.220},** the algorithm may resort to *Partial Settlement Preparation* function in order to solve any lack of securities with a minimum number of redeliveries submitted to partial settlement **{T2S.08.210} {T2S.08.050} {T2S.08.390},**
- If neither technical netting nor auto-collateralisation nor even partial settlement achieve to fill in the main lack or insufficient external guarantee headroom, the algorithm reports a failure to the *Daytime Series Manager* function.

If the algorithm succeeds in finding a settlable collection, it sends the latter to the *Daytime Validation, Provisioning and Booking* module.

Priority management

Reference Id	***SETT.R&O.OAF.2.1***

Each time an optimisation algorithm needs to sort unsettled *Settlement Transactions* before a selection the following criteria are used in decreasing weight order **{T2S.08.190}:**

- Level of priority **{T2S.07.130} {T2S.07.140} {T2S.07.150}** with the following order:
  - Reserved priority **{T2S.07.160},**
  - Top priority **{T2S.07.170},**
  - High priority **{T2S.07.180},**
  - Normal priority **{T2S.07.190},**
  - In case of linked settlement transactions, the level priority considered is the highest of the *Settlement Transactions* contained in the link se**t {T2S.09.130};**
- Oldest intended settlement date **{T2S.08.060} {T2S.08.070}.**

Conditional resort to Auto-collateralisation

Reference Id	***SETT.R&O.OAF.3.1***

If the collection contains at least one *Settlement Transaction* eligible to auto-collateralisation **{T2S.07.286},** the function checks that the conditions for requesting auto-collateralisation are met:

- No DVP cut-off for the currency of the business transaction has been received;
- In case of lack of securities, if the securities position is earmarked for a collateral purpose;
- In case of lack of cash, it checks:
  - If the DVP cut-off of the currency related to the T2S Dedicated Cash Account has not already been reached;
  - If the payment/settlement bank which owns the *T2S Dedicated Cash Account* is eligible for CB collateralisation;
  - If the headroom of the relevant auto-collateralisation limit set by the CB is sufficient to fill the lack **{T2S.07.274} {T2S.10.061};**
- In case of insufficient external guarantee headroom, it checks:

All rights reserved.

– If the client of the payment/settlement bank is eligible for client-collateralisation with this payment/ settlement bank;

– If the total amount of headroom for the credit memorandum balance (all types of limits included), is sufficient to cover the need for cash of the settlement transaction(s) related to the client of the payment/settlement bank **{T2S.07.286} {T2S.10.063} {T2S.10.064}.**

If all the relevant checks above are successful, the *Optimisation Algorithms Functions* resort to the Intraday Credit Provider functions. In return, the function indicates if the process of auto-collateralisation can solve the lack.

4 – Common Limit and Provision Checking Simulation

Introduction

Reference Id	***SETT.R&O.CLP.1.1***

The *Common Limit and Provision Checking Simulation* function is used in simulated mode to provide technical netting.

Framework

Reference Id	***SETT.R&O.CLP.2.1***

This function aims at verifying whether the settlement of a collection of *Settlement Transactions* is possible through the following process:

- The provisioning net flows are calculated and compared to the available quantities in the securities positions, available amounts in the cash balances, and available external guarantee headroom in the credit memorandum balances, taking into account the possible use of restricted positions per account before the settlement of the collection;
- The function then returns a status (settlable or not) and the list of the calculated provisioning net flows (with potential lacks on securities positions or cash balances [173] and potential insufficient external guarantee headroom) for the collection to the calling optimisation Algorithm **{T2S.07.272} {T2S.07.274} {T2S.07.278}.**

5 – Partial Settlement Preparation

Preparation

Reference Id	***SETT.R&O.PSP.1.1***

The *Optimisation Algorithms Functions* resort to the *Partial Settlement Preparation* function, if requested and allowed, according to an *Event (Start of Partial Settlement Window)* received from the *Scheduling* module **{T2S.08.220}.**

During a partial settlement window **{T2S.03.135} {T2S.07.080} {T2S.07.100},** the function determines the *Settlement Transactions* which can be partialised in order to solve or reduce the lacks of securities of the collection sent by the *Optimisation Algorithms* function **{T2S.08.050}.** In doing so:

- It checks that there is at least a lack of securities in the collection;

173 Those lacks are sorted so optimisation algorithms can determine which is to be solved first.

All rights reserved.

- It identifies for each lack of securities the *Settlement Transactions* which are allowed for partial settlement and which debit the securities positions of the lack **{T2S.08.210} {T2S.08.230} {T2S.08.250} {T2S.08.380}.**

Determination and application of the thresholds

Reference Id	***SETT.R&O.PSP.2.1***

Once the *Settlement Transactions* are identified, it determines the threshold value to apply according to the threshold type of each *Settlement Transaction* eligible for partial settlement **{T2S.08.250} {T2S.08.290} {T2S.08.300}:**

- If the Threshold Type of the *Settlement Transaction* is "Quantity", the threshold value is equal to the combination of the minimum settlement unit and the settlement unit multiple according to the underlying securities attributes in T2S static data **{T2S.08.310};**
- If the Threshold Type of the *Settlement Transaction* is *"Cash value",* the Threshold Value attribute is found in the *Partial Settlement Threshold* entity using **{T2S.08.315} {T2S.11.730}:**
  - Instrument Type = *"Unit-quoted securities"* or *"Nominal-quoted securities"* (based on the first character of the ISO10962 Classification of Financial Instruments set in T2S static data and the ISIN referenced into the settlement transaction),
  - Currency = currency of the cash amount of the transaction.

Then, the function applies the partial procedure as follows:

- In order to use the maximum amount of securities available on the securities accounts **{T2S.08.230},** it determines which *Settlement Transactions* can be partialised based on the threshold types, the defined threshold values and the available securities positions and cash balances **{T2S.08.380}.** If the Threshold Type is *"Cash value",* it also takes into account the minimum settlement unit and the settlement unit multiple for the partial settlement;
- It calculates the partial quantities and the partial amounts of the *Settlement Transactions* to be partialised;
- It attributes the same partial quantities/amounts for all the *Settlement Transactions* of the collection with T2S generated links **{T2S.08.390};**
- It fills in the link between the collection and *Settlement Transactions* with the partial quantity/amount of each relevant partialised *Settlement Transaction* of the collection.

6 – Recycling

Settled collection

Reference Id	***SETT.R&O.REC.1.1***

When the *Recycling* function receives a *Collection Status Information* from the *Daytime Settlement Optimisation Manager* function related to a newly settled collection:

- It cancels the collection received;
- It selects the following settlement transactions:

All rights reserved.

- Based on the net flows of the collection underlying the *Collection Status Information,* this function identifies all unsettled *Settlement Transactions* that may benefit from the increased resources and performs a provision check against the available resources to determine the ones to be recycled. **{T2S.08.160} {T2S.08.170},**
- All unsettled *Settlement Transactions* whose optimisation was previously deferred (see SETT.R&O.DSO.1.1) and is now enabled due to the settlement of a *Settlement Transaction* contained in the collection;

- It regroups them in new collections (each *Settlement Transaction* has its own collection or grouped with others with respect of their "with" link if relevant) and sends those collections for settlement to the *Daytime Validation, Provisioning and Booking* module, complying with the priorities defined in SETT.R&O.OAF.2.1 **{T2S.08.050}.** [174]

Increase limits Event

Reference Id	***SETT.R&O.REC.2.1***

When the *Recycling* function receives an *Increase Limits Event* from the *Standardisation and Preparation to Settlement* module:

- It selects the following settlement transactions based on the impacted limit from the *Increase Limits Event,* this function identifies all unsettled *Settlement Transactions* that may benefit from the increased resources and performs a provision check against the available resources to determine the ones to be recycled **{T2S.08.170};**
- It regroups them in new collections (each *Settlement Transaction* has its own collection or grouped with others with respect of their "with" link if relevant) and sends those collections for settlement to the *Daytime Validation, Provisioning and Booking* module, complying with the priorities defined in SETT.R&O.OAF.2.1 **{T2S.08.050}.** [174]

Massive Recycling Event

Reference Id	***SETT.R&O.REC.4.1***

When the *Recycling* function receives a Massive Recycling *Event* from *Scheduling* domain:

- It selects all the pending settlement transactions.
- It regroups them in new collections (each *Settlement Transaction* has its own collection or if relevant) and sends those collections for settlement to the *Daytime Validation, Provisioning and Booking* module, complying with the priorities defined in SETT.R&O.OAF.2.1 **{T2S.08.050}.**

7 – Partial Settlement Attempt

Reference Id	***SETT.R&O.PSA.1.1***

174 Due to parallelism in the recycling function, it may happen that, in some cases, the priorities may not be respected between two Settlement Transactions of the same recycling process. However, the main one, the highest priority criteria, is always respected, and the second one, the oldest intended settlement date, is always respected when at least one of the two Settlement Transactions is three days or more past its intended settlement date (i.e. its age is greater than or equal to three days). If the oldest intended settlement date is strictly considered, then so are the two remaining criteria (highest quantity or amount and earliest T2S timestamp).

All rights reserved.

The *Partial Settlement Attempt* function receives an *Event (Start of Partial Settlement Window)* from the *Scheduling* module **{T2S.08.220} {T2S.11.735} {T2S.03.165}** and:

- Selects all *Settlement Transactions* eligible to partial settlement which have failed in a previous settlement attempt due to lack of securities **{T2S.08.210}** and partially released Settlement Transactions**;**
- Creates as many collections as *Settlement Transactions* selected (each *Settlement Transaction* has its own collection or the *Settlement Transactions* with T2S generated links are grouped in a same collection) and sends those collections for settlement to the *Daytime Series Manager* function, complying with the priorities defined in SETT.R&O.OAF.2.1 **{T2S.08.050} {T2S.08.190}.**

This function ensures that all relevant *Settlement Transactions* received before start of partial settlement window and eligible for partial settlement are processed at least once during any partial settlement window **{T2S.08.220}.** It stops when all relevant *Settlement Transactions* have been submitted to *Daytime Validation, Provisioning and Booking* module.

All *Settlement Transactions* sent to *Daytime Validation, Provisioning and Booking* module by *Partial Settlement Attempt* function for partial settlement may be attempted again for settlement during the partial settlement window by the *Optimisation Algorithms Functions* if they fail to settle for the full or partial quantity.

3.5.7.4 Multi-currency aspects

The multi-currency capability is handled in the optimisation process with the management of every currency as a separately missing resource to be solved (as opposed to considering only lacks of cash in Euro) **{T2S.08.440} {T2S.08.450} {T2S.08.460}.**

As a result, no technical cross-currency cash netting is done: the optimisation processes consider net position on securities and on each currency **{T2S.08.470}.**

3.5.7.5 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Collection Status Information	In	Collection status information (collection settled/collection unsettled)	SETT:VPB	
Event (Start/Stop of Partial Settlement Window)	In	Event received at the beginning and at the end of partial settlement window	OPSR:Scheduling	
Collection (Collection Created)	Out	Collection of settlement transactions resulting from an optimisation process		SETT:VPB

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Increase Limits Event	In	Notification of a static data update for an increase of a limit.	SETT:SPS	

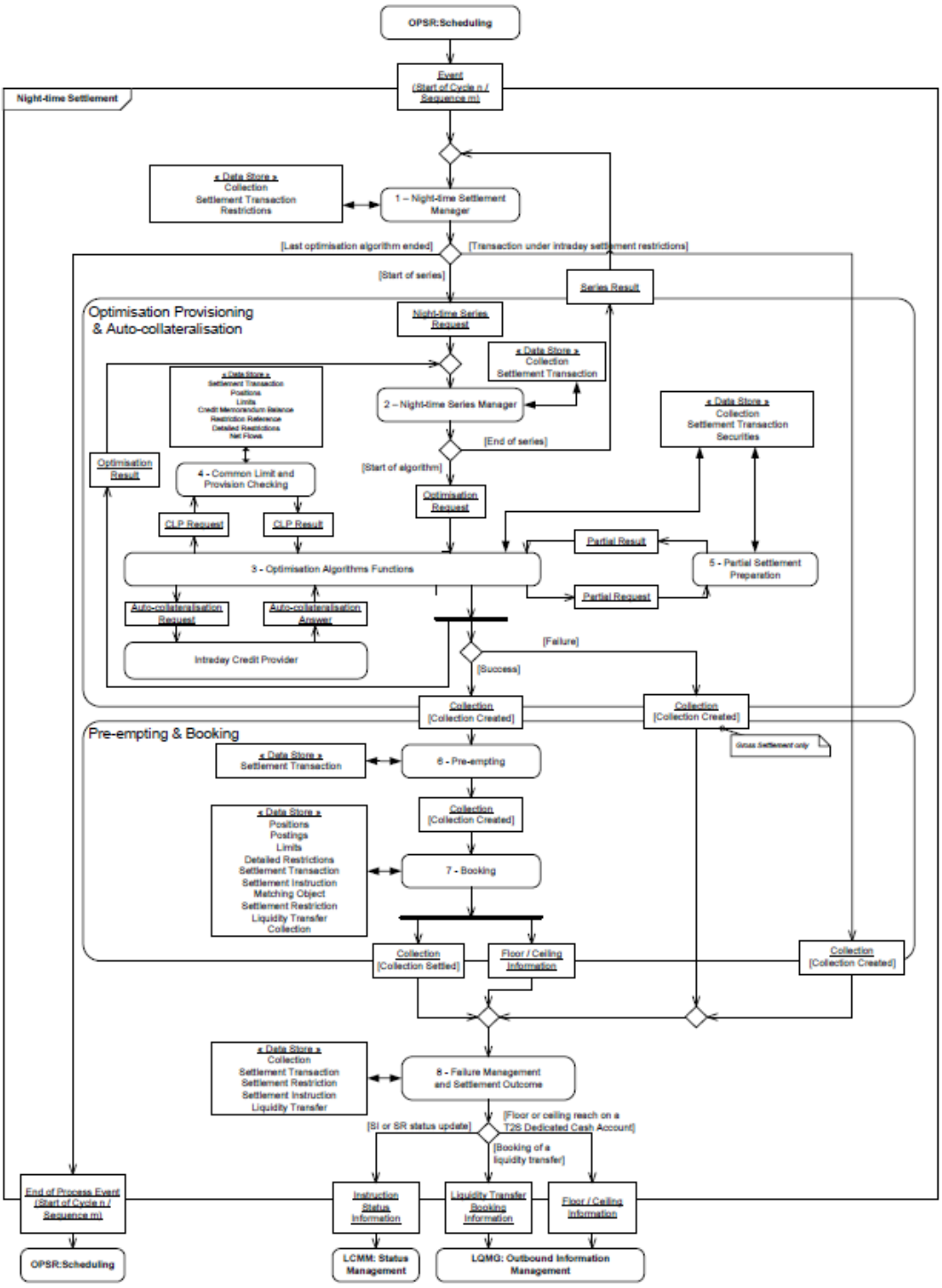
3.5.7.6 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
STATIC DATA			
Securities	Securities Valuation	Read	
	Securities	Read	
	Securities Code	Read	
DYNAMIC DATA			
Collection	Collection	Read/Write	
	CollectionTransaction	Read/Write	
	Cash Provisioning Net Flow	Read/Write	
	Securities Provisioning Net Flow	Read/Write	
	CMB Provisioning Net Flow	Read/Write	
Settlement Transaction	Settlement Transaction	Read	
	Restriction Reference	Read	
	Settlement Transaction Link-Set	Read	
	Settlement Transaction Link	Read	
	Partial Settlement Threshold	Read	

All rights reserved.

3.5.8 Night-time Settlement

3.5.8.1 Diagram of the module



3.5.8.2 Description of the module

This module runs during the night-time period only and performs the night-time settlement. It consists in a predefined number of sequences grouped into a predefined number of cycles.

All rights reserved.

Each sequence starts upon receiving an *Event (Start of Cycle n / Sequence m)* and runs a succession of optimisation procedures (series). Night-time series contain optimisation algorithms with different optimisation strategies. The conjunction of these different strategies yields the optimum balance [175] between volume and value of settlement with the available securities and cash resources **{T2S.08.010}.**

These collections are made of both:

- New *Settlement Transactions* created before the beginning of the sequence;
- And pending settlement transactions, notably the *Settlement Transactions* recycled at the Start of Day, which failed to settle in a previous settlement attempt **{T2S.08.080}.**

The *Night-time Settlement* module handles collections of *Settlement Transactions* in accordance with the links defined by the T2S parties **{T2S.05.147} {T2S.09.110}.**

Night-time Settlement Manager

At the beginning of a sequence, the *Night-time Settlement Manager* function (function 1) selects *Settlement Transactions* according to the considered sequence criteria and the applicable intraday settlement restrictions.

The *Night-time Settlement Manager* function notifies the *Scheduling* module when the last series ends.

Optimisation, Provisioning & Auto-collateralisation

Once this selection is performed, the *Night-time Series Manager* function (function 2) groups the selected *Settlement Transactions* into collections and sends them to the *Optimisation Algorithms Functions* for their optimisation. When the last optimisation algorithm ends, the *Night-time Series Manager* function notifies the *Night-time Settlement Manager* function.

Optimisation Algorithms Functions (function 3) looks for settlable collections, through the use of the *Common Limit and Provision checking* function (function 4) and, if needed, with resort to auto-collateralisation. At last, during sequence X, optimisation algorithms also resort to partial settlement in case of lack of securities with the *Partial Settlement Preparation* function (function 5) **{T2S.07.080} {T2S.08.040}.**

Pre-empting and Booking

When an optimisation algorithm finds a settlable collection, in order to maintain consistency from first check to final booking, the module resorts to all relevant technical means ensuring that the resources used for the current settlement attempt (i.e. settlement transactions, limit utilisations, securities positions and cash balances) cannot be modified by another settlement process. The collection found is then sent to the *Pre-empting* function (function 6) and then to the *Booking* function (function 7) for their actual settlement. The settlement outcome is reported through the *Failure Management and Settlement Outcome* function (function 8).

3.5.8.3 Description of the functions of the module

1 – Night-time settlement manager

Introduction

175 The URD defines the optimum balance between the maximisation of volume and value as aiming to avoid situations where only volume optimisation would be sought (that could lead to favour settlement of low value retail transactions at the detriment of transactions with higher value) or situations where only value optimisation would be sought (what could lead to favour the settlement of high value transactions at the detriment of many retail transactions with lower value).

All rights reserved.

Reference Id	***SETT.NTS.NTM.1.1***

This function receives an *Event (Start of Cycle n / Sequence m)* during the night-time settlement period **{T2S.08.020} {T2S.08.080}.** It then selects the *Settlement Transactions* related to the identified sequence **{T2S.03.100} {T2S.07.010} {T2S.07.020},** creates a collection that includes them and sends it, through a *Night-time Series Request*, to the *Night-time Series Manager* function. When the last series ends, the *Night-time Settlement Manager* function:

- When relevant, removes Intraday Restrictions having restriction type code "CAPR" (Corporate Actions Processing Restriction);
- Notifies the Scheduling module.

Sequence processing

Determination of the Settlement Transactions relevant to sequences

Reference Id	***SETT.NTS.NTM.2.1***

The function selects all the pending or created *Settlement Transactions* **{T2S.03.110}:**

- With an Applicable Instruction Entry Date/Time value earlier than the *Event (Start of Cycle n / Sequence m)* **{T2S.03.300};**
- Which are not on hold;
- All "With" linked *Settlement Transactions* of different ISO Transaction Code are dealt with in the first possible sequence where all the linked *Settlement Transactions* would be eligible separately.

Sequence 0

Reference Id	***SETT.NTS.NTM.2.2***

Sequence 0 handles:

- Liquidity transfers, identified by a Settlement Transaction Category set to *"Liquidity Transfer"*, from RTGS accounts to *T2S Dedicated Cash Accounts* and from a *T2S Dedicated Cash Account* to another *T2S Dedicated Cash Account* of the same T2S Party or between *T2S Dedicated Cash Accounts* of different payment banks but linked to same RTGS account **{T2S.03.070} {T2S.06.063} {T2S.06.114} {T2S.06.230};**
- Cash *Settlement Restriction* corresponding to CoSD blocking regeneration, identified by a Settlement Transaction Category set to "*Cash restriction regeneration*".

Sequence 1

Reference Id	***SETT.NTS.NTM.2.3***

Sequence 1 handles:

- Corporate Actions, identified by an ISO Transaction Code corresponding to Corporate Action **{T2S.07.030} {T2S.09.270};**
- All *Liquidity Transfers* identified by a Settlement Transaction Category set to *"Liquidity Transfer"* **{T2S.06.230}.**

All rights reserved.

Sequence 2

Reference Id	***SETT.NTS.NTM.2.4***

Sequence 2 handles:

- FOP transactions for rebalancing purposes (between securities accounts of the same T2S Party), identified by a Transaction Type Code set to *"FOP"* **{T2S.07.040};**
- All *Liquidity Transfers* identified by a Settlement Transaction Category set to "*Liquidity Transfer*" **{T2S.06.230};**
- Corporate Actions identified by an ISO Transaction Code corresponding to Corporate Action **{T2S.09.270};**
- And *Settlement Transactions* not settled in the previous sequence **{T2S.03.110}.**

Sequence 3

Reference Id	***SETT.NTS.NTM.2.5***

Sequence 3 handles:

- Central Bank Operations (CBO), identified by an ISO Transaction Code set to *"CNCB"* **{T2S.07.050};**
- All *Liquidity Transfers* identified by a Settlement Transaction Category set to *"Liquidity Transfer"* **{T2S.06.230};**
- Corporate Actions identified by an ISO Transaction Code corresponding to Corporate Action **{T2S.09.270};**
- FOP transactions, as per sequence 2, identified by a Transaction Type Code set to *"FOP"* **{T2S.07.040};**
- And *Settlement Transactions* not settled in the previous sequence **{T2S.03.110}.**

Sequence 4

Reference Id	***SETT.NTS.NTM.2.6***

Sequence 4 handles:

- all trading-related settlement transactions, including the ones related to settlement restrictions;
- *Settlement Transactions* not settled in the previous sequence **{T2S.07.060} {T2S.07.070}.**

Sequence X

Reference Id	***SETT.NTS.NTM.2.7***

Sequence X handles:

- all trading-related settlement transactions, including the ones related to settlement restrictions;
- partially released Settlement Transactions;
- and *Settlement Transactions* not settled in the previous sequence **{T2S.03.110};**

All rights reserved.

- with partial settlement, in case of lack of securities, for all transactions which are eligible to partial settlement processing **{T2S.08.220} {T2S.03.135} {T2S.07.080}.**

Sequence Y

Reference Id	***SETT.NTS.NTM.2.8***

Sequence Y handles the reimbursement of the multiple liquidity providers, i.e. *Settlement Transactions* with a Settlement Transaction Category value set to *"Liquidity Transfer"* and with a cash flow going from any *T2S Dedicated Cash Account* to any RTGS Dedicated transit account **{T2S.06.067}.**

Sequence Z

Reference Id	***SETT.NTS.NTM.2.9***

Sequence Z handles all standing and predefined *Liquidity Transfers* based on a night-time event, i.e. *Settlement Transactions* with a Settlement Transaction Category value set to *"Liquidity Transfer"* **{T2S.06.220}.**

Validation of settlement transactions

Reference Id	***SETT.NTS.NTM.3.1***

Additionally to the selection by the type of *Settlement Transaction* described above, the function verifies during its selection that no intraday *Settlement Restriction* applies on the following entities to which the *Settlement Transaction* is related:

- Securities **{T2S.07.260} {T2S.16.510};**
- Securities Account **{T2S.07.260} {T2S.16.680};**
- *T2S Dedicated Cash Account* **{T2S.07.260} {T2S.16.658} {T2S.16.680};**
- Party (set by a CSD or a Central Bank) **{T2S.07.260} {T2S.16.680}.**

The function excludes from the selection all *Settlement Transactions* for which at least one intraday *Settlement Restriction* is detected and sends them to the *Failure Management and Settlement Outcome* function with the reason for failure for each settlement transaction.

Processing of Sequences

Reference Id	***SETT.NTS.NTM.4.1***

For each sequence, the function:

- Sends a *Night-time Series Request* to the *Night-time Series Manager* function to inform it that a sequence of *Settlement Transactions* is ready to be executed;
- Sends one collection to the *Failure Management and Settlement Outcome* function with all *Settlement Transactions* that correspond to the sequence criteria but that were not selected due to at a least one intraday *Settlement Restriction,* each of those transactions carrying its own reason for failure;
- At the reception of a Series Result sent by the Night-time Series Manager function:
  - Checks if a Corporate Action on stock (CORP) FoP was settled during the series:
  - Using an ISIN under an Intraday Restriction having restriction type code "CAPR" (Corporate Actions Processing Restriction),

All rights reserved.

- On a Securities Account belonging to the CSD who set this IR;
- If such FoP exists, removes this Intraday Restriction on the ISIN at stake;
- Sends an End of Process Event (Start of Cycle n / Sequence m) to the Scheduling module, indicating that it has finished the processing of the sequence.

2 – Night-time Series Manager

Series of Optimisation Algorithms

Reference Id	***SETT.NTS.SMR.1.1***

The following table summarises the series of Optimisation Algorithms launched by the *Night-time Series Manager* and their related parameters:

TRIGGERING EVENT	PARAMETERS	OPTIMISATION ALGORITHMS FUNCTIONS
NIGHT-TIME OPTIMISATION SERIES		
Event (Start of Cycle n / Sequence m) sent by the *Scheduling* module with the sequence identifier	Duration	Mathematical optimisation algorithms Gross algorithm

The main features of the optimisation algorithms listed above are described in the *Settlement* domain introduction.

Parameters of the series

Reference Id	***SETT.NTS.SMR.2.1***

Series of Optimisation Algorithms use the following parameters to adapt and/or fine-tune the optimisation process: The "Duration" parameter is the total time given for the execution of the successive calls of *Optimisation Algorithms Functions* by the *Night-time Series Manager* function. If the series exceeds its allocated time, the *Night-time Series Manager* function stops it.

Night-time Optimisation Series

Reference Id	***SETT.NTS.SMR.3.1***

This series is launched when the function is triggered by a *Night-time Series Request.*

For the *Settlement Transactions* selected for the *Night-time Series Request* and based on the parameters of the series, the function sequentially:

- Runs the relevant algorithms for the cycle and sequence;
- Sends a *Series Result* to the *Night-time Settlement Manager* function after the execution of the last optimisation algorithm.

3 – Optimisation Algorithms functions

All rights reserved.

Unless explicitly stated, the same descriptions as specified in the *Daytime Recycling and Optimisation* module apply to night-time. The following description only highlights the differences between night-time and daytime.

Framework

Reference Id	***SETT.NTS.OAF.1.1***

The *Optimisation Algorithms Functions* listed above aim at identifying a settlable collection. These algorithms share a common framework.

During night-time and contrary to the daytime, the provisioning and the auto-collateralisation sets of functions are fully enclosed within the optimisation.

All the optimisation algorithms begin with a collection containing any number of *Settlement Transactions* and try to achieve a settlable collection. Every time the optimisation algorithm tries to add or subtract a *Settlement Transaction* to the collection, it calls the *Common Limit and Provision Checking* function to update all relevant net flows.

Optimisation algorithms iterate the following steps:

- Depending on the algorithm strategy, determination of the candidates *Settlement Transactions* for filling in the main lack or insufficient external guarantee headroom:
  - Among the whole stock for build-up based algorithms, as they will be added to the collection,
  - Among the collection's *Settlement Transactions* for de-selection based algorithms, as they will be subtracted from this collection **{T2S.08.120};**
- Sorting of the candidates based on *Settlement Transaction* priority and intended settlement date;
- Selection of the group of *Settlement Transactions* to add or subtract:
  - If the latter selection does not find *Settlement Transactions* that can fill in the main lack or insufficient external guarantee headroom, the algorithm resorts to auto-collateralisation **{T2S.08.040} {T2S.08.120},**
  - If auto-collateralisation fails and T2S is in a partial settlement window (in that case, sequence X) **{T2S.07.080} {T2S.08.220},** the algorithm resorts to *Partial Settlement Preparation* function in order to solve any lack of securities with a minimum number of redeliveries submitted to partial settlement **{T2S.08.040} {T2S.08.120} {T2S.08.390} {T2S.08.400} {T2S.09.120},**
  - If neither technical netting, nor auto-collateralisation nor even partial settlement achieves to fill in the main lack or insufficient external guarantee headroom, the algorithm reports a failure to the *Night-time Series Manager* function.

If an optimisation algorithm succeeds in finding a settlable collection, it sends the settlable collection directly to the *Pre-empting* function so as to ensure the potential pre-emption of incoming resources to complete a partially filled reservation (on cash or securities).

Priority management

Reference Id	***SETT.NTS.OAF.2.1***

All rights reserved.

Each time that an optimisation algorithm needs to sort unsettled *Settlement Transactions* before a selection or de-selection (reverse order is used in the latter case), during the night-time **{T2S.08.130}** the following criteria are used, according to this decreasing weight of order:

- Level of priority **{T2S.07.130} {T2S.07.140} {T2S.07.150}** with the following order:
  - Reserved priority **{T2S.07.160},**
  - Top priority **{T2S.07.170},**
  - High priority **{T2S.07.180},**
  - Normal priority **{T2S.07.190},**
  - In case of linked settlement transactions, the level priority considered is the highest of the *Settlement Transactions* contained in the link set **{T2S.09.130};**
- Oldest intended settlement date **{T2S.08.060} {T2S.08.070}.**

"Gross algorithm"

Reference Id	***SETT.NTS.OAF.3.1***

This algorithm performs gross settlement **{T2S.07.010}** for each pending or created settlement transaction. It uses the *Common Limit and Provision Checking* function to check their settlability. If this check is positive, the *Settlement Transaction* is added to a unique collection which is sent to the *Pre-empting and Booking* functions at the end of the algorithm. If not, the collection is sent as failure to the *Failure Management and Settlement Outcome* function.

This algorithm performs the above processing up to the point where it can't settle any more pending settlement transaction, thus emulating day-time recycling of incoming resources.

During sequence X, "Gross settlement" may resort to partial settlement, only in case of lack of securities.

Conditional resort to Auto-Collateralisation

Reference Id	***SETT.NTS.OAF.4.1***

If the collection contains at least one *Settlement Transaction* eligible to auto-collateralisation **{T2S.07.286},** the function checks that the conditions for requesting auto-collateralisation are met:

- In case of lack of securities, if the securities position is earmarked for a collateral purpose;
- In case of lack of cash, it checks:
  - If the payment/settlement bank which owns the *T2S Dedicated Cash Account* is eligible for CB collateralisation,
  - If the headroom of the relevant auto-collateralisation limit set by the CB is sufficient to fill the lack **{T2S.07.274} {T2S.10.061};**
- In case of insufficient external guarantee headroom, it checks:
  - If the client of the payment/settlement bank is eligible for client-collateralisation with this payment/settlement bank,

All rights reserved.

– If the total amount of headroom for the credit memorandum balance (all types of limits included), is sufficient to cover the need for cash of the settlement transaction(s) related to the client of the payment/settlement bank **{T2S.07.286} {T2S.10.063} {T2S.10.064}.**

If all the relevant checks above are successful, the *Optimisation Algorithms Functions* resort to the *Intraday Credit Provider* functions. In return, the function indicates if the process of auto-collateralisation can solve the lack.

4 – Common Limit and Provision Checking

Except for the processing hereunder, the function is the same as the function used in daytime, and is called as described above, see SETT.R&O.CLP.1.1.

Settlement Restrictions processing

Reference Id	***SETT.NTS.CLP.1.1***

The *Common Limit and Provision Checking* function is used in night mode to provide technical netting and provision check of any settlement transaction, including the one related to *Settlement Restrictions,* as optimisation process is fully integrated into night-time settlement **{T2S.09.180}.**

5 – Partial Settlement Preparation

The function is the same as the function used in daytime, and is called as described above, see SETT.R&O.PSP.1.1.

6 – Pre-empting

The function is the same as the function used in daytime, and is called as described above, see SETT.VPB.PRE.1.1.

7 – Booking

The function is the same as the function used in daytime, and is called as described above, see SETT.VPB.BKG.1.1.

8 – Failure Management and Settlement Outcome

The function is the same as the function used in daytime settlement, but contrary to the daytime (SETT.VPB.FMS.1.1) and since all possible optimisation have been already attempted, the remaining fails will not be proposed to any additional optimisation and the successful settlement will not trigger any recycling.

During the night-time, the *Failure Management and Settlement Outcome* function has also the specificity of cancelling the collection created during the night-time period.

3.5.8.4 Multi-currency aspects

The multi-currency capability is handled in the optimisation process with the management of every currency as a separately missing resource to be solved **{T2S.08.440} {T2S.08.450} {T2S.08.460}.**

As a result, no technical cross-currency cash netting is done: the optimisation processes consider net position on securities and on each currency **{T2S.08.470}.**

All rights reserved.

3.5.8.5 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Event (Start of cycle n / sequence m)	In	Event to launch a night-time sequence	OPSR:Scheduling	
End of Process Event (Start of cycle n / sequence m)	Out	Information event for end of sequence treatment		OPSR:Scheduling
Instruction Status Information	Out	In case of failure, settlement or partial settlement (with the partial quantity or amount corresponding to the quantity or amount actually settled during this partial settlement) Concerning settlement transaction category related to settlement restriction, contains also the restriction reference		LCMM:Status Management
Liquidity Transfer Booking Information	Out			LQMG:Outbound Information Management
Floor / Ceiling Information	Out			LQMG:Outbound Information Management

3.5.8.6 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
STATIC DATA			
Restrictions	Securities Account Restriction	Read	
	T2S Dedicated Cash Account Restriction	Read	
	Party Restriction	Read	
	Restriction type	Read	
	Security Restriction	Read	

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
Headroom	Limit	Read	
Credit Memorandum Balance	Credit Memorandum Balance	Read	
Securities	Securities Valuation	Read	
	Securities	Read	
	Securities Code	Read	
DYNAMIC DATA			
Collection	Collection	Read/Write	
	CollectionTransaction	Read/Write	
	Cash Provisioning Net Flow	Read/Write	
	Securities Provisioning Net Flow	Read/Write	
	CMB Provisioning Net Flow	Read/Write	
Detailed Restrictions	Detailed Securities Restric-tion	Read/Write	
	Detailed Cash Restriction	Read/Write	
Limits	Limit Utilisation	Write	
	Journaling of Limit Utilisation	Write	
Settlement Instruction	Settlement Instruction	Write	
Matching Object	Matching Object	Write	
Settlement Restriction	Settlement Restriction	Write	
Settlement Transaction	Settlement Transaction	Read/Write	
	Restriction Reference	Read	
	Settlement Transaction Link-Set	Read	
	Settlement Transaction Link	Read	
	Partial Settlement Threshold	Read	
Liquidity Transfer	Liquidity Transfer	Write	
Positions and Balances	Securities Position	Write	
	Securities Position Current	Write	
	Cash Balance	Write	
	Cash Balance Current	Write	
Postings	Securities Posting	Write	
	Cash Posting	Write	

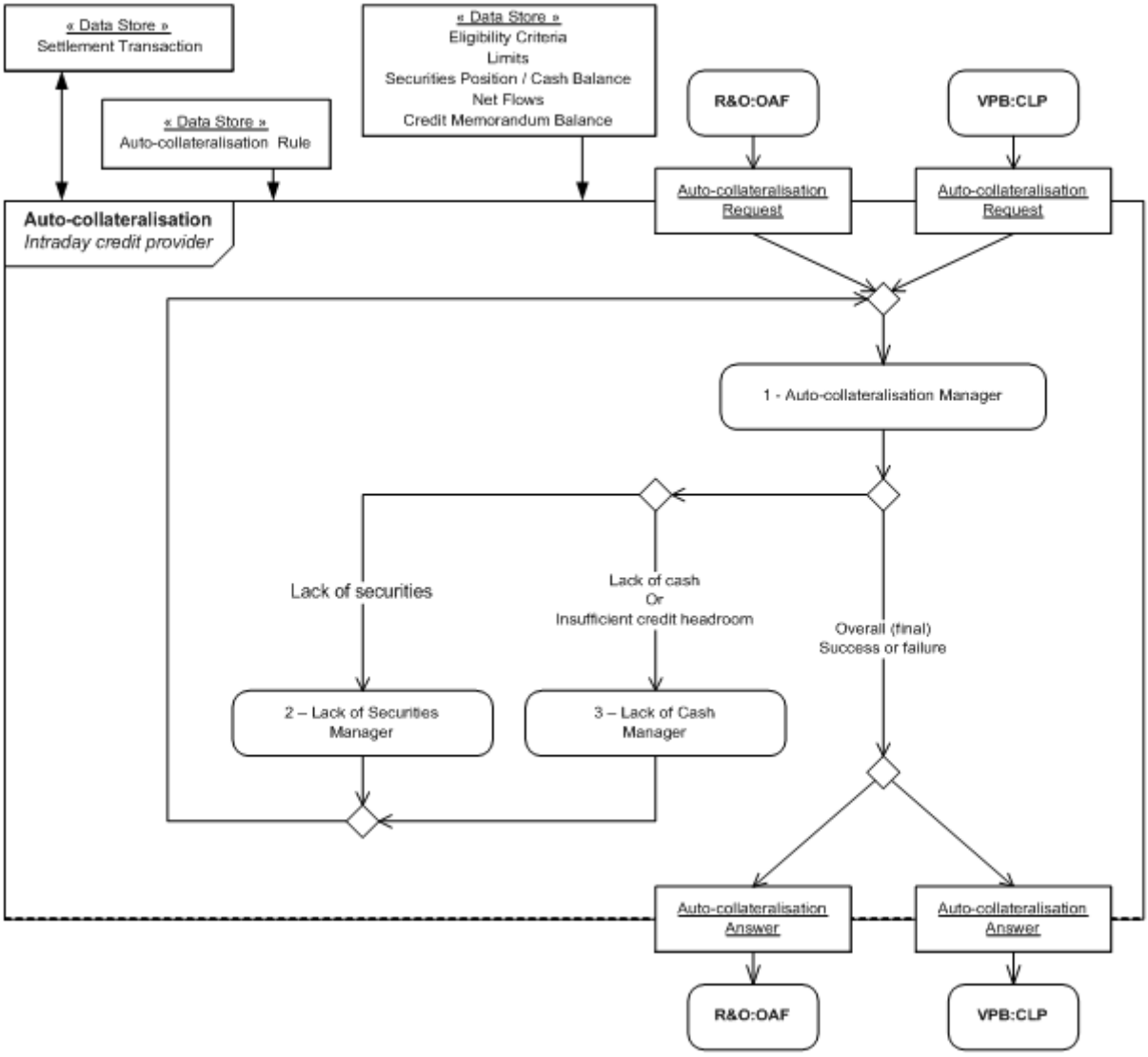
All rights reserved.

3.5.9 Auto-collateralisation

3.5.9.1 Diagrams of the module

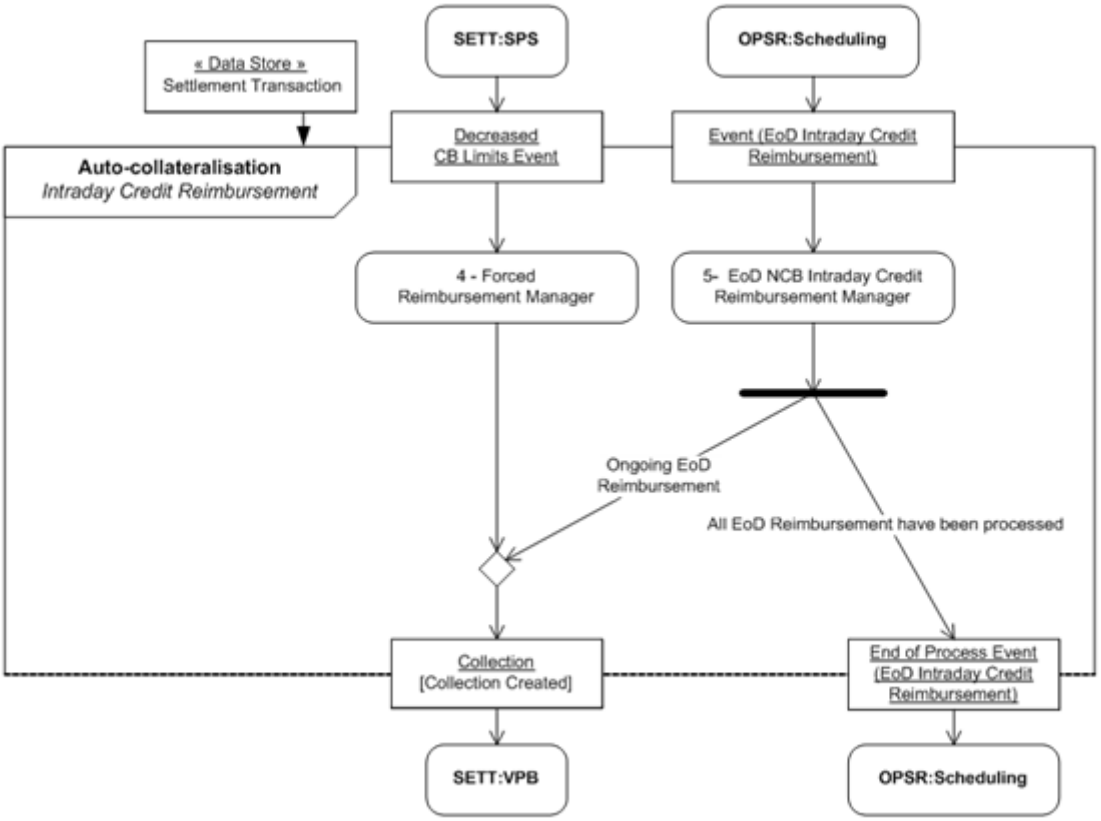
The diagram of the *Auto-collateralisation* module is split in two parts each describing a group of functions for more clarity.

Intraday Credit Provider



All rights reserved.

Intraday Credit Reimbursement



3.5.9.2 Description of the module

The core features of this module are:

- To provide intraday credit, through auto-collateralisation process with central bank (CB collateralisation) { or with payment/settlement bank (client-collateralisation) [176], during the night-time and daytime settlement periods in order to reduce the number of *Settlement Transactions* failing to settle in case of insufficient external guarantee headroom, lack of cash or securities **{T2S.08.480} {T2S.08.040} {T2S.08.050} {T2S.08.490};**
- To handle the reimbursement of intraday credit provided by the Central Bank when:
  - Following a decreased auto-collateralisation limit set by a Central Bank during the settlement day **{T2S.08.800}** [177]**,**
  - Due to the end of day process **{T2S.08.850};**
  - Dynamically, in case of failure during settlement due to securities already collateralised for CB collateralisation and/or client-collateralisation **{T2S.08.910}.**

176 When the specifications are common to CB collateralisation and client-collateralisation, we will refer to the process as "auto-collateralisation".

177 The decrease of an auto-collateralisation limit set by a settlement bank does not trigger any automated reimbursement.

All rights reserved.

The auto-collateralisation is provided for all T2S settlement currencies **{T2S.08.480}** for trading-related *Settlement Transactions* or corporate action related *Settlement Transactions* involving a debit on a *T2S Dedicated Cash Account* **{T2S.08.730}** excluding *Settlement Transactions* related to *Liquidity Transfers* and Cash *Settlement Restrictions..*

Intraday Credit Provider

The Intraday Credit Provider functions are in charge of the auto-collateralisation in the module and are triggered by an *Auto-collateralisation Request* containing the lacks and the insufficient external guarantee headrooms to be solved.

The *Auto-collateralisation Manager* function (function 1) monitors the resolution process for each lack and insufficient external guarantee h headroom, depending on their type.

When all lacks and insufficient external guarantee headrooms have been processed or when a final error has occurred, the *Auto-collateralisation Manager* function sends an answer to the module which sent the *Auto-collateralisation Request* including, when requested by the received request and relevant, the created collateral *Settlement Transactions.*

Intraday Credit Reimbursement

The Intraday Credit Reimbursement functions handle two possible cases of reimbursement of intraday credit, by identifying and selecting the reverse collateral transactions to be settled:

- The function *Forced Reimbursement Manager* function (function 4) handles the forced reimbursement of intraday credit due to an outstanding intraday credit amount already greater than an auto-collateralisation limit decreased by a Central Bank.
- The function *EOD CB Intraday Credit Reimbursement Manager* function (function 5) tries to settle the reverse collateral *Settlement Transactions* to reimburse all pending intraday credit with all the available liquidity and resorting to the relocation if need be.

In any case, a payment/settlement bank can release the reverse collateral transaction initially created, in order to reimburse its pending intraday credit **{T2S.08.827}.**

3.5.9.3 Description of the functions of the module

1 – Auto-Collateralisation Manager

Introduction

Reference Id	***SETT.ACO.ACM.1.1***

The *Auto-collateralisation Manager* function handles the process of lack and insufficient external guarantee headroom resolution through auto-collateralisation (i.e. provision of intraday credit, for lack of cash and for insufficient external guarantee headroom, or dynamic reimbursement of collateralised securities, for lack of securities) **{T2S.08.480}..**

This function:

- For each lack and insufficient external guarantee headroom received, runs the relevant resolution process ;

All rights reserved.

- In case of success, and if requested, creates the necessary collateral *Settlement Transactions* prepared during the previous step;
- Sends the *Auto-collateralisation Answer* to the requesting module, including the auto-collateralisation transaction created when relevant along with the reverse collateral *Settlement Transactions* selected, if any.

Triggering and options of auto-collateralisation manager

Reference Id	***SETT.ACO.ACM.1.2***

The *Auto-collateralisation Manager* function is triggered by an *Auto-collateralisation Request* that may include the request to create auto-collateralisation transactions for the purpose of CB collateralisation or client-collateralisation.

For *Auto-collateralisation Request* including the creation of settlement transactions, the *Auto-collateralisation* module provides back, when all lacks and insufficient external guarantee headrooms are filled in:

- The reverse collateral *Settlement Transactions* to be settled in order to fill in the received lack of securities with collateralised securities either for CB collateralisation or client-collateralisation;
- And/or the collateral *Settlement Transaction* to be settled, in order to provide the intraday credit needed to fill in the received lacks of cash and insufficient external guarantee headrooms, or in order to perform an automated substitution.

The *Auto-collateralisation Manager* function adds the created collateral *Settlement Transactions* in the received collection in order to submit them to settlement on an all-or-none basis.

Outcome of auto-collateralisation

Reference Id	***SETT.ACO.ACM.1.3***

The final outcome of the auto-collateralisation depends on whether the creation of collateral settlement transaction was requested:

- If the creation was requested, as soon as one fail occurs during the lack resolution process or insufficient external guarantee headroom resolution process (e.g. broken limit or insufficient available collateral), the *Auto-collateralisation Manager* function stops its process and sends back a negative *Auto-collateralisation Answer;*
- If the creation was not requested, the function sends an *Auto-collateralisation Answer* describing the capacity to fill in the received lacks and insufficient external guarantee headrooms. Even if a fail occurs during the resolution processes, the *Auto-collateralisation* Manager function continues to process other lacks and insufficient external guarantee headrooms, in order to draw a full picture of the *collateralisation possibilities.*

Resolution processes

Reference Id	***SETT.ACO.ACM.2.1***

The auto-collateralisation manager may, depending on the resources insufficient to settle the incoming collection:

All rights reserved.

- For a lack of securities, trigger the *Lack of Securities Manager* function. The latter may trigger in turn the *Lack of Cash Manager* function to perform a dynamic substitution of collateral **{T2S.08.910};**
- For a lack of cash or insufficient external guarantee headroom, trigger the *Lack of Cash Manager* function. The latter will try to complement the missing resource by issuing one or more new collateral settlement transactions

The resolution process always takes the lacks of securities into account first, as the solution to those may cause or aggravate lacks of cash and/or insufficient external guarantee headrooms.

Monitoring lack of securities

Reference Id	***SETT.ACO.ACM.2.2***

If the processed lack of securities cannot be filled in due to insufficient collateralised securities, the *Lack of Securities Manager* function issues a failure to the *Auto-collateralisation Manager* function.

If the processed lack of securities can be filled in with the release of collateralised securities, a selection of one or more reverse collateral *Settlement Transactions* is sent back to the *Auto-collateralisation Manager* function. The latter may trigger in turn once, or more if needed, the *Lack of Cash Manager* function to fill in possible lacks of cash or insufficiency of credit headroom caused by the reimbursement itself.

Monitoring lack of cash or insufficient external guarantee headroom

Reference Id	***SETT.ACO.ACM.2.3***

If the processed lack of cash or insufficient external guarantee headroom can be filled in with provision of intraday credit, the *Lack of Cash Manager* function sends the prepared collateral *Settlement Transactions* to the *Auto-collateralisation Manager* function.

If the processed lack of cash or insufficient external guarantee headroom cannot be filled in with provision of intraday credit, the *Lack of Cash Manager* sends a failure notification to the *Auto-collateralisation Manager* function. Whether or not the latter considers this failure as final depends on the options of the *Auto-collateralisation request.*

Finalisation of the auto-collateralisation solution

Reference Id	***SETT.ACO.ACM.3.1***

The Auto-collateralisation manager considers its processing complete:

- Either once all lacks and insufficient external guarantee headrooms related to the received *Auto-collateralisation Request* are filled in ;
- Or once a failure has been encountered on a lack of cash or on an insufficiency of credit headroom, if the creation of collateral Settlement Transactions was requested;
- Or once all the lacks and insufficiencies have been processed, whether successfully or not, if the creation of collateral *Settlement Transactions* wasn't requested.

Finalisation of the prepared Settlement Transactions

Reference Id	***SETT.ACO.ACM.3.2***

All rights reserved.

In case of success of the auto-collateralisation processing and if requested, the collateral *Settlement Transactions* are created by the *Auto-collateralisation Manager* function and added to the collection before sending its *Auto-collateralisation Answer:*

- Cash movement including :
  - Debited cash balance of the CB **{T2S.08.500}** and credited cash balance of the payment/settlement bank for a lack of cash,
  - Cash balance of the payment/settlement bank debited and credited simultaneously for an insufficient external guarantee headroom;
- Debited earmarked securities positions and credited securities position for the selected collateral **{T2S.08.710} {T2S.08.720} {T2S.08.572} {T2S.08.574}.**

In case of CB collateralisation, these pieces of information are set in compliance with the auto-collateralisation procedure (Repo, Pledge, Pledge Sub) used by the Central Bank which provides the intraday credit **{T2S.08.700} {T2S.08.572} {T2S.08.574}.**

In case of client-collateralisation, only Repo is available. The credited securities position pertains to the securities account dedicated by the settlement bank to auto-collateralisation with the concerned client **{T2S.08.507} {T2S.08.577}.**

Sending Auto-collateralisation answer

Reference Id	***SETT.ACO.ACM.3.3***

When the request was successfully processed, the *Auto-collateralisation Manager* function sends back the *Auto-collateralisation Answer* to the function which sent the request.

The *Auto-collateralisation Manager* function returns:

- In case of success, a positive *Auto-collateralisation Answer* with, if requested, the selected reverse collateral *Settlement Transactions* and the created collateral *Settlement Transactions* added into the collection received in the *Auto-collateralisation Request* **{T2S.08.570} {T2S.09.090};**
- In case of failure a negative *Auto-collateralisation Answer* indicating that the received collection cannot be settled with the auto-collateralisation with additional information aimed at tackling this failure.

2 – Lack of Securities Manager

Introduction

Reference Id	***SETT.ACO.LSM.1.1***

Triggered by the *Auto-collateralisation Manager* function, the *Lack of Securities Manager* function checks if enough collateralised securities can fill in the lack of securities and if the associated reverse collateral *Settlement Transaction* can be released accordingly.

Collateralised securities check

Reference Id	***SETT.ACO.LSM.2.1***

All rights reserved.

The *Lack of Securities Manager* function checks if there are reverse collateral *Settlement Transactions*, either on hold or pending in the system due to missing resource, which credit the securities position with the lack **{T2S.08.910}** and does not debit a *T2S Dedicated Cash Account* under an intraday settlement restriction.

When the sum of securities provided by these reverse collateral *Settlement Transactions* is insufficient to fill in the lack, the function sends a failure notification to the *Auto-collateralisation Manager* function.

When the sum of securities is sufficient to fill in the lack, the function:

- Selects the reverse collateral *Settlement Transactions* if they are related to a CB collateralisation;
- Selects the reverse collateral *Settlement Transactions* if they are related to a client-collateralisation and can be settled with the available quantity in the delivering securities position;

Additional lack of cash information for substitution

Reference Id	***SETT.ACO.LSM.3.1***

If there aren't enough securities on the delivering securities position of any reverse collateral *Settlement Transaction* regarding client collateralisation, the *Lack of Securities Manager* function triggers itself in an attempt to solve this new lack of securities, through the possible reimbursement of CB collateralisation [178].

Additional lack of cash information for substitution

Reference Id	***SETT.ACO.LSM.4.1***

If there is not enough cash or headroom to settle the reverse collateral *Settlement Transactions* necessary to fill in the lack of securities, an automated substitution is attempted (i.e. collateralisation of other securities positions) by providing new intraday credit using the standard process.

3 – Lack of Cash Manager [179]

Principle of the intraday credit capacity calculation

Introduction

Reference Id	***SETT.ACO.LCM.1.1***

The *Lack of Cash Manager* function identifies the *Settlement Transactions* (flow) and/or securities position (stock) that can be used as collateral either for the *T2S Dedicated Cash Account* with the lack **{T2S.08.730}** or for the credit memorandum balance with the insufficient external guarantee headroom.

For the available collateral identified, the function calculates the potential total amount of intraday credit which can be provided for the settlement of the underlying collection. This amount is called hereafter the intraday credit capacity (ICC), possibly split as on flow c or on stock.

Calculation of the intraday credit capacity through auto-collateralisation on flow

Reference Id	***SETT.ACO.LCM.1.2***

178 The securities missing that prevent the reverse collateral from settlement may have been used by the settlement bank for a NCB collateralisation. Considering the new lack will lead to select the reverse collateral *Settlement Transactions* related to this NCB collateralisation.

179 It is assumed that a Central Bank wishing to offer auto-collateralisation has previously met the pre-requisites in the module static data, i.e. that before being able to allow the access of a settlement bank to auto-collateralisation, the Central Bank has opened the requested central bank cash account, and defined all the parameters for the auto-collateralisation operations, according to its Repo, Pledge or Sub Pledge choice.
Same remark for the static data allowing client-collateralisation between a settlement bank and a settlement bank client.

All rights reserved.

For auto-collateralisation on flow, the function selects in the collection the *Settlement Transactions* eligible to auto-collateralisation:

- That credit a securities position:
  - In a securities account with a link for collateral supply with the considered *T2S Dedicated Cash Account* (case of CB collateralisation) or credit memorandum balance (case of client-collateralisation) **{T2S.08.505},**
  - Earmarked for an auto-collateralisation purpose in the relevant currency [180] **{T2S.05.143} {T2S.08.640} {T2S.08.650} {T2S.08.651};**
- Where the buying securities are:
  - Eligible for an auto-collateralisation in the relevant combination of currency and credit provider **{T2S.16.899},**
  - Eligible for the CSD of the credit provider,
  - Without an applicable valuation of the securities for close link identified with the benefiting T2S Party by the credit provider [181] **{T2S.08.690}.**

For the selected settlement transactions, the function:

- Calculates for each credited securities position, the corresponding flow of collateral (i.e. sum of the securities quantities contained in the selected settlement transactions);
- If the calculated flow of collateral is greater than the provisioning net flow received in the request, the collateral flow is adjusted with the value of the provisioning net flow [182];
- Finally, on the basis of the calculated flow of collateral, the intraday credit capacity on flow for the currency of the lack under examination is computed as follows **{T2S.06.720} {T2S.16.520}:**

ICC on flow	=	Flow of collateral	×	ISIN collateral price for the considered currency and credit provider

Calculation of the Intraday Credit Capacity through auto-collateralisation on stock

Reference Id	***SETT.ACO.LCM.1.3***

The auto-collateralisation on stock complements, when requested, the intraday credit capacity on flow. The function selects the securities positions:

- Belonging to a securities account:

180 The currency used by the T2S Dedicated cash account must be identical to the one linked to the Restriction Type that involves the securities position.

181 The benefiting T2S Party is either the owner of the T2S Dedicated cash account where the lack takes place (CB collateralisation) or the owner of the credited securities account (client-collateralisation)

182 The collateral flow is not calculated with all the *Settlement Transactions* of the underlying collection that impact the T2S Dedicated cash account under examination, but only with the *Settlement Transactions* eligible for the auto-collateralisation on flow that debit the T2S Dedicated cash account where the lack takes place. As a consequence, due to the netting effect, this collateral flow may be greater than the provisioning net flow (calculated with all the transactions). Therefore, in such a case the collateral flow that takes into account only purchased securities needs to be adjusted to the amount of the provisioning net flow.

All rights reserved.

- Linked for collateral supply to the considered *T2S Dedicated Cash Account* (case of CB collateralisation) or credit memorandum balance (case of client-collateralisation) **{T2S.08.630},**
- Without any applicable intraday settlement restriction;

- Earmarked for an auto-collateralisation purpose in the relevant currency **{T2S.08.610} {T2S.08.650} {T2S.08.651};**
- Related to securities:
  - Eligible for an auto-collateralisation purpose in the relevant combination of currency and credit provider **{T2S.16.899};**
  - Eligible for the CSD of the credit provider;
  - For which no close link is identified or an own-used valuation is present with the benefiting T2S Party [183] **{T2S.08.690},**
  - Without any intraday restriction.

The function retrieves in the dynamic data of the T2S system the collateral values of each selected securities position for the considered currency and credit provider either an Eurosystem Central Bank, a non-Eurosystem central bank or payment bank**{T2S.03.360} {T2S.06.720} {T2S.16.690}.**

Two valuations are possible. The standard valuation based on the price of the security in the collateral valuation currency where there is no close link between the credit consumer and the security provided as collateral. The own-use valuation the price of the security in the collateral valuation currency where there is a close link between the credit consumer and the security provided as collateral **{T2S.16.520}.**

It then computes the potential amount of intraday credit which can be obtained with the stock of securities usable as collateral on stock, hereafter called intraday credit capacity on stock, by summing the collateral values.

ICC on stock	=	Sum of the collateral value of each selected securities position for the considered currency and credit provider

Selection of the collateral securities

Reference Id	***SETT.ACO.LCM.2.1***

The *Lack of Cash Manager* function selects the collateral among the collateral flows and securities positions with an intraday credit capacity previously calculated in a way to fill in the lack of cash or to overcome the insufficiency of external guarantee headroom:

- The function selects first collateral on flows and, if necessary, complements it with collateral on stock **{T2S.08.600}.**

183 The benefiting T2S Party is either the owner of the T2S Dedicated cash account where the lack takes place (CB collateralisation) or the owner of a securities account credited by the underlying *Settlement Transactions* at the origin of the insufficient external guarantee headroom.

All rights reserved.

- If auto-collateralisation is possible on several eligible securities, the function seeks the one(s) which provides the lowest total amount of intraday credit, i.e. the minimum total collateral value **{T2S.08.040} {T2S.08.740}.**

The function ensures that the amount of the provided intraday credit:

- Does not exceed the remaining headroom in the applicable auto-collateralisation limits: auto-collateralisation limit set by the Central Bank on the T2S Dedicated cash account [184] having the lack in case of CB collateralisation **{T2S.10.061},** or client-collateralisation limit set on the considered credit memorandum balance by the payment bank providing the intraday credit **{T2S.10.062};**
- Is at least equal to the missing amount **{T2S.08.560} {T2S.08.750},** decreased by the headroom of the unsecured credit limit in case of client-collateralisation **{T2S.08.560} {T2S.08.755};** [185]

Respect of Minimum amount for auto-collateralisation

Reference Id	***SETT.ACO.LCM.2.2***

The function ensures that the amount of the provided intraday credit per collateral Settlement Transaction:

- Is at least equal to the minimum amount for auto-collateralisation {T2S.08.480} {T2S.10.065};

4 – Forced Reimbursement Manager

Introduction

Reference Id	***SETT.ACO.FRM.1.1***

This function handles the forced reimbursement of intraday credit due to an intraday credit already provided greater than a decreased auto-collateralisation limit set by a Central Bank.

The function is triggered by a *Decreased CB Limits Event* sent by the *Standardisation and Preparation to Settlement* module when an updated auto-collateralisation limit set by a Central Bank has a negative remaining amount. This triggering results in forced reimbursements according to the negative remaining amount of the limit and crediting the *T2S Dedicated Cash Account* identified by the credit memorandum balance for which the limit has been decreased.

Forced reimbursement: process

Reference Id	***SETT.ACO.FRM.2.1***

On a *Decreased CB Limits Event* triggering, the amount to reimburse is equal to the negative remaining amount.

In this case, the function:

- Searches the hold reverse collateral *Settlement Transactions* associated to the concerned T2S Dedicated cash account;

184 More exactly, the CB limit is set on a credit memorandum balance of the T2S Dedicated cash account. Only one credit memorandum balance of the T2S Dedicated cash account has the CB limit set on it.

185 Nevertheless, the function favours the selection of collateral versus the use of unsecured credit. The latter should be used only if the function cannot find enough collateral to fill in an insufficient external guarantee headroom.

All rights reserved.

- Selects among these reverse collateral *Settlement Transactions* those for which the sum of settlement amounts is closest to but higher than the negative remaining amount **{T2S.08.800};**
- Identifies these *Settlement Transactions* as "forced reimbursement" in order to give them a reserved priority in case of recycling.

Forced reimbursement: submission to settlement

Reference Id	***SETT.ACO.FRM.3.1***

Once all selected reverse collateral *Settlement Transactions* are released, the *Forced Reimbursement Manager* function creates a collection per reverse collateral *Settlement Transaction* selected.

These created collections are then submitted to settlement. In case of failure of the settlement attempt of a collection, the contained reverse collateral *Settlement Transaction* will be handled by *Daytime Recycling and Optimisation* module.

5 – EOD CB Intraday Credit Reimbursement Manager

Introduction

Reference Id	***SETT.ACO.CRM.1.1***

This function is triggered with an *EOD Intraday Credit Reimbursement* event as part of the real-time settlement closure. It aims at reimbursing all pending intraday credit obtained from a Central Bank with all the available liquidity by submitting to settlement all the reverse collateral *Settlement Transactions* not yet settled, whose currency corresponds to the currency of the received EOD Intraday Credit Reimbursement, whether on hold or not **{T2S.08.850}.**

Preparation of the end-of-day intraday credit reimbursement

Reference Id	***SETT.ACO.CRM.2.1***

To prepare the release of collateralised securities, the function searches all the reverse collateral *Settlement Transactions* which concern CB collateralisation and are unsettled.

For each involved T2S Dedicated cash account, the function checks if the liquidity available on the deliverable cash balance of the impacted *T2S Dedicated Cash Account* allows the reimbursement of their associated pending amount of intraday credit (i.e. the summed amount of the unsettled reverse collateral *Settlement Transactions* selected above).

If the available amount in the deliverable cash balance is insufficient to reimburse its pending amount of intraday credit, the function checks in others *T2S Dedicated Cash Accounts* hold by the same T2S Party, if some exceeding liquidity remains available after the reimbursement of their own associated intraday credit **{T2S.08.850}.**

Selection of the reverse settlement transactions

Reference Id	***SETT.ACO.CRM.3.1***

If the overall available liquidity found is sufficient to reimburse the net pending amount of intraday credit, the function:

All rights reserved.

- Selects the reverse collateral *Settlement Transactions* associated to those T2S Dedicated cash accounts;
- Generates, if necessary, additional *Liquidity Transfers Settlement Transactions* in order to transfer cash from the relevant T2S Dedicated cash accounts on which exceeding cash remains available after the reimbursement of their associated reverse collateral *Settlement Transactions* to the *T2S Dedicated Cash Account* [186] on which cash is insufficient to settle their associated reverse collateral settlement transactions.

Management of pending intraday credit

Reference Id	***SETT.ACO.CRM.4.1***

If the overall available liquidity found on the considered *T2S Dedicated Cash Accounts* is insufficient to reimburse the pending amount of intraday credit (including the transfer of cash from *T2S Dedicated Cash Accounts* having excess cash to *T2S Dedicated Cash Accounts* having a lack), the function tries to fill in the remaining lack of each concerned *T2S Dedicated Cash Account* by selecting collateral among the securities positions credited by the pending reverse collateral *Settlement Transactions* crediting the T2S Dedicated cash account [187]. The intraday credit capacity found must be the closest to, but higher than, the pending amount of intraday credit. Then the function prepares a collateral relocation *Settlement Transaction* **{T2S.08.860}** for each couple of *T2S Dedicated Cash Account* having a lack of cash and securities position selected to supply collateral guaranteeing the intraday credit that will be provided to fill in the lack.

For each collateral relocation settlement transaction:

- the securities movement:
  - Debits the securities position for the quantity of securities to be collateralised,
  - Credits the regular collateral securities account of the Central Bank;
- The cash movement:
  - Debits the deliverable cash balance of the T2S Central Bank account for an amount corresponding to the collateral values of the securities collateralised,
  - Credits the cash balance having the lack.

In the same time the function creates The T2S generated *Settlement Instructions* and their related *Matching Objects* related to these collateral relocation settlement transactions.

The function applies cross-border rules, if need be for this relocation, depending on the location of the settlement bank and/or the Central Bank involved in the RTGS and in T2S and depending on the applicable Central Bank auto-collateralisation rules.

This relocation can be followed by a collateral release or transfer instructed by the relevant Collateral Management System sent in a standard *Settlement Instruction* **{T2S.06.750}.**

Sending the prepared EOD intraday credit reimbursement settlement transactions

186 I.e. T2S Dedicated cash accounts hold by the same T2S Party with the same CB.

187 By using the same algorithms of selection than those used by the Lack of Cash Manager function.

All rights reserved.

Reference Id	***SETT.ACO.CRM.5.1***

The function creates new collections for the released reverse collateral *Settlement Transactions* and, when relevant, the *Settlement Transactions* related to the rebalancing *Liquidity Transfers* and *"DVP"* settlement transactions for collateral relocation.

Those created collections are then submitted for settlement.

Once all reverse collateral *Settlement Transactions* are processed, an *End of process Event* is sent to the *Scheduling* module.

3.5.9.4 Multi-currency aspects

Link between intraday credit currency and T2S dedicated cash account currency

T2S provides auto-collateralisation in all T2S settlement currencies depending on whether the central bank issuing the relevant currency agrees to resort to auto-collateralisation **{T2S.08.480}.**

In case of lack, the auto-collateralisation is possible only when the Central Bank which manages the *T2S Dedicated Cash Account* provides intraday credit through auto-collateralisation in the currency of the *T2S Dedicated Cash Account* concerned by a lack of cash.

Collateral value calculation for securities eligible as collateral in different currencies

The T2S static data stores the collateral price of securities for each currency for which the securities is eligible to auto-collateralisation. T2S is provided with these collateral prices by each relevant Collateral Management System that manages this security as eligible collateral for the auto-collateralisation. During the auto-collateralisation process, in case securities are considered as eligible collateral in different currencies, the collateral values are calculated using the collateral price based on the valuation price and own-use valuation price denominated in the currency of the needed intraday credit.

Dynamic reimbursement and automated substitution of collateralised securities in cross currency context

The dynamic reimbursement or automated substitution of collateralised securities is possible in T2S in a multi-currency context. In particular, when securities already collateralised with a National Central Bank or payment/settlement bank in a given currency need to be released in order to settle a trade on the same ISIN, the dynamic reimbursement or automated substitution facilities allow releasing the collateralised securities even if the cash side of the *Settlement Transaction* are denominated in another currency.

However, in this specific case, the dynamic reimbursement is attempted without using the cash proceeds of the underlying *Settlement Transaction* to reimburse the reverse collateral settlement transactions. Only the cash available on the T2S Dedicated cash that received the intraday credit can be used. If it is insufficient, an automated substitution is triggered.

The main features of the dynamic reimbursement or automated substitution of collateralised securities position in a cross-currency context are presented in the example below.

For example, when a T2S Party sells 100 ISIN1 for GBP 20 while its available quantity is equal to 15 ISIN1, a lack of securities is detected on an earmarked securities position for auto-collateralisation purpose during the provision-checking for 85 ISIN1 and a request is sent to the *Auto-collateralisation* module.

All rights reserved.

In order to fill in the detected lack of securities, the *Auto-collateralisation* module checks if sufficient collateralised securities exist and attempts first a dynamic reimbursement. If the collateralised securities cover an intraday credit in a different currency, the *Auto-collateralisation* module can use for the dynamic reimbursement only the amount available in the *T2S Dedicated Cash Account* which received the intraday credit. If the available amount is not sufficient, the cash proceeds of the underlying *Settlement Transaction* cannot be used since the currency is different from the denomination currency of the outstanding intraday credit.

In this example, 90 ISIN1 are collateralised to cover EUR 18 of pending intraday credit. EUR 17 are necessary to release the 85 ISIN1 requested to fill in the lack of securities. If the amount of cash available on the *T2S Dedicated Cash Account* which received the intraday credit is higher than or equal to EUR 17, the dynamic reimbursement can be done. Otherwise the dynamic reimbursement cannot be done since the GBP 20 received in the *Settlement Transaction* cannot be used to reimburse the intraday credit denominated in EUR.

If the dynamic reimbursement is not possible, an automated substitution is attempted. This substitution results in a new auto-collateralisation to provide the necessary amount in the *T2S Dedicated Cash Account* which receives the intraday credit (in the same currency as the original intraday credit and not in the currency of the trade). If the auto-collateralisation is feasible with other securities to be used as collateral, the substitution is done without any cross-currency substitution (i.e. the amount used for the reimbursement is in the same currency as the original auto-collateralisation).

In this example, if the holding in other securities eligible as collateral in euro is sufficient to provide the EUR 17, the substitution can be done by a *Settlement Transaction* which is included in the collection and a positive answer is sent to the *Daytime Validation, Provisioning and Booking* module which attempts a new settlement.

Otherwise a negative answer is sent back and the related *Settlement Transaction* is unsettled.

3.5.9.5 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Auto-collateralisation Request	IN	Request for an auto-collateralisation to resolve lacks on cash or securities during an actual settlement or an optimisation process	VPB:CLP NTS:CLP R&O:CLP	
Decreased CB Limits Event	IN	Request for a forced reimbursement due to decreased auto-collateralisation limit set by a Central Bank	SETT:SPS	
Event (EOD Intraday Credit Reimbursement)	IN	Event to launch the EOD CB intraday credit reimbursement process	OPSR:Scheduling	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Collection [Collection Created]	OUT	Collection submitted to settlement		SETT:VPB
Auto-collateralisation Answer	OUT	Positive or negative answer to an Auto-collateralisation request. In case of success, the received collection is enriched with the created or released settlement transactions		VPB:CLP NTS:CLP
End of Process Event (EOD Intraday Credit Reimbursement) event	OUT	End of process event sent when the EOD CB reimbursement process is performed		OPSR:Scheduling

3.5.9.6 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
STATIC DATA			
Credit Memorandum Balance	Credit Memorandum Balance	Read	Receiving collateral securities account Regular collateral securities account
	CMB Securities Account Link	Read	Type of the link between a T2S Dedicated cash account and a Securities Account (i.e. cash, or collateral)
Eligibility Criteria	Security Auto-collateralisation Eligibility	Read	ISIN Eligibility as collateral for the lack currency and credit provider
Limits	Limit	Read	Central Bank or payment/ settlement bank auto-collateralisation limits quantity or amount External guarantee limit Amount Unsecured credit limit Amount

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
Auto-collateralisation Rule	Auto-Collateralisation Rule	Read	T2S Central Bank cash account Applicable collateralisation procedure (Repo, Pledge, Pledge Sub-account)
Securities Position / Cash Balance	Payment Bank	Read	Central Bank or Payment/ Settlement-bank Auto-collateralisation Limits quantity or amount Remaining cash to be reimbursed
	Restriction Type	Read	Among other, allows the earmarking of a securities position or securities account for auto-collateralisation purpose. By this way, a T2S Party give its agreement to use a securities position as collateral on stock incoming settlement transactions as collateral on flow
	CMB Securities Account Link	Read	Link to use securities as collateral for a Credit Memorandum Balance
	T2S Dedicated Cash Account	Read	T2S Dedicated cash account where a lack takes place
	Securities Account	Read	Characteristic of the securities account
DYNAMIC DATA			
Limits	Limit Utilisation	Read	Remaining amount for a limit during a settlement day
Securities Position / Cash Balance	Cash Balance Current	Read	Cash Balance checked for releasing some collateralised securities

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
	Securities Position Current	Read	Securities positions checked for auto-collateralisation on stock purpose
Settlement Transaction	Settlement Transaction	Read/Write	Quantity which can be used as collateral on flow Creation of collateral settlement transaction Creation of collateral relocation settlement transaction Release of reverse collateral settlement transaction
	Settlement Transaction Link	Read	Link between collateral settlement transaction(s) and underlying settlement transactions which create the lack of cash
	Settlement Transaction LinkSet	Read	Link between collateral settlement transaction(s) and underlying settlement transactions which create the lack of cash
Net Flows	Cash Provisioning Net Flows	Read/Write	For identifying a possible lack of cash
	Securities Provisioning Net Flows	Read/Write	For identifying a possible lack of securities
	CMB Provisioning Net Flow	Read/Write	For identifying an insufficient external guarantee headroom

3.5.10 Processing of Settlement Instruction Use Cases

3.5.10.1 Selection of representative settlement instructions and liquidity transfers use cases for Settlement

This selection aims at limiting the description of processing to the use cases representing a variation of each criterion (defined as the list of representative SI use cases in the table below) and at illustrating use cases relevant for the *Settlement* domain selected within the table presented in the *Lifecycle Management and Matching* domain.

All rights reserved.

Use cases already presented in the *Lifecycle Management and Matching* domain are recalled on grey background.

The current section therefore only deals with the remaining use cases.

REPRESENTATIVE USE CASE	PROCESS DESCRIPTION FOCUS	CRITERIA						
		CSD CONFIG	COSD	INSTRUCTION TYPE	LINKED	MATCHING	PARTIAL	AUTOCOLLAT
UC-SI-1	Intra-CSD standard settlement	Intra-CSD	No	Settlement Instruction	No	Already Matched	No	Deliverable
UC-SI-2	Linked instructions	Intra-CSD	No	Settlement Instruction	Yes	Unmatched	No	Deliverable
UC-SI-3	Matching	Intra-CSD	No	Settlement Instruction	No	Unmatched	No	Deliverable
UC-SI-4	Partial settlement	Intra-CSD	No	Settlement Instruction	No	Already Matched	Yes	Deliverable
UC-SI-5	Auto-collateralisation	Intra-CSD	No	Settlement Instruction	No	Already Matched	No	Deliverable/ Earmarked
UC-SI-6	Conditional Security Delivery	Intra-CSD		Settlement Instruction	No	Already Matched	No	Deliverable
UC-SI-7a	Use of Blocking	Intra-CSD	No	Settlement Instruction	No	Not Required	No	Deliverable
UC-SI-7b	Blocking	Intra-CSD	No	Blocking Instruction	No	Not Required	No	Deliverable
UC-SI-7c	Earmarking	Intra-CSD	No	Blocking Instruction	No	Not Required	No	Deliverable
UC-SI-8	Reservation	Intra-CSD	No	Reservation Instruction	No	Not Required	No	Deliverable

All rights reserved.

REPRESENTATIVE USE CASE	PROCESS DESCRIPTION FOCUS	CRITERIA						
		CSD CONFIG	COSD	INSTRUCTION TYPE	LINKED	MATCHING	PARTIAL	AUTOCOLLAT
UC-SI-9	Settlement of a corporate action with associated liquidity transfer	Intra-CSD	No	Settlement Instruction	No	Not Required	No	Deliverable
UC-SI-10a	Cross CSD settlement	Cross-CSD	No	Settlement Instruction	No	Not Required	No	Deliverable
UC-SI-10b	In-out T2S settlement	In-out CSD	No	Settlement Instruction	No	Not Required	No	Deliverable
UC-SI-10c	Partial Release	Intra-CSD	No	Settlement Instruction	No	Not Required	Yes	Deliverable

3.5.10.2 Common features for the following processes

The following processing descriptions follow several common features except:

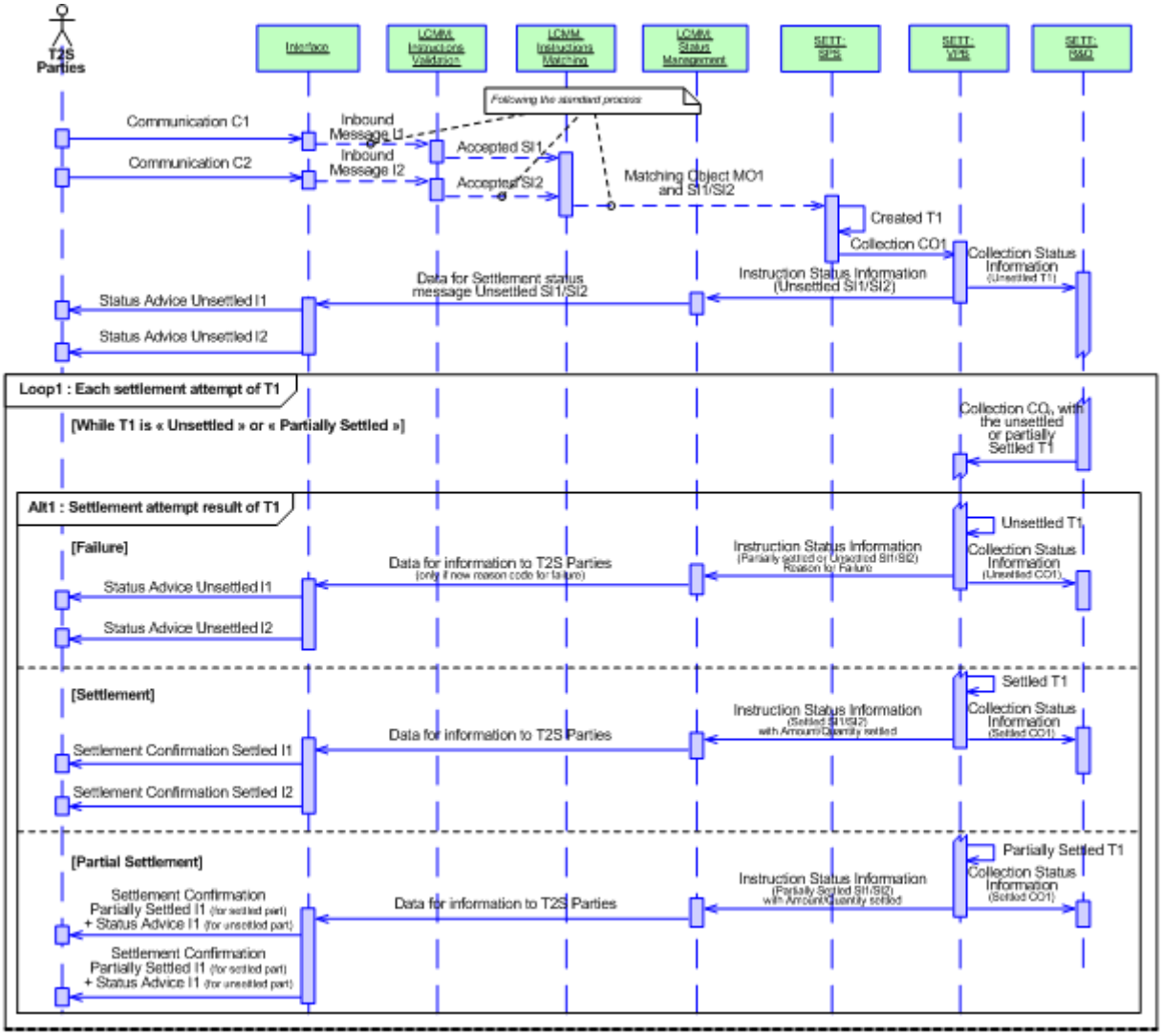
- The processing described is during the daytime settlement window;
- The underlying settlement instructions:
  - Correspond to an "intra" scenario,
  - Are received at their intended settlement date.

When necessary to the description of the processing, other features can be followed. In this case, the business assumption section mentions it explicitly.

All rights reserved.

3.5.10.3 Processing of UC-SI-4 Intra CSD-Standard DVP with Partial Settlement

DIAGRAM 25 - SETTLEMENT PROCESS WITH PARTIAL SETTLEMENT



This use case relates to the processing of *Settlement Instruction* that is eligible to partial settlement and part of a collection containing only a single settlement transaction.

Business assumption

The partial settlement is only possible during a partial settlement window started by an event received from the *Operational Services* domain.

Upon their entry in the *Interface* domain, the communications C1 and C2 follow the standard process as described in UC-SI-1 and UC-SI-3 to their entry in the *Standardisation and Preparation to Settlement* module.

Processing

On reception of the *Matching Object* MO1 and the related *Settlement Instructions* SI1/SI2, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 with all attributes requested for a potential partial settlement:

All rights reserved.

- A Partial Settlement Indicator, confirming the applicability of partial settlement by taking into account the Partial Flag of each original *Settlement Instruction* SI1 and SI2 as well as the absence of linked settlement instruction(s) to SI1 and/or SI2;
- The applicable Threshold Type by taking into account the threshold type of each original *Settlement Instruction* SI1 and SI2.

The *Daytime Validation, Provisioning and Booking* module applies a first settlement attempt on T1 through the collection CO1.

Upon a fail:

- Settlement Transaction Status of T1 and Settlement Status value of SI1 and SI2 are set to *"Unsettled"*;
- An *Instruction Status Information* is sent to the *Status Management* module;
- CO1 is sent to the *Daytime Recycling and Optimisation* module.

Upon reception of *Instruction Status Information,* the *Status Management* module sends to the *Interface* domain the necessary data in order to inform the involved T2S Parties (depending on their message subscriptions).

Loop – Each Settlement attempt of T1

The settlement of the *Settlement Transaction* T1 is attempted as many times as necessary during the partial settlement window, while T1 is not a hold or suspended *Settlement Transaction* and the status of the transaction is either unsettled or partially settled.

If T1 being *"Unsettled"* or *"Partially Settled",* the *Daytime Recycling and Optimisation* sends T1 through a new collection COn to the *Daytime Validation, Provisioning and Booking* module which, in case of lack of securities, calculates the relevant partial quantity (and by consequence the relevant partial amount) during the partial settlement window.

Alternative 1 - Settlement attempt result of T1: Failure

In this case, T1 does not settle:

- T1 Settlement Transaction Status remains *"Unsettled"* or *"Partially Settled"* and sent again to the *Daytime Recycling and Optimisation* module via *Collection Status Information;*
- The SI1 and SI2 Settlement Status remain *"Unsettled"* or *"Partially Settled";*
- The reason for settlement failure of the remaining amount/quantity is sent to the *Status Management* module with an *Instruction Status Information* in order to update the reason code associated to the *Settlement Instruction* settlement status.

Upon reception of *Instruction Status Information* and only in case of an update of the reason code, the *Status Management* module sends to the *Interface* domain the necessary data for the information on the settlement attempt of the involved T2S Parties (depending on their message subscriptions).

Alternative 1 - Settlement attempt result of T1: Settlement

In this case, T1 is settled for its total remaining amount/quantity:

- The *Daytime Validation, Provisioning and Booking* module sets the T1 Settlement Transaction Status to *"Settled"* and the SI1 and SI2 Settlement Status is set to *"Settled";*

All rights reserved.

- An *Instruction Status Information* is sent to the *Status Management* module to confirm that the original *Settlement Instructions* (SI1 and SI2) have their Settlement Status value set to *"Settled"*, along with the amount/quantity actually settled during this settlement;
- The following data are sent to the *Interface* domain for the information of the involved T2S Parties (depending on their message subscriptions):
  - An indicator mentioning that it is the last partial settlement for the underlying settlement instruction,
  - The amount/quantity actually settled during this partial settlement,
  - The Total amount/quantity previously settled equal to the matched amount or the original quantity,
  - The remaining amount/quantity to be settled equal to zero (the settled amount/quantity are equal to the matched amount or original quantity).

Alternative 1 - Settlement attempt result of T1: Partial Settlement

In this case, T1 is settled for a partial amount/quantity:

- The *Daytime Validation, Provisioning and Booking* module sets:
  - The T1 Settlement Transaction Status to *"Partially Settled"* and sends T1 again to the *Daytime Recycling and Optimisation* module,
  - The SI1 and SI2 Settlement Status to *"Partially Settled";*
- An *Instruction Status Information* is sent to the *Status Management* module to confirm that the original *Settlement Instructions* (SI1 and SI2) have their Settlement Status value set to *"Partially Settled",* along with the amount/quantity actually settled during this partial settlement;
- The following data are notably sent to the *Interface* domain for the information of the involved T2S Parties (depending on their message subscriptions):
  - The originally instructed quantity or matched amount,
  - The confirmation of the partial settlement of the settlement instruction,
  - The amount/quantity actually settled during this partial settlement,
  - The total amount/quantity previously settled till the present partial settlement if SI1 and SI2 have been previously partially settled (equal to the previous settled amount/quantity before its update for the present partial settlement),
  - The remaining amount/quantity to be settled (calculated by subtraction of the matched amount or original quantity and the settled amount/quantity after its update for the present partial settlement).

3.5.10.4 Processing of UC-SI-5 Intra CSD-Standard DVP with Auto-collateralisation

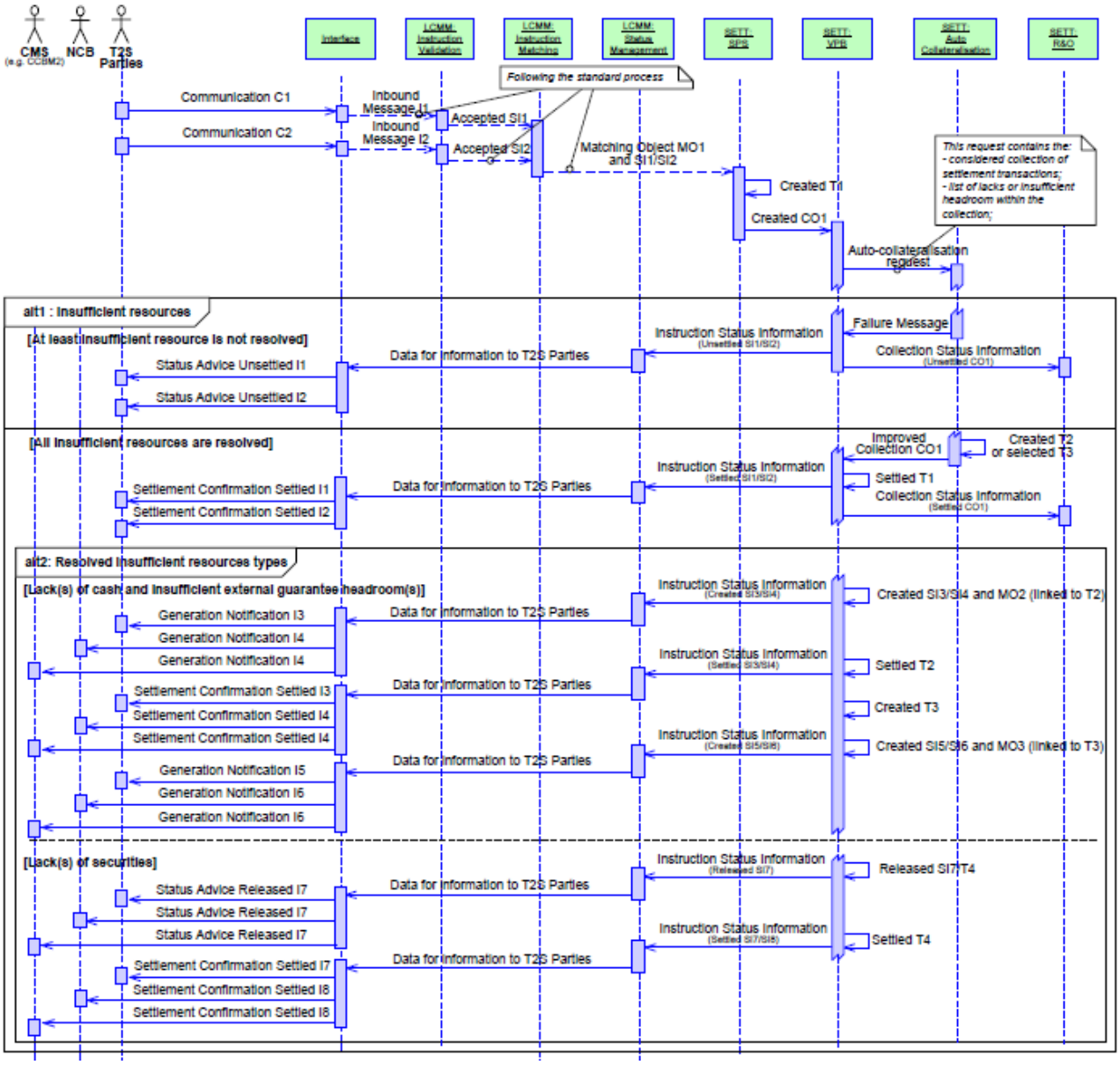
The *Auto-collateralisation* module is called in case of insufficient resources, i.e.:

- Lack of cash (to establish a new CB collateralisation for the provision of intraday credit by the Central Bank);
- Insufficient external guarantee headroom (to establish a new client-collateralisation for the provision of intraday credit by the payment/settlement bank);

All rights reserved.

- Lack of securities (to release collateralised securities by performing a dynamic reimbursement or an automated substitution by providing a new intraday credit).

DIAGRAM 26 - SETTLEMENT PROCESS WITH AUTO-COLLATERALISATION DURING BOOKING PROCESS



This use case relates to a situation where the lack (of cash or of securities) or insufficient external guarantee headroom is detected in the *Common Limit and Provision Checking* function in the *Daytime Validation, Provision and Booking* module during a settlement process.

Business assumption

This use case is described for "Repo" (for CB collateralisation or client-collateralisation) or "Pledge" (for CB collateralisation only) type of collateral settlement (according to auto-collateralisation CB rules). For the "Pledge Sub" (for CB collateralisation only), the differences are:

- As the "Pledge Sub" is handled as a restriction within the securities account, a *Settlement Restriction* is created for the T2S Party to collateralise the securities;

All rights reserved.

- The relevant messages are different too.

Upon their entry in the *Interface* domain, the communications C1 and C2 follow the standard process as described in UC-SI-1 and UC-SI-3 to their entry in the *Standardisation and Preparation to Settlement* module.

This focus starts with the point where:

- The *Common Limits and Provision Checking* function in the *Daytime Validation, Provisioning and Booking* module detects a lack or an insufficient external guarantee headroom;
- A request for auto-collateralisation is sent to the Intraday Credit Provider functions of *Auto-collateralisation* module for the identified lack or insufficient external guarantee headroom after checking the fulfilment of the triggering conditions (including the checks of the T2S Party's eligibility to auto-collateralisation with the credit provider).

Processing

The processing in the Auto-collateralisation module is different for a lack of cash or insufficient external guarantee headroom and for a lack of securities.

In case of a lack of cash or insufficient external guarantee headroom, the *Auto-collateralisation* module:

- Checks the eligibility of the securities to auto-collateralisation for the credit provider in the relevant currency identified with the benefiting T2S Party by the credit provider;
- Checks if the credit consumer has an allowed close link with a present own-used valuation or not allowed one with a null valuation;
- Values the requested collateral amount, resorting preferably to auto-collateralisation on flow – use of credited securities position which are earmarked for collateral purpose - complemented if necessary with auto-collateralisation on stock;
- Checks the collateral value against the auto-collateralisation headroom (set by the relevant CB or the payment/settlement bank) applicable to the relevant credit memorandum balance;
- If case of client-collateralisation, if the auto-collateralisation headroom is not sufficient, checks of the missing amount is covered by the available unsecured credit limit.

In case of a lack of securities, the *Auto-collateralisation* module:

- Checks if the hold reverse collateral settlement transaction(s) can fill in the considered lack if settled;
- Checks if the cash available on the *T2S Dedicated Cash Account* complemented, if necessary, with the incoming positive cash flow is sufficient to settle the considered reverse collateral settlement transaction;
- In case of client-collateralisation reimbursement, checks if the available external guarantee headroom complemented, if necessary, with the incoming positive cash flow is sufficient to settle the considered reverse collateral settlement transaction.

If the available cash or available external guarantee limit is insufficient to perform the dynamic reimbursement, the module attempts an automatic substitution with available collateral.

Alternative 1 – Insufficient resources: At least one insufficient resource is not resolved

All rights reserved.

The *Auto-collateralisation* module returns a failure message to the *Daytime Validation, Provisioning and Booking* module if the auto-collateralisation is not achievable.

The *Daytime Validation, Provisioning and Booking* module:

- Sets the Settlement Status of SI1 and SI2 to "*Unsettled*";
- Sends to the *Status Management* module an *Instruction Status Information* with the settlement result including the reason for failure.

Upon reception of the *Instruction Status Information*, the *Status Management* module informs the involved T2S parties (depending on their message subscriptions).

Alternative 1 - Insufficient resources: All insufficient resources are resolved

If the auto-collateralisation can be achieved resulting in the *Settlement Instruction* being settled, the *Auto-collateralisation* module creates the collateral *Settlement Transaction* T2.

The *Auto-collateralisation* module returns an improved collection CO1 including the original *Settlement Transactions* T1 with the linked collateral *Settlement Transaction* T2 or reverse [188].

During the booking process, for each collateral *Settlement Transaction* resolving a lack or insufficient guarantee headroom, an *Instruction Status Information* is sent to the *Status Management* module about the settlement of the *Settlement Instructions* SI1 and SI2.

Upon reception of the *Instruction Status Information,* the *Status Management* module informs the T2S parties and, in case of CB collateralisation, the CB and its Collateral Management System (CMS) or, in case of client-collateralisation, the payment/settlement bank with a message for the settlement of the *Settlement Instructions* SI1 and SI2 (depending on their message subscriptions).

Alternative 2 – Resolved insufficient resources types: Lacks of cash or insufficient external guarantee headrooms

- The relevant *Matching Object* MO2 and collateral *Settlement Instructions* SI3 and SI4 are created;
- An *Instruction Status Information* is sent to the *Status Management* module about the creation of the collateral *Settlement Instructions* SI3 and SI4;
- An *Instruction Status Information* is sent to the *Status Management* module about the settlement of the collateral *Settlement Instructions* SI3 and SI4;
- The reverse collateral *Settlement Transaction* T3 are created;
- The relevant *Matching Object* MO3 and reverse collateral *Settlement Instructions* are created; the *Settlement Instruction* SI5 (*Settlement Instruction* related to the Payment/Settlement Bank as credit receiver inc case of CB collateralisation or, credit provider in case of client-collateralisation) is created on hold;
- An *Instruction Status Information* is sent to the *Status Management* module about the creation of the collateral *Settlement Instructions* SI5 and SI6.

188 The reverse collateral settlement transaction(s) added to the collection are the ones related to the filling of a lack of securities.

All rights reserved.

Upon reception of the *Instruction Status Information,* the *Status Management* module informs the T2S parties and, in case of CB collateralisation, the CB and its Collateral Management System (CMS) or, in case of client-collateralisation, the payment/settlement bank with:

- A message for the creation of the collateral *Settlement Instructions* SI3 and SI4 (depending on their message subscriptions);
- A message for the settlement of the collateral *Settlement Instructions* SI3 and SI4 (depending on their message subscriptions);
- A message for the creation of the reverse collateral *Settlement Instructions* SI5 and SI6 (depending on their message subscriptions).

Alternative 2 – Resolved insufficient resources types: Lacks of securities

- The selected reverse collateral *Settlement Transaction* T4 is released as well as the associated reverse collateral *Settlement Instruction* SI7 related to the Payment/Settlement bank;
- An *Instruction Status Information* is sent to the *Status Management* module about the release of the collateral *Settlement Instructions* SI7;
- The reverse collateral *Settlement Transaction* T4 are settled;
- An *Instruction Status Information* is sent to the *Status Management* module about the settlement of the collateral *Settlement Instructions* SI7 and SI8.

Upon reception of the *Instruction Status Information,* the *Status Management* module informs the T2S parties and, in case of CB collateralisation, the CB and its Collateral Management System (CMS) or, in case of client-collateralisation, the payment/settlement bank with:

- A message for the release of the reverse collateral *Settlement Instructions* SI7 (depending on their message subscriptions);
- A message for the settlement of the reverse collateral *Settlement Instructions* SI7 and SI8 (depending on their message subscriptions).

3.5.10.5 Processing of UC-SI-6: Conditional Securities Delivery instruction

This use case covers the description of the processing of a CoSD *Settlement Instruction,* where final securities and/or cash booking are subject to CoSD rules defined in the reference data or where securities delivery is dependent on the successful completion of an additional action or event outside T2S (Conditional In/Out CoSD blocking).

Three steps are described here below:

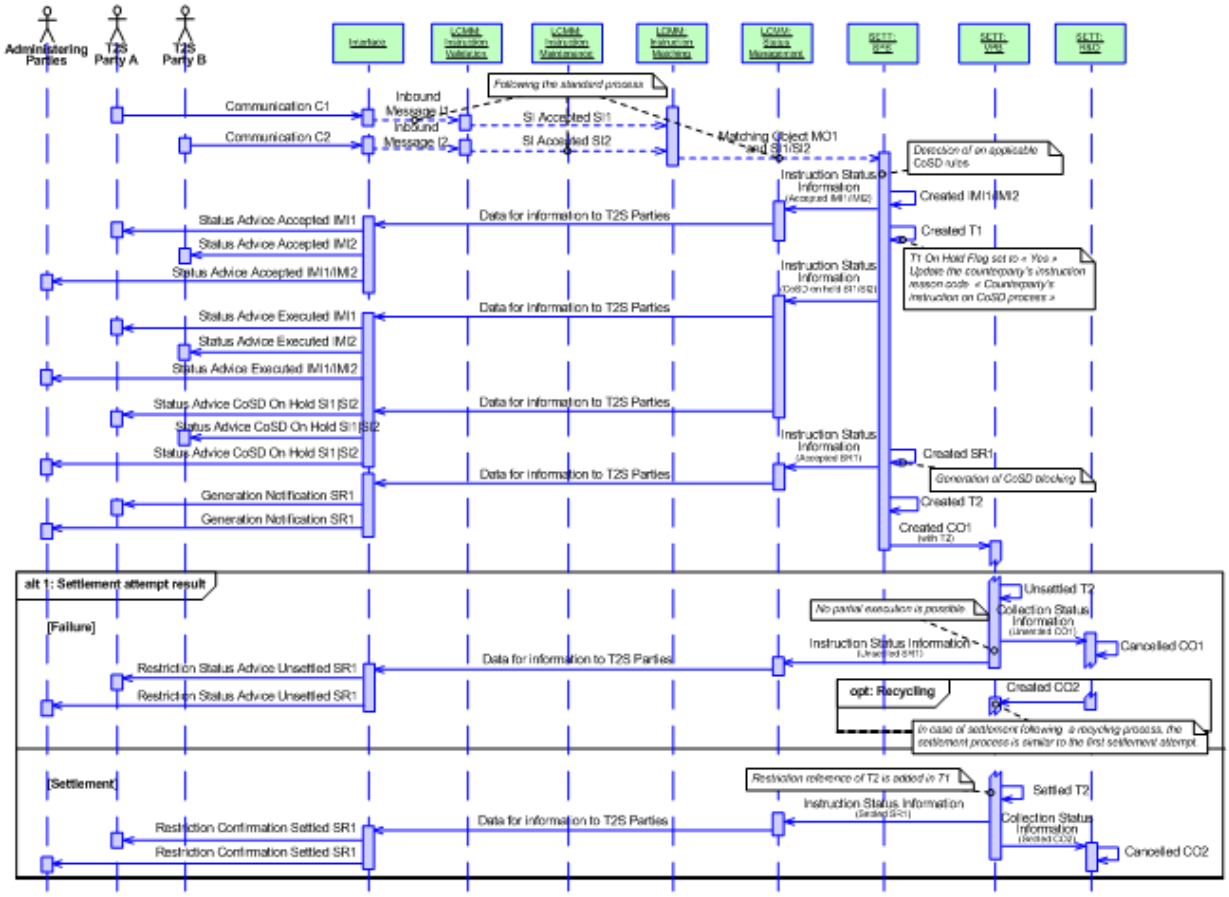
- The CoSD activation (a);
- The CoSD release (b);
- The CoSD cancellation (c).

In the three steps described below, only the daytime settlement is described.

a) CoSD Activation

All rights reserved.

Diagram 27 - Settlement process for CoSD Activation



This use case relates to the processing of *Settlement Instructions* identified as CoSD instructions whose securities delivery is to be blocked.

Business assumption

Upon their entry in the *Interface* domain, the communications C1 and C2 follow the standard process as described in UC-SI-1 and UC-SI-3 to their entry in the *Standardisation and Preparation to Settlement* module, except for the detection of a CoSD in the *CoSD Analyser* functions of the *Standardisation and Preparation to Settlement* module.

In this use case, only one condition applies to *Settlement Instruction* SI1.

Processing

When *Settlement Instructions* SI1 and SI2, and their related *Matching Object* MO1, enter the *Standardisation and Preparation to Settlement* module, this module checks if the CoSD functionality shall be activated on intended settlement date (i.e. if CoSD rules in the reference data apply, if CSDs belonging to the settlement chain are identified as external to T2S).

For the *Settlement Instructions* setting up CoSD activation, the *Standardisation and Preparation to Settlement* module:

- Generates two Hold/Release instructions IMI1 and IMI2, one per CoSD rule applicable to each T2S Party (i.e. T2S Party A and T2S party B), to hold the *Settlement Instructions* SI1 and SI2;

All rights reserved.

- Updates the counterparty's instruction reason code with the value *"Counterparty's instruction on CoSD process";*
- Creates the hold *Settlement Transaction* T1 with the quantity and the amount to be settled between the two participants involved in the CoSD settlement instructions;
- Creates one *Settlement Restriction* SR1 for blocking the securities;
- Generates one CoSD Blocking *Settlement Transaction* T2 corresponding to the *Settlement Restriction* SR1. T2 allows to block the securities to be used in order to settle SI1 and SI2 when they are released by the Administering Party (c.f. § b – CoSD Release);
- Associates the *Settlement Restrictions* SR1 to the *Settlement Instructions* SI1.

Then, both the T2S Actors and the Administering Party are informed of the conditions applying to their *Settlement Instructions,* including their holding for CoSD Activation purpose, and of the creation of the settlement restriction.

During the daytime, T2 is sent directly in a collection CO1 to the *Daytime Validation, Provisioning and Booking* module for a settlement attempt.

Alternative 1 - Settlement attempt result: Failure

Following a settlement attempt in the *Daytime Validation, Provisioning and Booking* module, if T2 is unsettled, the T2S Party A and the Administering Party are informed, via the *Interface* domain and depending on their message subscriptions, of the Settlement Status value *"Unsettled"* concerning the CoSD *Settlement Restriction* SR1.

Option: Recycling

Following a recycling and optimisation in the *Daytime Recycling and Optimisation* module, T2 is submitted to the *Daytime Validation, Provisioning and Booking* module in a new collection CO2 for a new settlement attempt following an optimisation process that has assessed that T2 may be settled successfully.

Alternative 1 - Settlement attempt result: Settlement

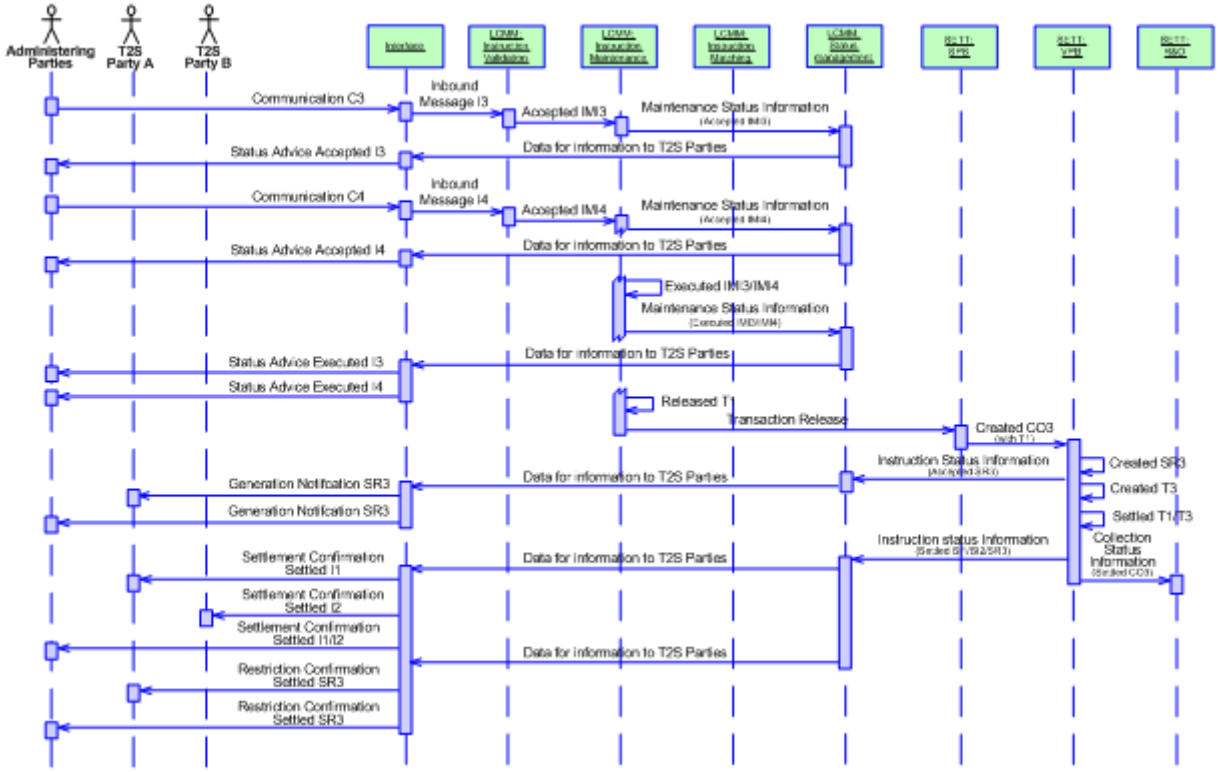
Following a settlement attempt in the *Daytime Validation, Provisioning and Booking* module, T2 is settled. The restriction reference of the CoSD blocking is generated and linked to the *Settlement Transaction* T2, and to the *Settlement Transaction* T1 created at the beginning of the process.

Then, both the T2S Party A and the Administering Party are informed, the restriction reference is communicated via the *Interface* domain and depending on their message subscriptions, of the Settlement Status value *"Settled"* concerning the CoSD *Settlement Restriction* SR1.

b) CoSD Release

All rights reserved.

DIAGRAM 28 - SETTLEMENT PROCESS FOR COSD RELEASE



This use case relates to the processing of the release instruction sent by the Administering Party to release SI1 and SI2, previously set to "On hold" during the CoSD activation.

Business assumption

The communication messages C3 and C4 corresponding to the release instructions releasing the *Settlement Instructions* SI1 and SI2 are submitted by the Administering Party.

These communication messages are sent from the *Interface* domain, as Inbound Messages I3 and I4 to the *Instruction Validation* module, in order to be validated and forwarded to the *Instruction Maintenance* module.

The *Lifecycle Management and Matching* domain has to wait to receive all the release instructions from the Administering Party prior to release the settlement instructions.

Processing

The *Instruction Maintenance* module records the reception of the release instructions IMI3 and IMI4 from the Administering Party.

Once the release instructions IMI3 and IMI4 are successfully validated, the Administering Party is informed via the *Interface* domain of the validation of its release instructions by a status advice value "Accepted".

Once all the release instructions are received, the Administering Party is informed, via the *Interface* domain and depending on their message subscriptions, of the status advice value "Executed".

At this step, the *Instruction Maintenance* module:

- Releases the *Settlement Instructions* SI1 and SI2 (CoSD Hold Status set to *"No"*);

All rights reserved.

- Sends a *Transaction Release* to the *Standardisation and Preparation to Settlement* module to release the *Settlement Transaction* T1.

Once the *Transaction Release* message is received in the *Standardisation and Preparation to Settlement* module, T1 is sent directly to the *Daytime Validation, Provisioning and Booking* module in a collection CO3 for a settlement attempt with the use of blocked securities corresponding to the CoSD blocking.

The *Daytime Validation, Provisioning and Booking* module then:

- Generates the related *Settlement Transaction* T3 to transfer the blocked securities to the deliverable securities position;
- Includes in the collection CO3 the *Settlement Transaction* T3 previously created;
- Creates one *Settlement Restriction* SR3 corresponding to T3;
- Sends information to the *Status Management* module for the information of Administering Party and T2S Party A about the creation of the *Settlement Restriction* SR3 (depending on their message subscriptions).

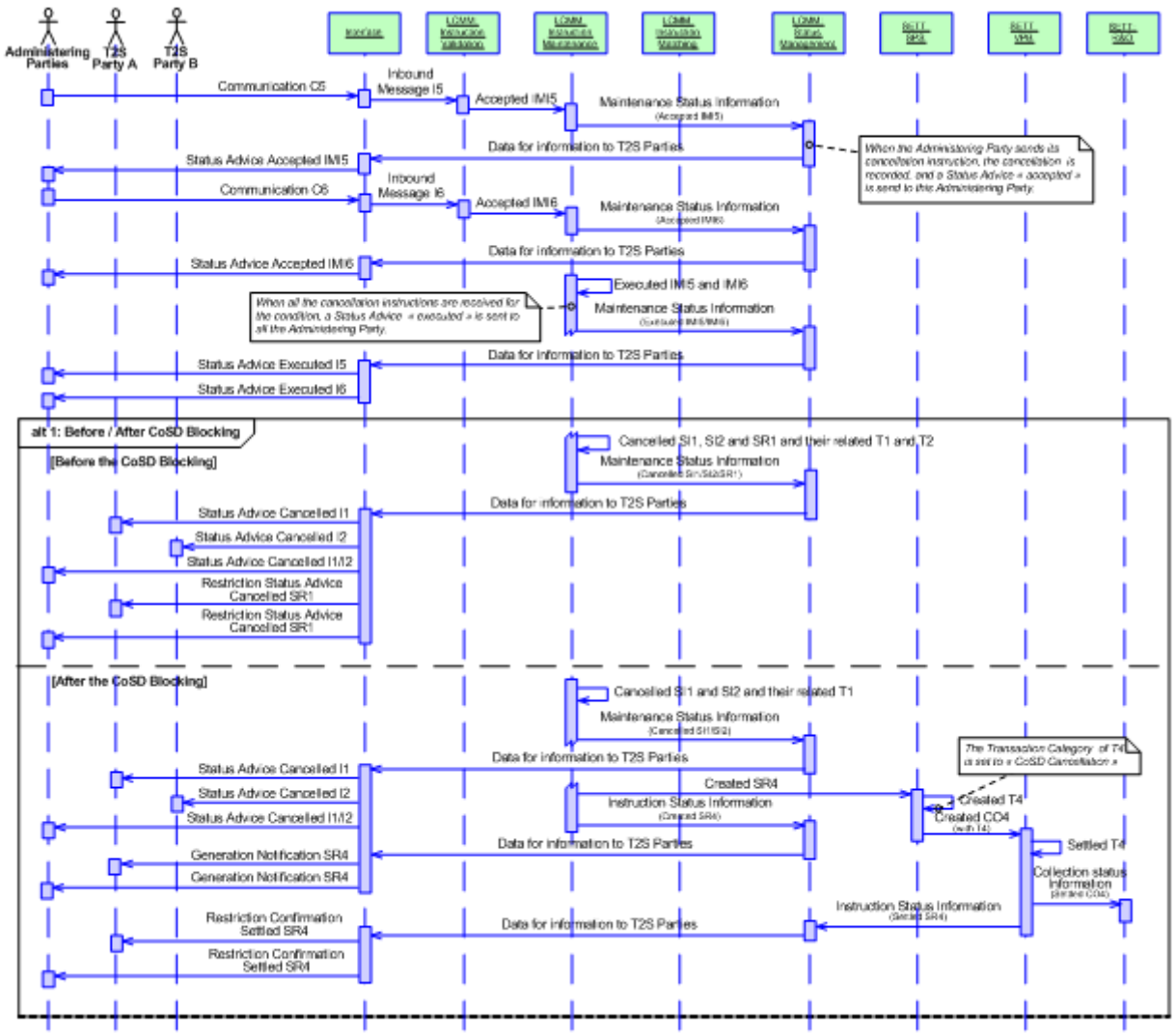
Once the collection CO3 is settled:

- The Settlement Transaction Status of the *Settlement Transactions* T1 and T3 are set to *"Settled";*
- An *Instruction Status Information* is sent to the *Status Management* module for the information of Administering Party and T2S Parties (depending on their message subscriptions).

c) CoSD Cancellation

All rights reserved.

DIAGRAM 29 - SETTLEMENT PROCESS FOR COSD CANCELLATION



This use case relates to the processing of the cancellation instructions sent by the Administering Party to cancel the SI1 and SI2, previously created during the CoSD activation.

Business assumption

The communication messages C5 and C6 corresponding to the cancellation instructions cancelling the *Settlement Instructions* SI1 and SI2, are submitted by the Administering Party.

These communication messages are sent from the *Interface* domain, as Inbound Message I5 and I6 to the *Instruction Validation* module in order to be validated and forwarded to the *Instruction Maintenance* module.

The *Lifecycle Management and Matching* domain has to wait to receive all the cancellation instructions from the Administering Party prior to cancel the *Settlement Instructions* SI1 and SI2.

Processing

The *Instruction Maintenance* module records the reception of the cancellation instructions C5 and C6 from the Administering Party.

All rights reserved.

Once the cancellation instructions IMI5 and IMI6 are successfully validated, the Administering Party is informed via the *Interface* domain of the validation of its cancellation instructions by a status advice value "Accepted".

Once all cancellation instructions are received, the Administering Party is informed, via the *Interface* domain and depending on their message subscriptions, of the status advice value "Executed".

At this step, the process differs depending on whether the CoSD blocking related to the *Settlement Instructions* has already taken place or not.

Alternative 1 - Before/After CoSD Blocking: Before the CoSD Blocking

If the CoSD blocking has not taken place (i.e. SR1/T2 are not yet settled), the *Instruction Maintenance* module:

- Cancels the *Settlement Instructions* SI1 and SI2 and their related *Settlement Transaction* T1;
- Cancels the *Settlement Restriction* SR1 and its related *Settlement Transaction* T2;
- Sends information to the *Status Management* module in order to inform both the T2S Parties and the Administering Party, via the *Interface* domain and depending on their message subscriptions, of the *Settlement Instructions* and restriction status values *"Cancelled".*

Alternative 1 - Before/After CoSD Blocking: After the CoSD Blocking

If the CoSD blocking has already taken place (i.e. the restriction references of the CoSD *Settlement Instructions* are created and linked to SI1 and SI2), then the *Instruction Maintenance* module:

- Cancels the *Settlement Instructions* SI1 and SI2 and their related *Settlement Transaction* T1;
- Sends information to the *Status Management* module in order to inform both the T2S Parties and the Administering Party, via the *Interface* domain and depending on their message subscriptions, of the *Settlement Instructions* status values *"Cancelled"*;
- Creates one *Settlement Restriction* SR4 to unblock the CoSD blocking for the securities;
- Sends information to the *Status Management* module for the information of Administering Party and T2S Party A about the creation of the *Settlement Restriction* SR4 depending on their message subscriptions;
- Sends information to the *Standardisation and Preparation to Settlement* module about the creation of the *Settlement Restriction* SR4.

Once information about the creation of the *Settlement Restriction* SR4 is received in the *Standardisation and Preparation to Settlement* module, the *Settlement Transaction* T4 is created associated to the *Settlement Restriction* SR4.

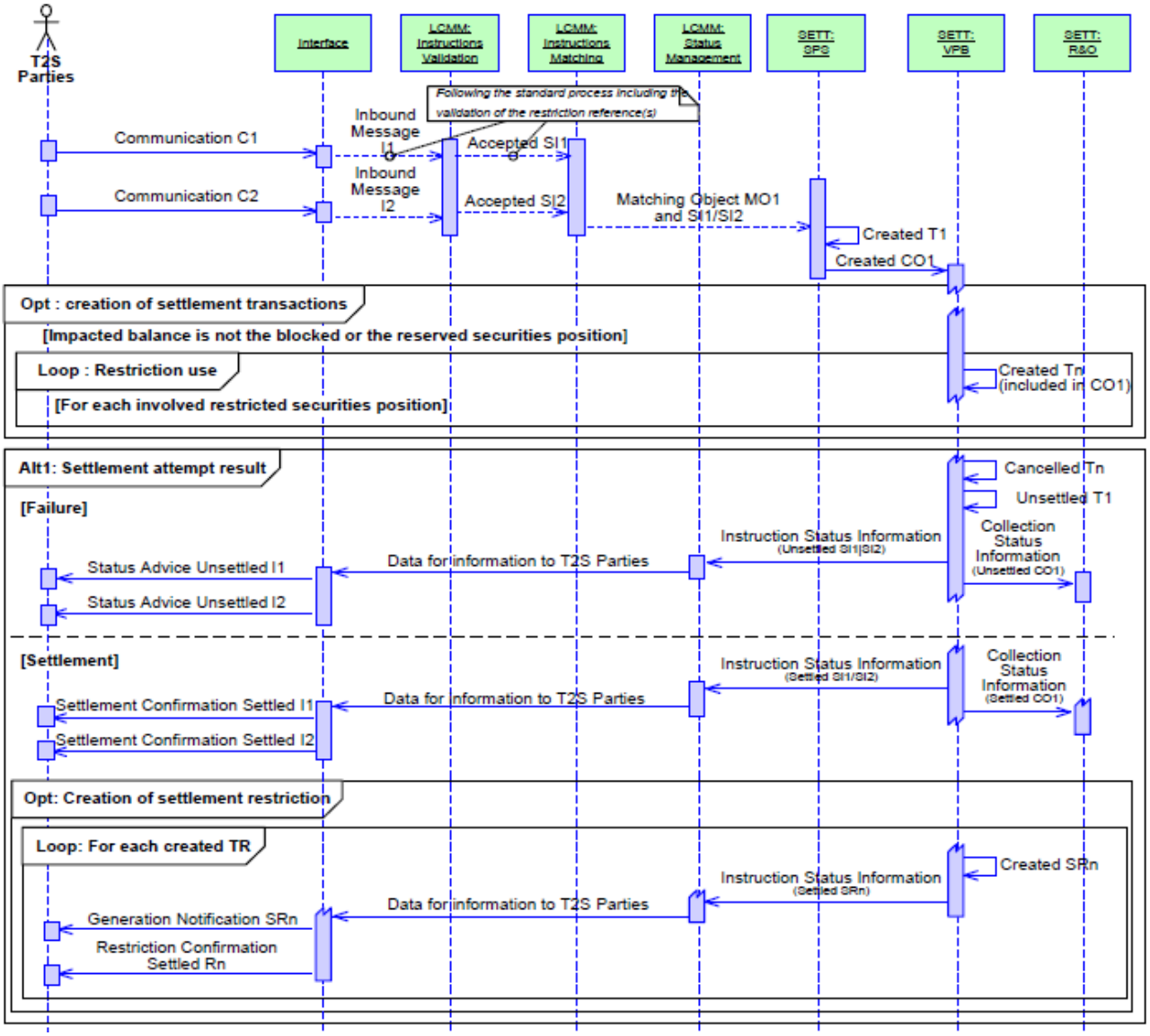
During the daytime, T4 is sent directly to the *Daytime Validation, Provisioning and Booking* module in the collection CO4 for a settlement attempt.

Upon the settlement of T4, the *Daytime Validation, Provisioning and Booking* module sends an *Instruction Status Information* to the *Status Management* module in order to inform both the T2S Party A and the Administering Party, via the *Interface* domain and depending on their message subscriptions, of the restriction status values *"Settled".*

All rights reserved.

3.5.10.6 Processing of UC-SI-7a: Use of blocking in a settlement instruction

DIAGRAM 30 - USE OF BLOCKING IN A SETTLEMENT INSTRUCTION



Business assumption

Upon their entry in the *Interface* domain, the communications C1 and C2 follow the standard process as described in UC-SI-1 and UC-SI-3 taking into account the restricted resources required to be used for the settlement.

The processing of cash blocking or reservation follows the same principle, except that the so-called optional parts are mandatory.

Processing

After the successful validation, including the checking of the restriction reference(s) when filled, the *Instruction Validation* module creates the *Settlement Instructions* SI1 and SI2.

All rights reserved.

The *Instruction Matching* module performs the matching, creates the related *Matching Object* MO1 and sends the *Settlement Instructions* SI1 and SI2 and their related *Matching Object* MO1 to the *Standardisation and Preparation to Settlement* module for further processing.

On the intended settlement date, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 from the *Matching Object* and the *Settlement Instructions* received.

During the daytime window, a collection CO1 containing the *Settlement Transaction* T1 is created and sent to the *Daytime Validation, Provisioning and Booking* module.

Before performing the provision-checking with all the resources available to settle T1, the *Daytime Validation, Provisioning and Booking* module prepares the use of restricted resources and checks the restriction reference(s):

- If the restriction reference(s) is (are) consistent with the deliver and/or receiver securities position, T1 is valid for further processing;
- If not, T1 is rejected and the collection fails.

Option 1: Creation of settlement transactions

If the impacted balance is a blocked or a reserved securities position, T1 is simply settled and the module does not create any additional settlement transaction.

If the impacted balance is neither a blocked, nor a reserved securities position, the module creates one *Settlement Transaction* Tn for each *securities position* identified by the *restriction type* related to the restriction reference(s) specified.

That *Settlement Transaction* provides the deliverable/earmarked securities position with the resources available into the detailed securities position related to the restriction reference(s).

The *Settlement Transactions* Tn, added to the collection CO1, notably contains:

- Deliver securities position identification;
- Receiver securities position identification;
- Settlement Transaction Category is set to *"Restriction Use";*
- Original quantity is equal to the sum of all the quantities (limited to the restriction references indicated in the settlement instruction) per securities position identified through a given restriction type;
- Link with the original trade *Settlement Transaction* T1.

Then, a provision-checking is carried out to check if the available quantity allows the settlement of T1:

- If the impacted balance is neither a blocked, nor a reserved securities position, the available quantities is checked regarding the *Detailed Securities Restriction* required to be used and, if necessary, is complemented with the deliverable/earmarked securities position;
- If the impacted balance is a blocked, or a reserved securities position, the available quantity is checked regarding the *Detailed Securities Restriction* required to be used and limited to the restriction references indicated in the settlement instruction.

The settlement attempt result depends on the results of this provision-checking.

All rights reserved.

Alternative 1 - Settlement attempt result: Failure

If the provision-checking is not satisfactory:

- The whole collection CO1 is unsettled;
- The *Settlement Transactions* Tn created to provide the restricted resources are cancelled;
- SI1 and SI2 Settlement Status value are set to *"Unsettled";*
- The unsettled collection is sent to the *Daytime Recycling and Optimisation* module for optimisation.

The *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending it an *Instruction Status Information.*

The T2S Parties are informed (depending on their message subscriptions) with a message which includes the reason for failure.

Alternative 1 - Settlement attempt result: Settlement

If the provision-checking is satisfactory:

- If the impacted balance is a blocked or a reserved securities position, T1 is settled with all the resources available into the restricted securities position concerned by the restriction references (i.e. sum of all the remaining quantity in each detailed securities restriction);
- If the impacted balance is neither a blocked nor a reserved securities position, T1 and all the linked *Settlement Transactions* Tn are settled in an "all or none" basis;
- Each detailed securities restriction(s) identified by each restriction reference specified is updated with the quantity actually used;
- Each securities position created/updated with the quantity actually booked;
- One securities posting is created per movement on each securities position;
- The original SI1 and SI2 settlement Instructions Settlement Status value are set to *"Settled".*

Option 1: Creation of settlement restrictions

For each created *Settlement Transactions* Tn a *Settlement Restriction* is created with the status *"Settled".*

The *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending it an *Instruction Status Information.*

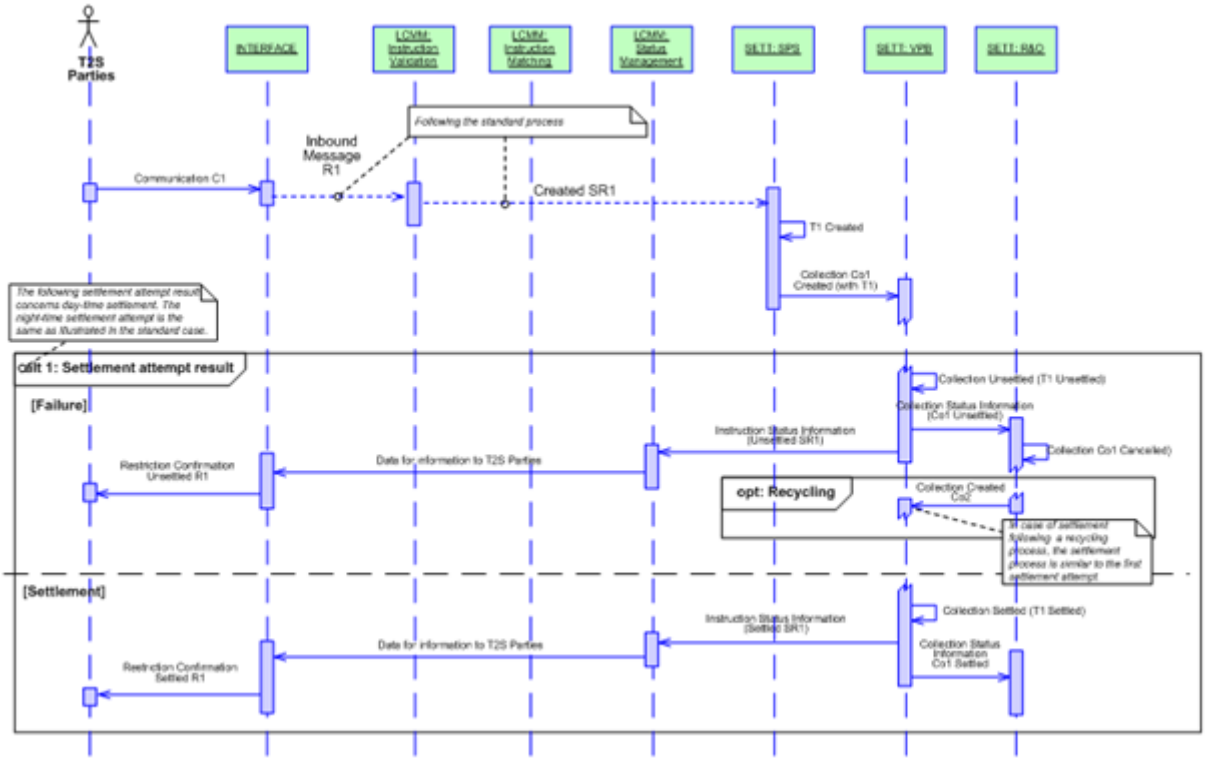
The T2S Parties are informed (depending on their message subscriptions) with a message related to each *Settlement Instruction* and per *Settlement Restriction* SRn created.

3.5.10.7 Processing of UC-SI-7b: Blocking

a) Set-up of a blocking position

All rights reserved.

DIAGRAM 31 - SET-UP OF A BLOCKING POSITION



Business assumption

This description relates to the processing of a securities blocking from the entry of the communication C1 (determined by the *Instruction Validation* module as a blocking settlement restriction) to the reporting of the settlement result to the T2S party.

The processing of a set up of a cash blocking, follows the same principle.

Processing

After a successful validation, the *Instruction Validation* module creates the *Settlement Restriction* SR1 (no matching required).

On the intended settlement date, the *Instruction Validation* module sends the *Settlement Restriction* SR1 to the *Standardisation and Preparation to Settlement* module for further processing.

When SR1 is received, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 from the original settlement restriction.

A collection CO1 containing the *Settlement Transaction* T1 is created and sent to the *Daytime Validation, Provisioning and Booking* module.

The provision-checking is carried out to determine the available quantity in the deliverable (or earmarked) securities position , and whether it allows the settlement of T1.

Alternative 1 - Settlement attempt result: Failure

If CO1 is unsettled, the *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

All rights reserved.

The T2S Parties are informed of the unsettled status (depending on their message subscriptions) with a status advice message.

A *Collection Status Information* is sent to the *Daytime Recycling and Optimisation* module to launch a standard optimisation process.

Alternative 1 - Settlement attempt result: Settlement

If CO1 is settled, the *Daytime Validation, Provisioning and Booking* function does the following:

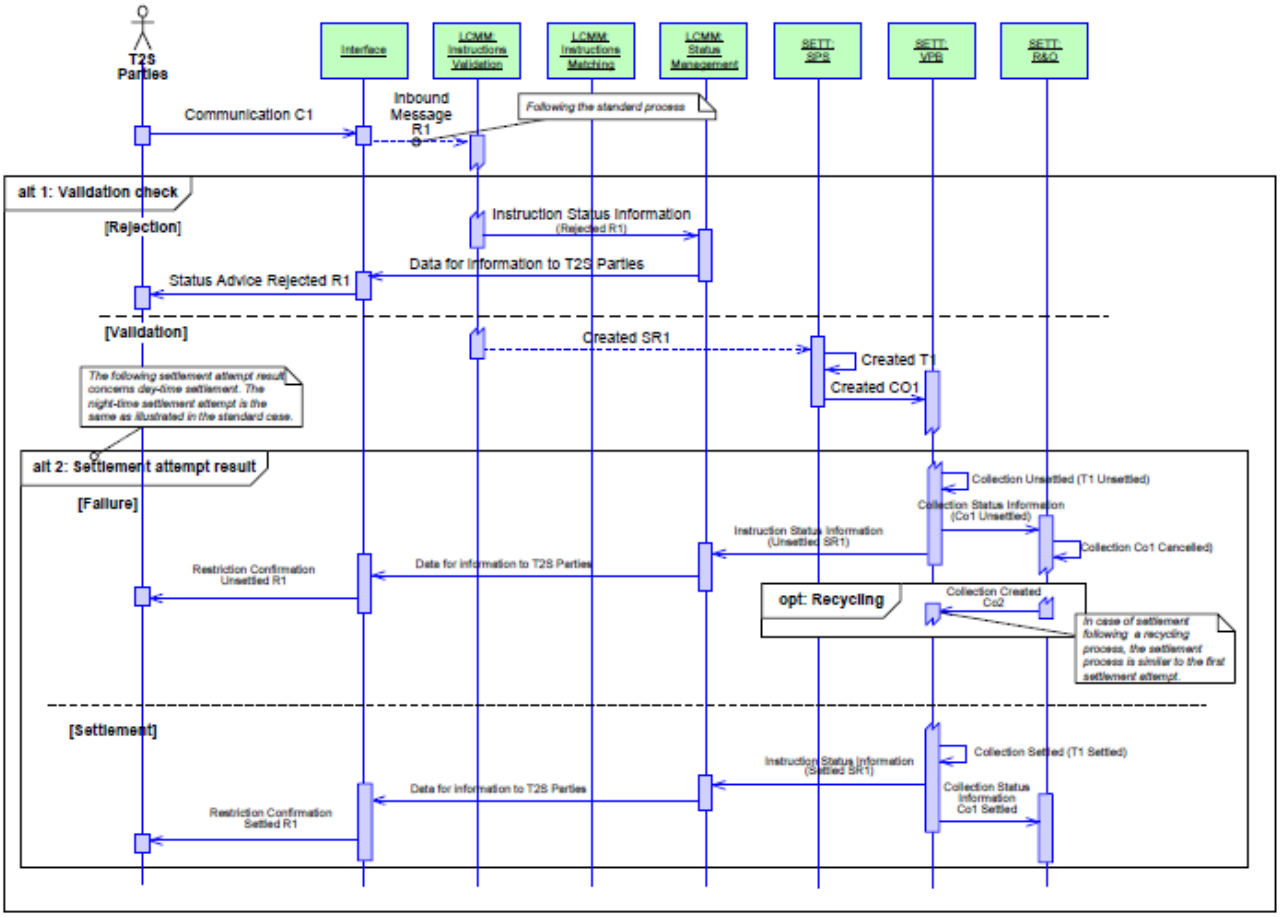
- It informs the *Status Management* module with the settlement result by sending an *Instruction Status Information*
- T1 is settled *per se*:
  - The *Detailed Securities Restriction* identified by a restriction reference is created with the available quantity,
  - Each Securities position is created/updated with the booked quantity,
  - The original SR1 *Settlement Restriction* Settlement Status value is set to *"Settled"*;
- Two securities postings are created related to the movements on each securities position;

The T2S Party is informed of the settled status (depending on his message subscriptions) with a settlement confirmation message including the created restriction reference.

b) Increase or decrease of a blocking position

All rights reserved.

DIAGRAM 32 - INCREASE OR DECREASE OF A BLOCKING POSITION



Business assumption

The description relates to the processing of an increase or a decrease of a securities blocking position from the entry of the communication C1 to the reporting of the settlement result to the T2S party.

The processing of an increase or decrease of a cash balances follows the same principle.

Processing

Firstly, the *Instruction Validation* module performs the validation of the received Communication C1, including the checking of the restriction reference.

Alternative 1 – Validation check: Rejection

If the Communication C1 is rejected (including if the indicating Restriction Reference is unknown), the *Instruction Validation* module informs the *Status Management* module by sending it an *Instruction Status Information*. The T2S Party is then informed (depending on their message subscriptions).

Alternative 1 – Validation check: Validation

If the Communication C1 is validated, the *Instruction Validation* module creates the *Settlement Restriction* SR1 (no matching required).

On the intended settlement date, the *Instruction Validation* module sends *Settlement Restriction* SR1 to the *Standardisation and Preparation to Settlement* module for further processing.

All rights reserved.

When SR1 is received, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 from the original *Settlement Restriction* with notably:

- Deliver securities position identification;
- Receiver securities position identification;
- Settlement Quantity of the original settlement restriction;
- Restriction reference provided by T2S party.

A collection CO1 containing the *Settlement Transaction* T1 is created and sent to the *Daytime Validation, Provisioning and Booking* module.

Next, the function checks the available quantity in the delivering securities position or detailed restriction, and whether it allows the settlement of T1.

Alternative 2 - Settlement attempt result: Failure

If CO1 is unsettled, the *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

The T2S Parties are informed of the unsettled status (depending on their message subscriptions) with a status advice message.

A *Collection Status Information* is sent to the *Daytime Recycling and Optimisation* module to launch a standard optimisation process.

Alternative 2 - Settlement attempt result: Settlement

If CO1 is settled, the *Daytime Validation, Provisioning and Booking* function does the following:

- It informs the *Status Management* module with the settlement result by sending an *Instruction Status Information*;
- T1 is settled *per se*:
  - The *Detailed Securities Restriction* identified by a restriction reference is created with the available quantity,
  - Each Securities position is created/updated with the booked quantity,
  - The original SR1 *Settlement Restriction* Settlement Status value is set to "*Settled*";
- Two securities postings are created related to the movements on each securities position.

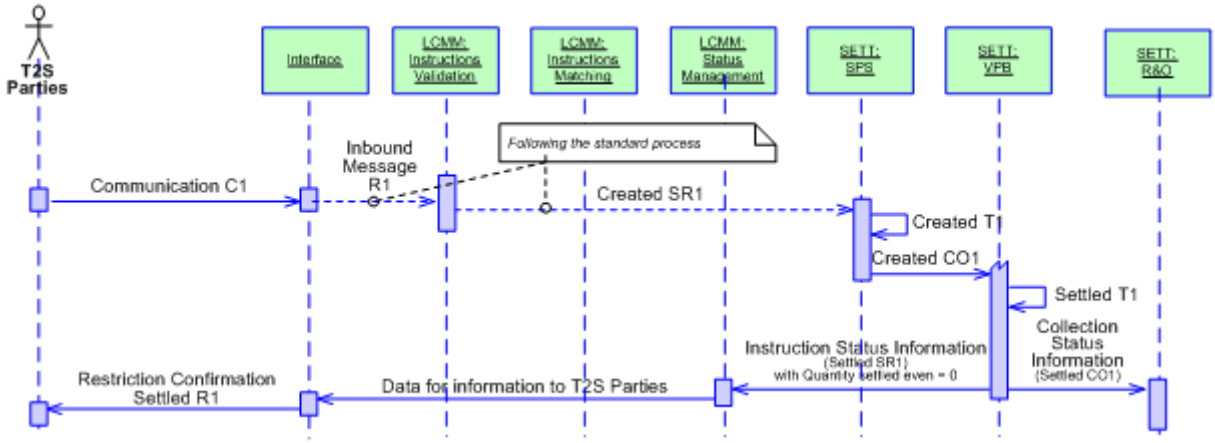
The T2S Party is informed of the settled status (depending on his message subscriptions) with a settlement confirmation message.

3.5.10.8 Processing of UC-SI-7c: Earmarking

a) Set-up or increase of an earmarking position with a settlement restriction

All rights reserved.

DIAGRAM 33 - SETUP OR INCREASE OF AN EARMARKING WITH A SETTLEMENT RESTRICTION



Business assumption

This description relates to the processing of an earmarking from the entry of the communication C1 (determined by the *Instruction Validation* module as an earmarking settlement restriction) to the reporting of the settlement result to the T2S party.

Processing

After a successful validation, the *Instruction Validation* module creates the *Settlement Restriction* SR1 (no matching required).

On the intended settlement date, the *Instruction Validation* module sends the *Settlement Restriction* SR1 to the *Standardisation and Preparation to Settlement* module for further processing.

When SR1 is received, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 from the original *Settlement Restriction* with notably:

- Deliver securities position identification;
- Receiver securities position identification;
- Settlement Quantity of the settlement Restriction.

A collection CO1 containing the *Settlement Transaction* T1 is created and sent to the *Daytime Validation, Provisioning and Booking* module.

The provision-checking is carried out to determine the available quantity in the deliverable securities position allowing the settlement of T1.

After a positive provision-checking, the *Booking* function performs the following processes:

- T1 is settled with the available quantity of securities moved from the deliverable securities position to the earmarked securities position;
  - If the earmarked securities position does not exist, it is a set up of earmarking. The earmarked securities position is created with the available quantity of securities,
  - If the earmarked securities position exists, it is an increase of earmarking. The earmarked securities position is updated with the available quantity of securities;

All rights reserved.

- Each securities position is created/updated with the quantity actually booked;
- Two securities postings are created related to the movements on each securities position;
- The original SR1 *Settlement Restriction* Settlement Status value is set to *"Settled"*.

The *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending it an *Instruction Status Information.*

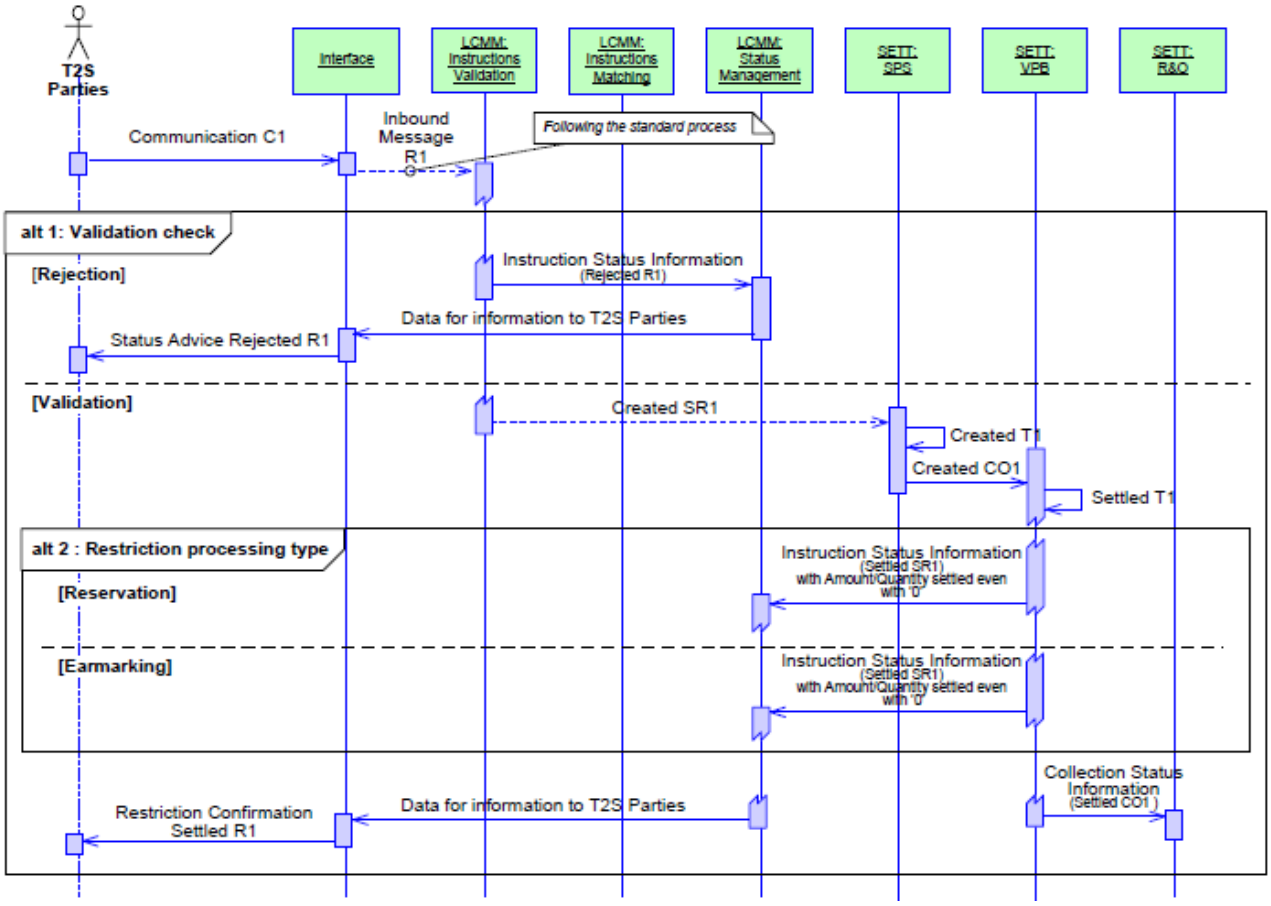
The T2S Party is informed (depending on their message subscriptions) with a message which includes:

- The settled quantity filled with quantity actually earmarked (including 0);
- The remaining quantity to be settled equals to 0.

After this settlement attempt, if SR1 has not been fully settled, the unsettled part is not attempted again (i.e. no recycling).

b) Decrease of a reservation or an earmarking position

DIAGRAM 34 - DECREASE OF A RESERVATION OR AN EARMARKING POSITION



Business assumption

The description relates to the processing of a decrease of a securities reservation or earmarking position from the entry of the communication C1 to the reporting of the settlement result to the T2S party.

The processing of a decrease of a cash reservation balances follows the same principle.

Processing

All rights reserved.

Firstly, the *Instruction Validation* module performs the validation of the received Communication C1, including the checking of the restriction reference when filled.

Alternative1 – Validation check: Rejection

If the Communication C1 is rejected (including if the indicating Restriction Reference is unknown), the *Instruction Validation* module informs the *Status Management* module by sending it an *Instruction Status Information.* The T2S Party is then informed (depending on their message subscriptions).

Alternative1 – Validation check: Validation

If the Communication C1 is validated, the *Instruction Validation* module creates the *Settlement Restriction* SR1 (no matching required).

On the intended settlement date, the *Instruction Validation* module sends *Settlement Restriction* SR1 to the *Standardisation and Preparation to Settlement* module for further processing.

When SR1 is received, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 from the original *Settlement Restriction* with notably:

- Deliver securities position identification;
- Receiver securities position identification;
- Settlement Quantity of the original settlement restriction;
- Restriction reference (Mandatory for reserved restriction type, never filled for earmarked one) provided by T2S party.

A collection CO1 containing the *Settlement Transaction* T1 is created and sent to the *Daytime Validation, Provisioning and Booking* module.

Before the booking, the restriction reference (when filled) is checked:

- If the restriction reference is consistent with the deliver securities position of T1, T1 is valid for further processing;
- If not, T1 is rejected and the collection fails.

Next, the function checks the available quantity in the reserved or earmarked *securities position.*

Then the booking process depends on the Settlement Transaction Category.

Alternative2 - Restriction processing type: reservation

Accordingly, the *Booking* function:

- Settles T1 with the available quantity moved from the deliver *securities position* to the receiver *securities position* [189];
- Updates the *Detailed Securities Restriction* identified by the restriction reference with the available quantity;
- Updates each securities position with the quantity actually booked;

189 In case of reservation partially filled, the *Restriction Reference* can still be active (since the To be Filled Quantity/ To be Filled Amount is not empty) but with a Remaining Quantity/ Remaining Amount equal to zero (i.e. the T2S Party has already use all the previously restricted securities or cash). In this case, T1 is settled for zero.

All rights reserved.

- Two securities postings are created related to the movements on each securities position;
- Sets the original SR1 Settlement status value to *"Settled"*.

The *Daytime Validation, Provisioning and Booking m*odule informs *Status Management* module with the settlement result by sending it an *Instruction Status Information.*

The T2S Party is informed (depending on their message subscriptions) with a message which includes:

- The settled quantity filled with quantity actually decreased from reserved position;
- The remaining quantity equals to 0.

Alternative 1 - Restriction processing type: earmarking

Accordingly, the *Booking* function:

- Settles T1 with the available earmarked quantity of securities moved from the deliver *securities position* to the receiver *securities position;*
- Updates each *securities position* with the quantity actually booked;
- Two securities postings are created related to the movements on each *securities position;*
- Sets the original SR1 Settlement Status value to *"Settled".*

The *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending it an *Instruction Status Information.*

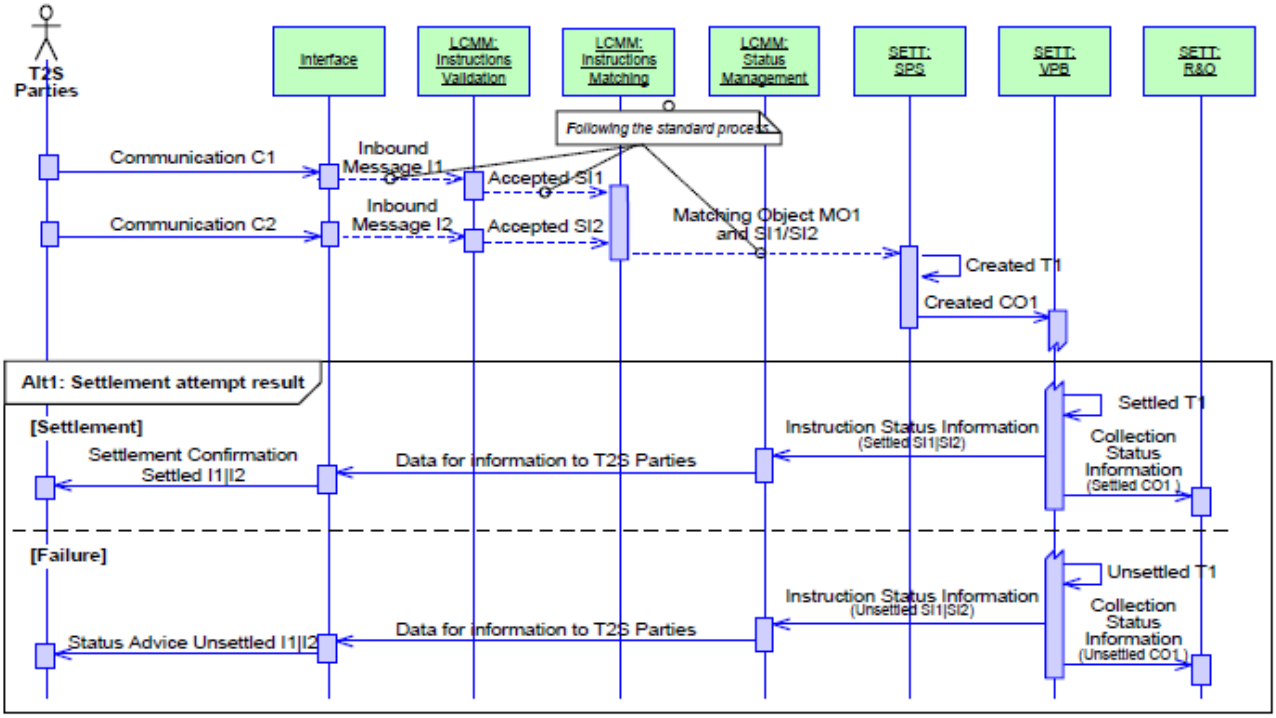
The T2S Party is informed (depending on their message subscriptions) with a message which includes:

- The settled quantity filled with quantity actually decreased from earmarked position;
- The remaining quantity equals to 0.

c) Increase/decrease of an earmarked securities position with a settlement instruction

All rights reserved.

DIAGRAM 35 - INCREASE/DECREASE OF EARMARKED SECURITIES POSITION WITH A SETTLEMENT INSTRUCTION



Business assumption

Upon their entry in the *Interface* domain, the communications C1 and C2 follow the standard process taking into account the restriction type specified to receive/deliver securities to/from a specific earmarked position.

Processing

After the successful validation, the *Instruction Validation* module creates the *Settlement Instructions* SI1 and SI2.

The *Instruction Matching* module performs the matching, creates the related *Matching Object* MO1 and sends the *Settlement Instructions* SI1 and SI2 and their related *Matching Object* MO1 to the *Standardisation and Preparation to Settlement* module for further processing.

On the intended settlement date, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1, to increase (or set up) or to decrease an earmarked securities position.

During the daytime window, a collection CO1 containing the *Settlement Transaction* T1 is created and sent to the *Daytime Validation, Provisioning and Booking* module.

The provision-checking is carried out to check if the available quantity in the deliverable or earmarked securities position allows the settlement of T1.

The settlement attempt result depends on the results of this provision-checking.

Alternative 1 - Settlement attempt result: Settlement

If the provision-checking is satisfactory:

- In case of increase (or set up), T1 is settled with the quantity of securities moved from the deliverable securities position to the earmarked securities position:

All rights reserved.

- If the earmarked securities position does not exist, it is created with the quantity received,
- If the earmarked securities position exists, it is increased with the quantity received;

- In case of decrease, T1 is settled with the quantity of securities moved from the earmarked securities position to a deliverable securities position;
- Each securities position and securities position current is created/updated with the quantity actually booked;
- Two securities postings are created related to the movements on each securities position;
- The original SI1 and SI2 *Settlement Instruction* Settlement Status value are set to *"Settled".*

The *Daytime Validation, Provisioning and Booking* module informs *Status Management* module with the settlement result by sending it an *Instruction Status Information.*

The T2S Party is informed (depending on their message subscriptions) with a message.

Alternative 1- Settlement attempt result: Failure

If the provision-checking is not satisfactory:

- T1 is unsettled;
- The original SI1 and SI2 *Settlement Instruction* Settlement Status value are set to *"Unsettled";*
- The collection containing unsettled T1 is sent to the *Daytime Recycling and Optimisation* module for optimisation.

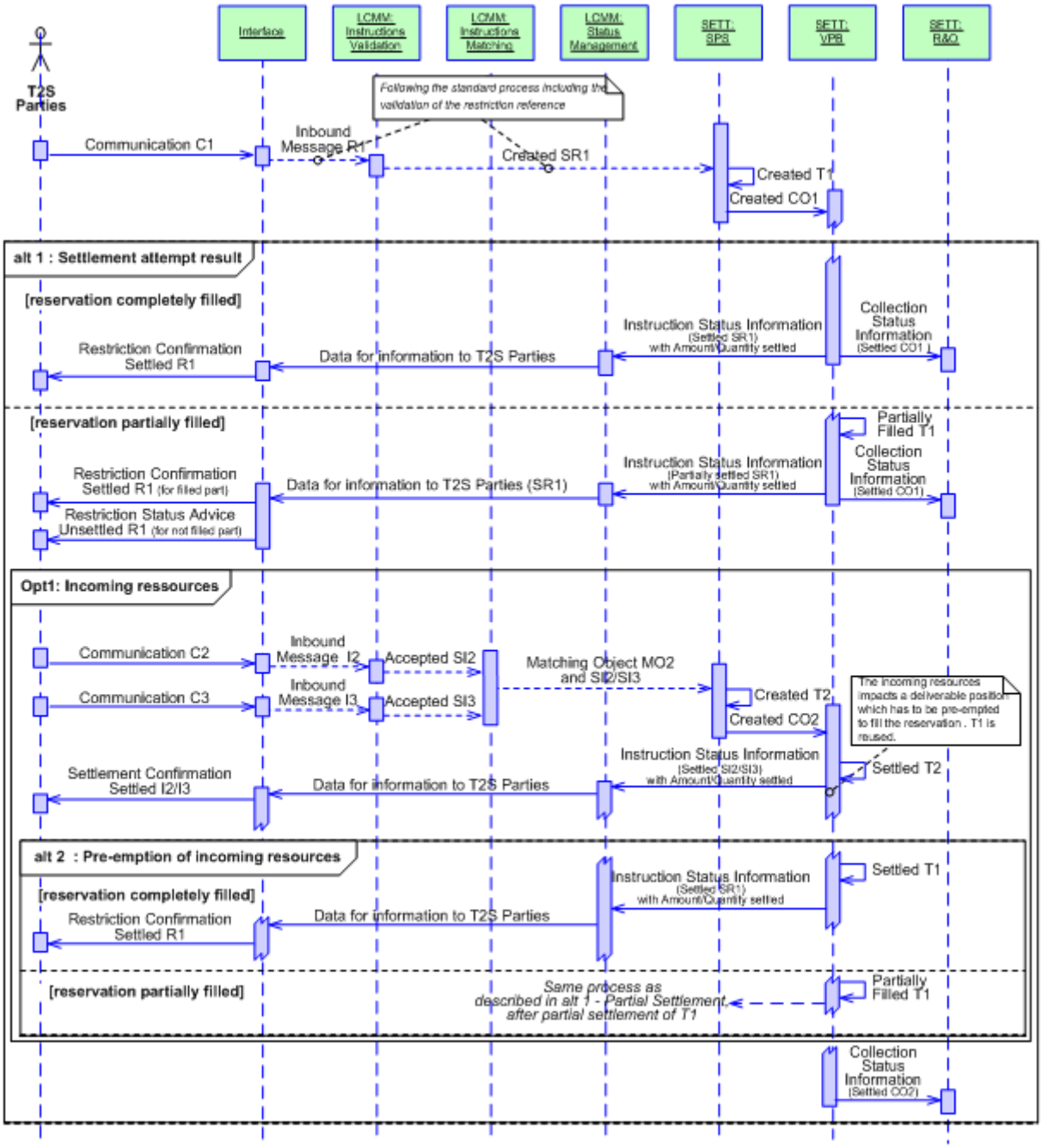
The *Daytime Validation, Provisioning and Booking* module informs *Status Management* module with the settlement result by sending it an *Instruction Status Information.*

The T2S Party is informed (depending on their message subscriptions) with a message.

All rights reserved.

3.5.10.9 Processing of UC-SI-8: Reservation (with additional pre-emption)

DIAGRAM 36 - SET-UP OR INCREASE OF A RESERVATION WITH ADDITIONAL PRE-EMPTION



Business assumption

This description relates to the processing of a reservation restriction from the entry of the communication C1 (determined by the *Instruction Validation* module) to the reporting of the settlement result to the T2S party.

The processing of a set up/increase of a cash restriction follows the same principle.

Processing

All rights reserved.

After a successful validation, including the checking of the restriction reference when filled, the *Instruction Validation* module creates the *Settlement Restriction* SR1 related to this message (no matching required).

On intended settlement date, the *Instruction Validation* module sends the *Settlement Restriction* SR1 to the *Standardisation and Preparation to Settlement* module for further processing.

When SR1 is received, the *Standardisation and Preparation to Settlement* module generates a *Settlement Transaction* T1 with notably:

- Deliver securities position identification;
- Receiver securities position identification;
- Settlement Quantity of the original settlement restriction;
- Restriction reference (optional) is equal to the internal restriction reference generated by T2S or not filled.

During the daytime window [190], a collection CO1 containing the *Settlement Transaction* T1 is created and sent to the *Daytime Validation, Provisioning and Booking* module.

The provision-checking is carried out to determine the available quantity in the deliverable (or earmarked) securities position allowing the settlement of T1 and to check the restriction reference if it is mentioned:

- If the restriction reference is consistent with the receiver securities position of T1, T1 is valid for further processing;
- If not, T1 is rejected and the collection fails.

The settlement attempt result depends on the result of the provision checking.

Alternative 1 - Settlement attempt result of T1 Reservation completely filled

In this case, the *Booking* function performs the following processes:

- T1 is settled with the available quantity of securities moved from the deliverable (or earmarked) securities position to the considered reserved securities position;
  - If no restriction reference is mentioned in the *Settlement Transaction* T1, it is a set up of reservation. The *Detailed Securities Restriction* identified by a restriction reference is created with the available quantity,
  - If a restriction reference is mentioned in the *Settlement Transaction* T1, it is an increase of reservation. The related *Detailed Securities Restriction* is updated with the available quantity;
- Each securities position is created/ updated with the quantity actually settled;
- Two securities posting are created related to the movements on each securities position;
- The original SR1 reservation *Settlement Restriction* Settlement Status value is set to "*Settled*".

The *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending and *Instruction Status Information.*

The T2S Party is informed (depending on their message subscriptions) with a message which includes:

190 During the night-time window, the *Settlement Transactions* related to restrictions are handled in the sequence managed by the *Night-time Settlement* module (Sequence 4).

All rights reserved.

- The restriction reference;
- The settled quantity filled with quantity actually reserved;
- No remaining quantity to be filled.

Alternative 1 - Settlement attempt result: Reservation partially filled

In this case, the *Booking* function performs the following processes:

- T1 is settled with the available quantity of securities moved from the deliverable (or earmarked) securities position to the considered reserved securities position;
  - If no restriction reference is mentioned in the *Settlement Transaction* T1, it is a set up of reservation. The *Detailed Securities Restriction* identified by a restriction reference is created with the available quantity,
  - If a restriction reference is mentioned in the *Settlement Transaction* T1, it is an increase of reservation. The related *Detailed Securities Restriction* is updated with the available quantity,
  - The To be Filled Quantity attribute of the *Detailed Securities Restriction* is updated; Each securities position is created/updated with the quantity actually settled,
  - The Position to Pre-empt attribute of the *Securities Position Current* is updated.
- Two securities posting are created related to the movements on each securities position;
- The original SR1 Settlement Status value is set to "*Partially Settled*".

The *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

The T2S Party is informed (depending on their message subscriptions) with a message for the settled part which includes:

- The restriction reference;
- The settled quantity filled with quantity actually reserved;
- The remaining quantity to be filled.

Option 1: Incoming resources

When a collection satisfying, the provision-checking provides resources on the deliverable (or earmarked) securities position related to a reservation to be filled, T2S prepares the pre-emption to complete the restriction.

Alternative 2 - Pre-emption of incoming resources: Reservation completely filled

In this case, the *Booking* function:

- Retrieves the *Settlement Transaction* T1 partially filled;
- According to the incoming resources, updates T1 with the quantity to be settled;
- Adds it to the collection CO1.

During the booking of the collection CO1, the *Daytime Validation, Provisioning and Booking* module:

All rights reserved.

- Decreases the missing quantity on the relevant restricted securities position (to pre-empt) and associated *Detailed Securities Restriction* (to be filled);
- Sets the original SR1 Settlement Status value to "*Partially Settled*" or "*Settled*".

The *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

The T2S Party is informed (depending on their message subscriptions) with a message for the settled part which includes:

- The settled quantity filled with quantity actually reserved;
- The remaining quantity to be filled.

Alternative 2 - Pre-emption of incoming resources: Reservation partially filled

The alternative 2 is repeated till the full completion of the reservation.

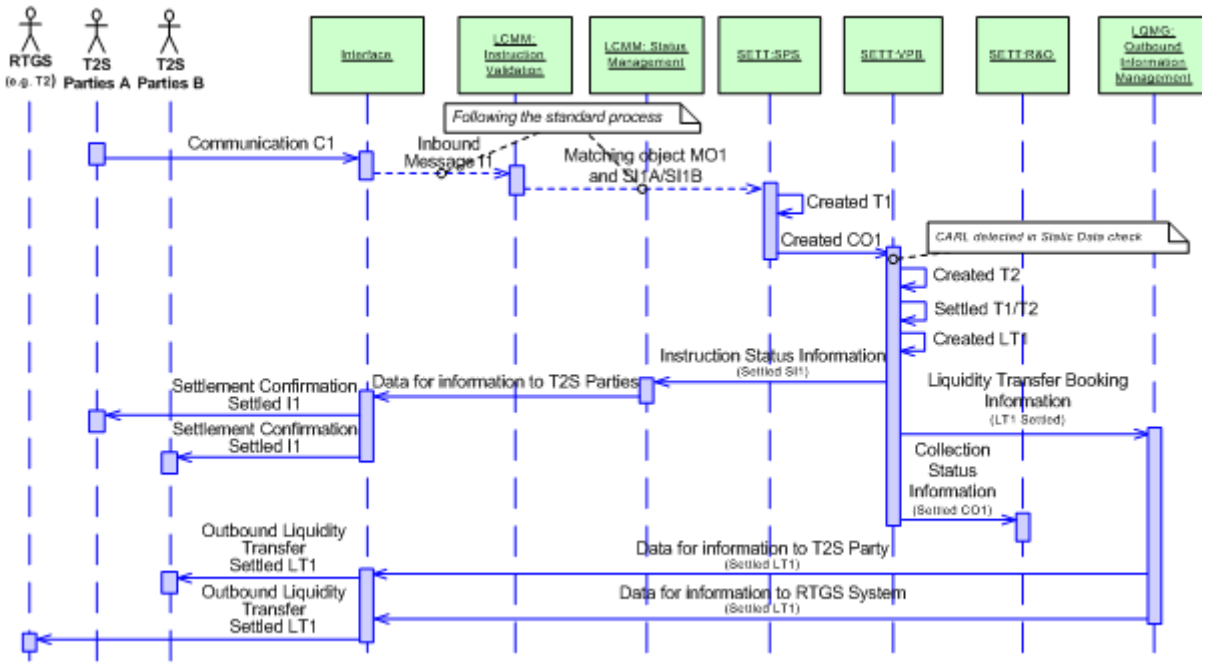
Specificities concerning the decrease of a reservation

A T2S party can cancel a restriction of reservation by sending a specific message.

In that case, if the reservation had not been totally filled, the considered To be Filled Quantity of the related *Detailed Securities Restriction* will be reset in order to avoid any additional pre-emption.

3.5.10.10 Processing of UC-SI-9 Settlement of a corporate action with associated liquidity transfer

Diagram 37 - Settlement of a Corporate Action with associated liquidity transfer



Business assumption

This description relates to the processing of a corporate action related instruction (C1) initiated by T2S Party A (CSD A):

- That results in a credit in the *T2S Dedicated Cash Account* of another T2S Party (T2S Party B);

All rights reserved.

- And is credited afterwards on the RTGS account linked to the *T2S Dedicated Cash Account* (used by T2S Party B) according to an option chosen by T2S Party B defined in reference data (T2S Party can opt for an automatic and immediate retransfer of cash proceeds from any corporate action(s)).

Settlement instructions received for corporate action imply only one debit and one credit of cash (i.e. only two T2S Parties are implied, CSD A and T2S Party B).

This use case is generalised for automated retransfer of cash proceeds from corporate actions.

Processing

T2S Party A (CSD A) initiates a corporate action related instruction.

Once the validation procedures have been successfully passed in the *Instruction Validation* module, the *Settlement Instructions* SI1A and SI1B and their related *Matching Object* MO1 are sent to the *Standardisation and Preparation to Settlement* module which:

- Generates a *Settlement Transaction* T1 for the *Settlement Instruction* SI1A/SI1B for corporate action settlement purpose;
- Creates a collection CO1 of *Settlement Transactions* with T1;
- Sends CO1 to the *Daytime Validation, Provisioning and Booking* module for a real-time settlement attempt.

At receipt of the collection CO1 including the *Settlement Transaction* T1 for which a corporate action purpose is specified, the *Daytime Validation, Provisioning and Booking* module detects in reference data that an option for an automatic and immediate retransfer of cash proceeds from any corporate action is defined for the *T2S Dedicated Cash Account* of Party B.

Then, the *Daytime Validation, Provisioning and Booking* module:

- Generates a *Settlement Transaction* T2 corresponding to the immediate *Liquidity Transfer* of the cash proceeds;
- The transfer amount is filled in with the settlement cash amount in the settlement instructions;
- Includes the new *Settlement Transaction* T2 in the collection CO1.

If the booking can be achieved, the *Daytime Validation, Provisioning and Booking* module creates the immediate *Liquidity Transfer* (LT1) corresponding to the *Settlement Transaction* T2.

Following the settlement attempt in the *Daytime Validation, Provisioning and Booking* module, if T1 and T2 are settled, the following pair of cash related bookings is carried out:

	DEBIT	**CREDIT**
For T1:	T2S Dedicated cash account of deliverer of cash	T2S Dedicated cash account of receiver of cash
For T2:	T2S Dedicated cash account of receiver of cash	RTGS Dedicated transit account in T2S

In addition:

All rights reserved.

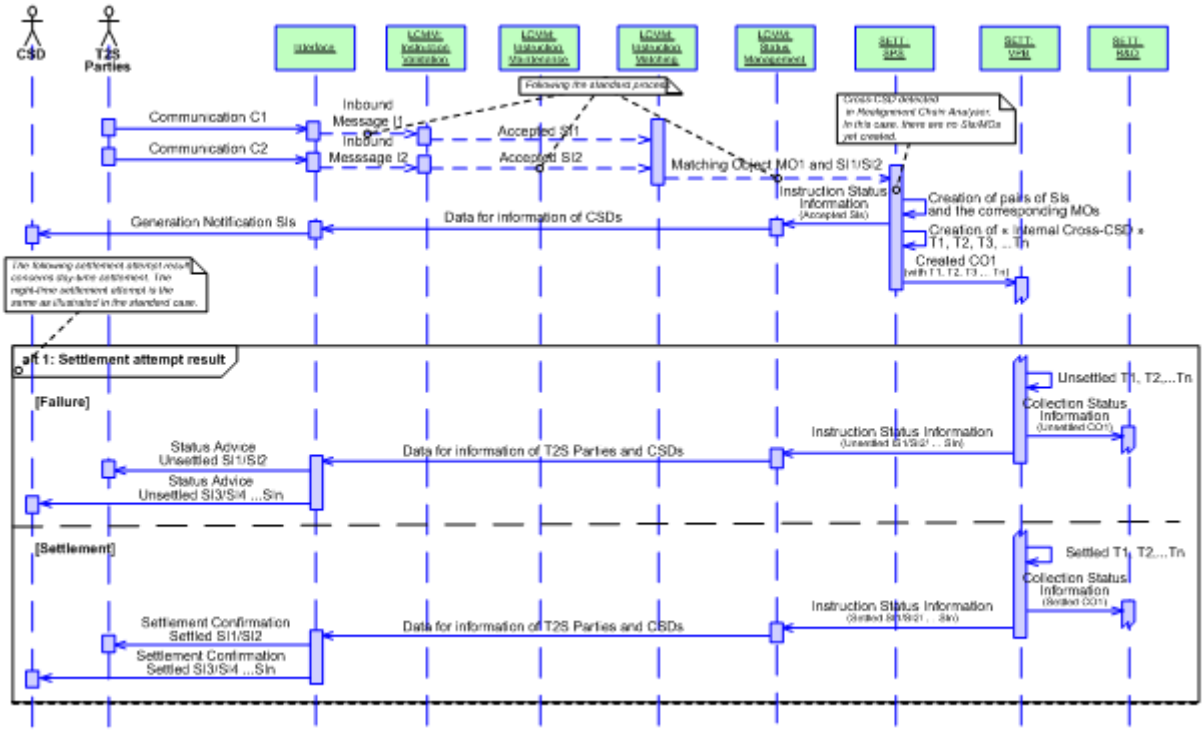
- The Settlement Status value of SI1A and SI1B are set to "*Settled*";
- The *Daytime Validation, Provisioning and Booking* module sends an *Instruction Status Information* to the *Status Management* module informing it of the successful settlement;
- The *Status Management* module sends data for a settlement confirmation message to *Interface* domain to inform T2S Party A (CSD A) and T2S Party B that SI1 is settled;
- The *Daytime Validation, Provisioning and Booking* module sends a *Liquidity Transfer Booking Information* on LT1 to the *Outbound Information Management* module.

This latter module then composes and sends an information message to the T2S Actor via the *Interface* domain which informs T2S Party B that the cash transfer was performed, as well as communicates information to RTGS System message to the RTGS system.

For further details, see UC-LT-2, regarding the management of this communication.

3.5.10.11 Processing of UC-SI-10a: Cross CSD Standard DVP

DIAGRAM 38 - SETTLEMENT PROCESS FOR CROSS CSD STANDARD DVP



Business assumption

Upon their entry in the *Interface* domain, the communications C1 and C2 follow the standard process as described in UC-SI-1 and UC-SI-3 to their entry in the *Standardisation and Preparation to Settlement* module.

Processing

After the successful validation, the *Instruction Validation* module creates the *Settlement Instructions* SI1 and SI2.

All rights reserved.

The *Instruction Matching* module performs the matching, creates the related *Matching Object* MO1 and sends the *Settlement Instructions* SI1 and SI2 and their related *Matching Object* MO1 to the *Standardisation and Preparation to Settlement* module for further processing.

On reception of the *Matching Object* MO1 and the related *Settlement Instructions* SI1/SI2, the *Standardisation and Preparation to Settlement* module identifies the incoming *Settlement Instructions* as Cross-CSD instructions.

In this use case, as all involved CSDs are in T2S, the "internal Cross-CSD" scenario is identified.

The concerned CSDs can be:

- Investor CSDs, i.e. the CSD of each party of the settlement transaction;
- Issuer CSDs, i.e. the CSD in which the security has been issued and distributed on behalf of the issuer.

In a first step, the *Standardisation and Preparation to Settlement* module:

- Creates pairs of T2S generated *Settlement Instructions* (pairs per CSD), as many as necessary, to ensure the realignment between the involved CSDs;
- Sends to the *Status Management* module the data for information of the CSD, corresponding to the new pairs of T2S generated *Settlement Instructions created;*
- Creates one *Matching Object* to each pair of T2S generated *Settlement Instructions* created;
- For each (T2S generated) *Settlement Instructions*, creates the corresponding *Settlement Transaction* with the Settlement Transaction Category set to *"Realignment".*

The *Status Management* module then generates the reporting to inform the involved CSDs (depending on their message subscriptions) about the creation of the new settlement instructions.

In a second step, the *Standardisation and Preparation to Settlement* module puts the *Settlement Transactions* previously created into a collection CO1 and sends it directly to the *Daytime Validation, Provisioning and Booking* module for a settlement attempt. All *Settlement Transactions* previously created must be settled in an "all or none" basis.

Alternative 1 - Settlement attempt result: Failure

If CO1 is unsettled, the *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

The T2S Parties are informed of the unsettled status (depending on their message subscriptions) with a status advice messages for each of the (T2S generated) settlement instructions.

A *Collection Status Information* is sent to the *Daytime Recycling and Optimisation* module to launch a standard optimisation process.

Alternative 1 - Settlement attempt result: Settlement

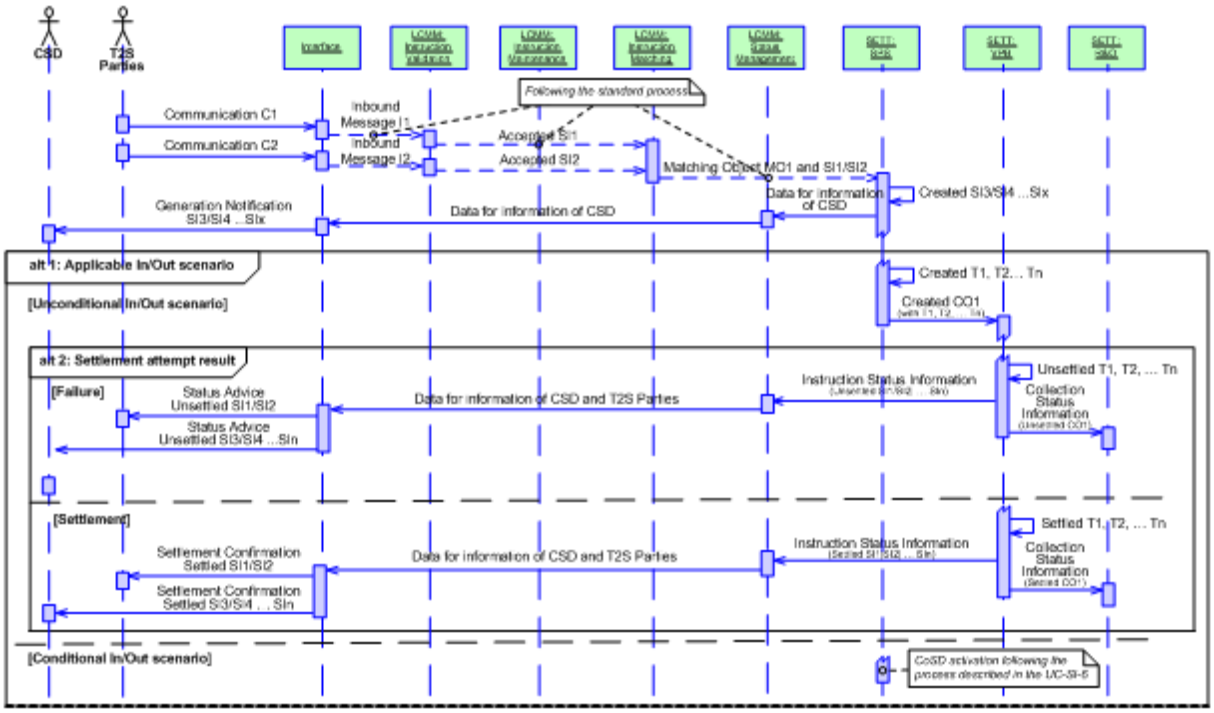
If CO1 is settled, the *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

The T2S Parties are informed of the settled status (depending on their message subscriptions) with a settlement confirmation messages for each of the (T2S generated) settlement instructions.

All rights reserved.

3.5.10.12 Processing of UC-SI-10b: In-out T2S Standard DVP

DIAGRAM 39 - SETTLEMENT PROCESS FOR IN-OUT T2S STANDARD DVP



Business assumption

Upon their entry in the *Interface* domain, the communications C1 and C2 follow the standard process as described in UC-SI-1 and UC-SI-3 to their entry in the *Standardisation and Preparation to Settlement* module.

Processing

After the successful validation, the *Instruction Validation* module creates the *Settlement Instructions* SI1 and SI2.

The *Instruction Matching* module performs the matching, creates the related *Matching Object* MO1 and sends the *Settlement Instructions* SI1 and SI2 and their related *Matching Object* MO1 to the *Standardisation and Preparation to Settlement* module for further processing.

On reception of the *Matching Object* MO1 and the related *Settlement Instructions* SI1/SI2, the *Standardisation and Preparation to Settlement* module identifies the incoming instructions as "In/Out" instructions.

In a first step, the *Standardisation and Preparation to Settlement* module:

- Creates pairs of T2S generated *Settlement Instructions* (pairs per CSD), as many as necessary, to ensure the realignment between the involved CSDs;
- Sends to the *Status Management* module the data for information of CSD, corresponding to the new T2S generated *Settlement Instructions* created;
- Creates one *Matching Object* to each pair of T2S generated *Settlement Instructions* created.

The *Status Management* module then generates the reporting to inform the involved CSDs (depending on their message subscriptions) about the creation of the new T2S generated settlement instructions.

All rights reserved.

In a second step, at the intended settlement date, the *Standardisation and Preparation to Settlement* module identifies if it is an "unconditional In/Out" or a "conditional In/Out" settlement scenario using CoSD rules defined in the reference data:

- For an "unconditional In/Out" settlement scenario, the (technical) Issuer CSD is inside T2S and at least one Investor CSD is outside. No actions outside T2S are required prior to settlement;
- For a "conditional In/Out" settlement scenario, the (technical) Issuer CSD is outside T2S, and one Investor CSD is inside (and the other Investor CSD is outside). The final securities delivery within T2S is dependent on actions outside T2S.

Alternative 1 – Applicable In/Out scenario: unconditional In/Out scenario

For an "unconditional In/Out" scenario, no additional action outside T2S is necessary prior to the settlement. The *Standardisation and Preparation to Settlement* module creates, for each (T2S generated) *Settlement Instructions,* the corresponding settlement transactions.

Those *Settlement Transactions* are then sent to the *Daytime Validation, Provisioning and Booking* module through the collection CO1 for a settlement attempt in an "all or none" basis.

Alternative 2 - Settlement attempt result: Failure

If the settlement attempt is failed, the *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

The T2S Parties are informed of the unsettled status (depending on their message subscriptions) with a status advice message for each of the (T2S generated) settlement instructions.

A *Collection Status Information* is sent to the *Daytime Recycling and Optimisation* module to launch a standard optimisation process.

Alternative 2 - Settlement attempt result: Settlement

If the settlement attempt is successful, the *Daytime Validation, Provisioning and Booking* module informs the *Status Management* module with the settlement result by sending an *Instruction Status Information.*

The T2S Parties are informed of the settled status (depending on their message subscriptions) with a settlement confirmation message for each of the (T2S generated) settlement instructions.

A *Collection Status Information* is sent to the *Daytime Recycling and Optimisation* module to launch a standard optimisation process.

Alternative 1 – Applicable In/Out scenario: conditional In/Out scenario

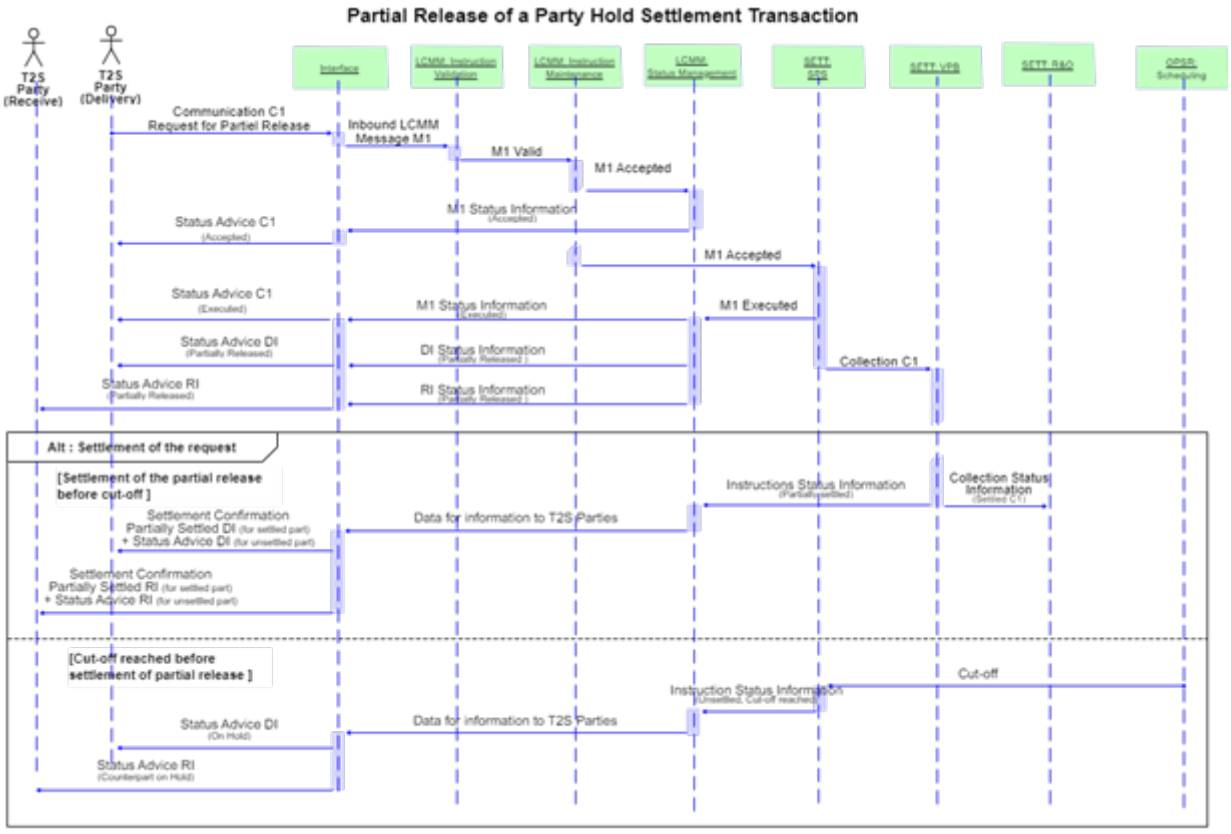
For a "conditional In/Out" scenario incoming *Settlement Instruction,* additional actions outside T2S are necessary prior to the settlement in T2S. The *Settlement Transaction* has to be created "on hold" and the resources necessary for this settlement blocked through a CoSD blocking, following the process described in the "CoSD Activation" section of the UC-SI-6.

Once the actions outsides T2S performed, the release of the *Settlement Instructions* will then be handled after the administering parties release following the process described in the "CoSD Release" section of the UC-SI-6.

All rights reserved.

3.5.10.13 Processing of UC-SI-10c: Partial Release

DIAGRAM 40 - SETTLEMENT PROCESS PARTIAL RELEASE{CR-827}



Business assumption

This description relates to a Communication C1 which contains a Partial Release Instruction MI1 communicated by aT2S Party (Delivery), requesting to release partially of a previously sent Settlement Instruction DI (Delivery Intra-CSD instruction). DI was matched with Settlement Instruction RI (Receiving Intra-CSD instruction) sent by T2S Party (Receive) to produce the Settlement Transaction ST1 "On Hold".

Processing

Once the format and syntax validations for Communication C1 are successfully passed, the Inbound LCMM Message M1 enters successively the Instruction Validation module and the Instruction Maintenance module which create the Partial Release Instruction and perform the following checks in order to identify if it can be processed:

- Underlying DI exists;
- Underlying DI is not awaiting approval / revoked;
- Underlying DI is not identified with CoSD Flag;
- Underlying DI is on Party Hold;
- Underlying DI is not an internally generated instruction;
- No other hold applies neither on DI nor on RI;

All rights reserved.

- Quantity to be released complies with SUM check;
- DI is delivering securities, matched and its ISD has been reached.

When these checks are successful, (i) the Status Management module is informed on the acceptance of MI1 to be reported to the T2S Party (Delivery), via the Interface domain (depending on the T2S Party's message subscription preferences) and (ii) the Partial Release Instruction is forwarded to the SPS module of Settlement domain, which performs the following checks in order to identify if it can be executed:

- Quantity to be released complies with partial settlement rules regarding MSU & SUM;
- DI is neither settled nor cancelled;
- DI is not identified with CoSD Flag;
- DI is on Party Hold;
- No other hold applies neither on DI nor on RI;
- DI and RI allow partial settlement;
- Neither DI nor RI is involved in any pool or any business link;
- Quantity must be lower than the original quantity of DI;
- ST resulting from matching of DI and RI has not reached its cut-off time

When these checks are successful, the Status Management module is informed on the execution of the Partial Release Instruction to be reported to the T2S Party (Delivery), via the Interface domain (depending on the T2S Party's message subscription preferences) and a collection C1 is created containing the settlement transaction

Then~~, if a Partial Settlement Window is open~~**{CR-827}**, a collection C1 containing ST1 to be settled up to the Quantity required by the Partial Release Instruction is sent to VPB module for settlement attempt.

Alternative - Settlement of the request: Settlement of the partially released Settlement Transaction before cut-off

If CO1 is settled, the Daytime Validation, Provisioning and Booking module informs the Status Management module with the settlement result by sending an Instruction Status Information.

T2S Parties (Delivery) and (Receive) are informed of the partially settled status (depending on their message subscriptions) with messages for respectively DI and RI: a settlement confirmation for the settled part and a status information for the unsettled part.

Alternative - Settlement of the request: Cut-off reached before settlement of partial release

If CO1 is unsettled when the cut-off applying to ST1 is reached, the Standardisation and Preparation to Settlement module informs the Status Management module by sending an Instruction Status Information.

The Partial Release Instruction is cancelled. DI and RI and are put back on "Party Hold" and submitted to an eligibility check.

T2S Parties (Delivery) and (Receive) are informed of the "Hold" status (depending on their message subscriptions) with a status advice messages for respectively DI and RI.

All rights reserved.

3.6 Liquidity Management

3.6.1 General Introduction

The *Liquidity Management* domain manages the transfer of liquidity:

- From *RTGS Accounts* to *T2S Dedicated Cash Accounts (Inbound Liquidity Transfers*);
- Between *T2S Dedicated Cash Accounts* (Internal *Liquidity Transfers*);
- From *T2S Dedicated Cash Accounts* to *External RTGS Accounts* (Outbound *Liquidity Transfers*);

T2S Actors have the possibility:

- To submit Internal and Outbound *Liquidity Transfers* for immediate execution in T2S and *Inbound Liquidity Transfers* for immediate execution via RTGS system to T2S [191] under certain attribute requirements **{T2S.06.231};**
- To configure *Liquidity Transfer Orders* to be triggered at a given point in time or on a given business event. Such *Liquidity Transfer Orders* can be configured to be executed only once (via Predefined *Liquidity Transfer Orders*) or at every occurrence of its triggering condition being satisfied (via Standing *Liquidity Transfer Orders*).

Standing and Predefined *Liquidity Transfer Orders* are maintained in CRDM. When a Standing or Predefined *Liquidity Transfer Order* is triggered by time or business events, an Immediate Outbound *Liquidity Transfer* is created and processed **{T2S.06.206}.**

Liquidity Management performs all the preparatory work for the booking of Immediate, Predefined and Standing *Liquidity Transfers* on the *T2S Dedicated Cash Accounts* **{T2S.06.200}.**

Furthermore, for Outbound *Liquidity Transfers, Liquidity Management* triggers the related communication between T2S and the involved RTGS system (e.g. TARGET2).

For liquidity management purposes, ISO-20022 compliant messages provided in the cash management standard (e.g. liquidity credit transfer) are used **{T2S.06.195}.**

The *Liquidity Management* domain comprises three modules:

- The *Liquidity Operations* module is in charge of formatting, validating, processing, generating and storing the data related to *Liquidity Transfers.* It is also responsible for preparing the information needed for booking of a *Liquidity Transfer;*
- The *Outbound Information Management* module is in charge of status management, forwarding the *Outbound Liquidity Transfers* and the management of the data exchanged with T2S Actors and RTGS systems during the processing of *Liquidity Transfers;*
- The *CB Business Procedures* module is in charge of End-of-Day procedures related to the management of *Liquidity Transfers.*

The *Liquidity Operations* module:

191 Non –T2S Actors can also initiate inbound liquidity transfers.

All rights reserved.

- Validates *Liquidity Transfers* (inbound, outbound and internal), and checks the permission of the sender (for Inbound, Outbound and Internal *Liquidity Transfers*);
- Creates *Liquidity Transfers* at the triggering of Standing and Predefined *Liquidity Transfer Orders;*
- Manages "multiple liquidity providers" functionality for the reimbursement of liquidity from one *T2S Dedicated Cash Account* to several *RTGS accounts* of liquidity providers at the end of night time settlement process. The necessary configuration is maintained in static data **{T2S.06.063} {T2S.06.067}.**

The *Outbound Information Management* module:

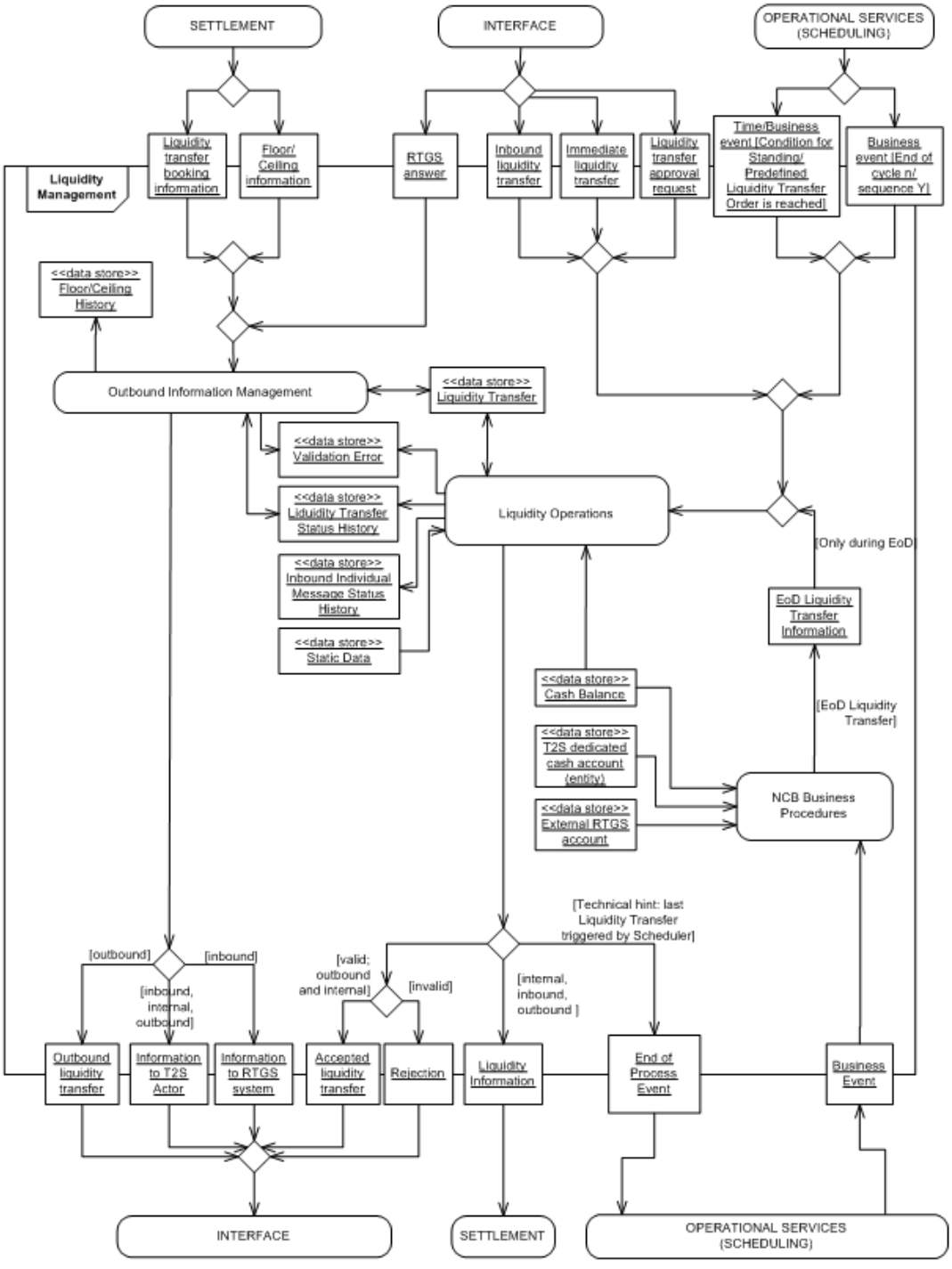
- Stores the Settlement information and the RTGS status of the processed *Liquidity Transfers,*
- Collects information on changes of the Settlement status and RTGS status of all *Liquidity Transfers* as well as floor/ceiling violation information on *T2S Dedicated Cash Accounts* to provide them to the T2S Actors;
- Forwards the Outbound *Liquidity Transfers.*

The *CB Business Procedures* module is in charge of following End-of-Day treatments. This module:

- Manages the transfer of additional liquidity from *RTGS accounts* in case of T2S Central bank Accounts;
- Prepares the cash sweep for transfer of remaining liquidity available on *T2S Dedicated Cash Accounts* to the relevant *RTGS accounts.*

All rights reserved.

Starting with the go live of CLM T2S will not initiate an "automated cash sweep" for ~~**Euro denominated**~~ Dedicated Cash Accounts **for all currencies connected to CLM**.**{CR-720}** T2S Users have the possibility to make use of a second optional cash sweep for their Euro denominated Dedicated Cash Accounts. [192]



Definition of a Liquidity Transfer

192 ~~**Please note that the described functionality won´t be available before CSLD go-live.**~~ **{CR-720}**

All rights reserved.

The term *"Liquidity Transfer"* covers all liquidity adjustments on *T2S Dedicated Cash Accounts* allowed according to the conditions listed below:

- Credited and debited *T2S Dedicated Cash Accounts* must be denominated in the same currency **{T2S.06.060};**
- *Liquidity Transfers* are possible from an RTGS account to a *T2S Dedicated Cash Account* (via *Inbound Liquidity Transfer*) and vice versa (via Outbound *Liquidity Transfer*) provided this is allowed by the relevant CB(s) **{T2S.06.060} {T2S.06.113} {T2S.06.150} {T2S.06.240} {T2S.06.390} {T2S.06.410};**
- In case the *Liquidity Transfer* is sent from T2S to the RTGS system (Outbound *Liquidity Transfer*) it is necessary that the receiving RTGS account is set up in CRDM;
- *Liquidity Transfers* are possible between two *T2S Dedicated Cash Accounts* (Internal Liquidity Transfer) linked to the same *RTGS account* or between two *T2S Dedicated Cash Accounts* belonging to the same Payment Bank (i.e. proprietary and clients' accounts) or if one or both of the involved accounts is a T2S Central Bank account. This rule is not applicable if one of the involved accounts is a Dedicated Transit Account i.e. if one of the involved accounts is a Dedicated Transit Account Internal Liquidity Transfers are always possible **{T2S.06.114} {T2S.06.241};**
- In T2S it is not foreseen to settle clean payments. Clean payments are cash transfers that are neither the counterpart of a securities transfer nor the proceeds of a corporate action **{T2S.06.130}.** As a consequence it is not possible to transfer liquidity between two *T2S Dedicated Cash Accounts* not belonging to the same Payment/Settlement Bank or to another T2S party for whom the Payment Bank acts as liquidity provider. In such a case the involved T2S Actors need to perform the transfer via the RTGS accounts of the respective account holders;
- Inbound and Outbound *Liquidity Transfers* rely on the use of RTGS Dedicated Transit Accounts [193].

Multi currency aspects of Liquidity Transfers

Since T2S is technically capable of providing cash settlement in Euro or non-Euro central bank money, *Liquidity Transfers* are possible in all T2S settlement currencies **{T2S.02.050}.**

A central bank must open an RTGS Dedicated Transit Account in T2S for each T2S settlement currency the respective CB is offering settlement for that currency, but in case of several CBs belonging to one RTGS system only one RTGS Dedicated Transit Account is opened. These accounts are managed in CRDM.

The respective currency is specified within the *Liquidity Transfer.* With this information *Liquidity Management* domain is able to assign the respective RTGS Dedicated Transit Account **{T2S.06.520}.**

Initiator of Liquidity transfers in T2S

Liquidity Transfers can be initiated by **{T2S.06.205} {T2S.06.240}:**

- The account holder (i.e. Payment/Settlement Bank) of the *T2S Dedicated Cash Account* that is debited;
- Another T2S party (e.g CSD) the account holder of the *T2S Dedicated Cash Account* that is debited has authorised to act on behalf of the account holder.

193 After a booking has taken place in one system on one of the dedicated transit accounts in this system, an outbound liquidity transfer is sent from this system to the other, which receives it as an inbound liquidity transfer

All rights reserved.

Four-Eyes principle

Immediate *Liquidity Transfers* can be initiated in T2S in U2A mode, using the Four-Eyes principle **{T2S.16.170} {T2S.16.230}.**

When an *Immediate/ Inbound Liquidity Transfer* sent under the Four-Eyes principle is waiting for the approval request from the Second T2S System User, the Approval Status value of the initial *Liquidity Transfer* is set to *"Awaiting for approval".*

The reception of the *Liquidity Transfer Approval Request* (either approval or revocation) from the Second T2S System User leads to a preliminary check of the Approval Request verifying the proper mapping between this Approval Request and the initial *Liquidity Transfer,* i.e. checking that the initial *Liquidity Transfer* exists in T2S and that its Approval Status value is *"Awaiting for Approval".*

- In case the preliminary check of the *Liquidity Transfer Approval Request* above is negative, then it leads to the rejection of this *Liquidity Transfer Approval Request ,* but not of the initial *Liquidity Transfer* awaiting for approval;
- If on the contrary, the preliminary check of the *Liquidity Transfer Approval Request* is positive, then:
  - If this Approval Request is a revocation request, the Approval Status of the initial *Liquidity Transfer* is set to *"Revoked";*
  - If this Approval Request is confirms the initial *Liquidity Transfer,* then the latter is submitted to a revalidation. Depending on the result of this revalidation, the Approval Status value of the initial *Liquidity Transfer* is set to:
    - *"Approved"* if the initial *Liquidity Transfer* is successfully revalidated;
    - *"Rejected"* if the initial *Liquidity Transfer* is not successfully revalidated.

Predefined and Standing *Liquidity Transfer Orders* are maintained in the reference data. After triggering a Predefined/ Standing *Liquidity Transfer Order* by a time or business event the Four-Eyes Principle is no longer taken into account.

The Four-Eyes Principle for initiating *Liquidity Transfers* is not foreseen in A2A mode.

Underlying transfer types in T2S

The following Underlying Transfer Types are defined **{T2S.06.270} {T2S.06.271} {T2S.06.290} {T2S.06.330} {T2S.06.410} {T2S.06.630} {T2S.06.390}:**

	CREATION "ON BEHALF"	FROM T2S TO RTGS (OUTBOUND)	FROM T2S TO T2S (INTERNAL)	EXECUTION	PARTIAL EXECUTION	FREQUENCY OF EXECUTION
IMMEDIATE LIQUIDITY TRANSFER	No	Yes	Yes	Immediate	No	once
	Yes	Yes	Yes	Immediate	Yes	once

All rights reserved.

	CREATION "ON BEHALF"	FROM T2S TO RTGS (OUTBOUND)	FROM T2S TO T2S (INTERNAL)	EXECUTION	PARTIAL EXECUTION	FREQUENCY OF EXECUTION
PREDEFINED LIQUIDITY TRANSFER ORDERS	Yes	Yes	No	Triggered by Date/Time Event, then immediate	Yes (after it is generated as an Immediate Liquidity Transfer)	once
STANDING LIQUIDITY TRANSFER ORDERS	Yes	Yes	No	Triggered by Date/Time Event, then immediate	Yes (after it is generated as an Immediate Liquidity Transfer)	on a regular basis

In the above mentioned table, *Liquidity Transfers* from RTGS to T2S are not included because the initiating step and the kind of *Liquidity Transfers* within the RTGS systems depend on the respective RTGS system. An *Inbound Liquidity Transfer* is always handled within T2S like an Immediate *Liquidity Transfer.*

Predefined and Standing *Liquidity Transfer Orders* are defined, changed and removed (amount changed to zero or Liquidity Transfer Order deleted) via CRDM**{T2S.06.231}.** At a predefined point in time or after a chosen business event, *Liquidity Management* interacts with CRDM order to receive all information needed for *Liquidity Transfer Orders.*

The following table shows which combination of underlying transfer type and transfer type are possible to be processed in T2S **{T2S.06.521}:**

	INBOUND LIQUIDITY TRANSFER	OUTBOUNG LIQUIDITY TRANSFER	INTERNAL LIQUIDITY TRANSFER
IMMEDIATE LIQUIDITY TRANSFER	Yes	Yes	Yes
PREDEFINED LIQUIDITY TRANSFER ORDERS	No	Yes	No
STANDING LIQUIDITY TRANSFER ORDERS	No	Yes	No

Real-Time processing of Liquidity Transfers

Liquidity Transfers in T2S are executed in real-time (except during the night-time settlement when a settlement cycle is running) **{T2S.06.230}** using the first-in-first-out principle **{T2S.06.620}.**

Information to T2S Actors

For each *Liquidity Transfer* the instructing T2S Actor receives information about the settlement result (*"Settled", "Partially settled"* or *"Unsettled"*) **{T2S.06.630}.** In case of standing and predefined Liquidity Transfer Orders, there is no separate notification about settlement status (only the credit/debit notification). Furthermore in case a Liquidity Transfer Order is unsettled the initiator is not informed.

All rights reserved.

In case T2S receives a "Rejection" or a duplicate/ unexpected [194] RTGS answer from the RTGS system an information message is sent to the respective CB.

If T2S receives a confirmation/rejection from an RTGS system [195] which cannot be reconciled with *a Liquidity Transfer,* an information message is sent to the respective CB.

Predefined Liquidity Transfer Orders

The T2S Actor who sets up the Predefined *Liquidity Transfer Order* in the reference data can define a time or select a business event in T2S at which the Predefined *Liquidity Transfer Order* should be executed **{T2S.06.280}.** The amount to be transferred can be either a specified amount or the whole available on the *T2S Dedicated Cash Account* at the moment the *Liquidity Transfer* takes place**. {T2S.16.660}.** It is not possible to establish a Predefined *Liquidity Transfer Order* to transfer liquidity between *T2S Dedicated Cash Accounts.*

A Payment/Settlement Bank (or CSD on behalf) can put in place (at maximum) one predefined *Liquidity Transfer Order* per *T2S Dedicated Cash Account* to be executed at a given time or on the occurrence of a business event. It is possible however, to define different Predefined *Liquidity Transfer Orders* at different points in time/for business events during the settlement day of T2S **{T2S.06.640}.**

In case of insufficient liquidity available on the *T2S Dedicated Cash Account* to be debited, the Predefined *Liquidity Transfer Order* (which is generated as an Immediate *Liquidity Transfer*) is partially executed, i.e. the whole available amount on the *T2S Dedicated Cash Account* is transferred. The Payment Bank/ Settlement Bank/ T2S Actors are informed accordingly about the Settlement Status of the *Liquidity Transfer.* In case of standing and predefined Liquidity Transfer Orders, there is no separate notification about settlement status (only the credit/debit notification). Furthermore in case a Liquidity Transfer Order is unsettled the initiator is not informed. The amount of liquidity not transferred is not stored, i.e. it is not transferred later, even if additional liquidity is credited on the *T2S Dedicated Cash Account* that was initially debited **{T2S.06.650}.**

Standing Liquidity Transfer Orders

Standing *Liquidity Transfer Orders* can be defined to transfer any specified amount or the whole amount **{T2S.16.660}** from a *T2S Dedicated Cash Account* to an *RTGS account.* This definition of a Standing *Liquidity Transfer Order* results in the triggering of a *Liquidity Transfer* on a repetitive basis at the specified point in time or upon the occurrence of the T2S business event during each settlement day in T2S **{T2S.06.410} {T2S.06.331}.**

It is not possible to establish a Standing *Liquidity Transfer Order* to transfer liquidity between *T2S Dedicated Cash Accounts.*

It is possible to set up more than one Standing *Liquidity Transfer Order* per *T2S Dedicated Cash Account* to be executed at a certain time or upon the occurrence of a given T2S Business Event **{T2S.06.340} {T2S.06.360} {T2S.06.660}.** The Immediate *Liquidity Transfers* stemming from the triggering of those Standing *Liquidity Transfer Orders* are executed in a random order [196].

---

194 An "Unexpected RTGS answer" is an RTGS answer ("Confirmation" or "Rejection"), which does not correspond to a Liquidity Transfer

195 In case T2S does not receive an answer from an RTGS system for an Outbound Liquidity Transfer, then T2S monitoring procedures shall apply

196 The order of the execution is technically influenced by the order of extracting from the data base all the standing liquidity transfer orders which should be executed at the same time.

All rights reserved.

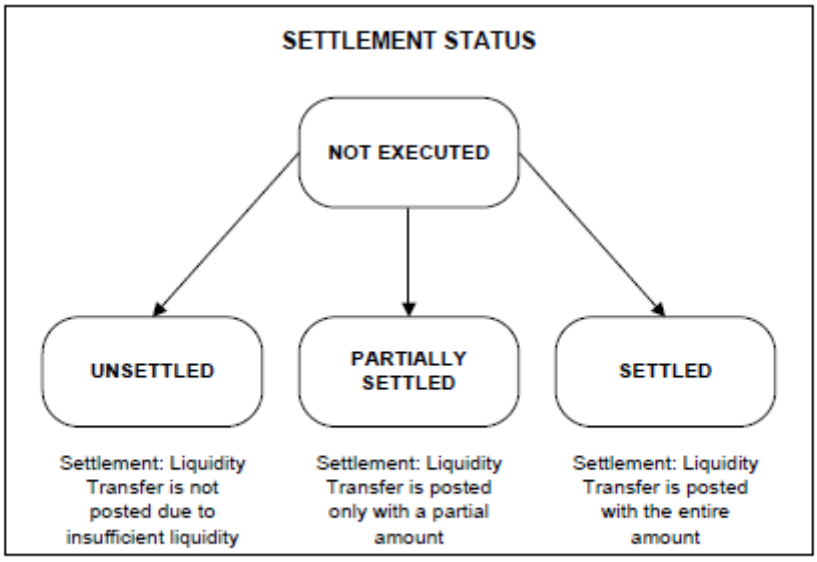
In case of insufficient liquidity available on the *T2S Dedicated Cash Account* to be debited, the Immediate *Liquidity Transfer* stemming from the triggered Standing *Liquidity Transfer Order* is partially executed. If several Standing *Liquidity Transfer Orders* are to be executed at the same time/business event and there is insufficient liquidity available on the *T2S Dedicated Cash Account* a maximum of one Standing *Liquidity Transfer Order* is partially executed. The others are either fully executed or not executed. In case of standing and predefined Liquidity Transfer Orders, there is no separate notification about settlement status (only the credit/debit notification). Furthermore in case a Liquidity Transfer Order is unsettled the initiator is not informed. The Payment/Settlement Bank/T2S Actors are informed accordingly of the Settlement Status of the Liquidity Transfer. The amount of liquidity not transferred is not stored, i.e. it is not transferred later via a reattempt of execution, even if additional liquidity is credited on the account that was initially debited **{T2S.06.670}.**

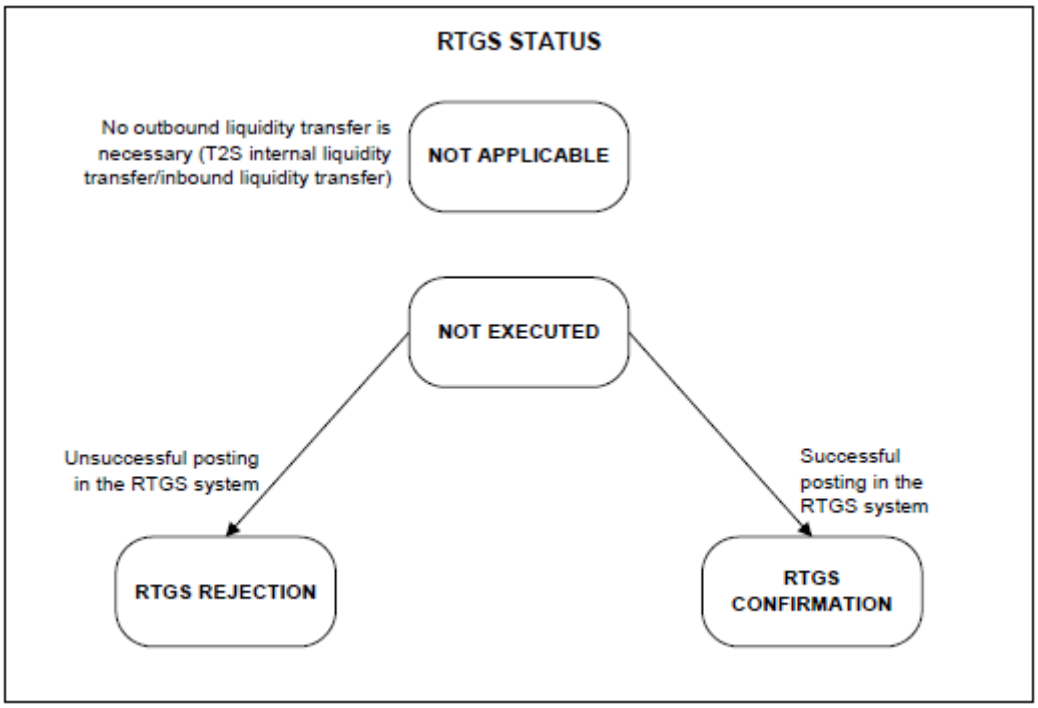
Liquidity Transfer status transition diagrams

Within the *Liquidity Management* domain the following statuses are managed **{T2S.06.232}:**

- Settlement Status keeps track on the booking result;
- RTGS Status keeps track on the result of communication of Outbound *Liquidity Transfers* to RTGS systems (via network) and its processing within the RTGS system based on the RTGS Answer from RTGS system;
- Approval Status keeps track related to the Four-Eyes principle.

All rights reserved.





SETTLEMENT STATUS	DESCRIPTION
"Not Executed"	Liquidity Transfer has been initiated but not yet processed by settlement (Initial status)
"Unsettled"	Liquidity Transfer is not posted in Settlement due to insufficient liquidity.
"Partially Settled"	Liquidity Transfer is only posted with a partial amount.
"Settled"	Liquidity Transfer is posted with the entire amount.
RTGS STATUS	DESCRIPTION
"Not Executed"	Liquidity Transfer has been initiated but not yet sent to RTGS system (Initial status)

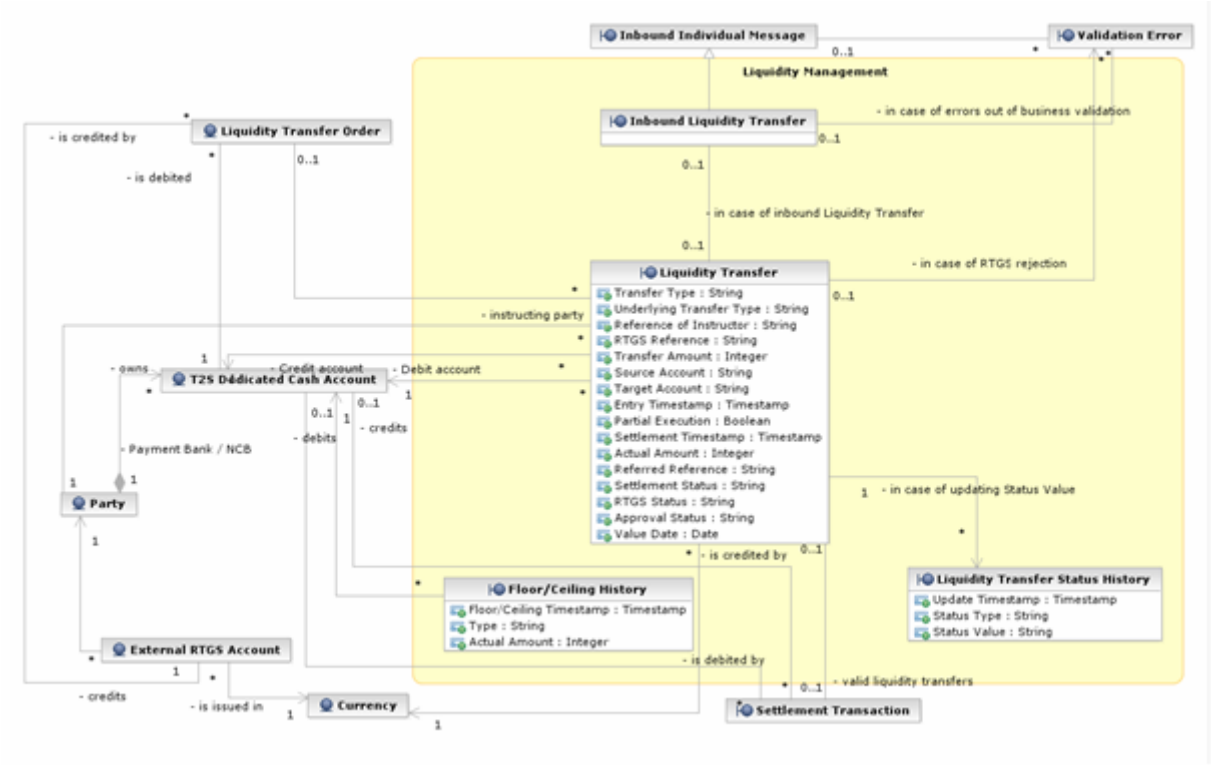
All rights reserved.

SETTLEMENT STATUS	DESCRIPTION
"Not Applicable"	In case of T2S Internal Liquidity Transfers or Inbound Liquidity Transfers no Outbound Liquidity Transfer is necessary.
"RTGS Rejection"	Liquidity Transfer is not booked in the RTGS system.
"RTGS Confirmation"	Liquidity Transfer is booked successfully in the RTGS system.
"RTGS Rebooking"	Liquidity Transfer was rejected and was rebooked to reimburse the debtor account of the Outgoing Liquidity Transfer.

For the description of the concept of the Approval Status in case of Four-Eyes principle and the associated transition diagram, see chapter "T2S Graphical User Interface".

3.6.2 Dynamic data managed by the domain

3.6.2.1 Diagram of the dynamic data



3.6.2.2 Description of the entities

1 – Liquidity Transfer

ATTRIBUTE	DESCRIPTION
Transfer Type	Possible values are (exhaustive list): ▪ Inbound Liquidity Transfer ▪ Outbound Liquidity Transfer

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▪ Internal Liquidity Transfer
Underlying Transfer Type	Identifies the Underlying transfer type of the Inbound/Outbound or Internal Liquidity Transfer. Possible values are (exhaustive list): ▪ Immediate Liquidity Transfer ▪ Predefined Liquidity Transfer Order ▪ Standing Liquidity Transfer Order
Reference of Instruction	Reference given by the original instructor of the Liquidity Transfer. That is: ▪ In case of Inbound Liquidity Transfers the reference send by the Payment Bank to RTGS system ▪ In case of Immediate Liquidity Transfer (Outbound or Internal) the reference send by the instructing T2S party ▪ In case of Standing or Predefined Liquidity Transfer Orders the Liquidity Transfer Order Reference out of the entity Liquidity Transfer Order
RTGS Reference	Reference assigned by the RTGS system for the Inbound Liquidity Transfer
Transfer Amount	Amount to be credited or debited with the Liquidity Transfer. (in case of all cash, the amount is zero)
Source Account	An account should be qualified by the type of identifier and the identifier itself, e.g. qualifying the T2S DCA means: 'Proprietary Account' plus [T2S Dedicated Cash Account].
Target Account	An account should be qualified by the type of identifier and the identifier itself, e.g. qualifying the T2S DCA means: 'Proprietary Account' plus [T2S Dedicated Cash Account].
Entry Timestamp	Timestamp specifying the date and the time the Liquidity Transfer entered or was generated in T2S.
Partial Execution	Flag if partial execution is possible or not.
Settlement Timestamp	Timestamp specifying the date and the time the settlement was attempted in T2S.
Actual Amount	Amount actually settled with the Liquidity Transfer
Referred Reference	Referred Reference is the reference of a settlement instruction to which the Liquidity Transfer is linked
Settlement Status	Possible values are (exhaustive list): ▪ Not executed ▪ Unsettled

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▪ Partially settled ▪ Settled
RTGS Status	Possible values are (exhaustive list): ▪ Not applicable ▪ Not executed ▪ RTGS Rejection ▪ RTGS Confirmation
Approval Status	Possible values are (exhaustive list): ▪ Awaiting approval ▪ Approved ▪ Rejected ▪ Revoked

For the entity "Inbound Liquidity Transfer" please refer to the description provided in the Interface domain.

2 – Liquidity Transfer Status History

ATTRIBUTE	DESCRIPTION
Update Timestamp	Timestamp for the latest update of the Liquidity Transfer status
Status Type	Type of status, which indicates the status of processing. Possible values (exhaustive list): ▪ Settlement Status ▪ RTGS Status ▪ Approval Status
Status Value	Value of the status depending on the Status Type: In case of Status Type = "Settlement status" possible values are (exhaustive list): ▪ Not executed ▪ Unsettled ▪ Partially settled ▪ Settled In case of Status Type = "RTGS status" possible values are (exhaustive list): ▪ Not executed ▪ Not applicable ▪ RTGS Rejection ▪ RTGS Confirmation

All rights reserved.

ATTRIBUTE	DESCRIPTION
	▪ RTGS Rebooking ▪ In case of Status Type = "Approval status" possible values are (exhaustive list): ▪ Awaiting approval ▪ Approved ▪ Rejected ▪ Revoked

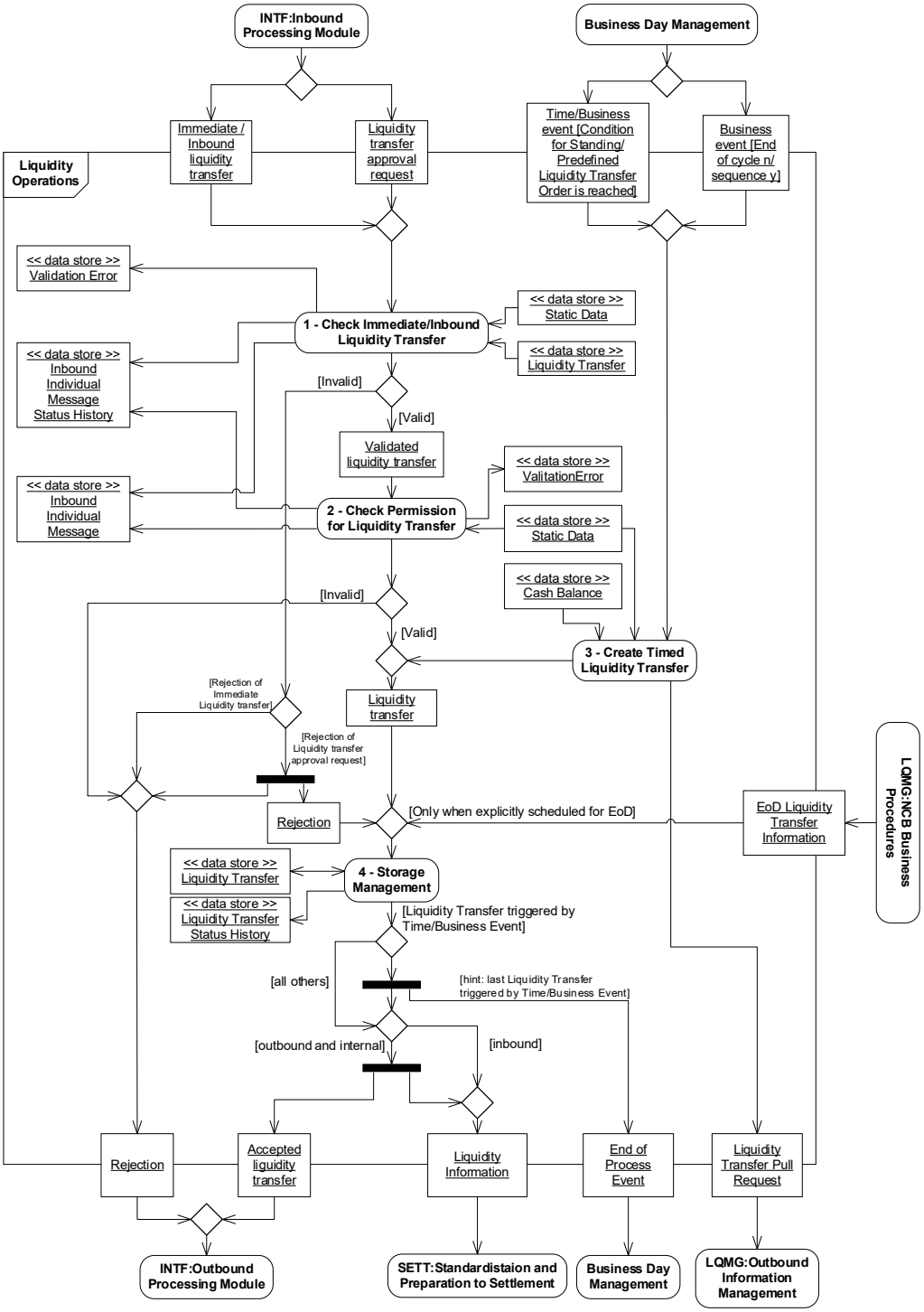
3 – Floor/Ceiling History

ATTRIBUTE	DESCRIPTION
Floor/Ceiling Timestamp	Timestamp when the floor or ceiling violation was triggered
Type	Type of triggering, possible values (exhaustive list): ▪ Floor ▪ Ceiling
Actual Amount	Amount after the posting that triggered the floor/ceiling violation.

All rights reserved.

3.6.3 Liquidity Operations

3.6.3.1 Diagram of the module



{CR-720}

3.6.3.2 Description of the module

Introduction

The *Liquidity Operations* module is in charge of formatting, validating, processing, generating and storing the data related to *Liquidity Transfers*:

- Inbound;
- Outbound (Including *Liquidity Transfers* created from Standing and Predefined *Liquidity Transfer Orders*);
- Internal.

The *Liquidity Operations* module consists of the following functions:

- *Check Immediate / Inbound Liquidity Transfer;*
- *Check Permission for Liquidity Transfer;*
- *Create Timed Liquidity Transfer;*
- *Storage Management.*

Multiple liquidity provider functionality

Based on Standing *Liquidity Transfer Orders* defined in the reference data, the module *Liquidity Operations* facilitates the execution of "multiple liquidity providers" functionality.

During start of settlement day and before start of night time settlement, liquidity can be transferred from different RTGS accounts (i.e. different liquidity providers) to a single *T2S Dedicated Cash Account*. These *Liquidity Transfers* should be executed before the T2S night-time settlement phase starts. The amounts of liquidity effectively transferred are used in the night time settlement process.

The "multiple liquidity provider" functionality caters to the process of refunding the remaining liquidity available in *T2S Dedicated Cash Accounts* to the different liquidity providers according to the configured priorities and the effectively transferred liquidity at the end of the night-time settlement phase **{T2S.06.067}**.

3.6.3.3 Description of the functions of the module

1 – Check Immediate / Inbound Liquidity Transfer

Triggers of the function

Reference Id	***LQMG.LIO.MLT.1.1***

The *Check Immediate / Inbound Liquidity Transfer* function receives *Immediate / Inbound Liquidity Transfers* and *Liquidity Transfer Approval Requests (*in case the Four-Eyes principle is used) *from* the *Inbound Processing* Module during the day, including those which are sent during SOD process **{T2S.03.070}**.

Validation/Revalidation

Reference Id	***LQMG.LIO.MLT.2.1***

The function validates all incoming *Immediate / Inbound Liquidity Transfers* and performs a preliminary check of *Liquidity Transfer Approval Requests* **{T2S.06.510}** **{T2S.06.520}** {**T2S.06.521}** **{T2S.06.530}** **{T2S.06.060}**.

All rights reserved.

In case of Liquidity Transfer Approval Requests LQMG does not have to validate if an underlying liquidity transfer exists and if the status is 'awaiting approval'. This is ensured by the Data Changes screen where the user can only choose existing LTs with the status 'awaiting approval'. Additionally Liquidity Transfer Approval requests can only exist for immediate liquidity transfers and not for inbound LTs (as always A2A).

The following paragraphs identify a non-exhaustive list of validations/ revalidations of *Immediate / Inbound Liquidity Transfers* to be performed (based on the reference data):

- The Business Sending Party is existing and an active T2S Party **{T2S.06.540}**;
- The involved *RTGS Accounts* are existing and active, in case of Outbound *Liquidity Transfer*;
- The involved *T2S Dedicated Cash Accounts* are existing and active (example: neither blocked by restrictions nor logically deleted) **{T2S.06.560}**;
- The involved *T2S Dedicated Cash Accounts* in case of an Internal *Liquidity Transfer* are linked to the same RTGS *account* or belong to the same Payment/Settlement Bank or one or both of the involved accounts is a T2S Central Bank account. This rule is not applicable if one of the involved accounts is a Dedicated Transit Account i.e. if one of the involved accounts is a Dedicated Transit Account Internal Liquidity Transfers are always possible **{T2S.06.114}** **{T2S.06.241}**;
- The currency is eligible as T2S settlement currency and is the same as the currency of the *T2S Dedicated Cash Accounts* involved **{T2S.06.530}**;
- The Transfer Amount is existing and positive **{T2S.06.590}**;
- There is no duplicate submission of the incoming *Immediate / Inbound Liquidity Transfer* based on a combination of the T2S Actor identifier, Transfer Amount, Dedicated Cash Accounts and the Order Reference assigned by the Business Sending Party. This check compares the incoming *Immediate / Inbound Liquidity Transfer* with those *Liquidity Transfers* not settled yet and those already in a predefined number of days in the past [197] **{T2S.06.570}**;
- When the first negative validation result is found, the function continues with the validation until all negative results are collected **{T2S.06.510}**.

Follow-ups of the validations

Reference Id	***LQMG.LIO.MLT.3.1***

The result of the validation of any *Immediate / Inbound Liquidity Transfer* determines the next steps to be performed by the *Check Immediate / Inbound Liquidity Transfer* function as follows:

- In case of positive validation, the function:
  - determines the RTGS Dedicated Transit Account to be credited (in case of Outbound *Liquidity Transfers*) or to debited (in case of Inbound Liquidity Transfer),
  - sets the partial execution flag (true or false) in case where the Immediate *Liquidity Transfer* is initiated by a CSD (or another party) authorised by the account holder of the account to be debited **{T2S.06.630}**,

197 This predefined number of days is a parameter set in T2S and is the same for the Settlement Restrictions, Settlement Instructions and all maintenance instructions (Hold/Release, Amendment and Cancellation Instructions).

All rights reserved.

- routes the validated *Immediate / Inbound Liquidity Transfer* via *Validated Liquidity Transfer* flow to the *Check Permission for Liquidity Transfer* function;

- In case of negative validation (e.g. account to be credited/debited not present or not active in T2S), the function:
  - creates and stores records *to the entity Validation Error*,
  - creates a Rejection including all the negative validation results to be forwarded via the Outbound Processing Module to the T2S Actor (according to the Message Subscription preferences) **{T2S.06.510} {T2S.06.580}**,
  - updates the Business Validation Status in the data store Inbound Individual Message and Inbound Individual Message Status History (by changing the Status Value to *"True"* for the Business Validation Status) **{T2S.06.232} {T2S.06.600} {T2S.06.610}**.

Follow-ups of the revalidations

Reference Id	***LQMG.LIO.MLT.4.1***

The revalidation of an *Immediate/ Inbound Liquidity Transfer* (after reception and positive validation of Liquidity Transfer Approval Request*)* determines the next steps to be performed by the *Check Immediate Liquidity Transfer* function as follows:

- In case of positive validation, the function routes the validated Liquidity Transfer Approval Request via Validated Liquidity Transfer flow to the *Check Permission for Liquidity Transfer* function;

- In case of negative validation (e.g. account to be credited/debited not present or not active in T2S), the function:
  - creates and stores records to the entity *Validation Error*,
  - creates a *Rejection* including all the negative validation results to be forwarded to the *Outbound Processing* Module in order to inform the T2S Actor and to the *Storage Management* function in order to update the status of the stored *Liquidity Transfer* **{T2S.06.510} {T2S.06.580}**,
  - updates the *Business Validation Status* in the data store *Inbound Individual Message and Inbound Individual Message Status History* (by changing the Status Value to "*True*" for the *Business Validation Status*) **{T2S.06.232} {T2S.06.600} {T2S.06.610}**.

2 – Check Permission for Liquidity Transfer

Triggers of the function

Reference Id	***LQMG.LIO.PLT.1.1***

The *Check Permission for Liquidity Transfer* function receives *Liquidity Transfers* and *Liquidity Transfer Approval Requests* that have passed validation positively.

Validation of Permissions

Reference Id	***LQMG.LIO.PLT.2.1***

All rights reserved.

The function checks (based on the reference data) that the T2S party instructing the *Liquidity Transfer* has sufficient privileges to instruct on the *T2S Dedicated Cash Account* to be debited **{T2S.06.210} {T2S.06.550} {T2S.06.060} {T2S.06.240} {T2S.06.540}**.

The validations to be performed include the case that the T2S System User is sending the *Immediate Liquidity Transfer* using a Four-Eyes principle.

Follow-ups of the validation of Permissions

Reference Id	***LQMG.LIO.PLT.3.1***

After completing the validation of permissions, the function:

- In case of positive validation, routes the *Liquidity Transfer* or the *Liquidity Transfer Approval Request* via *Liquidity transfer* flow to the *Storage Management* function;
- In case of negative validation:
  - creates and stores *Validation Error* records,
  - in case of a *Liquidity Transfer Approval Request* creates a *Rejection*. The function routes it via the *Interface* domain to the T2S Actor,
  - in case of an *Immediate Liquidity Transfer* creates a *Rejection and routes it via Interface domain* to the T2S Actor,
  - in case of an Inbound Liquidity Transfer creates a Rejection and routes it via Interface domain to the T2S Actor (RTGS system).

In both cases (positive and negative validation) the function updates the *Business Validation Status* in the data store *Inbound Individual Message and Inbound Individual Message Status History* (by changing the Status Value to "*True*" for the *Business Validation Status*) **{T2S.06.232} {T2S.06.610}**.

3 – Create Timed Liquidity Transfer

Triggers of the function

Reference Id	***LQMG.LIO.TLT.1.1***

The *Create Timed Liquidity Transfer* function generates an *Immediate Liquidity Transfer* **or a** *Liquidity Transfer Pull Request* after having received one of the following business or time events from the Scheduling module **{T2S.06.067} {T2S.06.280}**:

- *Business event [End of cycle n/ sequence Y]* (for refunding of liquidity after the end of last cycle in case the "multiple liquidity provider" functionality);
- *Time/Business event [Condition for Standing/Predefined Liquidity Transfer Order is reached]* [198].

Create Liquidity Transfer

Reference Id	***LQMG.LIO.TLT.2.1***

198 The conditions for automated retransfer of cash proceeds from Corporate Actions (CARL) and monetary policy Repo operations are dealt in SETT domain chapters of GFS

All rights reserved.

Based on the events received, this function retrieves the relevant Standing and Predefined *Liquidity Transfer Orders* from the Reference Data **{T2S.06.220} {T2S.06.280} T2S.06.340}**.

The function creates relevant *Liquidity Transfers* and sends them to the function *Storage Management* **{T2S.06.206}**.

Multiple liquidity provider functionality at the end of night time settlement phase

Reference Id	***LQMG.LIO.TLT.3.1***

For the execution of the multiple liquidity provider functionality this function **{T2S.06.067}**:

- Retrieves from the reference data , for each of the involved liquidity providers of the *T2S Dedicated Cash Account*, the sequence in which the liquidity providers are to be reimbursed (the main liquidity provider being the last one to be reimbursed);
- Retrieves from the entity *Cash Balance* the available balance of the involved *T2S Dedicated Cash Account*;
- Retrieves the cash postings that occurred on the specific *T2S Dedicated Cash Account* as well as the *Liquidity Transfers* each liquidity provider sent to/received from the *T2S Dedicated Cash Account* between SOD and the end of the night time settlement and calculates the net amount to be re-transferred to each Liquidity Provider;
- Creates the *Liquidity Transfers* and sends them to the function *Storage Management*.

Treatment of several Standing Liquidity Transfer Orders at the same time

Reference Id	***LQMG.LIO.TLT.4.1***

In the case that more than one Standing *Liquidity Transfer Order* per *T2S Dedicated Cash Account* are to be executed at a given time or on the occurrence of a given business event, those Standing *Liquidity Transfer Orders* are executed in a random order [199] **{T2S.06.360}**.

Create Liquidity Transfer Pull Request

Reference Id	**LQMG.LIO.TLT.5.1**

Based on the events received, this function retrieves the relevant Standing and Predefined Liquidity Transfer Orders from the Reference Data {T2S.06.220} {T2S.06.280} {T2S.06.340}.

The function creates relevant Liquidity Transfer Pull Requests and sends them to the function Outbound Liquidity Transfers {T2S.06.206}.{CR-720}

4 – Storage Management

Storage of Liquidity Transfers

Reference Id	***LQMG.LIO.SOM.1.1***

199 *The order of the execution is technically influenced by the order of reading all the standing liquidity transfer orders which should be executed at the same time*

All rights reserved.

The function stores all *Liquidity Transfers* and *End-of-Day Liquidity Transfers* (only during the End-of-Day period – see CB Business Procedure module) in the entity *Liquidity Transfer* following *Audit Trail requirements* **{T2S.06.600}**.

All records are created and updated in the entity *Liquidity Transfer* and in the entity *Liquidity Transfer Status History*.

The records are created with **{T2S.06.232}**:

- The Settlement status as *"Not Executed"*;
- The RTGS status as *"Not Executed"* for all Outbound *Liquidity Transfers* and *"Not Applicable"* for all *Inbound Liquidity Transfers* or *Internal Liquidity Transfers*.

Use of the Two-Eyes or Four-Eyes principle

Reference Id	***LQMG.LIO.SOM.2.1***

Storage Management validates whether an *Immediate Liquidity Transfer* was sent using Two-Eyes principle or Four-Eyes principle:

- In case of the Two-Eyes principle the Approval Status for the *Liquidity Transfer* is set to *"Approved"*;
- In case of Four-Eyes principle the Approval Status for the *Liquidity Transfer* is set to "Awaiting Approval".

When the *Storage Management* receives a valid *Liquidity Transfer Approval Request* (i.e. an *Approval Request* having successfully passed the preliminary checks performed by the *Check Immediate / Inbound Liquidity Transfer* function*),* the Approval status of the *Liquidity Transfer* is changed as follows:

- In case of a confirmation by the second T2S system user provided that the revalidation of the initial *Liquidity Transfer* is positive, the Approval Status of the initial Liquidity Transfer is changed to "*Approved"*;
- In case of a revocation by the second T2S system user the Approval Status is changed to *"Revoked"*.

In case the *Storage Management* function receives a *Rejection* flow (i.e. the validation/ revalidation of the initial *Liquidity Transfer* leads to a negative result) the Approval Status of the initial *Liquidity Transfer* is changed to *"Rejected"*.

Follow-ups of the storage

Reference Id	***LQMG.LIO.SOM.3.1***

The *Storage Management* function sends based on the stored *Liquidity Transfer* a *Liquidity Information* to the *Standardisation and Preparation to Settlement* module when processing the following inputs:

- *Immediate Liquidity Transfer* (using a Two-Eyes principle);
- *Liquidity Transfer Approval* Request (after the approval of an *Immediate Liquidity Transfer* using a Four-Eyes principle);
- *Inbound Liquidity Transfer*;
- Internally generated *Immediate Liquidity Transfer* (generated by the triggering of Standing or Predefined *Liquidity Transfer Orders*);

All rights reserved.

- *End-of-Day Liquidity Transfer Information* (for cash sweep);
- In case of internally generated *Immediate Liquidity Transfers* which should transfer "all cash" an "all cash flag" is forwarded via *Liquidity Information* module to the *Settlement* domain.
- In all cases –except Inbound Liquidity Transfers-, the validation result is forwarded to the *Interface* Domain via *Accepted Liquidity Transfer* for sending out to the initiating T2S Actor (according to the Message Subscription preferences). With the information given in the flow *Accepted Liquidity Transfer* the *Interface* Domain is able to assign the *Liquidity Transfer Approval Request* to the original immediate *Liquidity Transfer*.

The function Storage Management sends an *End of Process Event* to the *Scheduling* module in case the Outbound *Liquidity Transfer* has been triggered by the *Scheduling* module. Related to the "multiple liquidity provider" functionality the *Storage Management* function informs the *Scheduling* module via the *End of Process Event* after the last involved *Liquidity Transfer* has been forwarded to the *Settlement* domain.

3.6.3.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Immediate/ Inbound Liquidity Transfer	IN	Immediate Liquidity Transfer originally coming from T2S Actors / Inbound Liquidity Transfer coming from RTGS systems	INTF: Inbound Processing Module	
Liquidity Transfer Approval Request	IN	Approval requests coming from T2S Actors related to the Four-Eyes principle	INTF: Inbound Processing Module	
Time/Business Event [Condition for Standing/ Predefined Liquidity Transfer Orders is reached]	IN	Trigger for processing standing/ predefined orders	~~**OPSR: Scheduling**~~ **Business Day Management**	
Business Event [End of cycle n/ sequence y]	IN	End of cycle	~~**OPSR: Scheduling**~~ **Business Day Management** **{CR-720}**	
EoD Liquidity Transfer Information	IN	EoD Liquidity Transfer Information for processing EoD Liquidity Transfers	LQMG: CB Business Procedures	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Rejection	OUT	For failed business validation or missing permission		INTF: Outbound Processing Module
Liquidity Information	OUT	Request to create a settlement transaction		SETT:SPS
Accepted Liquidity Transfer	OUT	Information to T2S Actor, that the business validation has been successful		INTF: Outbound Processing Module
End of Process Event	OUT	Finalisation of former triggering event sent by scheduler		~~**OPSR: Scheduling**~~ **Business Day Management**
Liquidity Transfer Pull Request Information	**OUT**	**Information for Request to CLM to send an Inbound Liquidity Transfer**		**LQMG: Outbound Information Management**{CR-720}

3.6.3.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
CRDM	T2S Dedicated Cash Account T2S Dedicated Cash Account Restriction External RTGS Account External RTGS Account Restriction Liquidity Transfer Order Liquidity Transfer Order Link Liquidity Transfer Order Link Set Currency Party Party Code Party Restriction Role Privilege Object SD Secured Object	Read	

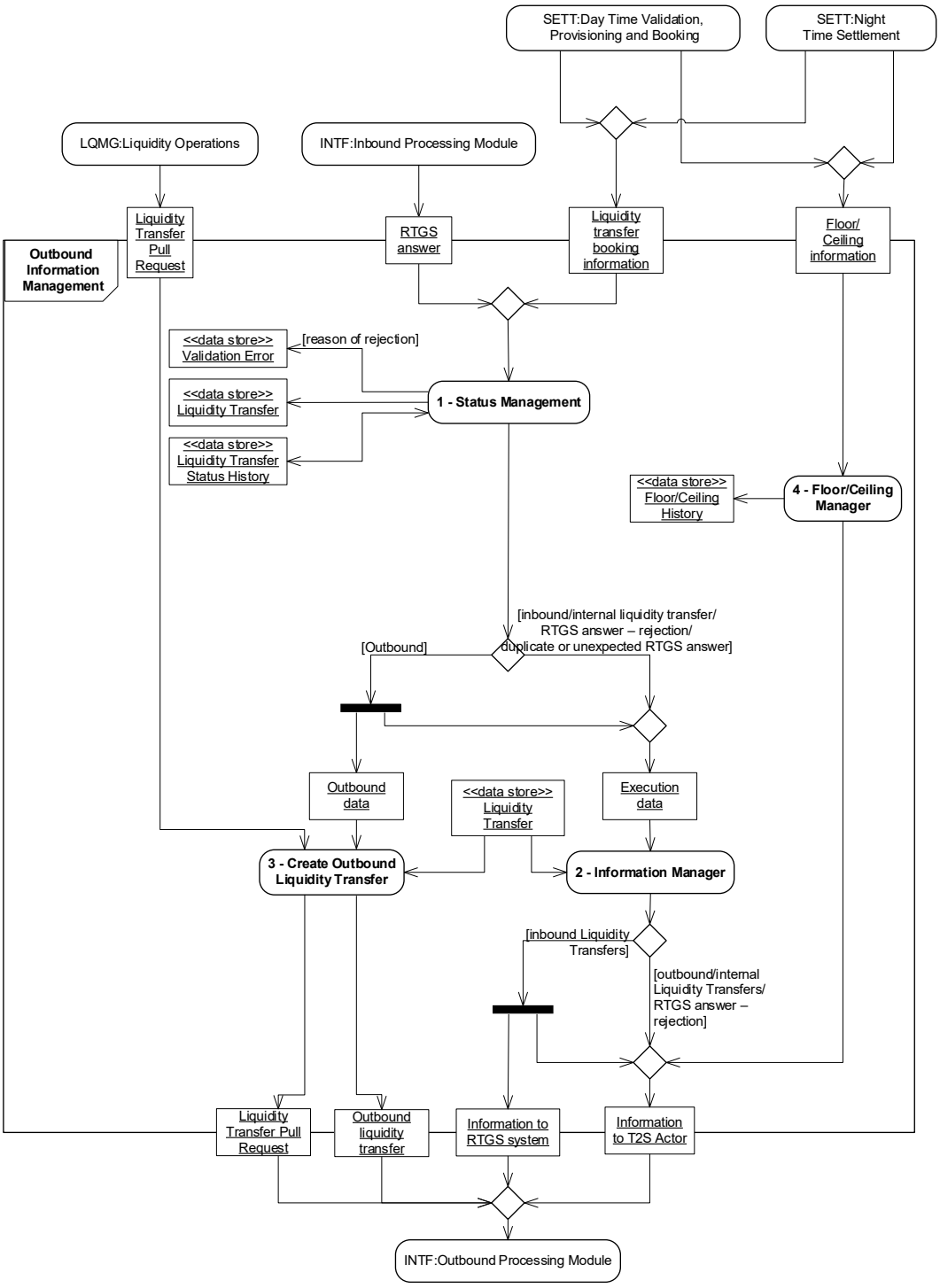
All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
	SD Secured Object Group T2S System Status		
Liquidity Transfer		Read/ Write	
Cash Balance		Read	
Validation Error		Write	
Liquidity Transfer Status History		Write	
Inbound Individual Message Status History		Write	
Inbound Individual Message		Write	

3.6.4 Outbound Information Management

All rights reserved.

3.6.4.1 Diagram of the module



{CR-720}

3.6.4.2 Description of the module

Purpose of the module

All rights reserved.

The *Outbound Information Management* module is in charge of the maintenance of the status information of *Liquidity Transfers* (as well as RTGS status) and management of all the data necessary to create Outbound *Liquidity Transfers* between T2S and any connected RTGS system. In addition, it provides T2S Actors, with information on changes of the Settlement Status of *Liquidity Transfers*, special posting events (i.e. floor/ceiling information) on *T2S Dedicated Cash Accounts* and rejection from RTGS systems.

The Outbound Information Management consists of four functions:

- *Status Management;*
- *Information Manager;*
- *Create Outbound Liquidity Transfer;*
- *Floor/Ceiling Manager.*

Triggers of the module

The Outbound Information Management receives the following incoming information:

- *RTGS Answer*;
- *Floor/Ceiling Information* (fall under predefined floor, excess of predefined ceiling);
- *Liquidity Transfer booking information*.
- *Liquidity Transfer Pull Request information* **{CR-720}**

3.6.4.3 Description of functions of the module

1 – Status Management

This function manages the status information of all the *Liquidity Transfers* after settlement in T2S. *Liquidity Transfer booking information* and *RTGS Answer* are processed by the *Status Management* function.

Update of settlement information

Reference Id	***LQMG.OIM.SMG.1.1***

According to the result of the settlement attempt, the flow *Liquidity Transfer Booking Information* from the *Day Time Validation, Provisioning and Booking* module or the *Night Time Settlement* module includes the necessary information to store the Settlement Timestamp and the Actual Amount of the *Liquidity Transfer* after its settlement processing (the *Settlement Status* is stored by the *Settlement* domain) **{T2S.06.630}**.

The following Settlement Statuses values are possible:

- *"Settled"* in case the *Liquidity Transfer* was booked with the full amount **{T2S.06.680}**;
- *"Partially Settled"* in case it was not possible to settle the whole amount of the *Liquidity Transfer* but only part of it;
- *"Unsettled"* in case there was no liquidity at all on the *T2S Dedicated Cash Account* or no partial settlement was possible, so the *Liquidity Transfer* was not settled.

These settlement statuses apply to all *Liquidity Transfers* except *Inbound Liquidity Transfers*. For *Inbound Liquidity Transfers* the only possible status is *"Settled"*, because booking cannot fail due to the fact that:

All rights reserved.

- All relevant validations are done by *Liquidity Operations* module before forwarding the *Liquidity Information* to *Settlement* domain for settlement **{T2S.06.690} {T2S.06.700}**;
- Such *Liquidity Transfer* debits a Dedicated Transit Account and credits a *Dedicated Cash Account* and the debit cannot fail, since a Dedicated Transit Account is a T2S Dedicated Cash Account owned by a central bank, for which the balance can be negative.

The following flows are sent to the next function for further treatment:

- The function sends *Outbound Data* to the function *Create Outbound Liquidity Transfer* in case the *Liquidity Transfer Booking Information* is referring to an *Outbound Liquidity Transfer*;
- The function sends *Execution Data* to the *Information Manager* function in all cases.

Update of RTGS Status for Outbound Liquidity Transfers

Reference Id	***LQMG.OIM.SMG.2.1***

For Outbound *Liquidity Transfers*, after the feedback from the RTGS system is received, the RTGS Status of the *Liquidity Transfer* is changed from *"Not executed"* to: **{T2S.06.710}**

- *"RTGS Confirmation"* in case a confirmation from the RTGS system was received that the Outbound *Liquidity Transfer* was credited successfully to the relevant account in the RTGS system.
- *"RTGS Rejection"* in case a rejection or an error message was received from the RTGS system indicating that it was not able to credit the relevant RTGS account in the Outbound *Liquidity Transfer*. Additionally, the reason for the rejection is stored in the data store *Validation Error*.
- When T2S receives an *RTGS answer* (*"Confirmation"* or *"Rejection"*), which does not correspond to a *Liquidity Transfer* in T2S, or a duplicate RTGS Answer this function logs an error in the Data store *Validation Error*. The duplicate RTGS answer is checked based on the Reference of the Liquidity Transfer.

The RTGS status is stored in the Data stores *Liquidity Transfer Status History* and Liquidity Transfer.

In case of a *"Rejection"* or a duplicate / unexpected RTGS Answer, the function sends *Execution Data* to the *Information Manager* function.

2 – Information Manager

Purpose of the function

Reference Id	***LQMG.OIM.IFM.1.1***

The *Information Manager* provides:

- T2S Actors with information on changes of the Settlement Status or RTGS Status of a *Liquidity Transfer* **{T2S.06.196} {T2S.06.680} {T2S.06.690} {T2S.06.700} {T2S.06.710}**;
- RTGS Systems with information on changes of the Settlement Status of an *Inbound Liquidity Transfer.*

Information to the T2S Actor

Reference Id	***LQMG.OIM.IFM.2.1***

All rights reserved.

After receiving information (*Execution Data*) [200] about a change of the Settlement Status or the RTGS Status of a *Liquidity Transfer*, the *Information Manager* function collects all business information needed to send this data as *Information to T2S Actor* to the *Outbound Processing Module* of the *Interface* domain to be forwarded to the T2S Actor according to the Message Subscription preferences **{T2S.06.196} {T2S.06.680} {T2S.06.690} {T2S.06.700} {T2S.06.710}**;

This functionality includes the information to the CB in case of *RTGS Answer "Rejection" and unexpected/duplicate RTGS Answers*. In case of standing/predefined Liquidity Transfer Orders there is no separate notification about the settlement status (only the credit/debit notification). In case a Liquidity Transfer Order is unsettled the initiator is not informed.

Booking information for the RTGS system

Reference Id	***LQMG.OIM.IFM.3.1***

After receiving information (*Execution Data*) about a change of the Settlement Status of an *Inbound Liquidity Transfer*, the Information Manager collects all business information needed to send this data as Information to RTGS system to the *Outbound Processing Module* of the *Interface* domain to be forwarded to the RTGS system according to the Message Subscription preferences **{T2S.06.710}**.

3 – Create Outbound Liquidity Transfer

Reference Id	***LQMG.OIM.OLT.1.1***

After receiving *Outbound Data* from the *Status Management* function the *Create Outbound Liquidity Transfer* function collects all business information needed from *Liquidity Transfer* and sends this data as *Outbound Liquidity Transfer* to the *Outbound Processing Module* of the *Interface* domain. **Data for** *Liquidity Transfer Pull Requests* **is also received from the** *Liquidity Operations* **module, which is then processed and sent to the** *Outbound Processing Module* **of the** *Interface* **domain.** **{CR-720}**

4 – Floor/Ceiling Manager

Reference Id	***LQMG.OIM.FCM.1.1***

After receiving *Floor/Ceiling Information from the Day Time Validation, Provisioning and Booking module or the Night Time Settlement module* the *Floor/Ceiling Manager* function stores the information received and sends this data as *Information to T2S Actor* to the *Outbound Processing Module* to be forwarded to the T2S Actor, if he has set up a Floor and Ceiling amount within the reference data**{T2S.06.233} {T2S.06.234} {T2S.06.680} {T2S.06.690} {T2S.06.700}**.

3.6.4.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Floor/Ceiling Information	IN	Information, that T2S Dedicated Cash Account balance falls un-	SETT: Validation, Provisioning and Booking and	

200 Outbound Data are needed to create an Outbound Liquidity Transfer. Execution Data are used to create Information to T2S Actor or Information to RTGS system.

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
		der the defined minimum amount (floor) or exceeds the defined maximum amount (ceiling)	SETT: Night Time Settlement	
Liquidity Transfer booking information	IN	Result of settlement	SETT: Validation, Provisioning and Booking and SETT: Night Time Settlement	
RTGS Answer	IN	Answer from RTGS, confirmation or rejection including a reason code	INTF: Inbound Processing Module	
Liquidity Transfer Pull Request Information	**IN**	**Information for Request to CLM to send an Inbound Liquidity Transfer**	**LQMG: Liquidity Operations**{CR-720}	
Information to T2S Actor	OUT	T2S Actor is informed about settlement status and/ or RTGS status and/or the alerts due to floor or ceiling violations		INTF: Outbound Processing Module
Outbound Liquidity Transfer	OUT	All business information of the Outbound Liquidity Transfer		INTF: Outbound Processing Module
Information to RTGS System	OUT	RTGS is informed about the settlement status of Inbound Liquidity Transfers		INTF: Outbound Processing Module
Liquidity Transfer Pull Request	**OUT**	**Request to CLM to send an Inbound Liquidity Transfer**		**INTF: Outbound Processing Module**{CR-720}

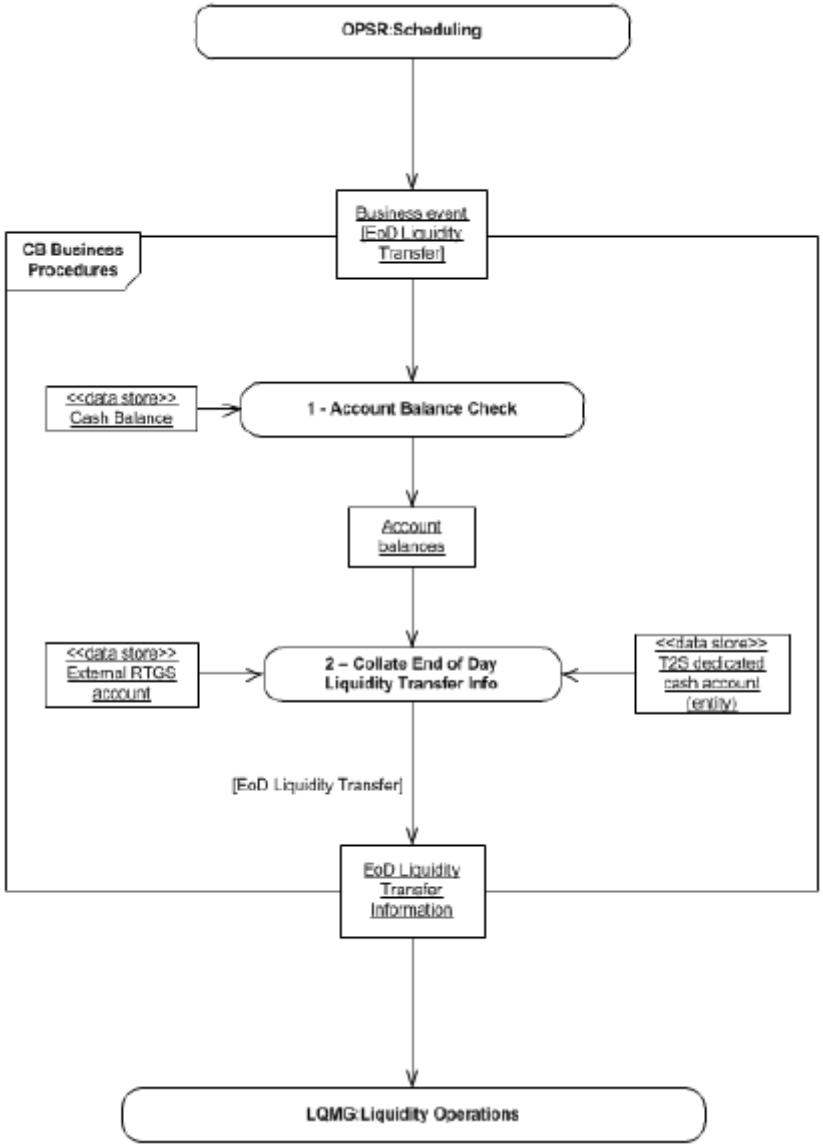
All rights reserved.

3.6.4.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
Liquidity Transfer Status History		Read / Write	
Validation Error		Write	
Liquidity Transfer		Read / Write	
Floor/Ceiling History		Write	

3.6.5 CB Business Procedures

3.6.5.1 Diagram of the module



All rights reserved.

3.6.5.2 Description of the module

Introduction

This module is in charge of creating *Liquidity Transfers* related to the End-of-Day procedures of T2S. **Please note that the activities described in this module are not related to the optional End-of-Day "second optional cash sweep", which is configured in CRDM as a** *Standing Liquidity Transfer Order*.

The CB Business Procedures module performs at End-of-Day the End-of-Day Liquidity Transfer for Cash Sweep. Based on *Business event [EoD Liquidity Transfer]* received from ~~**the Scheduling module**~~ **Business Day Management**Description of the module [575], the *CB Business Procedures* module checks the balances of all *T2S Dedicated Cash Accounts* (data store Cash Balance).**{CR-720}** The linked external *RTGS accounts* are retrieved from the reference data and the respective Outbound *Liquidity Transfers* are triggered (using the flow *EoD Liquidity Transfer Information*) via the *Liquidity Operations* module. Thus, all the remaining cash on each *T2S Dedicated Cash Account* is automatically transferred to the corresponding external RTGS account {**T2S.06.220} {T2S.03.180} {T2S.06.050}** In case of a negative balance on a central bank's *T2S Dedicated Cash Account* the Outbound *Liquidity Transfer* for the cash sweep is a direct debit transfer instead of a credit transfer.

The *CB Business Procedures* module consists of two functions:

- *Account Balance Check;*
- *Collate End-of-Day Liquidity Transfer Info.*

Please note that for all Currencies connected to CLM, Business Event [EoD Liquidity Transfer] will not be scheduled.{CR-720}

3.6.5.3 Description of the functions of the module

1 – Account Balance Check

Check of remaining liquidity on T2S Dedicated Cash Accounts

Reference Id	LQMG.NBP.ABC.1.1

After receiving a *Business event [EoD Liquidity Transfer]* the *Account Balance Check* function checks the balances of all *T2S Dedicated Cash Accounts* (including Central Bank *T2S Dedicated Cash Accounts*) except for the RTGS Dedicated Transit Accounts. For each *T2S Dedicated Cash Account* with a balance deviating from zero, the function forwards the *Account Balances* to the *Collate End-of-Day Liquidity Transfer Info* function. If the balance is zero, then nothing is done for this *T2S Dedicated Cash Account*.

2 – Collate End of Day Liquidity Transfer Info

Purpose of the function

Reference Id	LQMG.NBP.EDL.1.1

This function collates the End-of-Day Liquidity Transfer related information for the above mentioned steps.

Collate End-of-Day Liquidity Transfer info for Step 1 – End of Day Liquidity Transfer

All rights reserved.

Reference Id	LQMG.NBP.EDL.2.1
4	

After receiving the *Account Balances* of the *T2S Dedicated Cash Accounts* from *Account Balance Check* function, this function determines the linked *RTGS accounts*, collates for each *T2S Dedicated Cash Account* the *End of Day Liquidity Transfer Information* and forwards it to the *Liquidity Operations* module.

Background information procedure for Reconciliation procedure between T2S and connected RTGS systems (This is no functionality of the Collate End of Day)

Reference Id	LQMG.NBP.EDL.3.1

The RTGS systems can receive its RTGS Dedicated Transit Account balances and cash postings via the report "Statement of accounts at End-of-Day". This allows reconciling the postings and balances with the T2S Dedicated Transit Account within their RTGS system.

3.6.5.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Business Event [EoD Liquidity Transfer]	IN	See Introduction	OPSR:Scheduling	
EoD Liquidity Transfer Information	OUT	EoD Liquidity Transfer Information for processing EoD Liquidity Transfers		LQMG: Liquidity Operations

3.6.5.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
Cash Balances		Read	Balances for Liquidity Transfers at End-of-Day
External RTGS Account		Read	
T2S Dedicated Cash Account		Read	

3.6.6 Liquidity Transfers (LT) Use Cases

Scope

This section of use cases covers the liquidity transfers taking place between RTGS Accounts and T2S Dedicated Cash Accounts as well as between two T2S Dedicated Cash Accounts.

Specifically, the following is covered:

- Execution of Immediate liquidity transfers between T2S and RTGS

All rights reserved.

- Execution of Standing and Predefined liquidity transfer orders from T2S to RTGS as well as
- The End-of-Day liquidity transfers

Settlement of a cash leg of a securities transaction, Intraday credit provision through auto-collateralisation, the specific case for initiating immediate liquidity transfers in U2A mode using the 4-eyes principle and special conditions in the End-of-Day (e.g. reflecting a cross CB context) are not covered by the liquidity transfer use cases.

In addition, the liquidity transfer use cases do not cover cash-related cases where reference data maintenance and/or query request takes place. Creating, changing, deleting or querying of Standing and Predefined Liquidity Transfer Orders are also not covered.

Please see therefore the related functionality described in the respective chapters.

Criteria

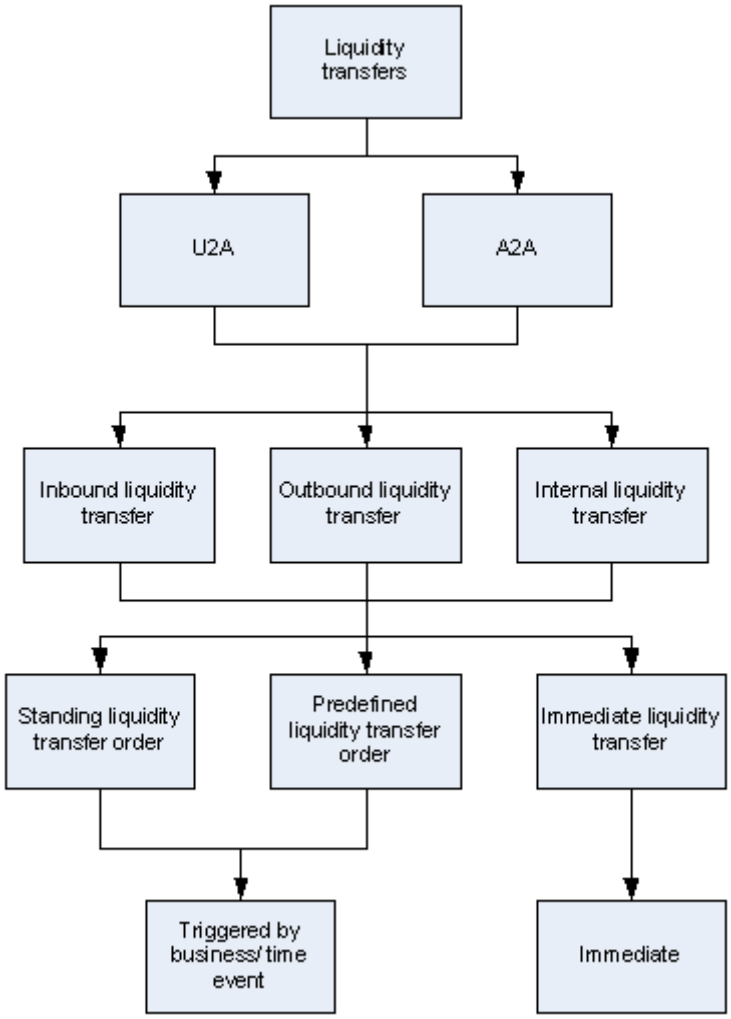
The criteria that characterise a liquidity transfer are summarised in the table below (exhaustive list):

CRITERIA	POSSIBLE VALUES	DESCRIPTION
Communication mode	U2A, A2A	See T2S URD glossary
Transfer Type	Inbound liquidity transfer, Outbound liquidity transfer, Internal liquidity transfer	Defines direction of liquidity transfer
Underlying Transfer Type	Immediate liquidity transfer, Predefined liquidity transfer order, Standing liquidity transfer order	See T2S URD glossary
Execution	Immediate, triggered by business/time event	Defines when the order is executed

List of Use Cases

The initiating point of Liquidity Transfer can be both user interaction and a triggering by the system (e.g. execution of standing orders).

All rights reserved.



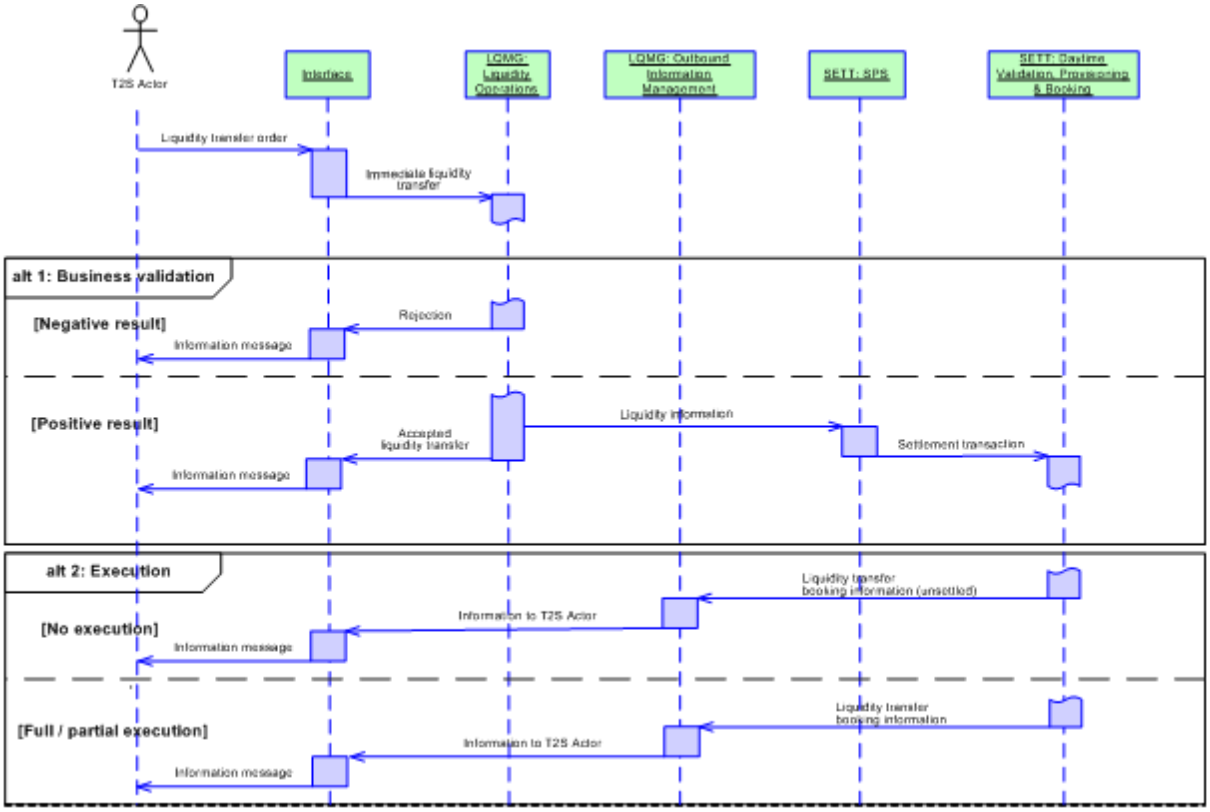
The Liquidity Management use cases are built by combining the criteria values as shown in the above tree: Not all combinations of criteria values constitute a valid liquidity transfer use case. Liquidity management use cases produced by relevant combinations of criteria's values are presented in the complete list of use cases in the appendix.

All rights reserved.

3.6.7 Processing of Liquidity Transfers Use Cases

3.6.7.1 Processing of UC-LT-1: Execution of Immediate Liquidity Transfers within T2S

DIAGRAM 41 - EXECUTION OF IMMEDIATE LTS WITHIN T2S



Business assumption

A *T2S Dedicated Cash Account* holder or another T2S Actor granted with the respective privilege sends an Immediate *Liquidity Transfer* to T2S to transfer liquidity between two *T2S Dedicated Cash Accounts* belonging to the same Payment Bank or between two *T2S Dedicated Cash Accounts* linked to the same *RTGS Account.*

Please note that all communication between the *Interface* domain and T2S Actors should be understood as

- "Encrypted inbound communication A2A" or "Encrypted outbound communication A2A" for A2A mode
- "U2A XHTML request via HTTPS" or "U2A XHTML response via HTTPS" for U2A mode

as described in the *Interface* domain chapter 3.2. For the sake of readability, more specific terms for these flows are used in the diagram above.

The execution of Immediate *Liquidity Transfers* within T2S is also applicable during Night time Settlement, but this Use Case covers only the Daytime processing.

Processing

After successful technical validations of the Immediate *Liquidity Transfer* in the *Interface* domain, the *Liquidity Operations* module carries out several Business Validations, for example:

All rights reserved.

- *T2S Dedicated Cash Accounts* are existing and active in the reference data;
- the involved *T2S Dedicated Cash Accounts* are linked to the same *RTGS Account* or belong to the same Payment Bank;
- all mandatory fields for *Liquidity Transfers* are filled;
- the currency is eligible as T2S settlement currency and is the same as the currency of the *T2S Dedicated Cash Accounts* involved;
- there is no duplicate submission of the incoming Immediate *Liquidity Transfer*;
- the T2S Actor instructing the Liquidity Transfer has sufficient privileges (e.g. based on a contractual agreement) to instruct on the *T2S Dedicated Cash Account* to be debited in case the instructing T2S Actor is not the account owner.

Alternative 1 - Business Validations: Negative Result

- In case of a negative result the Immediate *Liquidity Transfer* is rejected and a Rejection including all the negative validation results is sent to the *Interface* domain, which forwards an information message to the T2S Actor.

Alternative 1 - Business Validations: Positive Result

- If the validations are confirmed and successful, the *Liquidity Operations* Module creates a *Liquidity Transfer* and forwards the *Liquidity Information* (to debit one *T2S Dedicated Cash Account* – and to credit another one) to the Standardisation and Preparation to Settlement Module (*SPS)* within the *Settlement* domain.
- In addition, the positive validation result is sent to the *Interface* Domain which forwards an information message *Accepted Liquidity Transfer*, to the T2S Actor (according to the Message Subscription preferences).

The *SPS* Module processes the *Liquidity information* and makes the resulting *Settlement transaction* available for the *Daytime Validation, Provisioning and Booking* Module, which validates whether the booking is feasible. Three different scenarios are possible: full, partial and no execution:

Alternative 2 - Execution: No execution

- In case there is either no cash available (zero balance) on the *T2S Dedicated Cash Account* to be debited or insufficient cash with partial settlement not allowed, the *Settlement Transaction* could not be settled and no liquidity is transferred at all. In this case, the Settlement Status is updated to "unsettled" in *Settlement* and a *Liquidity transfer booking information* (unsettled) is sent to the *Outbound Information Management* Module which collects all business information and sends it to the *Interface*. A corresponding information message is sent to the initiating T2S Actor by the *Outbound Processing* Module within the *Interface* Domain (depending on the Message Subscription preferences).

Alternative 2 - Execution: Full/partial execution

- Full execution: The available cash on the *T2S Dedicated Cash Account* to be debited is sufficient to execute the Settlement Transaction.

All rights reserved.

- Upon a successful provision check, the *Booking* function within the *Daytime Validation, Provisioning and Booking* Module updates the balances of the *T2S Dedicated Cash Accounts* involved on a gross basis.
- After the successful booking and the update of the settlement status in Settlement, the Liquidity transfer booking information is sent to the *Outbound Information Management* Module within the *Liquidity Management* Domain.
- Afterwards the *Outbound Information Management* stores the settlement timestamp and the actual amount. In addition, it collects all business information and sends it to the *Outbound Processing* Module within the *Interface* domain, which sends an information message to the initiating T2S Actor (depending on the Message Subscription preferences).

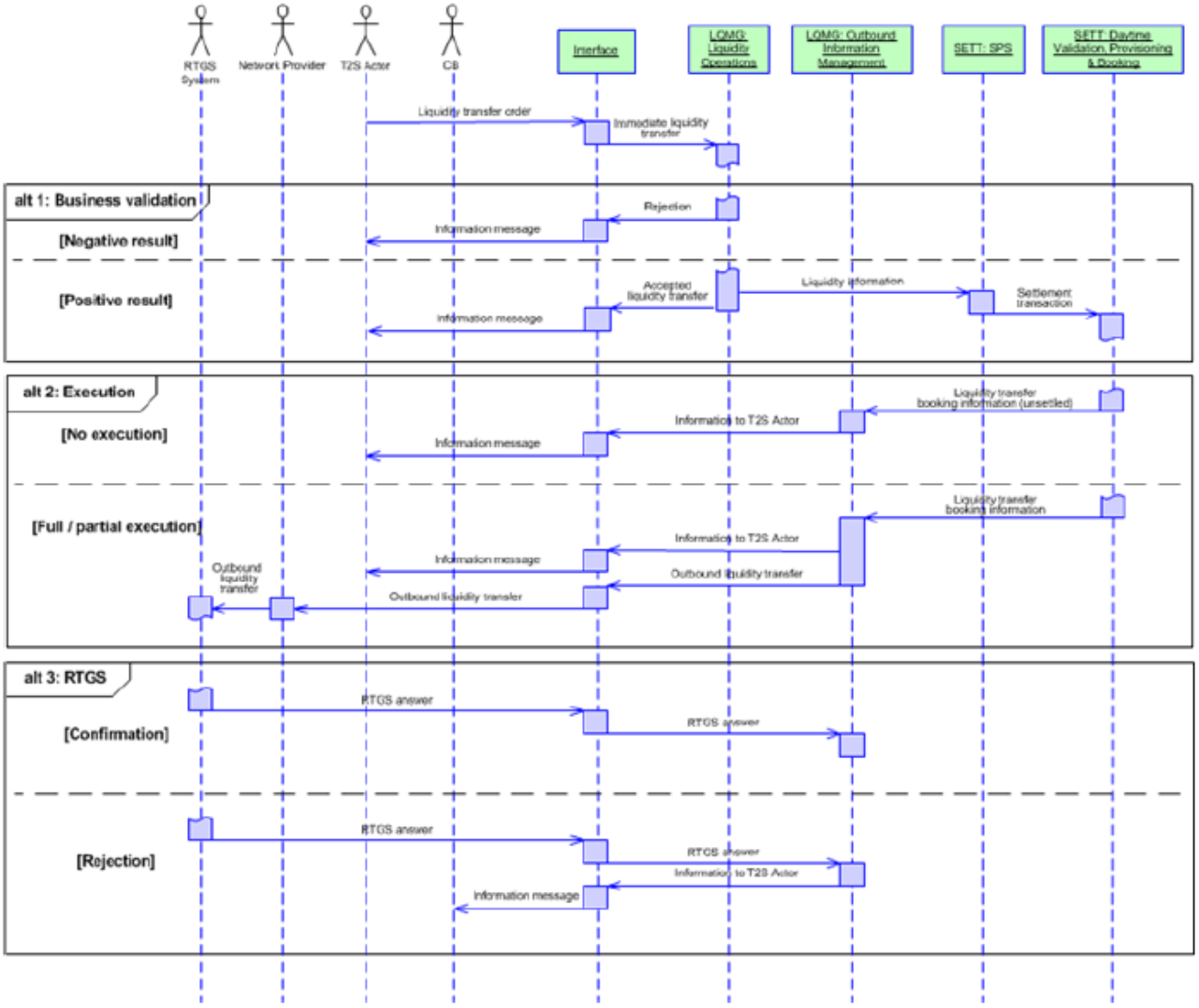
Partial execution: The available cash on the *T2S Dedicated Cash Account* of the Payment Bank is only partly sufficient to execute the *Settlement Transaction*.

- In this case, the cash amount of the *Liquidity Transfer Order* which cannot be settled due to unavailable cash in the *T2S Dedicated Cash Account*, is discarded. No further settlement attempt is done.
- For the available cash the processing steps are the same as described in the previous scenario for full execution. However, only in case of a T2S Actor granted with the respective privilege the attribute "Partial Execution" within the *Liquidity Transfer* is set to "*Yes*" and *Settlement* is aware that this order is subject to partial execution. The status of the *Liquidity Transfer* is set to "*partially settled*" by the *Settlement*.

All rights reserved.

3.6.7.2 Processing of UC-LT-2: Execution of Immediate Liquidity Transfers from T2S to RTGS

DIAGRAM 42 - EXECUTION OF IMMEDIATE LTS FROM T2S TO RTGS



Business assumption

A *T2S Dedicated Cash Account* holder or another T2S Actor granted with the respective privilege sends an Immediate *Liquidity Transfer* to T2S to transfer liquidity from T2S to RTGS.

Please note that all communication between the *Interface* domain and T2S Actors should be understood as:

- "Encrypted inbound communication A2A " or "Encrypted outbound communication A2A" for A2A mode
- "U2A XHTML request via HTTPS" or "U2A XHTML response via HTTPS" for U2A mode

as described in the Interface Domain chapter 3.2. For the sake of readability, more specific terms for these flows are used in the diagram above.

The Execution of Immediate Liquidity Transfers from T2S to RTGS is also applicable during Night time Settlement, but this Use Case covers only the Daytime processing.

Processing

All rights reserved.

After successful technical validations in the *Interface* domain the Immediate *Liquidity Transfer* enters the *Liquidity Operations* Module, which carries out several Business Validations. It is checked if for example:

- the involved *RTGS accounts*, *T2S Dedicated Cash Accounts* are existing and active in the reference data;
- all mandatory fields for *Liquidity Transfers* are filled;
- the currency is eligible as T2S settlement currency and is the same as the currency of the *T2S Dedicated Cash Accounts* involved;
- there is no duplicate submission of the incoming Immediate *Liquidity Transfer*;
- the T2S Actor instructing the *Liquidity Transfer* has sufficient privileges (e.g. based on a contractual agreement) to instruct on the *T2S Dedicated Cash Account* to be debited in case the instructing T2S Actor is not the account owner.

Alternative 1 - Business Validations: Negative Result

In case of a negative result the Immediate *Liquidity Transfer* is rejected and a Rejection including all the negative validation results is sent to the *Interface* domain, which forwards an information message to the T2S Actor.

Alternative 1 - Business Validations: Positive Result

If the validations are confirmed and successful, the *Liquidity Operations* Module creates a *Liquidity Transfer* and forwards the *Liquidity Information* (debit T2S Dedicated Cash Account - credit RTGS Dedicated Transit Account) to the *Standardisation Preparation to Settlement* module SPS) within the *Settlement* domain.

In addition, an *Accepted Liquidity Transfer* is sent to the *Interface* domain which is forwarded to the T2S Actor (according to the Message Subscription preferences).

The *Standardisation Preparation to Settlement* module processes the *Liquidity Information* and makes the resulting *Settlement Transaction* available for the *Daytime* [201] *Validation, Provisioning and Booking* module, which validates whether the booking is feasible. Three different scenarios are possible: full, partial and no execution.

Alternative 2 - Execution: No execution

- In case there is either no cash available (zero balance) on the *T2S Dedicated Cash Account* to be debited or insufficient cash with partial settlement not allowed the *Settlement Transaction* could not be settled and no liquidity is transferred at all. In this case:
  - The Settlement Status is set to "unsettled" by the *Settlement*;
  - A *Liquidity transfer booking information (unsettled)* is sent to the *Outbound Information Management* module;
  - Afterwards the *Outbound Information Management* module collects all business information and sends it to the *Outbound Processing Module* within the *Interface* domain, which then sends an information message to the initiating T2S Actor (depending on the Message Subscription preferences).

201 Immediate Liquidity Transfers from T2S to RTGS are also processed during night time, but this procedure is not part of this Use Case.

All rights reserved.

Alternative 2 - Execution: Full/ partial execution

- Full Execution: The available cash on the *T2S Dedicated Cash Account* to be debited is sufficient to execute the *Settlement Transaction*:
- Upon a successful provision check, the *Settlement Transaction* is sent to the *Booking* function within the *Daytime Validation, Provisioning and Booking* module.
  - The *Booking* function updates the balances of the accounts involved on a gross basis: the requested *T2S Dedicated Cash Account* is debited and the RTGS Dedicated Transit Account of the responsible RTGS system is credited.
  - After the successful booking of the *Settlement Transaction*, the Settlement Status is updated to *"Settled"* and a *Liquidity transfer booking information* is sent to the *Outbound Information Management* module within the *Liquidity Management* domain.
  - Afterwards the *Outbound Information Management* module stores the Settlement Timestamp and the Actual Amount. In addition, it collects all business information and sends it to the *Outbound Processing Module* within the *Interface* domain. The Outbound *Liquidity Transfer* is sent to the involved RTGS System for subsequent booking. An information message is sent to the initiating T2S Actor (depending on the Message Subscription preferences).
- Partial Execution: The available cash on the *T2S Dedicated Cash Account* is only partly sufficient to execute the *Settlement Transaction*:
  - In this case, no further settlement attempt is done for the cash amount which cannot be settled due to unavailable cash in the *T2S Dedicated Cash Account*.
  - The processing steps are the same as described above for full execution. However, only in case of another T2S Actor granted with the respective privilege the attribute Partial Execution of the *Liquidity Transfer* is set to *"Yes"* and Settlement is made aware that this *Liquidity Transfer* is subject to partial execution. In this case, the Settlement Status of the *Liquidity Transfer* is set to *"Partially settled"*.

Alternative 3 - RTGS: Confirmation

- A confirmation from the RTGS System implying that the *Liquidity Transfer* has been booked successfully within the RTGS system, it is forwarded from the *Interface* domain to the *Outbound Information Management* module in the *Liquidity Management* domain. There the RTGS Status of the *Liquidity Transfer* is updated from "*Not Executed"* to *"RTGS Confirmation"* accordingly.

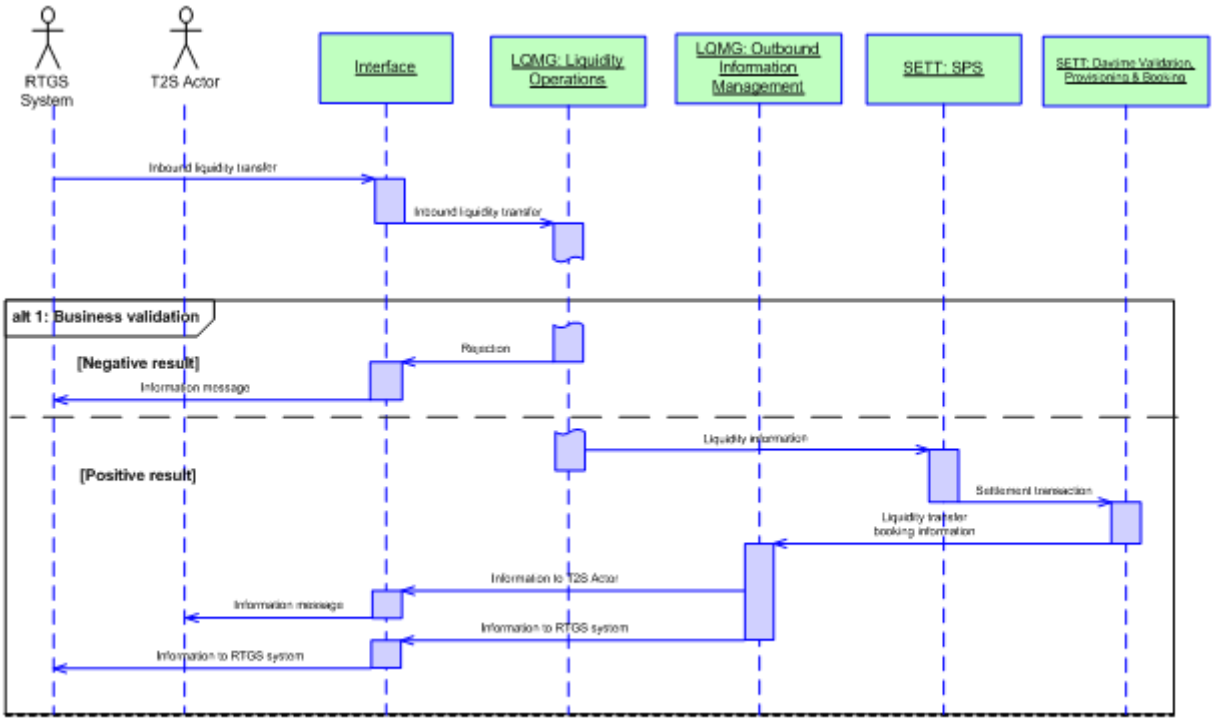
Alternative 3 - RTGS: Rejection

- A rejection from the RTGS System implying that the *Liquidity Transfer* is not processed within the RTGS system, it is forwarded from the *Interface* domain to the *Outbound Information Management* module in the *Liquidity Management* domain. There the RTGS Status of the *Liquidity Transfer* is updated from *"Not executed"* to *"RTGS Rejection"* and an *Information* to T2S Actor is sent back to the *Interface* which forwards it to the respective RTGS (according to the Message Subscription preferences).

All rights reserved.

3.6.7.3 Processing of UC-LT-3: Execution of Liquidity Transfers from RTGS to T2S

DIAGRAM 43 - EXECUTION OF LIQUIDITY TRANSFERS FROM RTGS TO T2S



Business assumption

The RTGS system's participant initiates a *Liquidity Transfer* from its RTGS Account in favour of a *T2S Dedicated Cash Account*. After the booking of the *Liquidity Transfer* within the RTGS System has taken place, regardless of which order type has been initiated by the RTGS System's participant, it must be perceived and processed as an *Inbound Liquidity Transfer* in T2S.

Please note that all communication between the *Interface* domain and T2S Actors should be understood as

- "Encrypted inbound communication A2A" or "Encrypted outbound communication A2A" for A2A mode
- "U2A XHTML request via HTTPS" or "U2A XHTML response via HTTPS" for U2A mode

as described in the Interface chapter 3.2. For the sake of readability, more specific terms for these flows are used in the diagram above.

The execution of Liquidity Transfers from RTGS to T2S is also applicable during Night time Settlement, but this Use Case covers only the Daytime processing.

Processing

Upon successful technical validations in the *Interface*, the *Liquidity Operations* module in the *Liquidity Management* domain carries out several Business Validations. It is checked if for example:

- the instructing party, *T2S Dedicated Cash Accounts* are existing and active in the reference data;
- all mandatory fields for *Liquidity Transfers* are filled;

All rights reserved.

- the currency is eligible as T2S settlement currency and is the same as the currency of the *T2S Dedicated Cash Accounts* involved [202];
- there is no duplicate submission of the incoming *Liquidity Transfer*;
- the RTGS System/ Operator instructing the *Liquidity Transfer* has sufficient privileges to instruct on the RTGS Dedicated Transit Account to be debited.

Alternative 1 - Business Validations: Negative Result

In case of a negative result the *Liquidity Transfer* is rejected and a *Rejection* including all negative validation results is sent to the *Interface* domain, which forwards an information message to the RTGS System.

Alternative 1 - Business Validations: Positive Result

If the validations are confirmed and successful, the *Liquidity Operations* module creates a *Liquidity Transfer* and forwards the *Liquidity Information* (credit *T2S Dedicated Cash Account* - debit RTGS Dedicated Transit Account) to the *Standardisation Preparation to Settlement* module (SPS) within the *Settlement* domain.

The *Standardisation Preparation to Settlement* module processes the *Liquidity Information* and makes the resulting *Settlement Transaction* available for the *Daytime Validation, Provisioning and Booking* module in charge of performing the settlement.

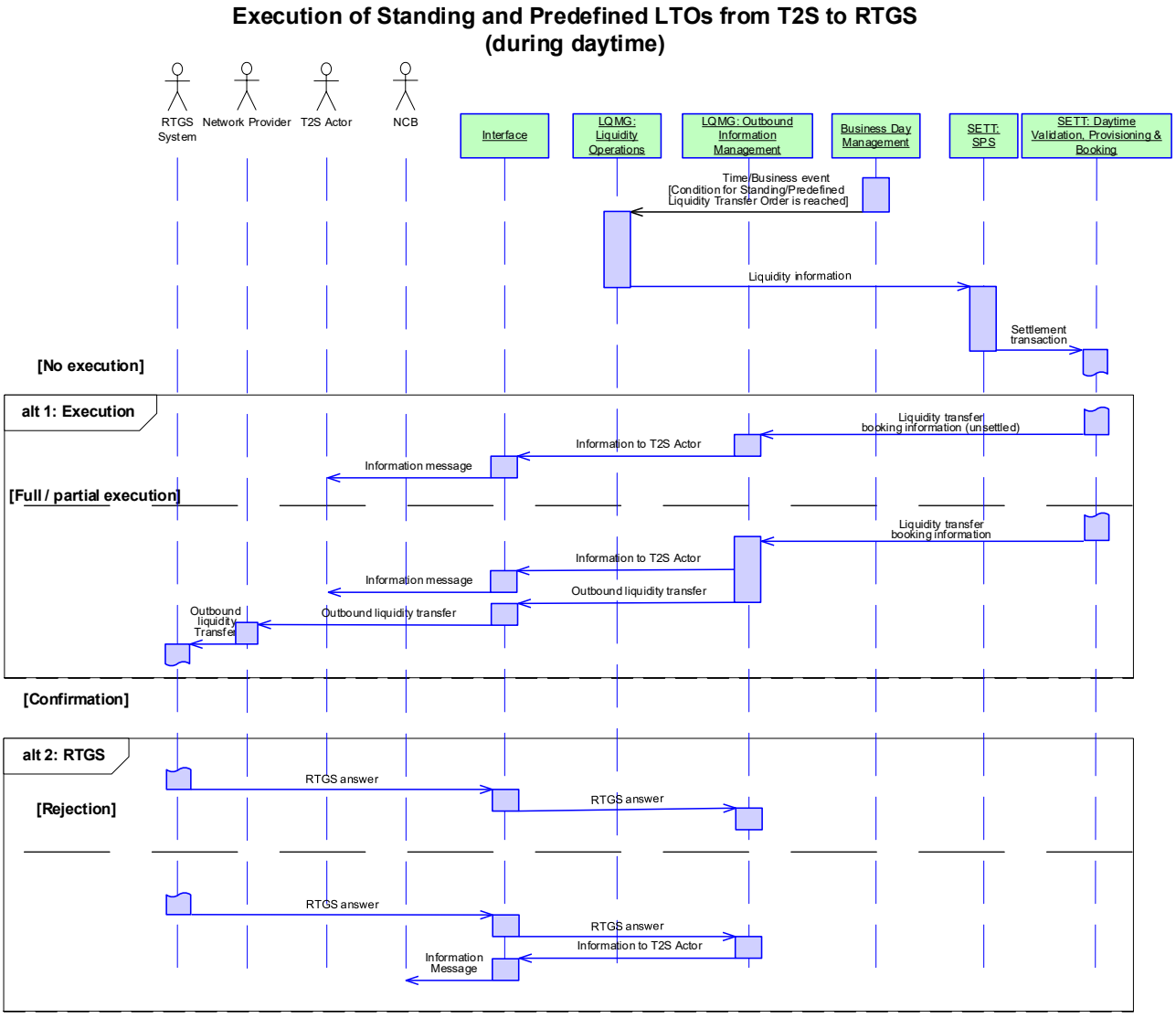
- Upon a successful provision check, the *Liquidity Transfer* is sent via the *Pre-empting* function to the *Booking* function within the *Daytime Validation, Provisioning and Booking* module.
- Then the *Booking* function updates on a gross basis the balances of the accounts involved: the requested RTGS Dedicated Transit Account is debited and the *T2S Dedicated Cash Account* is credited.
- After the successful booking and the update of the Settlement Status of the *Liquidity Transfer*, a *Liquidity Transfer Booking Information* is sent to the *Outbound Information Management* module. On this basis, this module stores the Settlement Timestamp and the Actual Amount. In addition, it collects all business information and sends it to the *Outbound Processing Module* within the *Interface* domain which then sends an Information message to the T2S Actor (depending on the Message Subscription preferences) and an Information to RTGS system.

202 In case the business day change is delayed or T2S has a currency specific closing date incoming Inbound Liquidity Transfers will be parked until the next business day for the specified currency is reached.

All rights reserved.

3.6.7.4 Processing of UC-LT-4: Execution of Standing and Predefined Liquidity Transfer Orders from T2S to RTGS

DIAGRAM 44 - EXECUTION OF STANDING AND PREDEFINED LIQUIDITY TRANSFER ORDERS FROM T2S TO RTGS{CR-720}



Business assumption

A *T2S Dedicated Cash Account* holder or another T2S Actor granted with the respective privilege is able to define Standing and Predefined *Liquidity Transfer Orders* to be executed during the settlement day of T2S. Standing and Predefined *Liquidity Transfer Orders* are maintained and stored in CRDM and are activated by the occurrence of the defined Time/Business event.

Please note that all communication between the *Interface* domain and T2S Actors should be understood as

- "Encrypted inbound communication A2A" or "Encrypted outbound communication A2A" for A2A mode
- "U2A XHTML request via HTTPS" or "U2A XHTML response via HTTPS" for U2A mode

as described in the Interface chapter 3.2. For the sake of readability, more specific terms for these flows are used in the diagram above.

All rights reserved.

The Execution of Standing and Predefined *Liquidity Transfer Orders* is applicable during Real time Settlement and Night time Settlement.

Processing

After receiving the events from the *Scheduling* module the *Liquidity Operations* module retrieves from the reference data the relevant Standing and Predefined *Liquidity Transfer Orders* foreseen for this point in time.

Afterwards the *Liquidity Operations* module creates a *Liquidity Transfer* and sends the *Liquidity Information* (debit *T2S Dedicated Cash Account* - credit RTGS Dedicated Transit Account) to the *Standardisation Preparation to Settlement* module (SPS) within the *Settlement* domain. The SPS Module processes the *Liquidity information* and makes the resulting Settlement Transaction available for the *Daytime Validation, Provisioning and Booking* module, which validates whether the booking is feasible.

Three different scenarios are possible:

Alternative 1 - Execution: No execution

In case there is either no cash available (zero balance) on the *T2S Dedicated Cash Account* to be debited the *Settlement Transaction* could not be settled and no liquidity is transferred at all. In this case, a *Liquidity transfer booking information* (Settlement Status: unsettled) is sent to the *Outbound Information Management* module. No further processing and no information to T2S Actor.

Alternative 1 - Execution: Full/ partial execution

- Full Execution: The available cash on the *T2S Dedicated Cash Account* to be debited is sufficient to execute the *Settlement Transaction*:
  - Upon a successful provision check, the *Settlement Transaction* is sent via the *Pre-empting* function to the *Booking* function within *the Daytime Validation, Provisioning and Booking* module.
  - Then the *Booking* function updates the balances of the accounts involved on a gross basis: the requested *T2S Dedicated Cash Account* is debited and the RTGS Dedicated Transit Account of the responsible RTGS system is credited.
  - After the successful booking and the update of the settlement status, a *Liquidity Transfer booking information* is sent to the *Outbound Information Management* module within the *Liquidity Management* domain.
  - Afterwards the *Outbound Information Management* module stores the Settlement Timestamp and the Actual Amount. In addition, it collects all business information and sends it to the *Outbound Processing Module* within the *Interface*. The *Outbound Liquidity Transfer* is sent to the involved RTGS System for subsequent booking in the RTGS. An *Information* to T2S Actor is sent to the *Interface*. Then the *Interface* sends an Information Message to the T2S Actor (holder of the debited *T2S Dedicated Cash Account* depending on the Message Subscription preferences).
- Partial Execution: The available cash on the *T2S Dedicated Cash Account* is only partly sufficient to execute the *Settlement Transaction*.

All rights reserved.

– The processing steps are the same as described above for full execution. The attribute Partial Execution within the Liquidity Transfer is set to "*Yes*" and *Settlement* domain is aware that this *Liquidity Transfer* is subject to partial settlement. In this case, the status of the *Liquidity Transfer* is set to "*partially settled*" by the *Settlement* domain and the Settlement timestamp and Actual amount are stored by the *Liquidity Management*.

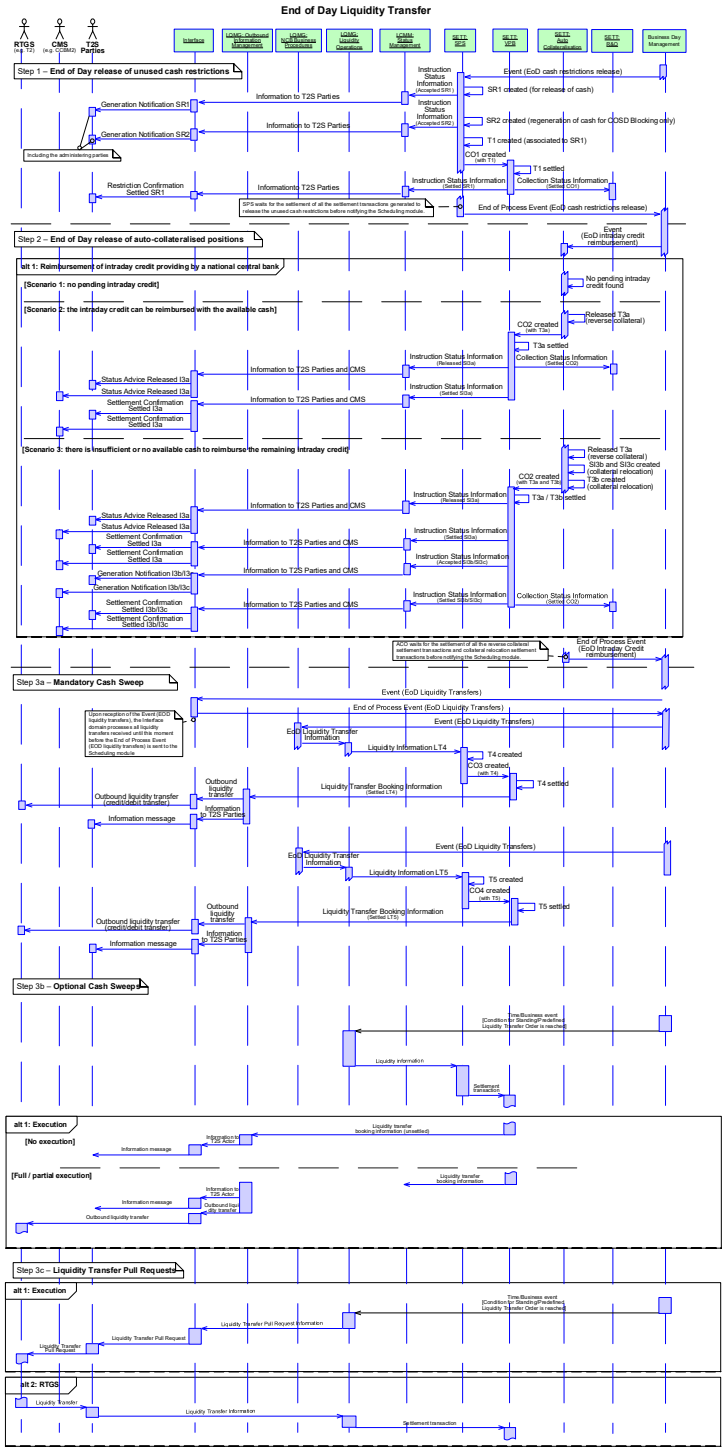
– In this case, no further settlement attempt is done for the cash amount which cannot be settled due to unavailable cash in the *T2S Dedicated Cash Account*.

  - Alternative 2 - RTGS: Confirmation
  - A confirmation from the RTGS System, that the *Liquidity Transfer* has been booked successfully, the RTGS Answer is forwarded from the *Interface* domain to the *Outbound Information Management* module in the *Liquidity Management* domain where the Settlement Status of the *Liquidity Transfer* is updated from "*Not Executed*" to "*RTGS Confirmation*".
  - Alternative 2 - RTGS: Rejection
  - A rejection from the RTGS System, the RTGS answer is forwarded from the *Interface* domain to the *Outbound Information Management* module in the *Liquidity Management* domain where the RTGS Status of the *Liquidity Transfer* is updated from "*Not Executed*" to "*RTGS Rejection*" and an *Information* to T2S Actor is sent back to the *Interface.* Then the *Interface* sends an *Information* message to the respective RTGS (according to the Message Subscription preferences).

All rights reserved.

3.6.7.5 Processing of UC-LT-5: End-of-Day Liquidity transfer

DIAGRAM 45 - END-OF-DAY LIQUIDITY TRANSFER{CR-720}



This use case relates to the End of Day processing of a T2S settlement day.

After the cut-offs processes (i.e. cut-offs related to Settlement Restrictions, FOP or BATM operations), the *Scheduling* module sends an Event (EOD cash restriction release) to the *Standardisation and Preparation to Settlement* module to initiate the End-of-Day processing.

Processing

All rights reserved.

At the end of the settlement day, the situation of the *T2S Dedicated Cash Account* could result in the following three different scenarios:

- 1 - No intraday credit was provided by a CB during the settlement day;
- 2 - The cash on the *T2S Dedicated Cash Account* is sufficient to fully reimburse the pending intraday credit provided by the CB;
- 3 - There is insufficient or no cash on the *T2S Dedicated Cash Account* to reimburse the pending intraday credit provided by the CB.

This use case covers only the case of pending intraday credit when the accounts are owned by the same settlement bank and managed by the same CB in T2S and the RTGS System. No Cross CB scenario is covered [203].

The End of Day processing is organised in three steps:

- Step 1 – End of Day release of unused cash restrictions;
- Step 2 – End of Day release of auto-collateralised positions;
- Step 3 – End of Day liquidity transfer (i.e. cash sweep).

Step 1 - End of Day release of unused cash restrictions

At the receipt of the Event (*EOD cash restriction release*) from the *Scheduling* module, the *EOD/SOD Processing* function of the *Standardisation and Preparation to Settlement* module selects the *Detailed Cash Restrictions* which have not been used.

For each selected *Detailed Cash Restriction*, the *EOD/SOD Processing* function:

- Creates a *Settlement Restriction* SR1 to release this *Detailed Cash Restriction* for the current settlement day;
- Only for released COSD Blocking, creates a *Settlement Restriction* SR2 to regenerate the released *COSD Blocking*, with an intended settlement date set to the next settlement day;
- Sends the *Instruction Status Information* related to the *Settlement Restrictions* created SR1 and SR2 to the *Status Management* module;
- Sends the *Settlement Restriction* SR1 created to the *Settlement Transaction Manager* function of the *Standardisation and Preparation to Settlement* module.

Upon receipt of the *Instruction Status Information*, the *Status Management* module sends to the *Interface* domain the necessary data for the information of the involved T2S Parties, including the administering parties in case of CoSD blocking, (depending on their message subscriptions) about the creation of the *Settlement Restrictions* SR1 and SR2.

Then, the *Settlement Transaction Manager* function:

203 The Cross CB scenario will be described later on when exchanges between T2S and Collateral management Systems will be fully clarified

All rights reserved.

- Creates a *Settlement Transaction* T1 associated to SR1 [204] with the Settlement Transaction Category set to *"Restriction Management"*;
- Sends T1 in a *Collection* CO1 to the *Daytime Validation, Provisioning and Booking* module for a settlement attempt.

Upon successful settlement of the *Settlement Transaction* T1, the *Daytime Validation, Provisioning and Booking* module sends:

- An *Instruction Status* Information to the Status Management module;
- A *Collection Status* Information to the Daytime Recycling and Optimisation module.

Upon receipt of the *Instruction Status Information*, the *Status Management* module sends to the *Interface* domain the necessary data for the information of the involved T2S Parties (depending on their message subscriptions) about the settlement of the *Settlement Restriction* SR1.

The *EOD/SOD Processing* function of the *Auto-collateralisation* module checks that all generated cash restriction release *Settlement Transactions* have been settled and notifies the *Scheduling* module with the *End of Process Event (EOD cash restrictions release)*.

Step 2 - End of Day release of auto-collateralised positions

At the receipt of the *Event (EOD Intraday Credit Reimbursement)* from the *Scheduling* module, the *Auto-collateralisation* module determines the applicable scenario by:

- Searching for each intraday credit provided by a CB the reverse collateral *Settlement Transactions* (and associated *Settlement Instructions*) already created;
- Checking the available cash to settle the pending reverse collateral *Settlement Transactions* (including potential rebalancing between the *T2S Dedicated Cash Account* of the involved T2S Party).

Alternative 1 – Reimbursement of intraday credit provided by a CB- Scenario 1: no pending intraday credit

In this scenario, the *Auto-collateralisation* module does not trigger any specific action for the considered *T2S Dedicated Cash Account*.

Alternative 1 – Reimbursement of intraday credit provided by a CB - Scenario 2: intraday credit can be fully reimbursed with the cash available (including potential cash rebalancing from another T2S DCA of the T2S Party)

In this scenario, *Auto-collateralisation* module releases the reverse *Settlement Transaction* T3a and sends it in a collection CO2 to the *Daytime Validation, Provisioning and Booking* module.

Upon successful settlement, the *Daytime Validation, Provisioning and Booking* module:

- Sets to *"Settled"* the Settlement Transaction Status of T3a and the Settlement Status of its associated reverse collateral *Settlement Instructions*;
- Sends a *Collection Status Information* to the *Daytime Recycling and Optimisation* module.

204 The *Settlement Transaction* associated to SR2 will be created on the next settlement day when the *Standardisation and Preparation to Settlement*-module receives the SR2 from the *Lifecycle Management and Matching* domain.

All rights reserved.

Upon receipt of the *Instruction Status Information*, the *Status Management* module sends to the *Interface* domain the necessary data for the information of the involved T2S Parties and CMS (depending on their message subscriptions) about the settlement of the reverse collateral *Settlement Instruction* SI3a.

Alternative 1 – Reimbursement of intraday credit provided by a CB - Scenario 3: there is insufficient or no available cash to reimburse the remaining intraday credit

In this scenario, for each *T2S Dedicated Cash Account* with insufficient or no available cash for reimbursement (including the fact that potential cash rebalancing from another T2S DCA of the T2S Party is not possible since it does not have sufficient liquidity), the *Auto-collateralisation* module:

- Releases the associated reverse collateral *Settlement Transaction* T3a corresponding to the release of the collateral (transfer of the collateralised securities from the CB auto-collateralisation securities account to the *Securities Account* of the settlement bank) and cash (from T2S DCA of the T2S party which initially received the intraday credit to the CB central bank cash account);
- For the amount of cash needed to reimburse the pending intraday credit, creates DVP *Settlement Transaction* T3b (and associated *Settlement Instructions* SI3b and SI3c with their related *Matching Object*) corresponding to the relocation of collateral (transfer of the collateralised securities from the *Securities Account* of the settlement bank to the CB regular collateral securities account) and cash (from CB central bank cash account to the relevant T2S DCA of the T2S party);
- Sends the *Instruction Status Information* related to the *Settlement Instructions* created SI3b and SI3c to the *Status Management* module;
- Creates a *Collection* CO2 with the *Settlement Transactions* T3a and T3b and sends the collection CO2 to the *Daytime Validation, Provisioning and Booking* module for settlement.

Upon receipt of the *Instruction Status Information*, the *Status Management* module sends to the *Interface* domain the necessary data for the information of the involved T2S Parties and CMS (depending on their message subscriptions) about the creation of the *Settlement Instructions* SI3b and SI3c.

Upon successful settlement, the *Daytime Validation, Provisioning and Booking* module:

- Sets to *"Settled"* the T3a and T3b Settlement Transaction Status and the Settlement Status of the associated reverse collateral *Settlement Instructions*;
- Sends a *Collection Status Information* to the *Daytime Recycling and Optimisation* module.

Upon receipt of the *Instruction Status Information*, the *Status Management* module sends to the *Interface* domain the necessary data for the information of the involved T2S Parties and CMS [205] (depending on their message subscriptions) about the settlement of the *Settlement Instructions* SI3a, SI3b and SI3c.

At the end of the process, the *Auto-collateralisation* module checks that all reimbursement *Settlement Transactions* have been settled and notifies the *Scheduling* module with the *End of Process Event (EOD Intraday Credit reimbursement)*.

Step 3 - End of Day liquidity transfer (i.e. cash sweep)

Step 3a: Mandatory Cash Sweep

205 On the receipt of this notification, the CMS then handles the necessary action in the RTGS system to update the credit line.

All rights reserved.

After receiving the *Event (EOD Liquidity Transfers)* from the ~~**Scheduling module**~~ *Business Day Management*, the *CB Business Procedures* module checks the cash balances of all *T2S Dedicated Cash Accounts*.

For all *T2S Dedicated Cash Accounts* (except those with a balance equal zero), the *CB Business Procedure* module sends the *EOD data* (transfer amount, T2S Dedicated Cash Account, currency, RTGS Account) to *Liquidity Operations* module to reset the accounts to zero. An Outbound *Liquidity Transfer* LT4 is created and a *Liquidity Information* is sent to the *Standardisation and Preparation to Settlement* module.

At the receipt of the *Liquidity Information*, the *Standardisation and Preparation to Settlement* module creates the associated *Settlement Transaction* T4 with Settlement Transaction Category *"Liquidity Transfer"*.

Then, the *Standardisation and Preparation to Settlement* module sends the *Settlement Transaction* T4 in a *Collection* CO3 to the *Daytime Validation, Provisioning and Booking* module.

Upon successful settlement, the *Daytime Validation, Provisioning and Booking* module:

- Sets to *"Settled"* the Settlement Transaction Status of T4 and the Settlement Status of the *Liquidity Transfer* LT4;
- Sends a *Collection Status Information* to the *Daytime Recycling and Optimisation* module;
- Sends a *Liquidity Transfer Booking Information* to the *Outbound Information Management* module.

On this basis, the Outbound *Liquidity Transfer* is forwarded to the *Interface* and further on sent to the RTGS System (as Credit or Debit Transfer [206]), as described in LT-UC-4. On the same basis an *Information message* is sent to the T2S Actors (depending on their Message Subscription preferences).

Thereafter the *NCB Business Procedures* module receives the *Event (EOD Liquidity Transfers)* and checks the cash balances of all *T2S Dedicated Cash Accounts*.

After all relevant bookings on the *T2S Dedicated Cash Accounts* are processed, the accounts have a cash balance of zero.

Step 3b: Optional Cash Sweeps

The optional cash sweeps will zero out positive account balances and are configured according to the needs of the T2S Actor in CRDM as a Standing or Predefined Liquidity Transfer Order.

After receiving the *Event (EoD Liquidity Transfers - optional cash sweep)* **or the** *Event (Second EoD Liquidity Transfers – optional)* **from** *Business Day Management*, **the** *Liquidity Operations* **module is notified. It creates an Outbound** *Liquidity Transfer* **LT4 to set any positive balance to zero and a** *Liquidity Information* **is sent to the** *Standardisation and Preparation to Settlement* **module.**

Then, the *Standardisation and Preparation to Settlement* **module sends the** *Settlement Transaction* **T4 in a Collection CO3 to the** *Daytime Validation, Provisioning and Booking* **module.**

Upon successful settlement, the *Daytime Validation, Provisioning and Booking* **module:**

- **Sets to "Settled" the Settlement Transaction Status of T4 and the Settlement Status of the** *Liquidity Transfer* **LT4;**
- **Sends a** *Collection Status Information* **to the** *Daytime Recycling and Optimisation* **module;**

206 A liquidity debit transfer can happen for the case where a central bank cash account has a negative balance and this negative balance has to be swept to its linked RTGS account in the RTGS system.

All rights reserved.

- **Sends a** *Liquidity Transfer Booking Information* to the *Outbound Information Management* **module.**

On this basis, the *Outbound Liquidity Transfer* **is forwarded to the Interface and further on sent to the RTGS System (as Credit Transfer), as described in LT-UC-4. On the same basis, an** *Information message* **is sent to the T2S Actors (depending on their Message Subscription preferences).**

Thereafter the *Liquidity Operations* **module receives an RTGS Response as described in LT-UC-4 and processes it accordingly.**

After all relevant bookings on the *T2S Dedicated Cash Accounts* **are successfully processed, the accounts have a cash balance of zero.**

Step 3c: Liquidity Transfer Pull Requests

The Liquidity Transfer Pull Requests will zero out negative account balances and are configured according to the needs of the T2S Actor in CRDM as a *Standing or Predefined Liquidity Transfer Order*.

After receiving the *Event (Second EoD Liquidity Transfers – optional)* **from** *Business Day Management*, **the** *Liquidity Operations* **module sends a** *Liquidity Transfer Pull Request Information* **to the** *Outbound Information Management* **module to zero out negative balances.**

On this basis, the *Liquidity Transfer Pull Request* **is created by the** *Outbound Information* **Management module to be forwarded to the** *Interface* **and further on sent to the RTGS System.**

After successful processing in the RTGS System, the Interface receives an *inbound Liquidity Transfer* **which will be processed as described in UC-LT-3.**

After all relevant bookings on the *T2S Dedicated Cash Accounts* **are successfully processed, the accounts have a cash balance of zero.{CR-720}**

3.7 Statistics, Queries, Reports and Legal Archiving

3.7.1 General Introduction

The *Statistics, queries, reports and archives domain* provides a number of functionalities to the T2S System User. It is split up in four modules:

- *Statistical Information;*
- *Query management;*
- *Report management;*
- *Legal Archiving.*

Statistical Information

The *Statistical Information* module provides T2S system users with tools for statistical analysis as a support for decision-making. It provides the T2S operator and the CSDs with statistics oriented to the management of the different components of T2S. On an optional basis, it offers statistics for analysis and regulatory reporting to CSDs, CBs and authorised directly connected parties.

All rights reserved.

The functions of this module provide the capabilities to perform basic statistical query and reporting activities as well as more sophisticated multi-dimensional analysis. Functions for loading and managing the relevant statistical data stores are also available in this module.

Query Management

The architectural concept of T2S foresees an operational database (OBD) and an informational database (IDB) in order to provide high performance T2S processes. The IDB is periodically synchronised with the ODB and provides information for the Query and Report Management module.

The *Query Management* module allows different categories of pre-defined real-time and historical queries on the production data. All User Queries are available for all CSDs in T2S, directly connected settlement banks and payment banks (according to access rights) and CBs. User Queries are processed in real time, based on the latest available data. Due to the architectural concept with ODB and IDB for SETT domain (chapter 3.5 [▸ 370]) and the underlying synchronisation process it is possible, that query responses concerning account balances do not always return the very latest information available in the ODB, but the very latest information available in the IDB. Query responses returning information from other domains are not affected by such synchronisation processes. The response to a User Query always contains the timestamp specifying the T2S system time when the response was created.

Report Management

The architectural concept of T2S foresees an operational database (OBD) and an informational database (IDB) in order to provide high performance T2S processes. The IDB is periodically synchronised with the ODB and provides information for the Query and Report Management module.

T2S provides T2S actors with a number of pre-defined reports for periodical information on the production data but the *Report Management* module does not cover any regulatory reporting. Reports are available for all T2S Actors. They can be provided indirectly through the CSDs/ CBs or sent directly to the connected parties. A report may contain information on individual accounts or on a set of accounts.

The creation of T2S reports can be triggered either by a business event (e.g. end-of-day) or a time event (at a fixed time). The reports are stored until the next generation is created which replaces the report. Due to the architectural concept with ODB and IDB (please refer to chapter 3.5.1 [▸ 370]) and the underlying synchronisation process it is possible, that reports concerning account balances do not always return the very latest information available in the ODB.

Legal Archiving

The *Legal Archiving* module addresses needs for a central archive for audit and regulatory requirements made available to T2S parties via their CSDs or CBs. It stores processed inbound and outbound messages in their original format for a given period, and retrieves these archived data on request.

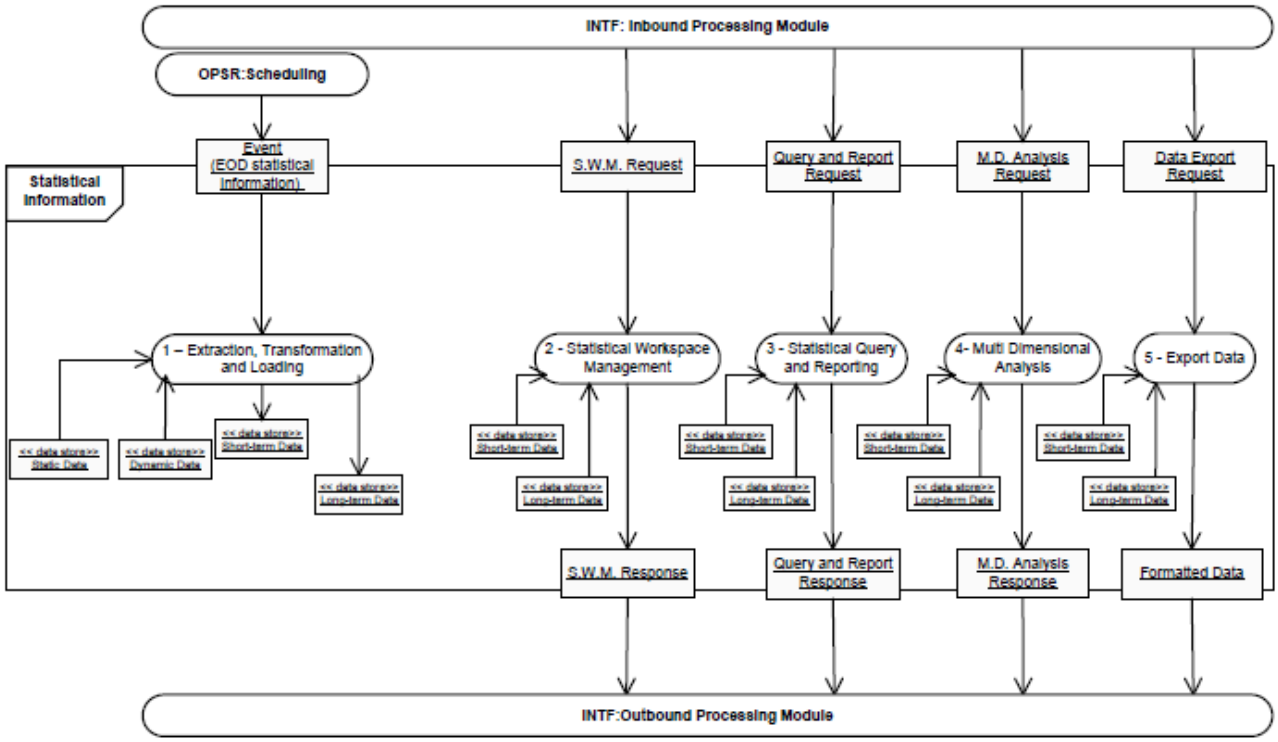
The T2S operator submits requests on behalf of CSDs and CBs to the *Legal Archiving* module. Archived data are made available through restitution files within a maximum time-frame of three days, in compliance with the retrieval requirements.

All rights reserved.

Reference and transactional data as of the last business day with data revision and history stand in the DWH. As a consequence, the Legal Archiving module does not address neither the needs for Data Revision (replication of reference data structures as revision tables) and Data History (data stored with "valid from" and "valid to" date attributes) nor the requirements regarding archiving on behalf of CSDs.

3.7.2 Statistical Information

3.7.2.1 Diagram of the module



3.7.2.2 Description of the module

The *Statistical Information* module is expected to provide T2S system users (i.e. the T2S operator and, on an optional basis, CSDs, CBs, and authorised directly connected parties) with business intelligence tools to be used for statistical analysis and as a decision support system.

It stores information on accounts (including position changes and event information), on instructions and on queries and reports (including volumes generated). [207]

The scope of this module is twofold:

- To provide statistics to the T2S operator and the CSDs on the level of use of the different components of T2S over time, to support the operational management of the system. Such statistics are based on a "short term" statistical information (STSI) repository including data up to three months and the whole instruction life history (including all status changes and relevant timestamps) [208].

207 The latter is not applicable for DWH.

208 This period is meant as three months after the day the concerned data item (e.g. a reference data object, a settlement instruction and so forth) has reached its final status (e.g. settled, cancelled, deleted).

All rights reserved.

- To offer to CSDs, CBs, T2S operator and possibly authorised directly connected parties historical data and statistics for regulatory reporting purposes and, on an optional basis, wider scope statistical analysis. These statistics are based on a repository storing detailed business data with their final status. These statistics are available on T2S static and transactional data as of the last business day, and for a 10-year period, which shall be configurable. The DWH stores settlements having reached the "final" status (e.g. for settlements: only for "cancelled" or "settled" statuses), based on a settlement-related archive. It gives users friendly access **{T2S.17.070} {T2S.17.080} {T2S.17.090}**.

Both repositories are separated from the live data repositories, in order to provide an easy access to high quality and business oriented data without the risk of impacting the performance of the operational settlement environment. They are based on a business intelligence data model, which relies on the data model of the T2S production environment, but it is more appropriate for statistical queries and multi-dimensional analysis.

In the present document, one scope (short term) is covered by the single *Statistical Information* module. Nevertheless, this general description will be followed, during the next phase of the project, by detailed specifications aiming at serving as a basis of the design of the technical (development and infrastructure) solution(s): (i) one covering the access to historical data and statistics needed for the monitoring and management of T2S by the T2S operator and the CSDs and (ii) the second covering access to historical data and statistics for regulatory reporting purposes by CSDs and CBs, T2S operator and possibly authorised directly connected parties and, on an optional basis, wider scope statistical analysis.

The *Statistical Information* module provides functions to manage historical and statistical data repositories, in order to provide access to these data and to perform historical and statistical query and reporting and multi-dimensional analysis:

- *Extraction, Transformation and Loading* (function 1) loads and manages the creation and update of statistical data repositories with T2S static and dynamic data.
- *Statistical Workspace Management* (function 2) creates and manages workspaces for each user profile, i.e. basic and advanced statistical users [209];
- *Statistical Query and Reporting* (function 3) retrieves relevant data, transforms them into the appropriate context, and displays them in a readable format. In this context, T2S system users just issue queries or run reports, selecting them from a pre-determined set of available statistical queries and reports;
- *Multidimensional Analysis* (function 4) enables T2S system users to extend the capabilities of statistical query and reporting. That is, rather than submitting queries or running reports, data are represented as business-oriented objects (e.g. settlement instruction, party) and they are categorised by different dimensions (e.g. instruction type, party type) that the user can browse to perform more complex statistical analysis.
- *Export Data* (function 5) enables T2S system users to export queries and analysis results into several possible formats (e.g. XML, .xls, .pdf, .csv and so forth).

The "short term" data repository is based on the following components:

209 See later on in this section for a definition of statistical workspace and of the available user profiles.

All rights reserved.

- Detailed Data Repository, including all "atomic" data of interest for statistical analysis for the specific category of T2S system users performing such analysis and with the same level of granularity of the operational data.
- Aggregated Data Repository, including all the data of interest for statistical analysis for the specific category of the T2S system users performing such analysis and with a lower level of granularity (obtained by means of detailed data aggregation) compared to the operational data. Aggregated data are built on the basis of detailed data. Starting from a given piece of aggregated information, it is possible to use a drill-down functionality to retrieve the underlying detailed data.
- T2S System User Data Repositories, a set of repositories each of which including detailed and aggregated data of interest for a specific category of T2S system users only (i.e. T2S operator, CSDs, CBs).
- Statistical Workspaces Repository, defining the relevant set of business-oriented objects to be used for statistical analysis (e.g. securities account, DVP transaction, cash amount and so forth) and their mapping to the physical objects available in the repositories (i.e. tables, columns and so on). It also includes the definition of statistical queries and reports that authorised users create on the basis of the available business-oriented objects. Each category of T2S system users can be assigned one or more statistical workspaces.

These repositories provide a description of the different data components managed by the *Statistical Information* module from a logical viewpoint. This description does not mandate any specific type of physical implementation. This means, for example, that the T2S system users data repositories may eventually be implemented just as a logical subset (view) of the same physical data base. Regardless the actual implementation of the repositories, the same concept of data segregation based on system entities (see section 3.3.6 [▸ 123]) must apply in terms of data ownership.

The specific set of available statistical data and functions depends on the specific access profile of each T2S system user. As to the set of accessible data, the T2S operator can access all the data needed to perform its business and technical monitoring of the overall T2S, while CSDs and CBs would access their own data and the data of their respective participants.

For instance, the T2S operator has a set of indicators at his disposal to determine possible performance issues within T2S (e.g. abnormal transit time). At the same time, system users of the CSDs and the CBs have the possibility to retrieve statistical information such as liquidity trends and the number of settled transactions, in order to detect any kind of problems or needs on the business side.

With respect to the set of usable functions, three main profiles are envisaged:

- Basic Statistical User: this is the T2S system user performing statistical query and reporting. Typically, they can simply view a set of standard statistical queries and reports run automatically and periodically or run a pre-defined set of statistical queries and reports with the possibility to input a specific set of search criteria. They cannot create new statistical queries or reports autonomously;
- Advanced Statistical User: this is the T2S system user performing multi-dimensional analysis. Advanced statistical users analyse their data via the relevant workspace defined by the statistical workspace administrator. They can also create new statistical queries and reports and make them available to the basic statistical users (or to a specific sub-set of them);

All rights reserved.

- Statistical Workspace Administrator: this T2S system user is in charge of defining the statistical workspaces for the advanced and basic statistical users, i.e. the set of business-oriented objects they use for their statistical analysis. The statistical workspace administrator is a T2S system user of the T2S operator having the necessary knowledge both of the physical data base structure and of the underlying business concepts, making it possible to define the mapping mentioned in the description of the statistical workspace repository. The statistical workspace administrator performs this task in User-to-Application mode.

Each T2S system user can be linked to one or many profiles.

3.7.2.3 Description of the functions of the module

1 – Extraction, Transformation and Loading

Reference Id	***SQRA.STI.ETL.1.1***

This function manages a three step process consisting, at each step, in extracting data from a source repository, transforming them to fit the relevant business needs and loading them in a destination repository **{T2S.15.010}**.

More in details:

- In the first step, the function extracts raw data from the operational database, transforms them and finally loads them into the detailed short-term and long-term data repositories. At this stage, data transformations are mainly data type conversions or functions applied to a single attribute or a set of attributes at row level, without any changes in the level of granularity (i.e. aggregation) compared to the operational data. They are required only to prevent possible performance issues (e.g. pre-calculating a numeric attribute at row level) and they should avoid any unnecessary data redundancy;
- During the second step, data are extracted from the detailed data repositories, transformed and finally loaded into the aggregated data repositories. At this stage, data transformations are typically aggregations and summations, with a decreasing level of granularity compared to the operational data;
- In the final step, the function extracts data from the detailed and aggregated data repositories, transforms them and finally loads them into the relevant T2S system user repositories.

The E*xtraction, Transformation and Loading* function is triggered at the end of day, by an EOD Statistical Information event, to load on a daily basis the relevant data into the "short term repositories.

2 –Statistical Workspace Management

Reference Id	***SQRA.STI.SWM.1.1***

This function allows statistical workspace administrators to create and manage workspaces for each user profile, i.e. basic and advanced statistical users.

Each statistical workspace can be described as a set of business-oriented objects defined by means of a set of quantitative and qualitative variables. For instance, an object called securities transaction might be defined as a set of qualitative variables such as transaction type (i.e. DVP, FOP, etc.), ISIN, CSD participant

All rights reserved.

and period (according to the time hierarchy year, quarter and month) and quantitative variables such as securities amount and cash amount. Moreover, multiple objects might be linked by specific relationships (e.g., securities transaction may be linked to securities account and T2S dedicated cash account).

Statistical workspace management allows statistical workspace administrators to maintain both workspaces and business-oriented objects and to make them available to basic and advanced statistical users for their analysis. Statistical workspace administrators can perform these tasks using the following sub-functions:

Statistical Workspace Maintenance

This feature enables statistical workspace administrators to maintain statistical workspaces, i.e. to create new workspaces or to update or delete existing workspaces.

Statistical Workspace Assignment

This feature enables statistical workspace administrators to grant advanced statistical users with access to statistical workspaces (i.e. all the business-oriented objects belonging to the granted workspace) and to revoke that access.

Statistical Workspace Access

This feature enables authorised T2S system users (i.e. statistical workspace administrators and authorised advanced statistical users) to access statistical workspaces, i.e. to view information concerning the workspaces and all the business-oriented objects belonging to those workspaces.

Business-Oriented Object Maintenance

This feature enables statistical workspace administrators to maintain business-oriented objects, i.e. to create new objects or to update or delete existing objects. The deletion of an object just refers to the deletion of its definition from the relevant statistical workspace and it does not imply the deletion of its underlying data. So, any statistics can be reproduced, if need be. In any case, the statistical workspace administrator should perform any business-oriented object maintenance operations just on the basis of the needs expressed by basic statistical users (in terms of required statistical queries and reports) and by advanced statistical users (for their multi-dimensional analysis activities).

Business-Oriented Object Assignment

This feature enables statistical workspace administrators to grant advanced statistical users with access to specific business-oriented objects defined in the relevant statistical workspaces (so that it can be used for multi-dimensional analysis purpose) and to revoke that access. It is also possible to grant or revoke access to a specific sub-set of the object's attributes only.

Business-Oriented Object Access

This feature enables authorised T2S system users (i.e. statistical workspace administrators and authorised advanced statistical users) to access business-oriented objects defined in the relevant statistical workspace, i.e. to view information concerning the objects and their relationships with other objects in the same workspace.

Statistical workspace administrators and advanced statistical users can use the relevant features related to *Statistical Workspace Management* in user-to-application mode only.

3 – Statistical Query and Reporting

All rights reserved.

Reference Id	***SQRA.STI.SQR.1.1***

This function allows basic statistical users to perform their query and reporting analysis, i.e. to view a set of standard statistical queries and reports executed automatically and periodically or to run a pre-defined (by the advanced statistical user) set of queries and reports with the possibility to input a specific set of search criteria.

Each basic statistical user is able to view and run only the set of queries and reports for which the advanced statistical user has granted access.

The statistical reports showing time series analysis **{T2S.15.040}** are available both as tables and in several graphic formats (e.g. charts, pie-chart).

Hereafter, the main features for statistical query and reporting are described **{T2S.15.030}**:

Statistical Queries and Reports Browsing

This feature allows the basic statistical user to browse the set of available statistical queries and reports and select one for access or execution.

Statistical Queries and Reports Access

This feature allows the basic statistical users to retrieve the definition of a statistical query or a report.

Statistical Queries and Reports Execution

This feature allows the basic statistical user to run a statistical query or report (possibly entering a set of input parameters), and returns the relevant retrieved data set. This includes both real-time execution of queries and reports, and background processing of queries and reports with a notification at the end of the process itself. In the case of background processing, it is also possible to schedule a periodic execution of queries and reports (e.g. at a pre-defined time of the day), so that the authorised users can just get the result data set when it is available, without the need to run each time these queries and reports and to wait for their execution.

A basic statistical user can use all the features related to *Statistical Query and Reporting* in user-to-application mode only.

4 – Multi Dimensional Analysis

Reference Id	***SQRA.STI.SQR.1.1***

This function allows advanced statistical users to perform their multi-dimensional analysis **{T2S.15.020}**, i.e. to explore data at different levels of detail, which can be determined dynamically. Complex relationships among data can be analysed through an iterative process that includes drilling down to lower levels of detail or rolling up to higher levels of summation and aggregation. Advanced statistical users analyse their data via the relevant statistical workspace defined by the statistical workspace administrator and create new statistical queries and reports, making them available to the basic statistical users (or to a specific sub-set of them) **{T2S.15.040}**.

All rights reserved.

Going into further detail, advanced statistical users can browse, explore and analyse their business-oriented objects via drill-down, roll-up and pivoting functions. Moreover, they can create, update, run and delete statistical queries and report that they can then publish to make them available to selected basic statistical users **{T2S.15.030}**.

Hereafter, the main sub-functions for multi-dimensional analysis are described:

Statistical Queries and Reports Maintenance

This feature enables advanced statistical users to maintain statistical queries and reports, i.e. to create new queries and reports or to update or delete existing queries and reports. When creating a new query or report, advanced statistical users can explore and analyse their business-oriented objects using drill-down, roll-up and pivoting functions and they can also combine different linked objects, in order to create the data set that is retrieved by the query or by the report.

Statistical Queries and Reports Assignment

This feature enables advanced statistical users to grant basic statistical users with access to specific statistical queries and reports (i.e. with the possibility to view and/or run queries and reports), and to revoke that access.

Statistical Queries and Reports Access

This feature enables authorised T2S system users (i.e. advanced statistical users and authorised basic statistical users) to access statistical queries and reports, i.e. to view information concerning the definition of those queries and reports.

An advanced statistical user can use all the features related to *Multi Dimensional Analysis* in user-to-application mode only.

5 – Export Data

Reference Id	***SQRA.STI.SQR.1.1***

This function allows both basic and advanced statistical users to export the result of their analysis (i.e. the result of their statistical queries and reports) in many different formats (e.g. XML, .xls, .pdf, .csv and so forth), for further utilisation with other tools (e.g. MS Office). Users can export these data directly from the GUI, i.e. in user-to-application mode, just saving their result data set according to one of the available formats.

3.7.2.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Event (EOD statistical information)	In		OPSR: Scheduling	
S.W.M. Request	In	Request from authorised T2S system users for creating/managing statistical workspace	INTF: Inbound Processing Module	

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
S.W.M. Response	Out	Statistical workspace creation/management response to be provided to authorised T2S system users		INTF: Outbound Processing Module
Query and Report Request	In	Request from allowed T2S system users for a statistical query/report	INTF: Inbound Processing Module	
Query and Report Response	Out	Statistical query/report results to be provided to authorised T2S system users		INTF: Outbound Processing Module
M.D. Analysis Request	In	Request from authorised T2S system users for performing Multidimensional Analysis	INTF: Inbound Processing Module	
M.D. Analysis Response	Out	Multidimensional Analysis results to be provided to authorised T2S system users		INTF: Outbound Processing Module
Data Export Request	In	Request from authorised T2S system users for exporting query/report results in different formats	INTF: Inbound Processing Module	
Formatted Data	Out	Query/report results exported in different format to authorised T2S system users		INTF: Outbound Processing Module

3.7.2.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
REFERENCE **DATA**			
Reference Data	All	Read	Data extracted from all domains of T2S for statistical purposes
DYNAMIC DATA			

All rights reserved.

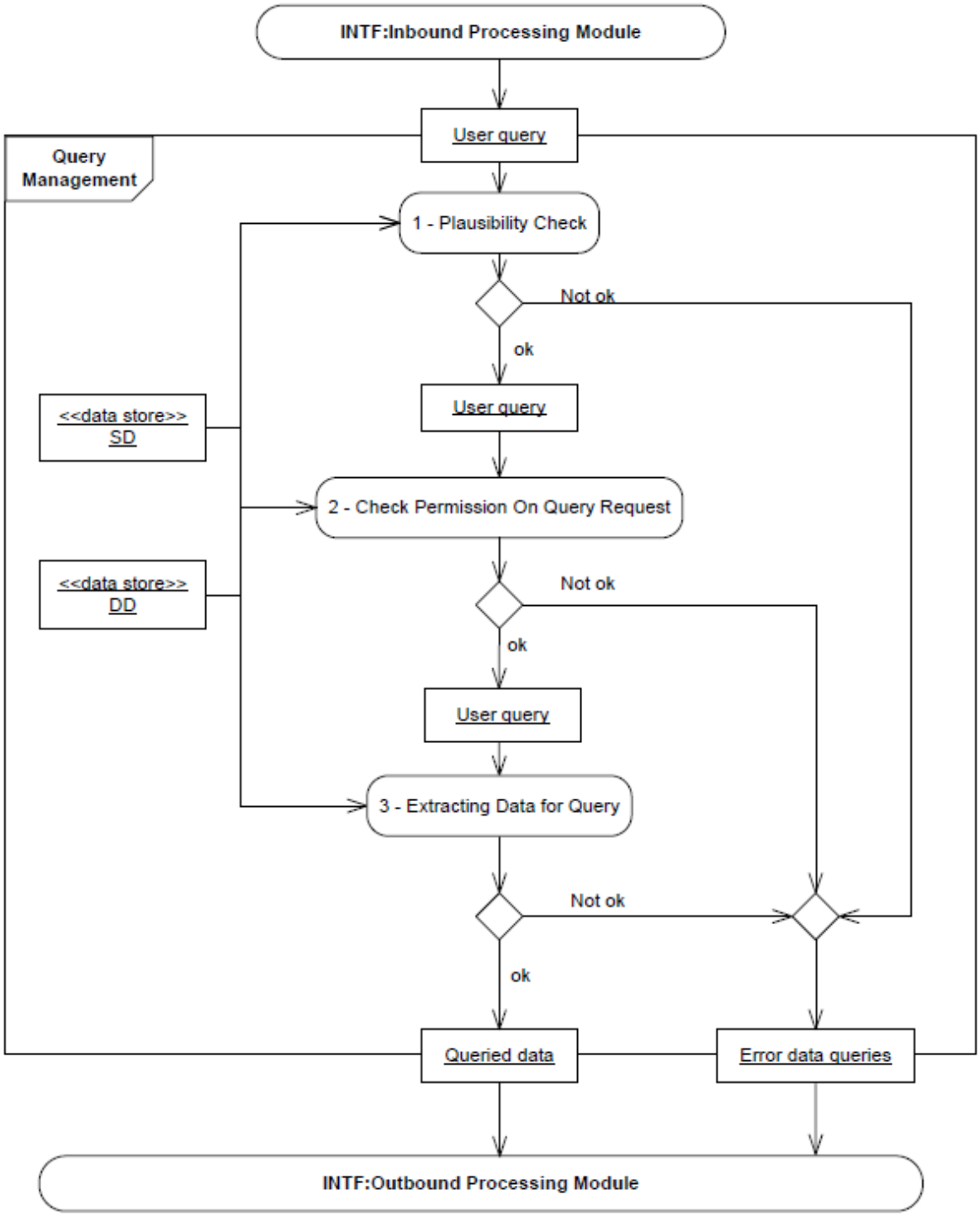
DATA	DATA ENTITIES	ACCESS MODE	COMMENT
Dynamic Data	All	Read	Data extracted from all domains of T2S for statistical purposes
Short-term Data	To be defined	Read/Write	Statistical data up to three months.

All rights reserved.

3.7.3 Query Management

3.7.3.1 Static Functional Description

Diagram of the module



Description of the module

The *Query Management* allows different categories of pre-defined real-time and historical queries on the production data. A T2S System user can query historical data for a period of the last three calendar months. Historical data beyond 3 calendar months can be accessed through the functions provided by the *Statistical Information* module of the domain *Statistical Information, Queries, Reports and Archiving* as described above.

In order to fulfil the User Requirement **{T2S.17.140}**, which demands the processing of 95% of the basic User queries with simple criteria within 3 seconds at maximum, all of the available queries have been categorised accordingly. The results have been derived from an overall assessment based on the selection parameters (their number, cardinality and mandatory/optional use) and on the expected volume of returned data (number of returned fields, number of returned data sets depending on the privileges and roles of the requesting T2S System user in accordance with the hierarchical level of its party).

The following types of queries are available (non-exhaustive list):

QUERY NAME	TYPE OF DATA	CATEGORISATION		URD REFERENCE
		BASIC	COMPLEX	
Securities Settlement Instruction Queries				
Settlement Instruction Query	real-time		x	**{T2S.14.261}** **{T2S.14.262}** **{T2S.14.263}** **{T2S.14.264}**
Settlement Instruction Current Status Query	real-time		x	**{T2S.14.263}** **{T2S.14.264}** **{T2S.14.271}** **{T2S.14.272}**
Settlement Instruction Status Audit Trail Query	real-time		x	**{T2S.14.263}** **{T2S.14.275}** **{T2S.14.277}**
Settlement Instruction Audit Trail Query	historic	x		**{T2S.14.281}** **{T2S.14.282}**
Securities Account Position Queries				
Securities Account Position Query	real-time	x		**{T2S.04.130}** **{T2S.04.160}** **{T2S.14.060}** **{T2S.14.070}** **{T2S.14.080}**

All rights reserved.

QUERY NAME	TYPE OF DATA	CATEGORISATION		URD REFERENCE
		BASIC	COMPLEX	
				{T2S.14.090} **{T2S.14.100}** **{T2S.14.110}** **{T2S.14.120}** **{T2S.14.130}** **{T2S.14.140}** **{T2S.14.145}** **{T2S.14.210}** **{T2S.14.220}** **{T2S.14.230}** **{T2S.14.240}** **{T2S.14.250}** **{T2S.14.960}**
Securities Account Position History Query	historic	x		**{T2S.04.130}** **{T2S.04.160}** **{T2S.14.060}** **{T2S.14.070}** **{T2S.14.080}** **{T2S.14.160}** **{T2S.14.170}** **{T2S.14.190}** **{T2S.14.200}** **{T2S.14.210}** **{T2S.14.215}** **{T2S.14.220}** **{T2S.14.230}** **{T2S.14.240}** **{T2S.14.250}** **{T2S.14.960}**
Cash Related Queries				

All rights reserved.

QUERY NAME	TYPE OF DATA	CATEGORISATION		URD REFERENCE
		BASIC	COMPLEX	
T2S Dedicated Cash Account Balance Query	real-time	x		**{T2S.04.070}** **{T2S.14.080}** **{T2S.14.811}** **{T2S.14.960}**
T2S Dedicated Cash Account Posting Query	historic		x	**{T2S.14.861}**
Immediate Liquidity Transfer Order List Query	real-time		x	**{T2S.14.940}**
Immediate Liquidity Transfer Order Detail Query	real-time	x		**{T2S.14.950}**
Outstanding Auto-Collateralisation Credit Query	real-time	x		**{T2S.14.840}**
T2S Overall Liquidity Query	real-time		x	**{T2S.14.080}** **{T2S.14.860}** **{T2S.14.870}**
Cash Forecast Query	real-time		x	**{T2S.14.890}**
Limit Query	real-time		x	**{T2S.14.930}**
Limit Utilisation Journal Query	historic	x		**{T2S.14.933}**
Limit Utilisation Query	real-time		x	**{T2S.14.935}**
Total collateral value per T2S Dedicated Cash Account Query	real-time		x	**{T2S.14.830}**
Collateral Value per T2S Dedicated Cash Account Query	real-time		x	**{T2S.14.831}**

All rights reserved.

QUERY NAME	TYPE OF DATA	CATEGORISATION		URD REFERENCE
		BASIC	COMPLEX	
Collateral Value of a Security Query	real-time	x		**{T2S.14.832}**
Reference Data Queries				
Data Changes Queries	real-time		x	
Reference Data Audit Trail Query	real-time		x	**{T2S.14.530}**
Securities Reference Data Query	real-time		x	**{T2S.14.540}** **{T2S.16.380}**
ISIN List Query	real-time		x	**{T2S.14.550}**
Securities Deviating Nominal Query	real-time		x	**{T2S.14.553}**
Securities CSD Link Query	real-time		x	**{T2S.14.557}**
Party Reference Data Query	real-time		x	**{T2S.14.560}**
Party List Query	real-time	x		**{T2S.14.563}**
Restricted Party Query	real-time	x		**{T2S.14.567}**
Securities Account Reference Data Query	real-time		x	**{T2S.14.600}**
Securities Account List Query	real-time		x	**{T2S.14.605}**
Cash Account Reference Data Query	real-time		x	**{T2S.14.640}**
T2S Dedicated Cash Account List Query	real-time	x		**{T2S.14.650}**
Liquidity Transfer Order List Query	real-time		x	**{T2S.14.936}**
Liquidity Transfer Order Detail Query	real-time		x	**{T2S.14.937}**

All rights reserved.

QUERY NAME	TYPE OF DATA	CATEGORISATION		URD REFERENCE
		BASIC	COMPLEX	
Total amount of standing and pre-defined orders Query	real-time		x	**{T2S.14.938}**
Liquidity Transfer Order Link Set Query	real-time		x	**{T2S.14.975}**
Liquidity Transfer Order of a Liquid-ity Transfer Order Link Set Query	real-time		x	**{T2S.14.939}**
T2S Calendar Query	real-time	x		**{T2S.14.690}**
T2S Diary Query	real-time	x		**{T2S.14.700}** **{T2S.14.970}**
System Entity Query	real-time	x		**{T2S.14.020}** **{T2S.11.090}** **{T2S.11.100}** **{T2S.14.710}**
Attribute Domain Query	real-time	x		**{T2S.11.280}** **{T2S.14.730}**
Attribute Value Query	real-time	x		**{T2S.11.330}**
Privilege Query	real-time	x		**{T2S.14.020}** **{T2S.14.740}**
Role Query	real-time	x		**{T2S.14.020}** **{T2S.14.760}**
T2S System User Query (T2S Actor Query)	real-time	x		**{T2S.14.770}**
Restriction Query	real-time	x		**{T2S.14.020}** **{T2S.14.800}**
SWIFT BIC Code Query	real-time	x		**{T2S.14.020}** **{T2S.14.565}**

All rights reserved.

QUERY NAME	TYPE OF DATA	CATEGORISATION		URD REFERENCE
		BASIC	COMPLEX	
Report Configuration List Query	real-time	x		
Report Configuration Detail Query	real-time	x		
Dynamic Data Queries				
Report Query	real-time		x	**{T2S.13.190}**
Cumulative Invoice Query	real-time	x		
Itemised Billing Data Query	real-time		x	

The various queries are described in more detail in the latest versions of the User Detailed Functional Specification (UDFS) and User Handbook (UHB) for the Graphical User Interface. Should additional queries be deemed requested in the future, they will be assessed on an ad hoc basis and developed provided that they do not have any material impact on the dynamic data model and on the reference data management included in the present GFS version.

In addition to the queries listed above T2S provides auxiliary queries in order to monitor further business objects.

Please note that the list above does not show the query possibilities which are available to T2S Operators only.

Availability of Queries

- All User Queries are available for all T2S System Users with the appropriate privilege **{T2S.14.030}**. When processing queries, the *Query Management* module takes into account all privileges as defined in *Reference Data* **{T2S.14.060}** **{T2S.14.525}**.

Communication mode and response time

- In general queries shall be available in U2A mode as well as in A2A mode. But there are some exceptions; the following queries and the respective responses (preliminary list) are foreseen to be available in U2A mode only: SWIFT BIC query, System Entity query, Privilege query, Role query, Restriction Query **{T2S.14.020}**. The User queries and their responses are set up as XML messages (compliant with the ISO20022 standards). The use of proprietary messages is avoided **{T2S.14.010}**. The response time within T2S to handle User queries is at maximum 3 seconds for 95 % of the queries (for basic User queries with simple criteria) **{T2S.17.140}**. Whether a query counts as simple or complex is stated in the list given above.

Processing during night-time settlement

All rights reserved.

- All User queries are processed in real time, based on the latest available data **{T2S.14.050}**. An exception from this rule is the processing of securities position and cash balance queries during night-time settlement sequences **{T2S.14.080}**. For further information on queuing of balance queries in the *Interface* please refer to the following functions of the *Interface* domain:
  - *Inbound Processing* Module for U2A: *Check for U2A Availability* function for User queries sent in U2A (Rejection of the query),
  - *Inbound Processing* Module for A2A: *Check for A2A Inbound Queuing* function for User queries sent in A2A (Queuing of the query) **{T2S.14.080}**.

Timeliness of data

- When required, the response to a User query contains a timestamp that specifies the T2S system time when the data selection was actually performed **{T2S.14.250} {T2S.14.830} {T2S.14.831} {T2S.14.832}**. Concerning queries for securities positions and cash balances, please see chapter 3.7.1 [▸ 595] for the architectural concept of OBD/IBD and its effect on Query Management.

Description of the functions of the module

1 – Plausibility Check

Reference Id	***SQRA.QUM.PLA.1.1***

This function performs a plausibility check on the received User query. Using logical checks it identifies zero-result queries (e.g. data range 01/01/08 < date and 01/01/08 > date) or queries containing not plausible characters to reduce unnecessary data store accesses. Furthermore User queries referring to currencies not recognised by T2S or non valid ISINs are blocked. The reference values for the checks are stored in a *Reference Data* data store (e.g. the currencies recognised by T2S or the ISINs of securities allowed for settlement in T2S) or in a *Dynamic Data* data store (e.g. an *Individual settlement instruction* reference).

Any User query failing one of these checks is forwarded as the *Error data queries* to the *Interface*. Otherwise the User query is forwarded to the function *Check Permission on Query Request*.

2 – Check Permission On Query Request

Reference Id	***SQRA.QUM.CPQ.1.1***

T2S System user privileges are checked against the access authorisations stored in the *Reference Data* data store and in *Dynamic Data* data store, guaranteeing that the requesting T2S System user only receives the information/data in an extent that complies with its permissions, which means that T2S Actors shall perform queries based on their privileges and roles **{T2S.14.060} {T2S.14.525}**.

If compliant with the individual privileges of a T2S system user which were granted by the responsible administrators, the queries on settlement instructions can return the following information **{T2S.14.259}**:

- to a T2S Actor all instructions that have been sent either by itself or by T2S Parties, who are authorised to do so on behalf of its party,
- to a T2S Actor of a CSD additionally all instructions referring to accounts legally attributed to the CSD,
- to a T2S System user of a CB,
- if the CB acts as CSD participant, (like to any other CSD participant) all own instructions, or additionally,

All rights reserved.

- if the CB acts as CSD as well, (like to any other CSD) all instructions referring to accounts legally attributed to it.

A check whether the requesting T2S System user is provided with the required privilege is done for each selection parameter which might be restricted by an access privilege. If the sender of the User query is not authorised to access the data he requested, the flow *Error Data Queries* is forwarded to the *Interface*. *Error Data Queries* flow is as well forwarded in case the requesting T2S System user is not allowed to access all specified selection parameters (e. g. if several securities account numbers are specified and the requesting T2S System user is not allowed to access all of the specified accounts, the *Error Data Queries* is created).

If the sender of the User query does not specify a value for an optional selection parameter which is relevant for the check permission on this field, *Check Permission on Query Request* function sets these selection parameters. In case a mandatory selection parameter which is relevant for the *Check Permission* is not specified the query is rejected. Therefore the function identifies all values which are in line with the access rights of the requesting T2S System user. These values supplement the data forwarded to the function *Extracting Data for Query*.

All User queries which did not cause error(s) are forwarded to the function *Extracting Data for Query*.

3 – Extracting Data for Query

General Information on Extracting Data for Query

Reference Id	***SQRA.QUM.EDQ.1.1***

The *Extracting Data for Query* function extracts the requested data from the relevant data stores (i. e. Reference and/or *Dynamic Data* stores) and sends the results as *Queried data* to the *Interface*. Concerning queries for securities positions and cash balances please see chapter 3.7.1 [▸ 595] for the architectural concept of OBD/IBD and its effect on Query Management.

Combination of search criteria

Reference Id	***SQRA.QUM.EDQ.1.2***

Most User queries allow the specification of search criteria for various fields. For these selections the following rules apply:

- In case more than one field is specified the criteria are combined using the logical AND operation (e. g. *SWIFT BIC Code Query*: in case the requesting T2S System user specifies the Financial Institution Name as well as the City, the BIC of the Financial Institution with the specified Name in the specified City is returned; i. e. the different selection criteria are combined with a logical AND).
- If a User query allows the specification of a list of values for a certain field, these restrictions are combined using a logical OR operator. Fields not specified allow any value (e. g. *Securities Account Position Query*: in case the requesting T2S System user specified several securities account numbers, each of these account numbers is a valid value for the *Extracting Data for Query* function; i.e. the different values are combined with a logical OR, each of the specified accounts is returned).
- For fields that allow the specification of a range for the field's value an upper and a lower limit can be specified (result includes boundary values). Depending on the actual User query it may be sufficient if only one value is given (e.g. to select all instructions sent after a specific time).

All rights reserved.

- In User queries that support the specification of a relation the actual comparison operator (</<=/=/>=/>/<>) and a fixed value for comparison can be specified.
- It is possible to query with the starting character of a name or a description and obtain a similar behaviour of a wild card usage.

Only records that match the selection criteria are returned **{T2S.14.230}**.

If the extraction from the data store fails the flow *Error data queries* is forwarded to the *Interface*.

Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
User query	IN		INTF: Inbound Processing Module	
Queried data	OUT			INTF: Outbound Processing Module
Error data queries	OUT			INTF: Outbound Processing Module

Data accessed by the module

DATA	ACCESS MODE	STATUS	COMMENTS
Reference Data	Read	All	According to the diagram of the module, all functions of the *Query Management* Module are able to access the Static Data data store. *Query Management* Module processes User queries across several business areas and thus accesses diverse static data stores, e.g. party data, securities data.
Dynamic Data	Read	All	According to the diagram of the module, all functions are able to access all the *Dynamic Data* data stores. *Query Management* Module processes User queries across several business areas and thus accesses dynamic data stores of different business fields.

All rights reserved.

3.7.3.2 Queries (QU) Use Cases

Scope

This category of use cases describes the situations when a T2S Actor sends different queries to the system. All possible types of queries are covered, e.g. queries to monitor securities positions, cash balances, instruction status and reference data.

Criteria

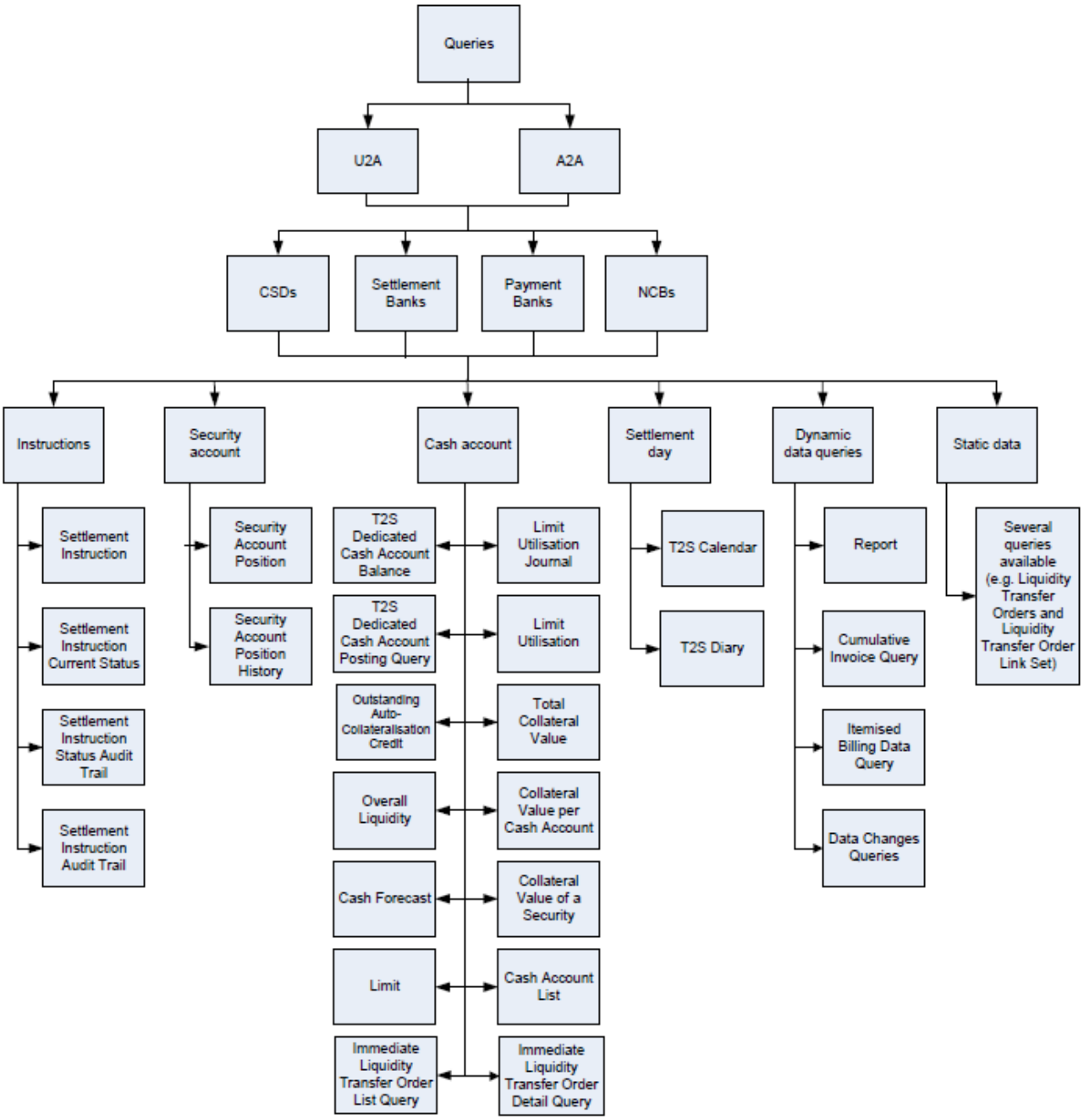
The criteria used to identify the use cases are: the communication mode, the instructing party, the query category and the query type. Possible values are the following (non-exhaustive list):

CRITERIA	POSSIBLE VALUES	DESCRIPTION
Communication mode	U2A, A2A	See Interface domain
Query category	Instructions, securities account, cash account, reference data, settlement day, dynamic data queries	The subject of the query as a broad category
Query type	For a preliminary list of queries please refer to the table in chapter 3.7.3.3. "Static Functional Description", for an overview please see also the diagram "List of Use Cases" below.	The concrete query type.

List of Use Cases

All rights reserved.

The criteria described above are reported in the following tree:



Not all of the combinations of values are relevant from a business point of view. The preliminary list of use cases is presented in the appendix "List of Use Cases". This list of queries mentioned in the diagram is not exhaustive.

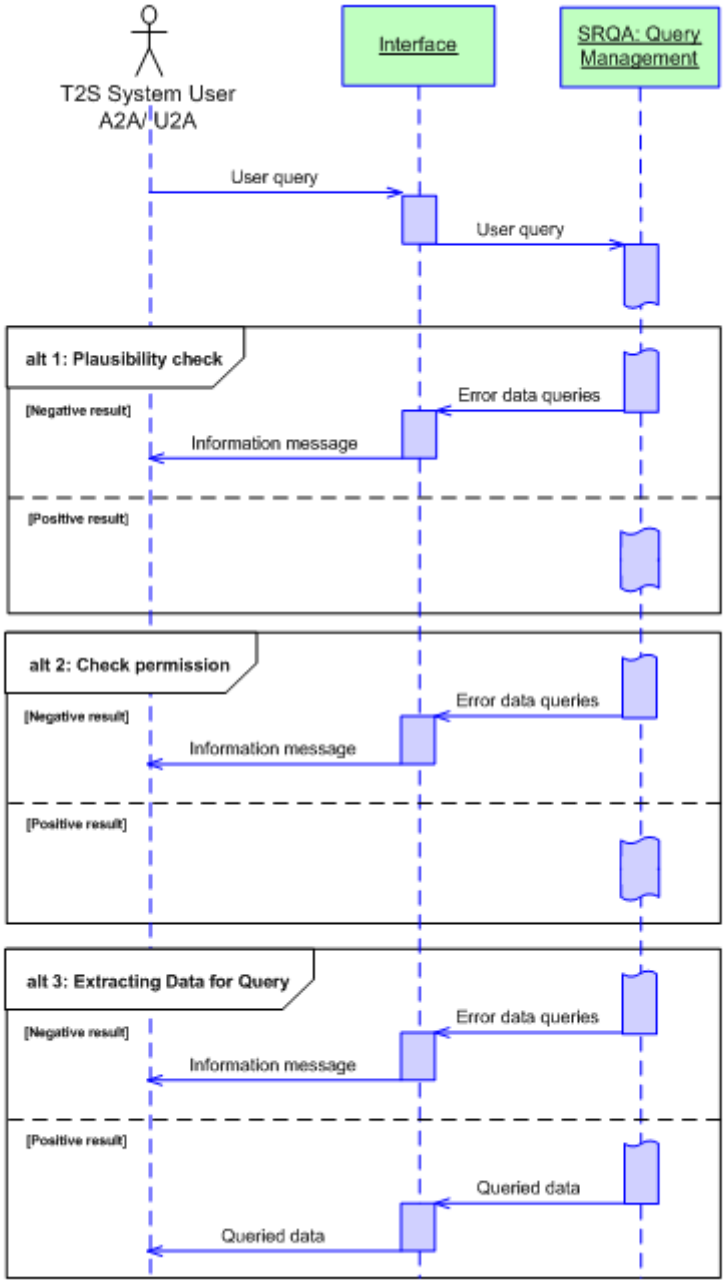
3.7.3.3 Processing of Queries use cases

Processing of UC-QU-1: Query on Settlement Instructions

Considering the uniformity of the Query Management process, only one representative use case is presented.

All rights reserved.

DIAGRAM 46 - UC-QU-1: QUERY ON SETTLEMENT INSTRUCTIONS



Business assumption

A T2S System user sends a query on the status of *Pending Settlement Instructions.*

Please note that all communication between the Interface domain and T2S Actors should be understood as

- *Encrypted Inbound Communication* or *Encrypted Outbound Communication* for A2A mode
- *XHTML request via HTTPS* or *XHTML response via HTTPS* for U2A mode

as described in the *Interface* chapter. For the sake of readability, more specific terms for these flows are used in the diagram above.

Processing

All rights reserved.

- A T2S System user sends a *query* (here: settlement instruction query) which enters the *Interface*. Once the technical checks and validations in the *Interface* are successfully passed the user query is forwarded to the *Query Management* module.
- As an exception to this general process it should be noted that queries received in A2A mode during the Maintenance Window are queued in the *Interface*. For U2A communication, T2S is unavailable during the Maintenance Window. Furthermore, securities position and cash balance queries received in A2A mode during the night-time settlement sequences are queued in the *Interface* as well. They are responded at the end of a night time settlement cycle. Securities position and cash balance queries received in U2A mode during night-time settlement sequences are not available and are not queued.

Within the *Query Management* module the *Plausibility Check* function verifies the plausibility of the query. Using logical checks it tries to identify zero-result queries or queries containing not plausible characters to reduce unnecessary accesses to data stores. For example, queries which contain non valid ISINs or currencies neither eligible nor externally recognised for T2S are filtered by the function *Plausibility Check.* The reference values for the plausibility checks are stored in the *Reference Data* data store (e. g. plausibility check for a Party Identifier of Instructing Party) or the *Dynamic Data* data store (e. g. plausibility check for Unique Instruction Reference).

Alternative 1 - Plausibility check: Negative Result

- Plausibility check fails: If a query fails one of the checks the flow *Error data queries* is forwarded to the *Interface*. Subsequently an information message is sent to the requesting T2S System user via the *Interface*.

Alternative 1 - Plausibility check: Positive Result

- Plausibility check is successful: The valid user query on a settlement instruction is forwarded to the function *Check Permission on Query Request.*

The function *Check Permission on Query Request* ensures that users can only query data which they are permitted to access. The access authorisations (i. e. the required privileges) are stored in the *Reference Data* data store.

Alternative 2 – Check Permission: Negative Result

- If the sender of the settlement instruction query is not authorised to access the data, *Error data queries* is forwarded to the *Interface*. Subsequently an information message is sent to the requesting T2S System user via the *Interface*.

Alternative 2 – Check Permission: Positive Result

- If the check is successful the query is forwarded to the *Extracting Data for Query* function.

The *Extracting Data for Query* function extracts the requested data from the data store.

Alternative 3 – Extracting Data for Query: Negative Result

- If the extraction from the data store fails, *Error Data Queries* is forwarded to the *Interface*. Subsequently an information message is sent to the requesting T2S System user via the *Interface*.

Alternative 3 – Extracting Data for Query: Positive Result

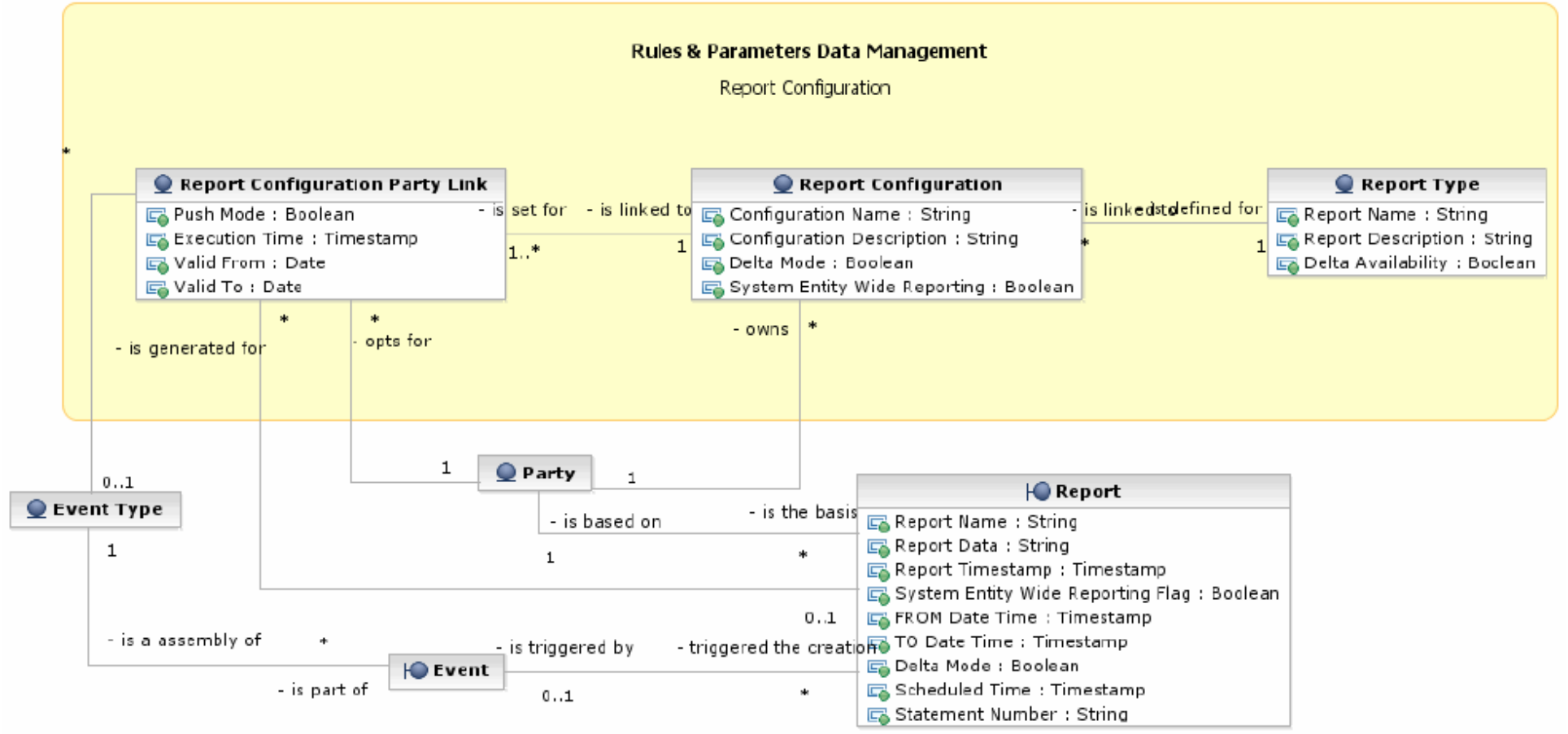
All rights reserved.

- If the extraction from the data store is successful the *Queried data* is sent via the *Interface* to the requesting T2S System user.

All rights reserved.

3.7.4 Report Management

3.7.4.1 Dynamic Data managed by the module



Report

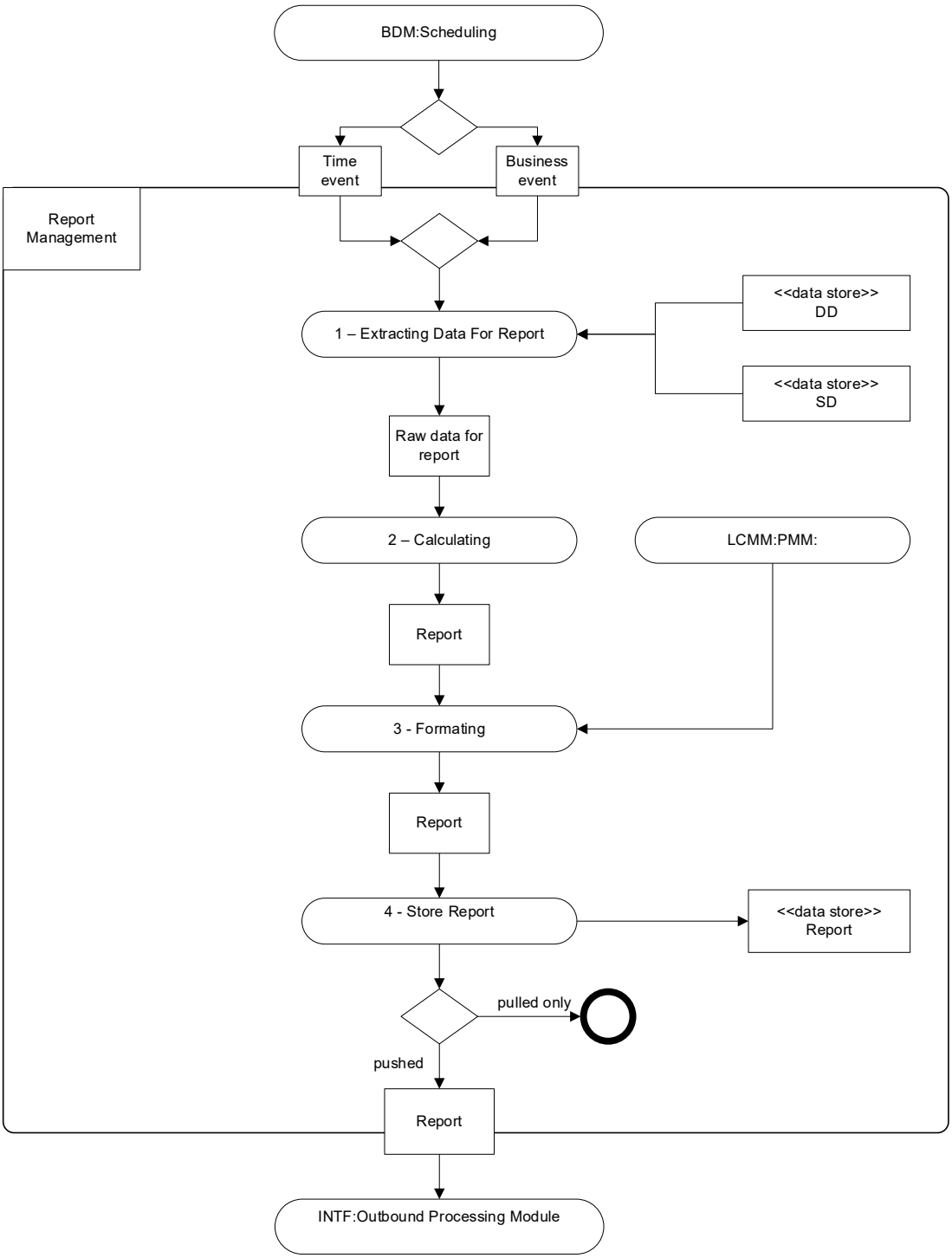
This dynamic data entity contains reports which are created by *Report Management* module.

ATTRIBUTE	DESCRIPTION
Report Name	Name of the stored Report
Report Data	Content of the Report stored as flat file.
Report Timestamp	Timestamp reflecting date and time of the creation of the report.
System Entity Wide Reporting Flag	Boolean attribute specifying whether the report should include information for the entire system entity data scope.
FROM Date Time	Information on the point in time, when the queried report period starts.
TO Date Time	Information on the point in time, when the queried report period ends.
Delta Mode	It stores the information whether the created report is a full or a delta report. Possible values are: - True (i.e. delta report) - False (i.e. full report)
Scheduled Time	Configured point in time which triggered the creation of the report.
Statement Number	Statement number of the report (composed of the business date and Report Type (2-letter-abbreviation depending on report name and Delta/Full-Flag) plus sequential number), which is unique in T2S.

The other entities depicted in the diagram above are static data entities and described in the Reference Data chapter.

3.7.4.2 Static Functional Description

Diagram of the module



Description of the module

General information

T2S provides T2S actors with a number of pre-defined reports for periodical information on the production data. The *Report Management* module, however, does not cover any regulatory reporting. The reports provided are delivered as XML messages and comply to the largest possible extent with ISO 20022 standard. In cases where the standard does not fit T2S demands, proprietary messages are defined **{T2S.13.160}.**

The reports are based on the latest data available **{T2S.13.200}.**

List of Report Types

T2S offers the several reports. Some of them are available as a delta version **{T2S.13.190}.** Delta reports allow identifying intraday new, modified and deleted information. The various reports are described in more detail in the latest versions of the User Detailed Functional Specification (UDFS) and User Handbook (UHB) for the Graphical User Interface. Should additional reports be deemed requested in the future, they will be assessed on an ad hoc basis and developed provided that they do not have any material impact on the dynamic data model and on the reference data management included in the present GFS version.

REPORT NAME	COMPLETE REPORT	DELTA REPORT	URD REFERENCES
Statement of Holdings	x	x	**{T2S.13.230}** **{T2S.13.190}**
Statement of Transactions	x	x	**{T2S.13.240}** **{T2S.13.190}**
Statement of Pending Instructions	x	x	**{T2S.13.250}** **{T2S.13.190}**
Statement of Settlement Allegements	x	x	**{T2S.13.260}** **{T2S.13.190}**
Statement of Accounts	x		**{T2S.13.270}**
Statement of Static Data	x		**{T2S.13.280}**
Current Settlement Day Cash Information Report	x		**{T2S.13.300}**
Following Settlement Day Cash Forecast Report	x		**{T2S.13.310}**

Note: The former Billing Data Report **{T2S.13.290}** is covered by the Itemised Billing Data which is created by the Billing Module according to the same principles as those used in the Report Management module.

Preconditions for report creation

For the creation of reports, *Report Management* module relies on the *Reference Data.* Which event triggers which report for the data of which party is stored as *Report Configuration* in the *Reference Data.* T2S also offers the possibility to configure system entity wide reports, which cover the data of all parties belonging to the specified CSD/CB. Moreover it is stored whether the report shall be created as a complete version or as a delta version. It is also stored which parties subscribed to which pre-defined Report. Parties can specify, that they want to receive the report directly after creation (push mode) or opt for a download at a later point in time (pull mode). In case a party opts for a download at a later point in time it has to send a *Report Query* to T2S which is processed and answered by *Query Management* Module.

All rights reserved.

Report Management Module creates only reports which are pre-defined in the *Report Configuration Reference Data* data store.

Availability of reports

All reports are available in the user-to-application mode and in the application-to-application mode **{T2S.13.170}.** All reports are available for all T2S Actors **{T2S.13.180}.** The privilege to subscribe to a certain report can be granted to the parties or users. Reports can be sent to the report receiving parties, containing information on individual accounts or on a set of accounts **{T2S.13.210}.**

T2S reports can either be generated on occurrence of a business event (e.g. end-of-day) or on a time event (i.e. at a fixed time) **{T2S.13.190}.** And T2S reports can be sent to its intended recipients.

A report generated according to an existing *Report Configuration* is available for download until it is replaced by the next generation of a report based on the same *Report Configuration.*

The last generated version of a report is available for three calendar months after its creation according to the general timeline for accessibility of data

If need be on T2S actor's side to receive a prior report (again), it is accessible via the *Report Query* of the *Query Management* Module **{T2S.13.190}.**

The list of report types and a short description of each report's content can be found in the latest versions of the User Detailed Functional Specification (UDFS) and User Handbook (UHB) for the Graphical User Interface.

Description of the functions of the module

1 – Extracting Data For Report

Information on the permission to receive a report

Reference Id	***SQRA.REM.EDR.1.1***

Directly connected T2S parties can receive reports on:

- their own securities and cash balances and those of their clients;
- instructions submitted by themselves or those clients;
- instructions that refer to the respective securities or/and cash accounts.

CBs in T2S can receive reports on:

- cash balances within their sphere of responsibility
- reference data within their sphere of responsibility

A CSD, CB or Payment Bank can also act as a participant of a CSD in T2S. In this case a report based on this party provides data of the CSD participant party. It is a report as any other report for a CSD participant of this CSD in T2S.

When a CB acts as a CSD in T2S a report based on this party provides data of the CSD party. It is a report as any other report for a CSD in T2S.

CSDs in T2S can receive reports on:

- Instructions submitted by itself or by its directly T2S connected clients;

All rights reserved.

- Securities positions of their securities accounts or securities accounts of their parties;
- reference data (own data and those of T2S parties who permit).

Where a CSD acts as an Investor CSD, it is treated like a participant of the respective Issuer CSD in T2S **{T2S.13.220}.**

In general a party can receive reports on those data which are within its default data scope..

Behaviour of Extracting Data for Report

Reference Id	***SQRA.REM.EDR.1.2***

This function is triggered by a Time event or Business Event sent from the *Scheduling* module. An Event is only sent in case a report generation is foreseen and valid, i.e. a *Report Configuration* is linked to it, at least one party subscribed to it and the current business day is within the validity period. When *Extracting Data for Report* function receives the event, the processing for all linked configurations (i. e. each combination of report type and party) is started.

The list of report receiving parties itself is irrelevant for the initiation of the report creation. That means the generation of the report is done independently from the fact how many parties have subscribed to it and whether they wish to get it directly after creation or not; necessary for the creation is the subscription of a report by at least one party.

Considering the report type and the party, *Extracting Data for Report* function accesses the respective *Dynamic* and *Reference Data* stores in order to extract all data that is necessary for the report generation. Delta reports contain similar information as the complete reports. The difference between both is the time scope which is regarded. Delta reports provide information that changed since the previous report was created, i. e. it includes cancellations, modifications and new items. Therefore the timestamp of the previous report is considered. Please note that delta reports only report on data changes of the current business day. If there is no previous report for the current business day, the SoD is considered as the starting point in time for the reporting period

Combined with the timestamp of the data extraction these data are forwarded as the *Raw Data for Report Data* flow to the *Calculating Report* function of this module.

2 – Calculating Report

Reference Id	***SQRA.REM.CAR.1.1***

The received *Raw Data for Report Data* flow is processed, e.g. sums, products, totals or averages are calculated. The resulting Report data, including raw data and calculated values, are sent to the function *Formatting Report.*

3 – Formatting Report

Reference Id	***SQRA.REM.FOR.1.1***

The received *Report Data* (raw data and calculated data) are formatted, sorted (order of columns is defined, the sorting of rows is performed) and grouped (e.g. totals are added to each group). Finally the finished report is forwarded to the function *Store Report.*

4 – Store Report

All rights reserved.

Storing the Report

Reference Id	***SQRA.REM.STR.1.1***

This function stores the newly *Generated Report* in the *Report Data Store.* The Report is linked to the respective *Report Configuration.*

Forwarding the Report

Reference Id	***SQRA.REM.STR.1.2***

In case at least one report receiving party requires receiving the *Report* directly after generation, the *Store Report* function forwards the *Report* to the *Outbound Processing* module of the *Interface*. Otherwise the processing ends at this point, as the *Report* does not need to be forwarded.

Description of the Input/Output of the module

FLOW	IN/ OUT	DESCRIPTION	FROM	TO
Business event/Time event	IN	Event triggering the report creation.	OPSR: Scheduling	
Report	OUT	A Report is only forwarded to the Interface in case at least one report receiving party requires a push of the Report.		INTF: Outbound Processing Module

Data accessed by the module

DATA	ACCESS MODE	STATUS	COMMENTS
Reference Data	Read		Reference Data for report creation and source data for creating the report
Dynamic Data	Read		Source data for creating the report
Report	Write		Store generated reports for later downloads

3.7.4.3 Reports (RE) Use Cases

Scope

This category of use cases covers the situations where T2S creates reports – either at a fixed time or triggered by a business event - and make it available for T2S Actors. Depending on the T2S Actors subscription preferences the report is sent out either directly after creation or stored for later retrieval.

Criteria

The criteria that characterise reports use cases are: the communication mode, the information basis, the report triggering, the report category and (optional) the report type. These criteria are summarised in the table below with their possible values (exhaustive list):

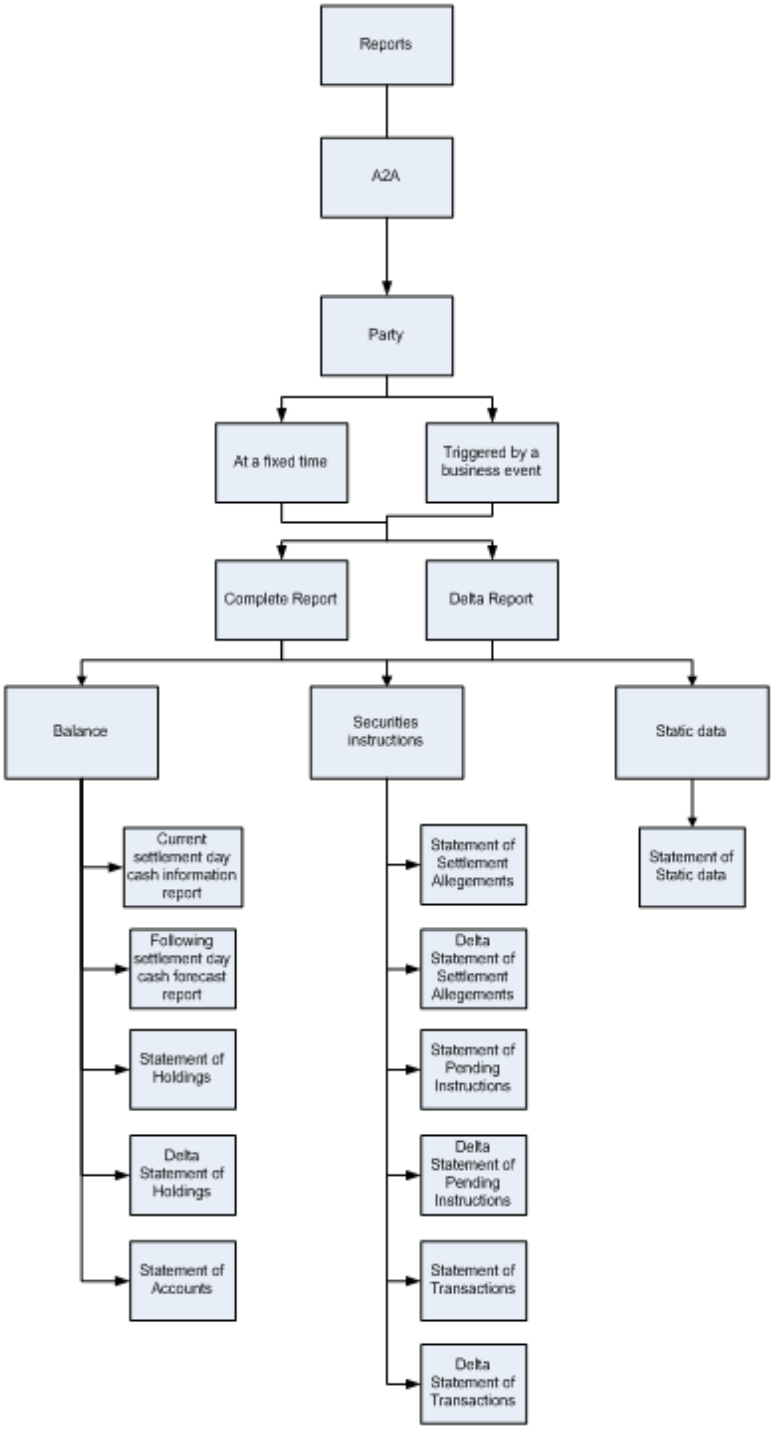
All rights reserved.

CRITERIA	POSSIBLE VALUES	DESCRIPTION
Communication mode	A2A	See Interface domain
Information basis	Party	Depending on the Report Type the created report contains information on T2S dedicated cash accounts, securities accounts etc. of a specified party
Report triggering	At a fixed time, triggered by a business event	Creation time of a report
Report category	Securities instructions, balance, static data	Business segment of report types
Report Classification	Complete report, delta report	Delta reports allow to identify intraday new, modified and deleted information since the creation of the previous report.
Report type	Current settlement day cash information report, following settlement day cash forecast report, statement of allegements, statement of pending instructions, statement of holdings, statement of transactions, statement of static data, statement of accounts	The concrete report type

List of Use Case

The criteria described above are reported in the following tree:

All rights reserved.



Not all the combinations of values are relevant from a business point of view. The preliminary list of use cases is presented in the appendix.

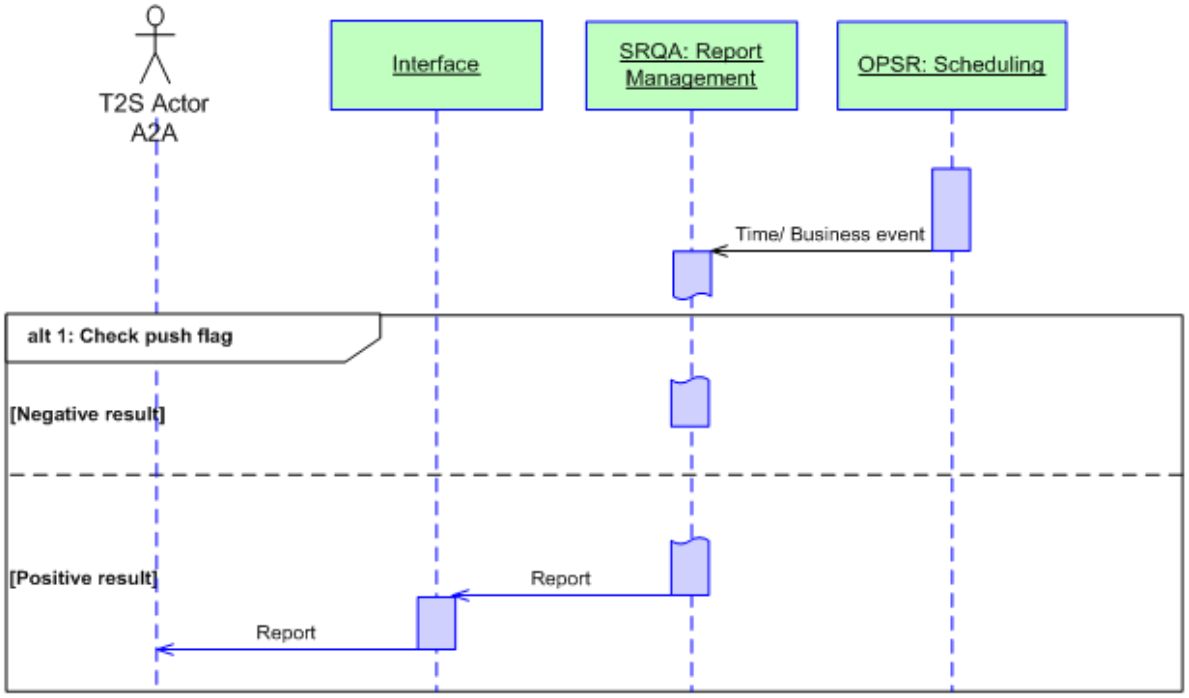
3.7.4.4 Processing of Reports Use Cases

Considering the high uniformity of the *Report Management* process, only one representative use case is defined.

Processing of UC-RE-1: Current Settlement Day Cash Information Report

All rights reserved.

Diagram 47 - Processing of UC-RE-1: Current Settlement Day Cash Information Report



Business assumption

T2S produces an automated report on the T2S dedicated cash accounts of a certain party at a fixed time or by a business event (e.g. End of night-time cycle event) for the cash forecast of the current settlement day.

Please note that the communication between the Interface domain and T2S Actors should be understood as *Encrypted outbound communication* for A2A mode as described in the *Interface* chapter. For the sake of readability, a more specific term for this flow is used in the diagram above.

Processing

Based on a time/business event the *Report Management* Module is triggered by the *Scheduling* Module during the business day. The relevant business data is extracted, processed, formatted and stored in the *Report Data* store. Then the report recipient list for each report (as defined in the *Report Configuration*) is checked. This list contains the information:

- which party requires to receive the report directly after generation in push mode and
- which party downloads the generated report in pull mode via *Report Query* of *Query Management* Module.

Alternative 1: Check push flag: Negative result.

- In this case no party wants to get the report pushed; so the report is not sent out to any party and stored in the Report Data store.

Alternative 1: Check push flag: Positive result.

- The created report is sent out via the Outbound Processing Module in Interface to the T2S Actor A2A.

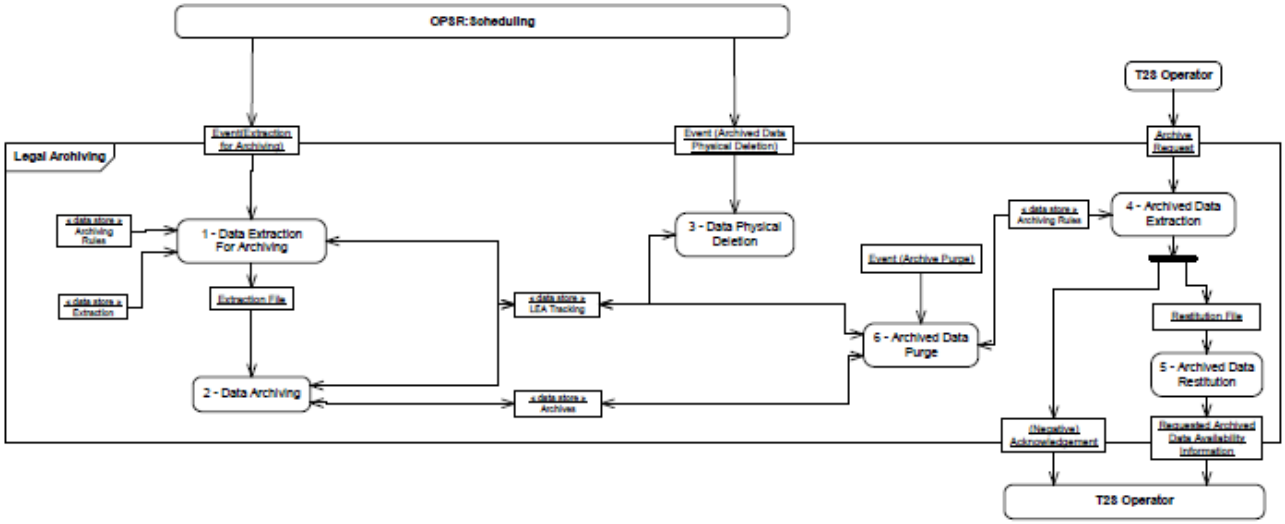
All rights reserved.

3.7.5 Legal Archiving

3.7.5.1 Static functional description

Diagram of the module

Below is the Activity Diagram modelling the interactions between the functions within the module.



Description of the module

This module addresses needs for a central archive for audit and regulatory requirements made available to T2S parties via their CSDs or CBs.

It stores, for a harmonised 10 years period, processed inbound and outbound messages in their original format and answers to the extraction request to these data **{T2S.17.070}.**

To avoid duplication of data and storage, static and transactional data older than three months with data revision (replication of reference data structures as revision tables) and data history (data stored with "valid from" and "valid to" date attributes) stand in the long-term statistical data store of the Statistical Information module **{T2S.17.100}.**

Besides, it is recalled that the retrieval of archived data aims only at meeting requirements from a legal nature and therefore the answer to an archive request is not to be expected on a "real time" basis (response time up to 3 days) **{T2S.17.130}.**

Archiving processing within the LEA module is based on archiving rules that defines:

- When messages are archived according to:
  - Archiving time which defines the delay before messages are archived by this module. The archiving time is configurable with up to 5 business day **{T2S.17.090},**
  - Archiving duration defines the period for which messages are kept as archives in the Archive data store before they are purged. The archiving duration is configurable and currently harmonised to 10 years **{T2S.17.070};**
- How messages are archived according to their ISO message description, their related business date and their related CSD or CB.

All rights reserved.

Functions of LEA module manage a LEA Tracking data store to keep track of each process result. The Data Archiving function creates the LEA Tracking data store, which all LEA functions update. For each file a status and a date are stored. The description of the database will be done during the implementation phase.

Six functions handle the different steps of the archiving processing:

- *Data Extraction for Archiving* extracts the data from T2S data stores, creates *Extraction Files* of processed inbound and outbound messages;
- *Data Archiving* organises extracted data into Archiving Files and handles their archiving;
- *Data Physical Deletion* manages the physical deletion of processed inbound and outbound messages that are already archived by the *Data Archiving* function;
- *Archive Data Extraction* extracts the archived files on an *Archive Request* and edits the archived processed inbound and outbound messages in a *Restitution File;*
- *Archived Data Restitution* posts the *Restitution Files* and informs the requester;
- *Archived Data Purge* deletes the archived files when archiving duration is reached.

Description of the functions of the module

1 – Data Extraction For Archiving

Data Extraction

Reference Id	***SQRA.LEA.DEA.1.1***

During the settlement day, triggered by an *Event (Extraction for Archiving)* from the *Scheduling* module, the function **{T2S.17.080}:**

- Selects from T2S data stores all the processed inbound and outbound messages and *Flat Files* for reports at end of day (EOD) period according to archiving rules;
- Copies them into *Extraction Files* with indexes to facilitate its archiving [210];
- Sends created Extraction Files to the *Data Archiving* function;
- Keeps track of the *Data Extraction for Archiving* process by creating, for the given Business Date, the LEA Tracking data store which contains, for each *Extraction File,* information on its creation and content.

2 – Data Archiving

Data archiving process

Reference Id	***SQRA.LEA.DAR.1.1***

Triggered by an *Extraction Files* received from the *Data Extraction for Archiving,* the function:

- Creates archiving files, for an archiving date, and fills in them with data from *Extraction Files* according to archiving rules, and also *Flat Files* for reports at end of day (EOD) period;
- Stores archiving files in the Archive data store for a duration defined in the archiving rules;

210 These indexes concern the ISO message description, a settlement date, and a CSD or NCB owner reference.

All rights reserved.

- Keeps track of the *Data Archiving* process by updating LEA Tracking data store with its process result.
- In cases where the standard does not fit T2S demands, proprietary messages are defined **{T2S.13.160}**.

Additionally CSDs may decide to receive some specific reports via flat file instead of XML. Description about the information services of T2S using flat files is given in more detail in the latest versions of the User Detailed Functional Specification (UDFS).

- The reports are based on the latest data available **{T2S.13.200}**.

3 – Data Physical Deletion

Data physical deletion process

Reference Id	***SQRA.LEA.DPD.1.1***

Triggered by an *Event (Archived Data Physical Deletion)* from the *Scheduling* module, the function:

- Determines the archiving date for physical deletion;
- Checks if each archiving file related to the archiving date have been successfully archived by the *Data Archiving function* according to the LEA Tracking data store;
- Proceeds to the physical deletion of processed inbound and outbound messages, contained in each archiving file for related archiving date, from T2S data stores;
- Keeps track of the *Data Physical Deletion* process by updating the LEA Tracking data store with its process result **{T2S.16.300}.**

4 – Archived Data Extraction

Archive request

Reference Id	***SQRA.LEA.ADE.1.1***

An *Archive Request,* sent by a T2S operator on behalf a CSDs or CBs or directly connected parties or for the T2S operator's own purpose triggers this function **{T2S.17.120}.**

On receipt of this request, the function carries out an initial check to validate whether the type of requested archived message corresponds to an archived one by checking the applicable archiving rule.

Depending on the result of this first check, a positive or negative *Acknowledgment* is sent to the T2S operator.

Data Extraction

Reference Id	***SQRA.LEA.ADE.2.1***

IF the initial check is successful, then the function extracts the requested archived messages in a second step by:

- Searching the relevant archive file using on index on the relevant file;
- Extracting and sending to the *Archived Data Restitution* function, through a *Restitution File,* the requested archived messages from the considered archive file.

5 – Archived Data Restitution

All rights reserved.

Archived data restitution process

Reference Id	***SQRA.LEA.ADR.1.1***

On receipt of the *Restitution file* and in a maximum of 3 days after the reception of the request **{T2S.17.130},** the function:

- Makes available the *Restitution File* to the T2S Operator;
- Sends a *Requested archived data availability* information to the T2S Operator in a way to inform of the availability of the restitution file in T2S.

Once the T2S Operator has received this information, it sends the *Restitution File* to the requester by a dedicated channel outside of the T2S Interface.

6 – Archived Data Purge

Archived data purge process

Reference Id	***SQRA.LEA.ADP.1.1***

On reception of an Event (*Archive Purge*) sent by the Legal Archiving module internal timer, the function checks the archived files for which the archiving duration is reached and purges them **{T2S.17.070}.**

Description of the Input/Output of the module

FLOW	IN/ OUT	DESCRIPTION	FROM	TO
Event (Extraction for Archiving)	In	Received daily to launch the extraction of messages from the T2S data store	OPSR:Scheduling	
Event (Archived Data Physical Deletion)	In	Received daily to launch the deletion of archived messages from the T2S data store	OPSR:Scheduling	
Archive Request	In	Received as a request for archived data sent by the T2S Operator	T2S Operator	
Acknowledgement	Out	Sent to inform the T2S Operator of the successful or failure result of the first check on its archive request		T2S Operator
Requested Archived Data Availability Information	Out	Sent to inform the T2S Operator of the availability of the restitution file		T2S Operator

Data accessed by the module

DATA	ACCESS MODE	COMMENT
DYNAMIC DATA		
Archiving Rules	Read	Refer to the Description of the module section

All rights reserved.

DATA	ACCESS MODE	COMMENT
LEA Tracking	Write	LEA Tracking data store contains information concerning the processing result of LEA functions
Extraction	Read	Data store where processed inbound and outbound messages are extracted from in order to be archived
Archives	Write/ Read	Data store where archiving files are stored

3.8 Operational Services

3.8.1 General Introduction

The *Operational Services* domain provides a number of functionalities to the T2S System User. It is split up in four modules, only the first two modules being subject to detailed functional description:

- *Scheduling;*
- *Operational Monitoring;*
- *Data Migration;*
- *Billing.*

Scheduling

The *Scheduling* module is responsible for managing the schedule of events of the T2S operating day as well as the change of the business date. This module allows the T2S Operator to set up the configuration of all the events of the operating day and the definition of dependencies between events. The *Scheduling* module is also in charge of communicating events to the other domains and modules to trigger processes in T2S.

The *Scheduling* module also provides the T2S Operator with features to manually modify the daily schedule (e.g. by changing an event's scheduled time) and the T2S closing days. Finally, the module can be queried to get information on the daily schedule and the current status of the operating day.

Operational Monitoring

The *Operational Monitoring* module provides tools for monitoring, information provisioning and real-time problem detection on both operational and technical levels. *Operational Monitoring* concerns the detection of functional or operational problems and the monitoring and provisioning of service performance indicators, whereas technical monitoring refers to the detection of hardware and software problems.

The module also offers online access to the Trouble Management System for the CSDs and the authorised T2S parties.

Data Migration

The *Data Migration* module provides the migration functionality which allows the CSDs and the CBs to automatically transfer the major part of relevant data (e.g. securities account data, party reference data, securities reference data, T2S Dedicated Cash account data…) to T2S **{T2S.21.300} {T2S.21.310}**. The CSDs and CBs can send their data via the standard channels of communication: via A2A using the supported standard XML messages or files, via U2A using the standard T2S GUI functionality to enter data required for

All rights reserved.

its migration. Alternatively, the CSDs can provide the T2S Operator outside T2S (e.g. via secured e-mail) with structured files (flat files, Excel files). The T2S Operator uses the data migration module to convert those structured files into standard T2S XML format and sends the data to T2S.

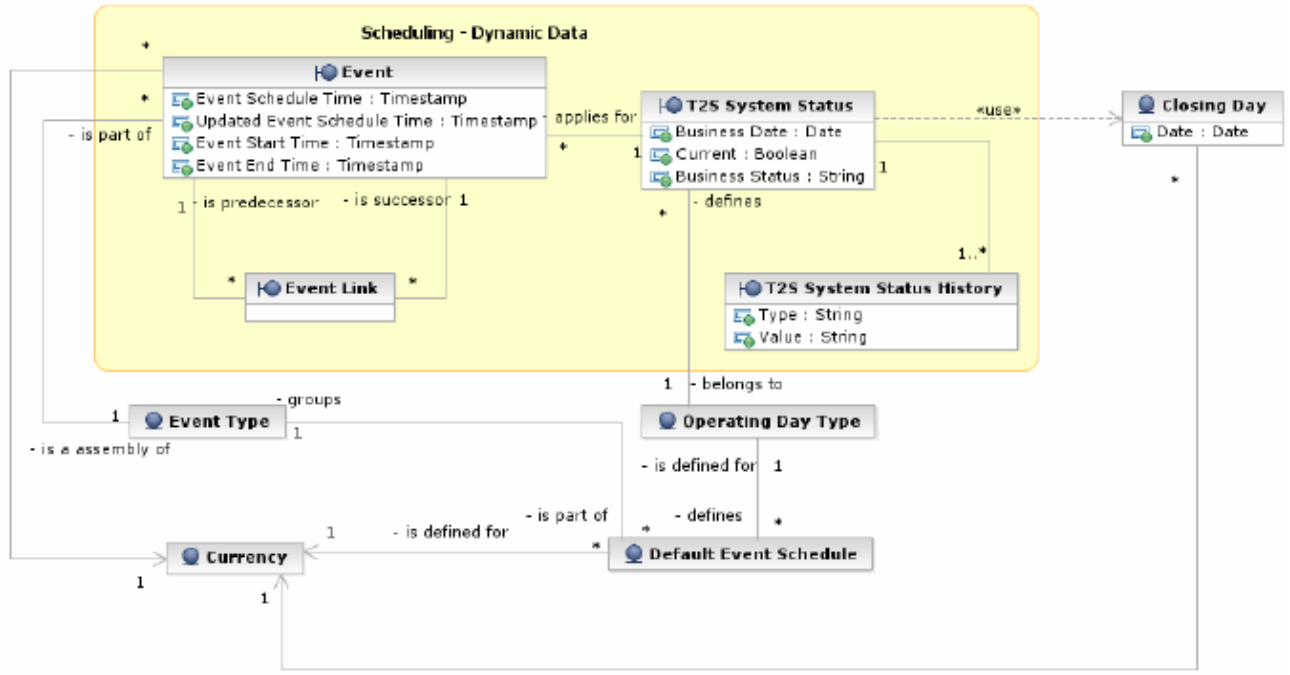
Billing

The *Billing* module produces the invoices for all CSDs in a monthly cycle covering the period of one month's activities. They are distributed in an electronic form after the completion of the last business day of the month.

T2S Parties (i.e. individual CSD participants) are invoiced by the respective CSDs based on the information provided by T2S and complemented by additional data possessed by the CSDs.

3.8.2 Scheduling

3.8.2.1 Dynamic data managed by the module



This section contains the data model descriptions of the entities, their relationships and their attributes for the dynamic data of the Scheduling module (see the *Rules and Parameters Data Management* module for a detailed description of the relevant reference data entities).

1 – Event

This entity includes all the information related to *Events* scheduled and triggered during the actual operating day. Each Event refers to a specific occurrence in the business day that is planned for a certain time and triggers processes within selected modules of T2S. The Operating Day Planner function (see section 3.8.2.2), according to the Operating Day Type and the Default Event Schedule information stored as reference data (see section 3.3.11.8) generates new instances in this entity for each change of business date. During the

All rights reserved.

operational day the T2S Operator is allowed to input additional *Events* (i.e. additional occurrences of previously defined Event Types) as well as to re-schedule and complete existing events depending on the contingent operational or business needs of a given day.

ATTRIBUTE	DESCRIPTION
Event Schedule Time	Scheduled time for each Event
Updated Event Schedule Time	Scheduled time for each Event following an update
Event Start Time	Actual start time of the Event
Event End Time	Actual end time of the Event

For the purpose of creating a list of *Events* for a given operating day, each instance of this entity has a link to the relevant *Event Type* and *T2S System Status*. In addition, *Events* may be linked to Currencies, for currency-specific events **{T2S.03.303}**. A currency-specific *Event* is identified by a Boolean flag in the *Event Type* entity. Finally, the T2S Operator also can link *Events* to define predecessor/successor constraints.

2 – T2S System Status

This entity defines the current Business Date and Business Status, i.e. the current phase of the operating day. The *Operating Day Planner* function calculates the next operating day, taking into account (i.e. skipping) the T2S closing days.

ATTRIBUTE	DESCRIPTION
Business Date	Date of the current operating day or of a past operating day.
Current Date	Boolean attribute specifying whether the business date refers to the current operating day.
Business Status	It specifies the current T2S business status **{T2S.03.010}**. The exhaustive list of possible values is as follows (listed in the order in which they normally occur during the operating day): ▮ SOD Period ▮ First Night-time Cycle (specifying progressive number) ▮ Last Night-time Cycle ▮ Maintenance Window ▮ Real-time settlement period ▮ Real-time settlement closure period ▮ EOD Period

Each occurrence of *T2S System Status* is linked to one or many occurrences of *T2S System Status History*, storing the full history of the T2S business status transitions throughout the operating day. In addition, each occurrence of *T2S System Status* is linked to all the relevant *Event* occurrences and to the related *Operating Day Type*, which the *Business Date Change* function uses during the creation of each new operating day schedule.

2 – T2S System Status History

All rights reserved.

This entity stores the full history of the T2S business status transitions throughout the operating day.Every time T2S starts a new operating day, it creates a new occurrence for *T2S System Status* and for *T2S System Status History* and it stores the value assigned to the initial T2S business status in both entities. Every time T2S changes the business status of the operating day, it creates a new revision of the relevant occurrence of *T2S System Status History* to store the new value of the T2S business status, propagating it also into the Business Status attribute of the *T2S System Status* entity.

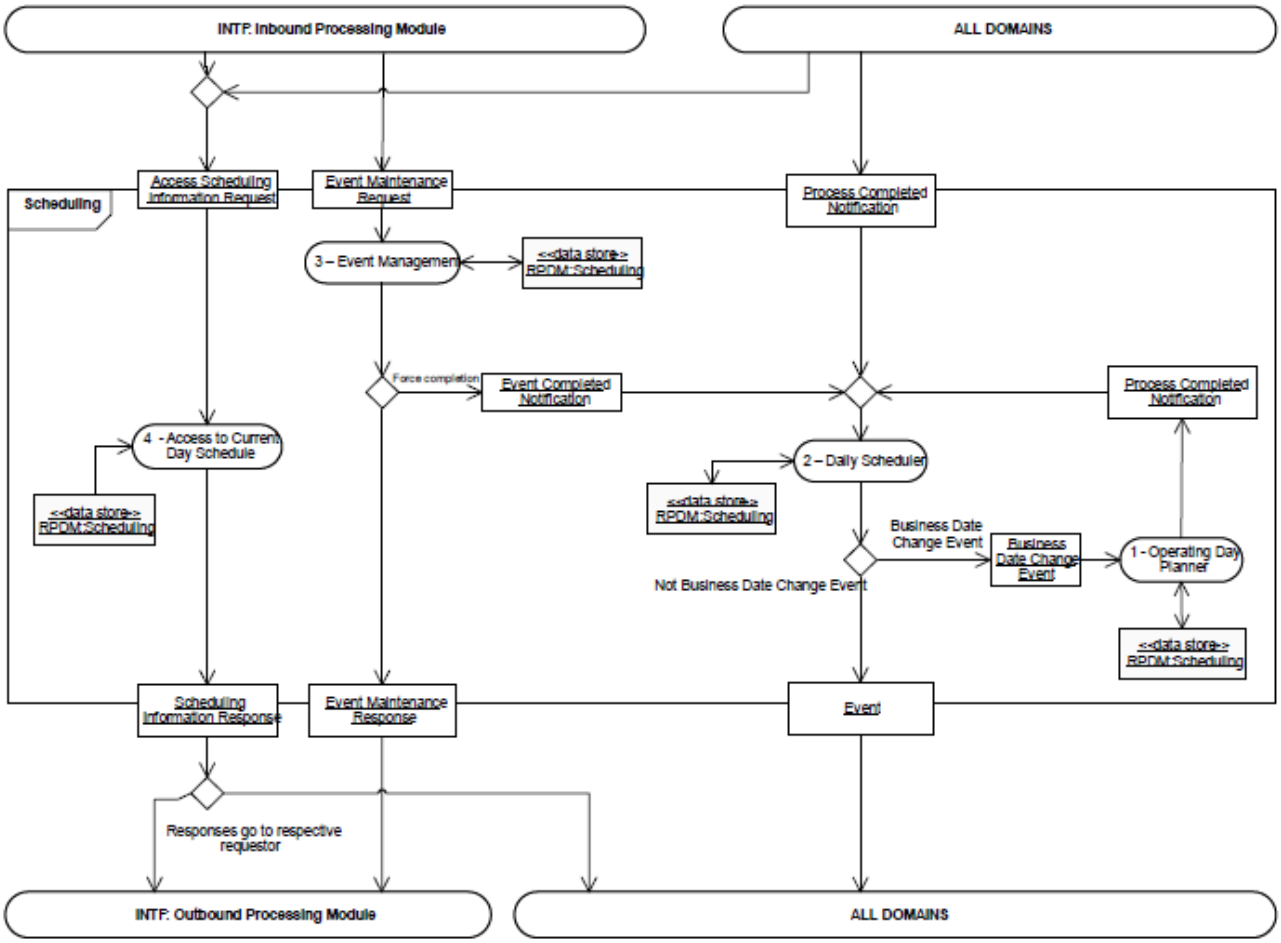
ATTRIBUTE	DESCRIPTION
Type	It specifies the status type of the operating day. The only possible value for this attribute is "Business Status".
Value	It specifies the status value assigned to the status type. The exhaustive list of possible values for the "Business Status" status type is as follows: ▮ SOD Period; ▮ First Night-time Cycle (specifying progressive number); ▮ Last Night-time Cycle ▮ Maintenance Window ▮ Real-time settlement period; ▮ Real-time settlement closure period; ▮ EOD Period.

Each occurrence of *T2S System Status History* is linked to the relevant occurrence of *T2S System Status*.

All rights reserved.

3.8.2.2 Static Functional description

Diagram of the module



Description of the module

The *Scheduling* module manages the operating day *Events* and the daily *T2S System Status* change in T2S.

The scheduler activates planned *Events* based on time only (started simply upon reaching the planned time) or based on a sequence (started upon completion of all linked predecessors and, if needed, upon reaching the Event Scheduled Time). Furthermore, the T2S Operator may generate unplanned *Events* and insert them into the daily plan throughout the day by specifying a scheduled time. These unplanned Events may not include links to specific predecessors or successors; they may either be inserted without any predecessor/successor links or as successors to all Events that precede them based on the scheduled time.

Events can be related to processes that need to be triggered within other modules (e.g. the revalidation process in the *Instructions Validation* module) or within an external workload manager application. The *Scheduling* module does not directly activate these processes; its scope is limited to generating and sending to the relevant module a notification that a certain *Event* has been reached. The appropriate module then manages the related processes. Once an *Event* reaches its Event Scheduled Time (and all the relevant predecessor events are completed), the *Daily Scheduler* function generates a notification message for the Event and routes it to the relevant module (or within the *Scheduling* module) in order to trigger the related pro-

All rights reserved.

cesses. Depending on the *Event Type*, the *Scheduling* module might also expect a notification from the relevant module once the processes complete. In this case, the scheduler does not consider the *Event* completed until it has received the responses from all the relevant modules.

At the end of each operating day, the Daily Scheduler function sends a *Business Date Change Event* to the *Operating Day Planner* function, which then sets the new Business Date for the system and generates the schedule for the next operating day. Each operating day is divided into different phases represented by one Status (e.g. *"EOD Period"*, *"Maintenance Window"* and so forth) **{T2S.03.010}**. The Status of the operating day is updated by the *Daily Scheduler* function on the basis of the completion of some specific *Events* triggering specific status transitions.

An example is: The completion of the *"Start of 1st Night-time Cycle" Event*, meaning the completion of all the processes triggered by this *Event*, shall trigger a status transition for changing the Status of the operating day from *"Start of Day Period"* to *"1st Night-time Cycle"*.

It is also possible to have *Events* that are generated dynamically throughout the operating day or inserted manually by the T2S Operator.

Besides the Event Scheduled Time, each *Event* is characterised by its sequencing dependencies with other *Events* of the same operating day schedule. If present, a dependency postpones the execution of the successor *Event* (and the related processes) until the relevant predecessor *Events* are completed [211]. Only after all predecessors of a successor complete the Scheduling module trigger the successor. These dependencies have priority over the actual scheduled time of the *Events*; i.e. if *Event* A is a predecessor and *Event* B is a successor, the latter must wait for the completion of the former even if this means starting past its scheduled time. Furthermore, for Event Types which foresee this possibility, the T2S operator can link specific *Events* (both scheduled by default and inserted dynamically in the current operating day schedule) to specific currencies **{T2S.03.303}**.

T2S has an internal business date, independent of the system date in the operating system. When setting a new *Business Date*, the *Scheduling* module takes into account the *Closing Days*, which are skipped. The use of Closing Days allows the T2S Operator to manually intervene on the business date change process **{T2S.11.010}**. Furthermore, the T2S operator is able to modify the current operating day event schedule during run-time, by inserting, updating, rescheduling and forcing completion of *Event* instances. In the T2S production environment, manual interventions which influence the business date are limited to business contingency situations.

The *Scheduling* module notifies all the *Events* which cause a change in the operating day Status to the *Interface* domain, so that the latter can send a message indicating the new Status at each change, according to the message subscription configuration for each interested party **{T2S.13.136}**.

Below is a tentative list of the possible *Events* that may appear on the daily schedule. Note that the following list should not be considered exhaustive [212]:

211 "Completion of an event" means the completion of the process triggered by the event itself.

212 Refer to the annex on the T2S operational day for a description of the interdependencies between events.

All rights reserved.

EVENT TYPE	DESTINATION DOMAIN:MODULE	CURRENCY-SPECIFIC
Beginning of Day-time	SETT: Standardisation and Preparation to Settlement INTF: Inbound Processing LCMM: Instruction Validation	No
Beginning of Night-time	(not communicated to external domains)	No
Business Date Change	OPSR: Scheduling	No
Event (according to an entry in SD Report Configuration) for report creation is reached	SQRA: Report Management	No
End of cycle	LCMM: Status Management INTF: Inbound Processing	No
End of cycle reporting	SQRA: Report Management	No
End of Day	(not communicated to external domains)	No
End of night-time reporting	SQRA: Report Management	No
Billing request	OPSR: Billing	No
EoD intraday credit reimbursement	SETT: Auto-Collateralisation	No
EoD Liquidity Transfers	INTF: Inbound Processing LQMG:CB Business Procedures	No
EoD Statements and Reports	SQRA: Report Management	No
Start of cycle n / sequence 0	INTF: Inbound Processing SETT: Night-Time Settlement Manager LQMG: Liquidity Operations	No
Start of cycle n / sequence 1/2/3/4	INTF: Inbound Processing LQMG: Liquidity Operations LCMM: Instruction Validation LCMM: Instruction Matching SETT: Night-time Settlement Manager	No
Start of cycle n / sequence X	LCMM: Instruction Validation SETT: Night-Time Settlement Manager	No
Start of cycle n / sequence Y	LQMG: Liquidity Operations SETT: Night-Time Settlement Manager	No

All rights reserved.

EVENT TYPE	DESTINATION DOMAIN:MODULE	CURRENCY-SPECIFIC
Start of cycle n / sequence Z	LQMG: Liquidity Operations SETT: Night-Time Settlement Manager	No
Start of Day	CRDM: All LCMM: Instruction Validation SETT: Standardisation and Preparation to Settlement	No
Start of SD Update	CRDM: All	No
Start Queuing System Status	INTF: Inbound Processing Module	No
End Queuing System Status	INTF: Inbound Processing Module	No
Stop Collection Creation	SETT: Standardisation and Preparation to Settlement	No
Condition for standing/predefined liquidity transfer orders	LQMG: Liquidity Operations	No
Archived Data Physical Deletion	SQRA: Legal Archiving	No
EoD cash settlement restrictions release	SETT: Standardisation and Preparation to Settlement	No
EoD Recycling and Purging	LCMM: Instruction Maintenance	No
Intraday DVP cut-off	LCMM: Instruction Validation LCMM: Instruction Matching SETT: Standardisation and Preparation to Settlement	Yes
Intraday FoP cut-off	LCMM: Instruction Validation LCMM: Instruction Matching SETT: Standardisation and Preparation to Settlement	No
Intraday BATM cut-off	LCMM: Instruction Validation LCMM: Instruction Matching SETT: Standardisation and Preparation to Settlement	Yes
Intraday CBO cut-off	LCMM: Instruction Validation LCMM: Instruction Matching SETT: Standardisation and Preparation to Settlement	Yes

All rights reserved.

EVENT TYPE	DESTINATION DOMAIN:MODULE	CURRENCY-SPECIFIC
Intraday cash settlement restrictions/ securities settlement restrictions cut-off	LCMM: Instruction Validation SETT: Standardisation and Preparation to Settlement	No
Matching Allegement Sending	LCMM: Status Management	No
Start of partial settlement window	SETT: Day-time Recycling and Optimisation	No
Stop of partial settlement window	SETT: Day-time Recycling and Optimisation	No
External system available	INTF: Communication	No
External system unavailable	INTF: Communication	No
Reference Data Update Reporting	SQRA: All	No
EoD statistical information	SQRA: Statistical Information	No
Maintenance window started	INTF: Inbound Processing	No
Extraction for archiving	SQRA: Legal Archiving	No

As said before, depending on the *Event Type*, the *Scheduling* module might also expect a notification from the relevant module once the processes complete. The following table provides a tentative list of these notifications, not be considered exhaustive at this stage:

PROCESS COMPLETED NOTIFICATION	SOURCE DOMAIN:MODULE
End of Process Event (EOD Recycling and Purging)	LCMM: Instruction Maintenance
End of Process Event (End of Cycle)	LCMM: Status Management
Event (End of Process)	LQMG: Liquidity Operations
Process Completed Notification	CRDM : All
End of Process (EOD Intraday Credit Reimbursement)	SETT: Auto-collateralisation
End of Process (Start of Cycle / Sequence)	SETT: Night-time Settlement
End of Process Intraday DVP Cut-off	SETT: Standardisation and Preparation to Settlement
End of Process Intraday FOP Cut-off	SETT: Standardisation and Preparation to Settlement
End of Process Intraday DATM Cut-off	SETT: Standardisation and Preparation to Settlement

All rights reserved.

PROCESS COMPLETED NOTIFICATION	SOURCE DOMAIN:MODULE
End of Process Intraday CBO Cut-off	SETT: Standardisation and Preparation to Settlement
End of Process Intraday Cash Settlement Restriction Cut-off	SETT: Standardisation and Preparation to Settlement
End of Process Intraday Securities Settlement Restriction Cut-off	SETT: Standardisation and Preparation to Settlement
End of Process EOD Cash Restriction Release	SETT: Standardisation and Preparation to Settlement
End of Process Stop Collection Creation	SETT: Standardisation and Preparation to Settlement
End of Process SOD	SETT: Standardisation and Preparation to Settlement
End of Process Beginning of Daytime	SETT: Standardisation and Preparation to Settlement

Description of the functions of the module

At the end of each operating day, the *Operating Day Planner* function calculates the new *Business Date* and generates the schedule for the next business day. Throughout the operating day, the *Daily Scheduler* function triggers *Event* instances once the scheduled time is reached and/or the predecessor *Events* are completed. The Access to Current Day Schedule function provides information to other domains and to the users, while the T2S Operator uses the *Event Management* function for manual interventions.

1 – Operating Day Planner

Reference Id	***OPSR.SCH.BDP.1.1***

An EOD event activates the *Operating Day Planner* function in the daily schedule. It calculates the new *Business Date* **{T2S.11.005}** by increasing the current Business Date by 1 day, skipping Saturdays, Sundays and dates included in the entity *Closing Day*. This allows the T2S Operator to define a Closing Day within Reference Data whenever there is the need to manually intervene on the business date change process **{T2S.11.010}**. The function then loads in the *Event* entity the new operating day schedule on the basis of the *Default Event Schedule* of the current *Operating Day Type* **{T2S.11.020}**. Upon successful completion of the procedure, the function sends a notification to the *Daily Scheduler* function, marking the successful completion of the process.

2 – Daily Scheduler

The *Daily Scheduler* function has two purposes, described below:

Start Next Scheduled Processes

Reference Id	***OPSR.SCH.DSP.1.1***

All rights reserved.

A dedicated long running process triggers this function on a regular basis (e.g. every minute) to perform a check and identify the *Event* instances that have been reached, depending on the current operating day schedule. The function performs a check on both the Event Scheduled Time and possible predecessors for each *Event* **{T2S.11.040}**. Firstly, it checks the system time against the scheduled time for each *Event*. Then, if an *Event* has reached its Event Scheduled Time and it has any predecessors, the *Daily Scheduler* function also checks that all predecessor events have been successfully completed. If this is the case, then the Daily Scheduler function generates a notification message for the Event and routes it to the relevant module or function, so that the related process (transaction or workload manager application) can be started **{T2S.11.030}**.

When an Event is reached, the function also updates the *Event* instance Event Start Time for the current operating day. As part of the processes to trigger, certain *Events* may modify the current T2S business Status (e.g. the completion of the *"Start of 1st Night-time Cycle" Event*, meaning the completion of all the processes triggered by this *Event*, shall trigger a status transition changing the Status of the operating day to *"1st Night-time Cycle"*).

The *Daily Scheduler* function manages the Status changes based on the Event Type of the relevant completed *Events*.

Receive Start Next Scheduled Processes

Reference Id	***OPSR.SCH.DSP.1.2***

Once *Scheduling* module triggers a process, the function expects notifications from the relevant modules when the process are completed. Once the function receives all relevant notifications, it updates the Event End Time for the current operating day.

Once the EOD cut-off time is reached for the EOD/SOD procedure, the *Daily Scheduler* function generates the EOD time *Events*, based on their Event Schedule Time and any dependencies with other *Events*, and sends them to the relevant T2S domains to activate the related EOD processes.

3 – Event Management

Reference Id	***OPSR.SCH.ODM.1.1***

The *Event Management* function allows the T2S Operator to perform manual changes to the schedule of the current day **{T2S.11.040}**. It is possible for the T2S Operator to perform the following operations:

- Insert a new *Event* instance;
- Change a single Event Scheduled Time (provided it has not been reached yet) by moving the *Event* forward or backward in the current day schedule. If the updated *Event* is moved past or ahead of another *Event*, the modification is still carried out and a warning message is displayed to warn the T2S Operator of this change in priorities. If the *Event* contains predecessor or successor constraints, these are automatically changed based on the *Event*'s new relative position in the daily schedule (e.g. if a predecessor Event is moved past its successor, the priority constraint between those two Events ceases to exist);
- Reschedule a portion of the daily schedule (i.e. an *Event* and all *Events* following it in the daily schedule, provided the first *Event* has not been reached yet), without changing their dependencies;

All rights reserved.

- Force completion (close) of an *Event* instance before its actual occurrence. The effect of this action is simply to generate an *Event Completed Notification* to the *Daily Scheduler* function indicating that the *Event* is complete. The management of the related processes is left entirely up to the relevant modules. With regards to *Events* which are related to processes triggered in other modules, two cases are possible. Manual completion can be done when the *Event* has not yet been reached (in which case the *Scheduling* module sends no notification to the relevant module once the Event Scheduled Time is reached) or if it is already started but not yet completed (in which case the *Event* is considered complete by the *Scheduling* module, but the actual process is managed by the relevant module). Forcing the completion of an *Event* causes all dependent *Events* (successors) to be executed as if the first one had been completed normally. The T2S operator may employ this feature exclusively with its use limited to testing and contingency situations.

4 – Access to Current Day Schedule

Reference Id	***OPSR.SCH.ACS.1.1***

This function provides detailed information on the schedule of the current operating day in response to enquiries by T2S Actors or other domains.

Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Access Scheduling Information Request	In	Request for information concerning current operating day schedule	All domains, INTF: Inbound Processing Module	
Event Maintenance Request	In	Manual intervention from T2S operator or event owner on current operating day schedule	All domains, INTF: Inbound Processing Module	
Scheduling Information Response	Out	Data concerning current operating day schedule		All domains, INTF: Outbound Processing Module
Event Maintenance Response	Out	Outcome of manual intervention		All domains, INTF: Outbound Processing Module
Event	Out	Event to be delivered to the relevant domain/module.		All domains
Process completed notification	In	Notification of process completion received by the relevant domain/module.	All domains	

All rights reserved.

Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
REFERENCE DATA			
Scheduling Rules and Parameters	Default Event Schedule	Read	Default schedule based on dependencies between T2S processes
	Operating Day Type	Read	Identifier of operating day type. Used to generate the T2S operating day schedule
	Event Type	Read	Information on the various types of Operating Day Events
	Event Currency	Read/Write	Valid currency/ies for a specific Event.
	T2S Closing Day	Read/Write	Contains information on the days on which the T2S system is closed. Information is organised by date and a flag to determine whether a closing day is recurrent (on an annual basis) or a one-time occurrence. This information is used in the determination of the new business date.
DYNAMIC DATA			
Scheduling Dynamic Data	Operating Day	Read/Write	Information necessary to manage the current Operating Day plan. When a new Operating Day starts, the table is compiled with the time schedule extracted from the Reference data Default Event Schedule
	T2S System Status	Read/Write	Contains the current business date of T2S to be provided all over the T2S system. At each business date change, the new business date is calculated in-

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
			creasing by 1 the previous one and taking into account information from static entity T2S Closing Day (i.e. closing days are skipped). It also contains the current T2S business status.

3.8.2.3 Events (EV) Use Cases

Scope

This category of use cases describes the possible scenarios in which the *Scheduling* module is activated.

Criteria

The criteria used to identify the use cases are: Action Type, and Action Category. The possible criteria values are:

CRITERIA	POSSIBLE VALUES	COMMENT
Action Type	Event Management, Daily Scheduler	Type of request to be processed.
Action Category	Insert, Update, Complete, Execute	Category of request to be processed

3.8.2.4 Processing of Events Use Cases

Processing of Events Use Cases

The processing of the following representative use cases is described:

- UC-EV-1 Event Management;
- UC-EV-2 Daily Scheduler.

Processing of UC-EV-1 Event Management

This use case allows a T2S Operator to insert/update/complete *Event* instances in the daily plan, where the operating day is identified in terms of *Event* instances planned for the specified *Business Date* to be triggered.

For the new instance of a specific *Event* inserted it is possible to update the Event Scheduled Time. The update is allowed not only for a single *Event* but also for a set of instances.

The T2S Operator can complete an *Event* instance planned for the current *Business Date*. This causes the *Event* to be marked with an Event End Time timestamp, which corresponds to completing it in the daily plan. Such an *Event* is not triggered upon being reached.

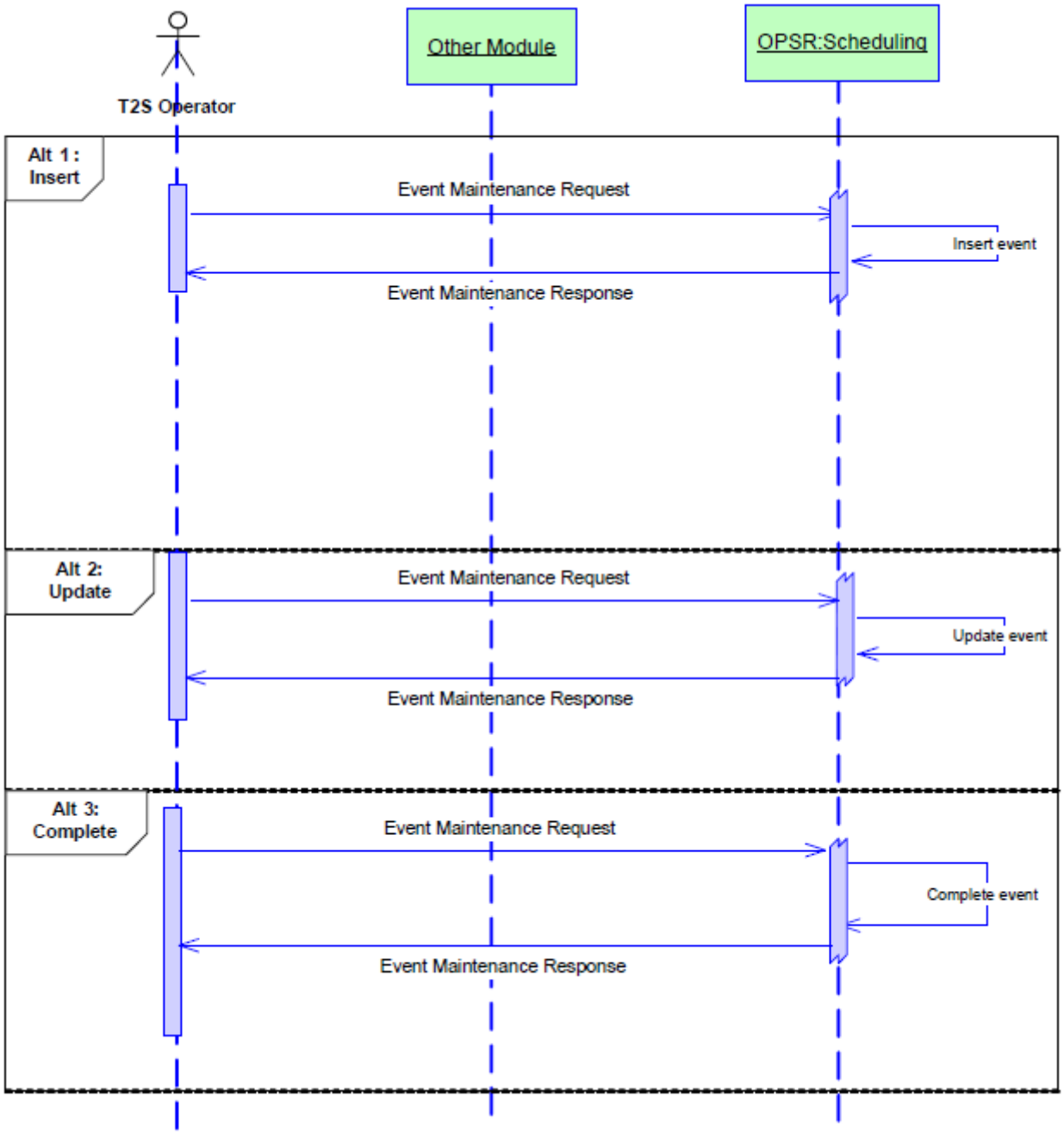
Business assumption

As far as Event Management use case is concerned, one possible scenario is identified with three possible types of request: insert, update and complete of planned event.

All rights reserved.

Processing

The following sequence diagram details the scenario:



Processing of UC-EV-2 Daily Scheduler

On a regular basis, *Events* to be triggered are identified and they are notified to the relevant modules.

Business assumption

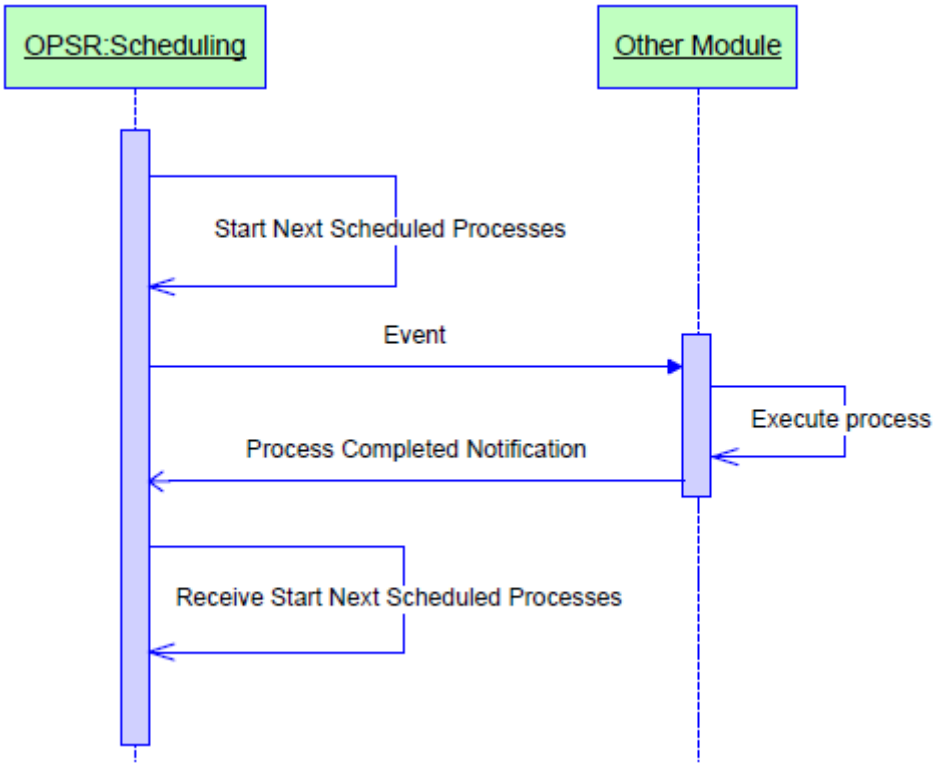
The actual date/time for an *Event* is reached [213].

213 The actual triggering time of an event might be different from the planned scheduled time owing to the presence of predecessor events not yet completed (see the description of the Daily Scheduler function for more information).

All rights reserved.

Processing

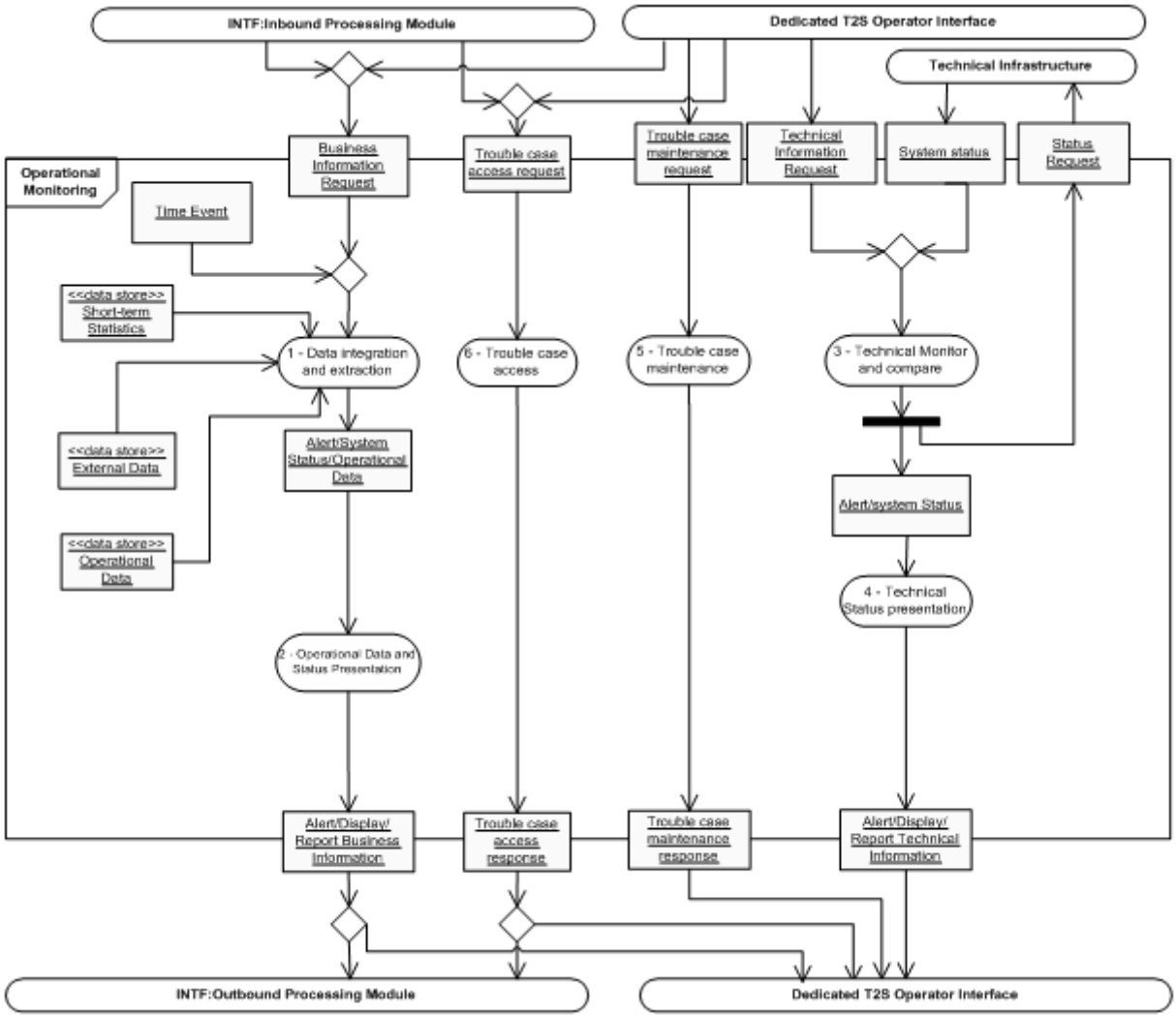
When the actual date/time of an *Event* is reached, the *Event* is notified to the relevant module, so that it can trigger the relevant process. Upon completion of the process, the interested module notifies back this information to the *Scheduling* module.



All rights reserved.

3.8.3 Operational Monitoring

3.8.3.1 Diagram of the module



3.8.3.2 Description of the module

The purpose of this module [214] is to provide the T2S operator with the capabilities to perform:

- Operational monitoring: the detection of functional or operational problems in real-time, monitoring of service performance indicators, and provision of information for crisis management scenarios;
- Technical monitoring: the detection of hardware and software problems through the real-time monitoring of all the technical components the T2S platform uses, including the network connections.

The *Operational Monitoring* module provides the following set of high-level functions:

214 Owing to the specificity of this module, dedicated to the operational monitoring of the platform, the description of its functions is more focused on the technical features and the type of information provided for this purpose, rather than detailing the business logic of such functions.

All rights reserved.

- Observing the behaviour of the T2S production environment and raising alarms at the appropriate level of priority and at the earliest point in time, in case deviations from the normal operating day are detected, where the normal operating day is derived from the behaviour of the system over a defined time period;
- Reporting aggregated, up-to-date report of the monitoring information to be routed and used for problem determination and solving by the appropriate team in the organisation of the T2S operator;
- Notifying the status of the T2S platform to the T2S operator so that the T2S operator can trigger the appropriate corrective actions, when required;
- Logging all information required for reporting Service Performance Indicators in order to produce Service Information reporting and Monthly Service Level Agreement reporting;
- Providing and tracking information about the status of an incident or problem, logging its history, and documenting the analysis and solution;
- Providing up-to-date, meaningful and comprehensive information to the crisis manager for the decision making process in crisis and contingency situations.

This module also provides access to other T2S system users for specific tasks. For example, CSDs and authorised T2S parties will have online access to the Trouble Management System.

The operational monitoring uses its own specific interfaces and authentication and authorisation mechanisms for the access of and interaction with the T2S operator, owing to the specific technical architecture and features of such tools. Therefore, the "dedicated T2S operator interface" is not an additional dedicated interface for the T2S operator, but rather the specific interfaces provided with these specific technical tools).

Operational Monitoring integrates data extracted from the short-term statistical repositories and from the T2S operational data base with that of other external sources of information, i.e. with news and markets data feed provided by financial channels, in order to obtain a meaningful overview of the status of the T2S platform for crisis management. News services needed are all which are directly (e.g. financial and economic information) or indirectly (e.g. and inter alia breaking news, international politics) related to the system management in its broadest approach with the aim to allow a proactive and prompt response.

3.8.3.3 Description of the functions of the module

1 – Data integration and extraction

Reference Id	***OPSR.OPM.DIE.1.1***

The aim of this function is to integrate data coming from different sources (the operational data base, the "short-term" statistical information repositories and external data sources as described above), extracting, merging and organising data in the format required for the presentation to the T2S system users both in push and pull mode (e.g. raising alarms or answering to specific requests), in order to monitor the operational and technical status of the T2S actors [215] for activities related to T2S and the performance of the platform for the real time verification of the respect of the agreed service levels.

215 The operational and technical status of a T2S actor may be, for example, the number of messages sent by a directly connected participant in the last x minutes or simply the status (open, closed) of another system interacting with T2S.

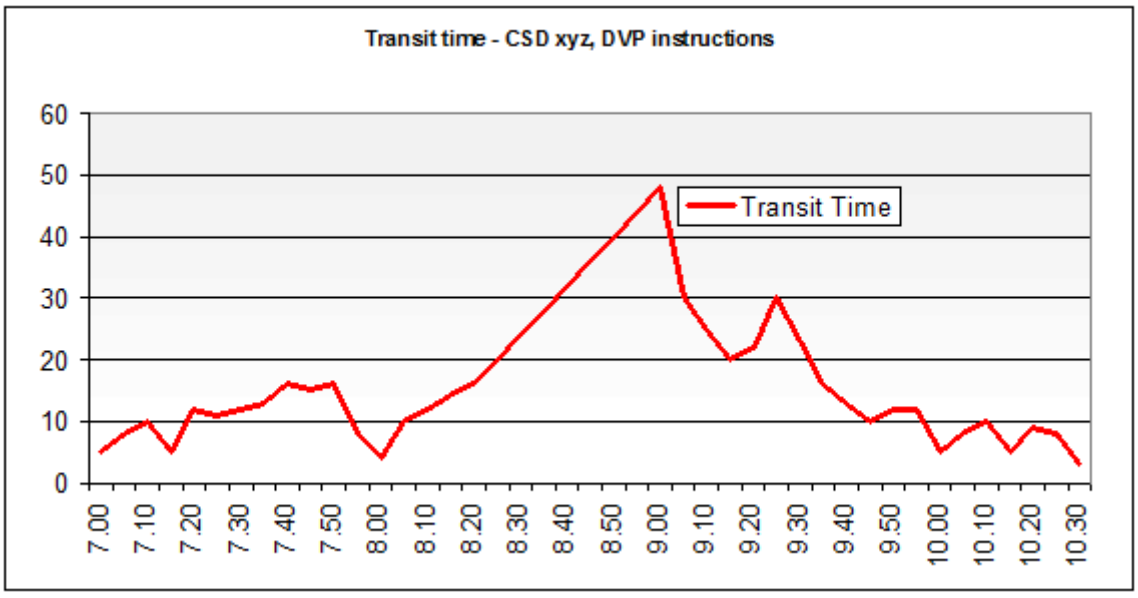
All rights reserved.

This function also performs the data retrieval for the Monthly Service Level Agreement reporting and the Service Information reporting **{T2S.20.100} {T2S.20.110}**. The first report is related to all the indicators defined in the SLA documentation, whereas the second includes some additional indicators, out of the scope of the Service Level Agreement, that can be used as an input for the improvement of the services provided.

The function queries the performance-related data stored in the "short-term" statistical information repositories to compare the current behaviour of the platform (and of its components) with a set of predefined thresholds and with data related to the last three months, in order to signal to the T2S operator any deviation from the normal expected behaviour (see the *Operational Data and Status Presentation* function for more information on the possible ways this information can be provided to the T2S operator). In what follows, a non exhaustive possible list of key indicators for different T2S components is described, together with the criteria used to raise alarms for the T2S operator, when applicable. The following list is tentative and should not be considered exhaustive; the final list will be detailed on the basis of the Service Level Agreement.

Overall T2S monitoring

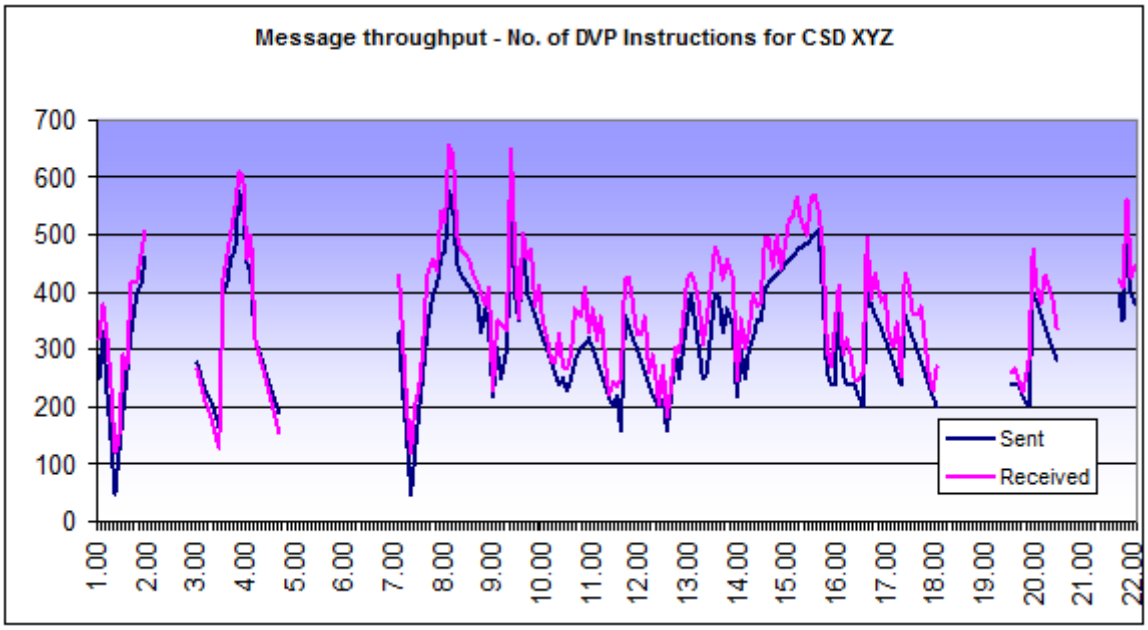
Indicators belonging to this area provide information on the different message flows and ongoing processes in T2S - the actual timestamps to be considered are to be defined as part of the Service Level Agreement. For instance:



- Instruction transit time - The aim of this indicator is to monitor the trend of instructions processing on the T2S platform during the T2S operating day. It is defined as the maximum interval, measured in 1 minute, between the arrival of the relevant input message and the dispatch of the related notification. Instructions not processed for legitimate reasons are excluded. In case of irregular and excessive increase of the concerned value, an alarm is raised. The preceding diagram provides an example showing the transit time related to a specific CSD and an instruction type.

All rights reserved.

- Instruction transit time breakdown – These indicators provides more detailed figures concerning the different processing steps of an instruction within T2S, taking into account the subsequent timestamps of these steps (e.g. receipt of the message, technical acceptance of the message, creation of the business information in the back-end module or rejection resulting from a failed business validation and so forth). Pre-defined reference thresholds can help also in this case to identify what components of the T2S platform are experiencing a performance issue.
- Incoming / outgoing messages – These indicators are defined as the number of messages arrived to (or sent from) the platform within the last x minutes/hours. If these figures differ by a certain amount from the pre-defined reference thresholds, an alarm is raised. The following diagram provides an example showing the incoming and outgoing message related to a specific CSD and message type for the current operating day.



- Most active T2S parties. This indicator provides the list of the x most active T2S parties, based on different parameters (e.g. number of settled instructions, total settled amount, number of sent messages).
- Network alarms. Several alarms can be defined in order to cope with many different network problems. For instance, specific alarms can be raised when:
  - there are messages that cannot be sent from the platform to the network (i.e. an acknowledgement from the network is still missing);
  - there are messages that cannot be delivered to the T2S actor (i.e. an acknowledgement from the T2S actor is still missing);
  - the connection with TARGET2 (or other RTGS systems) or collateral management systems is not working properly (i.e. there are messages that can not be sent from the platform to the external system or vice versa);

Scheduling module monitoring

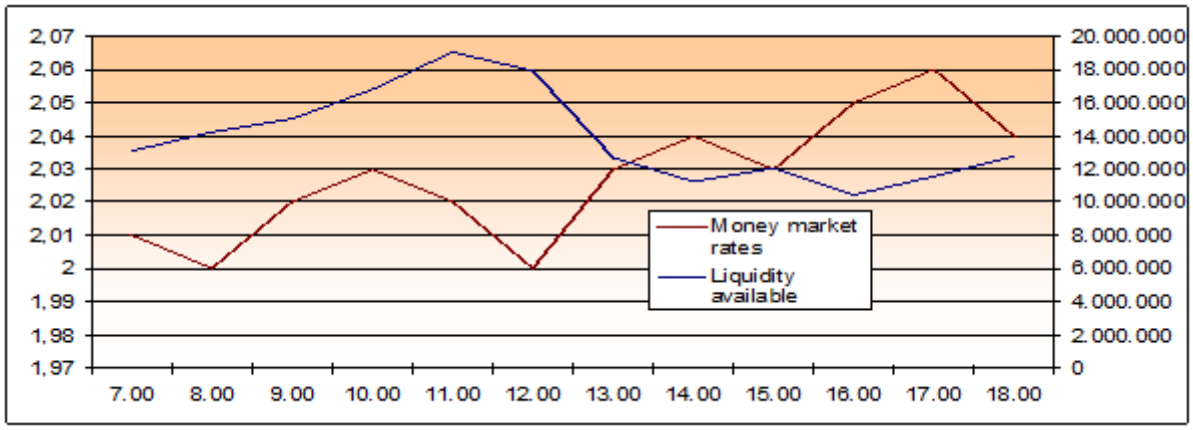
All rights reserved.

Monitoring in this area provides information about the current T2S business status, i.e. detailed information about planned, revised and actual timestamps of the scheduled events. Monitoring raises the appropriate alarms when scheduling does not trigger, or complete events by the scheduled deadline.

Liquidity management domain monitoring

Indicators belonging to this area provide information concerning liquidity availability and usage within the platform throughout the operating day. For instance:

- Available liquidity. This indicator shows the amount of available liquidity per CB and/or currency, i.e. of all the available liquidity in T2S dedicated cash accounts owned by T2S parties per CB and/or currency. The following diagram provides an example showing the trend of the available liquidity for a specific CB, compared with the money market rates.



- Intraday liquidity. This indicator provides information concerning the usage of intraday liquidity during the operating day and can be helpful in order to have an early warning of possible problems in the flow of liquidity.
- Most active payment/settlement banks. This indicator provides the list of the x most active payment banks, based on different parameters (e.g. number of settled payments, total settled amount, and number of sent messages).

Settlement domain monitoring

Indicators belonging to this area provide information on the behaviour and the performance of the T2S settlement engine. For instance:

- Last transaction settlement timestamp. If the time span between this indicator and the current time is greater than a given threshold (that can be different for different time periods of the T2S operating day), an alarm is raised.
- Securities settlement volume. This indicator is calculated both in absolute figures (number of settled instructions, pending matched instructions, pending settlement transactions) and percentages (based on the total number of instructions and transactions processed by T2S).
- Securities settlement amount. This indicator is calculated both in absolute figures (total settled amount) and percentages (based on the total amount related to instructions processed by T2S).

All rights reserved.

- Optimisation algorithm performance. It refers to information about the current status of the algorithm and its performance (processing time and number of settled transactions per each settlement attempt).

Lifecycle management and matching domain monitoring

This area provides a view on the lifecycle of an instruction, graphically representing the different paths that a *Settlement Instruction* or maintenance instruction can take through the system. For each processing step, an indicator displays the volume of instructions currently in the status related for that processing step (e.g. number of unmatched settlement instructions for the current operating day). If one or more of these indicators reach an abnormal value compared to a predefined threshold an alarm is raised.

Interface domain monitoring

Indicators belonging to this area provide information on the different inbound and outbound message flows through the T2S *Interface*. For instance:

- Inbound communication. These indicators are defined as the number of messages pending technical validation or the number of messages pending business validation.
- Four-Eyes operations. This indicator is defined as the number of pending Four-Eyes approvals.
- Query indicators. These indicators are defined as the number of queries and the average query response time per type of query and within the last x minutes/hours.
- In all cases, when these figures differ by a certain amount from the pre-defined reference thresholds, an alarm is raised.

Reference Data domain monitoring

- Indicators belonging to this area provide information on the reference data changes. For instance:
- Night-time reference data changes. This indicator is defined as the number of reference data changes that have to be processed between the current night-time settlement sequence and the next one.
- Next operating day reference data changes. This indicator is defined as the number of reference data changes that will become valid from the next operating day.
- In all cases, when these figures differ by a certain amount from the pre-defined reference thresholds, an alarm is raised. Alerts may also be defined on the basis of ad hoc consistency checks to performed either intra-day (on request) or at pre-defined points in time during the operating day.

Trouble management system monitoring

Indicators belonging to this area provide information on the performance and workload of the Trouble Management System, e.g.

- Trouble management performance. This indicator measures, for each trouble case category (request/ incident/problem), the average time between the opening and closure of a trouble case. Indicators are also separated by case priority.
- Trouble cases volume. This indicator measures the amount of cases in a status other than "closed" in any given moment; it may be monitored or only produced on demand. Cases are organised by category.

All rights reserved.

- Maximum resolution time. This indicator monitors the time span from the opening of the oldest non-closed case. If this value surpasses a set threshold (which is defined separately for each case priority), an alarm is raised.

2 – Operational data and status presentation

Reference Id	***OPSR.OPM.ODP.1.1***

This function presents the output of the monitoring activity in an organised and human-readable form to the T2S operator **{T2S.20.040}**, according to the following features:

- Delivery: Data delivery and updates are processed in real-time push mode, without additional delay, filtering, aggregation or other processes that could affect the quality of data.
- Interfaces: The T2S operator can access data using a web browser or a native client.
- Formats: The relevant formats are pages (plain text information such as a sentence reporting on the status of a specific T2S component, or an alarm stored in a dedicated log), tables and graphs.
- Comparative monitoring: It is possible to plot different time series in the same screen.
- Customise front-end: The presentation of data can be customised by the T2S system user.
- Export: It is possible to export data as plain and/or formatted text for further processing with other tools (e.g. MS Office applications).
- Alarms: The application can generate alarms triggered by the crossing of configurable thresholds; the triggered alarms remain active as long as the relevant triggering condition is present. It is also possible to view a history log of all raised alarms, including their start and end timestamps **{T2S.20.130}**. Alarms can be raised as a visual alert in a monitoring screen, an entry in dedicated log or an e-mail sent to a pre-defined list of interested recipients.

Reports related to Service Information reporting and Monthly Service Desk Management Information reporting are provided in this function **{T2S.20.100}** **{T2S.20.110}** and are based on the data retrieved by the *Data integration and extraction* function.

3 – Technical monitoring and compare

Reference Id	***OPSR.OPM.TMC.1.1***

This function provides the T2S operator with the possibility to monitor every single component of the T2S platform with real-time information. This includes all hardware and network devices, plus all the software components of the platform, including operating systems (CPUs usage, memory usage), data base management systems (memory usage, tablespaces), web application servers, and all the processes related to the T2S applications. The information provided helps the T2S operator to identify any malfunction in the platform; the available information is role-based, with different data and levels of detail shown depending on the roles and privileges granted to each T2S system user. Moreover, this information is detailed enough to allow the T2S operator to understand the business impact of the malfunction and the underlying technical implications, so to provide the T2S operator with an indication as to the actions to be taken to solve the relevant incident **{T2S.20.130}**.

All rights reserved.

Information (e.g. operational and technical status, alarms) can be presented and grouped at different levels (i.e. entire T2S platform, domain, module, function).

4 – Technical status presentation

Reference Id	***OPSR.OPM.TSP.1.1***

This function provides a graphical display of the components of the T2S platform and their status in order to provide a global view on the whole system for monitoring purposes. From an aggregated view of the processes it is possible to drill down on a single process or sub-process and obtain specific status displays with increasing levels of detail. When an alarm is raised, the relevant process in the technical status view immediately displays an alert; by opening the detailed view it is possible to read the alert message and receive information on the event that triggered it. Finally, it is possible to automatically send e-mail notifications to the T2S operator when an alarm is raised.

5 – Trouble case maintenance

Reference Id	***OPSR.OPM.TCM.1.1***

This function encompasses aspects related to the management and processing of trouble cases via a specific Trouble Management System (TMS) application {**T2S.20.070}**. Trouble cases may be defined as:

- Service request: a case opened to submit requests to the T2S support teams, usually for information on any specific issue or to request interventions that are part of standard operation (e.g. running specific applications or issuing new user passwords). Request cases are not linked to interruptions or reductions in the quality of the service.
- Incident: a case reporting any event which does not fall within standard operation and causes an interruption or reduction in quality of the service. The resulting intervention is limited to the individual Incident and the case is closed upon restoring normal service operation; further investigations and interventions, if necessary, may be carried out by opening a linked Problem case.
- Problem: a case opened to further the analysis related to one or more Incidents after they have been closed, normally to investigate the underlying cause and minimise or nullify possible future impacts **{T2S.20.140}**.

Regardless of their category, a Case ID number along with information detailing the specific case identifies a trouble case. A specific field displays the status of the case (whether it is a newly opened case, assigned to a specific T2S support unit, being worked on, resolved or closed) and the team it is currently assigned to for treatment. Each case records timestamps for each intervention, most notably for its opening and closure. A "security impact" flag is available to mark the cases which have potential bearing on security issues. Finally, it is possible to categorise trouble cases by increasing levels of priority, depending on the urgency of the matter, the impact on the system's functionality and the timeframe by which a solution is expected.

6 – Trouble case access

Reference Id	***OPSR.OPM.TCA.1.1***

All rights reserved.

This function allows read-only access to Trouble cases, for informational purposes, to all T2S actors linked to a specific Trouble Case. While the T2S operator directly acts on Trouble Cases, this function allows all involved parties to constantly monitor the status of the case.

3.8.3.4 Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Business information request	In	Request from T2S Actor / Request from T2S Operator	INTF: Inbound Processing Module / Dedicated T2S Operator Interface	
Technical information request	In	Request from T2S Operator	Dedicated T2S Operator Interface	
System status	In	Information on the system status for monitoring and diagnostic purposes	Technical infrastructure	
Status request	Out	Request for information on the system status		Technical infrastructure
Alert/Display/Report Technical Information	Out	Information sent to the T2S Actor via the Interface domain		INTF: Outbound Processing Module
Trouble case maintenance request	In	Information needed to maintain (i.e. to create, update or delete) a trouble case.	Dedicated T2S Operator Interface	
Trouble case maintenance response	Out	Information concerning the result of a specific trouble case maintenance request		Interface / Dedicated T2S Operator Interface
Trouble case access request	In	Information needed to retrieve a specific trouble case	Interface / Dedicated T2S Operator Interface	
Trouble case access response	Out	Information concerning a trouble case to be sent to the T2S Actor or the T2S Operator		Interface / Dedicated T2S Operator Interface

All rights reserved.

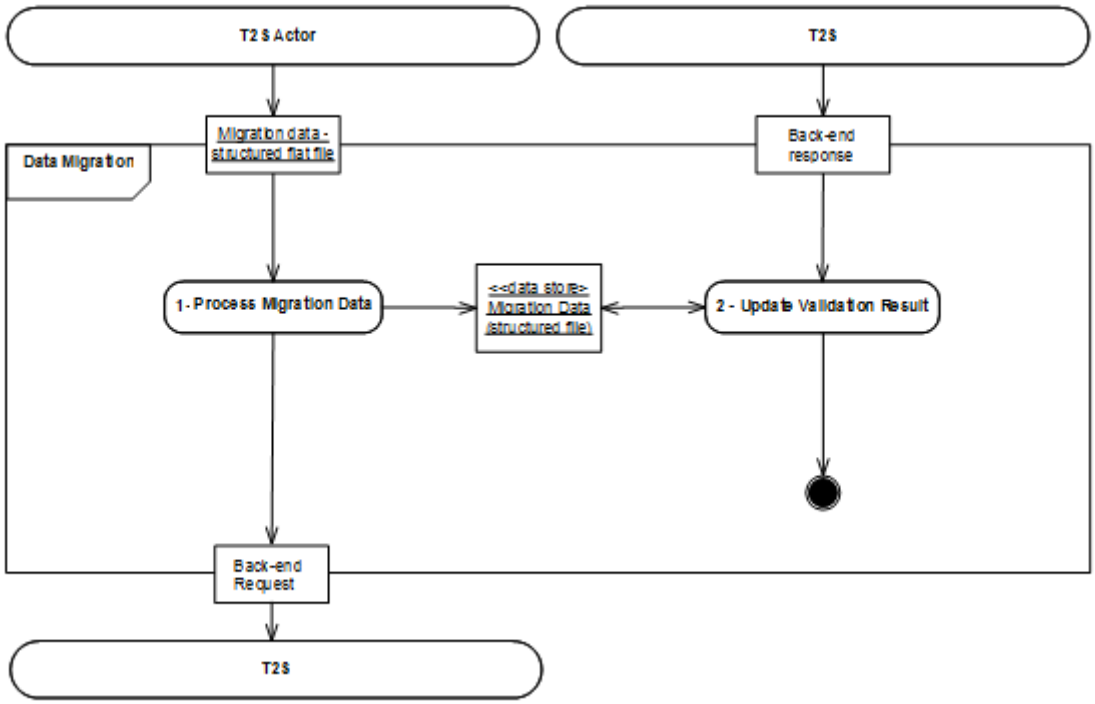
3.8.3.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENT
REFERENCE **DATA**			
Short-term Statistics	n/a	Read	Short-term statistical data to support the T2S operator for the proper management of the system.
External Data	n/a	Read	Information from external sources to complement T2S data, such as news and markets data feeds
Operational Data	n/a	Read	T2S operational reference data needed for operational monitoring purposes.
DYNAMIC DATA			
Short-term Statistics	n/a	Read	Short-term statistical data to support the T2S operator for the proper management of the system.
External Data	n/a	Read	Information from external sources to complement T2S data, such as news and markets data feeds
Operational Data	n/a	Read	T2S operational dynamic data needed for operational monitoring purposes.

All rights reserved.

3.8.4 Data Migration

3.8.4.1 Diagram of the module



3.8.4.2 Description of the module

Introduction

The Data Migration in general covers the need to migrate data from the respective CSDs and CBs to T2S **{T2S.21.280} {T2S.16.490}**. These data include:

- Party reference data;
- Securities reference data;
- Securities account reference data;
- T2S Dedicated Cash account data;
- Rules and parameters:
  - Restriction Types,
  - Users,
  - Conditional Securities Delivery Rule sets,
  - Message Subscription Rule sets,
  - Report Configurations;
- FOP Settlement Instructions (to be executed for the generation of the initial Securities positions) {T2S.21.320};
- Pending Settlement Instructions;

All rights reserved.

- Settlement Restrictions.

The CSDs and CBs can send their data for the migration via the standard channels of communication:

- Via A2A using the supported standard XML messages or files;
- Via U2A using the standard T2S GUI functionality to enter data required for its migration;
- Alternatively, the CSDs can use a dedicated date migration tool to upload structured files (flat files, Excel files) **{T2S.21.290}**. The T2S Operator interacts with the data migration tool application to release those structured files general formal checks and back-end modules to process.

The data migration tool adds to the migration data (structured file) the result of the T2S back-end modules processing.

Enriched structured files, including migration status, are made available to the related CSD/CB via the data migration tool application.

The Data Migration module consists of the following two functions:

- *Process Migration Data;*
- *Update Validation Result.*

3.8.4.3 Description of the functions of the module

1 – Process Migration Data

Reference Id	***OPSR.DAM.CMD.1.1***

After having received the Migration Data (structured flat file) the *Process Migration Data* function

- Stores the incoming structured file in the Migration Data data store;
- Checks formal structure of single records;
- Forwards single records proper request to the related back-end module for processing ;

In case the format check of a single data record is not possible, the function data migration tool marks the record as rejected and continues with the release of following data records.

2 – Update Validation Result

Reference Id	***OPSR.DAM.UVR.1.1***

The *Update Validation Result* function:

- Receives and stores processing results from the standard processes of the different modules described in the GFS;
- Adds for each single data record a status in the Migration Data store which specifies the result of the migration along with reported errors;
- Makes available the enriched structured file to the originating CSD/CB.

All rights reserved.

3.8.4.4 Description of the Input/Output of the module

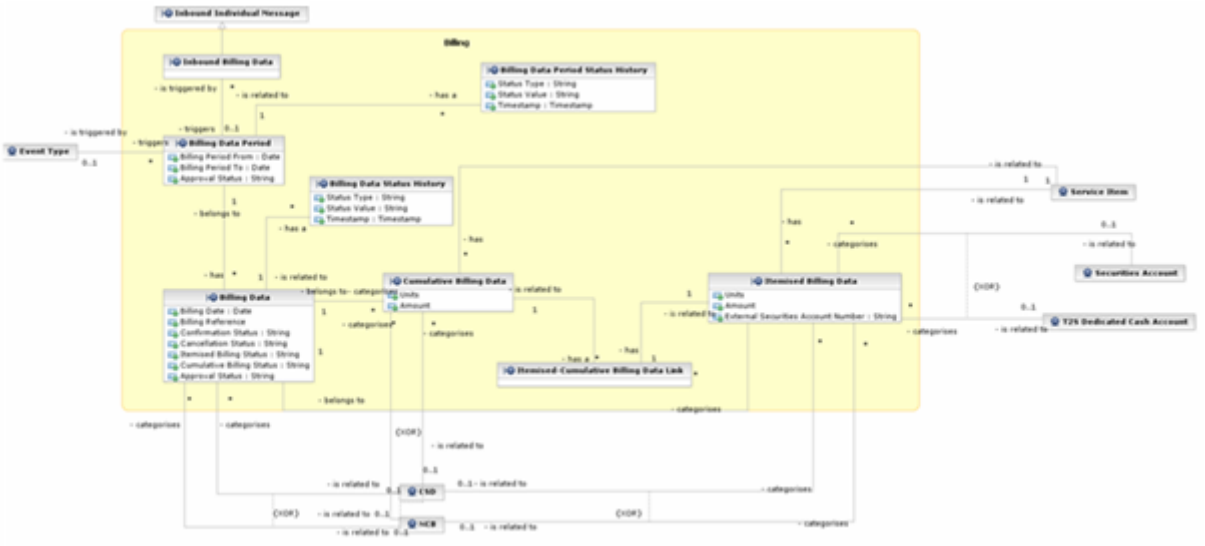
FLOW	IN/OUT	DESCRIPTION	FROM	TO
Migration Data –structured flat files	IN	Structured Migration Data Files	T2S Actor	
Back-end response	IN	Responses resulting from the standard back-end modules process.	T2S	
Back-end request	OUT	Request for the back-end module out of a single record of the migration data.		T2S

3.8.4.5 Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
Migration Data (structured file)		Read/Write	Stored for each single data set of a Structured Migration Data File

3.8.5 Billing

3.8.5.1 Dynamic Data managed by the module



3.8.5.2 Description of the entities

1 – Inbound Billing Data

All rights reserved.

It is a child entity to inbound individual message of the *Interface* domain where the invoice processing request and the invoice approval request are stored.

2 – Billing Data Period

ATTRIBUTE	DESCRIPTION
Billing Period From	Start date of the billing period (contained in the Billing Event or the Invoice Processing Request)
Billing Period To	End date of the billing period (contained in the Billing Event or the Invoice Processing Request)
Approval Status	Approval Status for Invoice Processing requests "generation". For the concept of Approval Status in case of Four-Eyes principle and the associated transition diagram see description in chapter "T2S Graphical User Interface".

3 – Billing Data Period Status History

ATTRIBUTE	DESCRIPTION
Status Type	Type of status which indicates the status of processing. Possible value is approval status (for invoice processing requests "generation")
Status Value	For the concept of Approval Status in case of Four-Eyes principle and the associated transition diagram see description in chapter "T2S Graphical User Interface"
Timestamp	System date and time of the status update

4 – Billing Data

ATTRIBUTE	DESCRIPTION
Billing Date	Issue date of the bill. It is the current business date when the confirmation takes place
Billing Reference	Unique reference of the bill. The reference is automatically generated
Confirmation status	Possible values are (exhaustive list): not confirmed confirmed
Cancellation status	Possible values are (exhaustive list): not cancelled cancelled
Itemised billing status	Possible values are (exhaustive list): not generated generated
Cumulative billing status	Possible values are (exhaustive list): not generated

All rights reserved.

ATTRIBUTE	DESCRIPTION
	generated
Approval status	For the concept of Approval Status in case of Four-Eyes principle and the associated transition diagram see description in chapter "T2S Graphical User Interface".

5 – Billing Data Status History

ATTRIBUTE	DESCRIPTION
Status Type	Type of status which indicates the status of processing. Possible values are (exhaustive list): ▌ Confirmation status ▌ Cancellation status ▌ Itemised billing status ▌ Cumulative billing status ▌ Approval status (for invoice processing request "confirmation" or "cancellation")
Status Value	Value of the status depending on the Status Type. Possible values are (exhaustive lists): In case of Status Type = "Confirmation status": ▌ not confirmed ▌ confirmed In case of Status Type = "Cancellation status": ▌ not cancelled ▌ cancelled In case of Status Type = "Itemised billing status": ▌ not generated ▌ generated In case of Status Type = "Cumulative billing status": ▌ not generated ▌ generated In case of Status Type = "Approval Status" For the concept of Approval Status in case of Four-Eyes principle and the associated transition diagram see description in chapter "T2S Graphical User Interface".
Timestamp	System date and time of the status update

6 – Itemised Billing Data

All rights reserved.

Itemised Billing Data provides aggregated information on the level of single *securities account*(s) or *T2S Dedicated Cash Account*(s) and on Party level.

ATTRIBUTE	DESCRIPTION
Units	Total number of units per service item.
Amount (excl. VAT)	Calculated amount (excl. VAT) (units multiplied with unit price)
External Securities Account Number	Used in External Settlement Scenario (to provide precise data on itemised level)

7 – Cumulative Billing Data

Cumulative Billing Data provides aggregated information on the level of CSD or CB. The hereafter listed attributes are sums of the underlying *Itemised Billing Data* attributes.

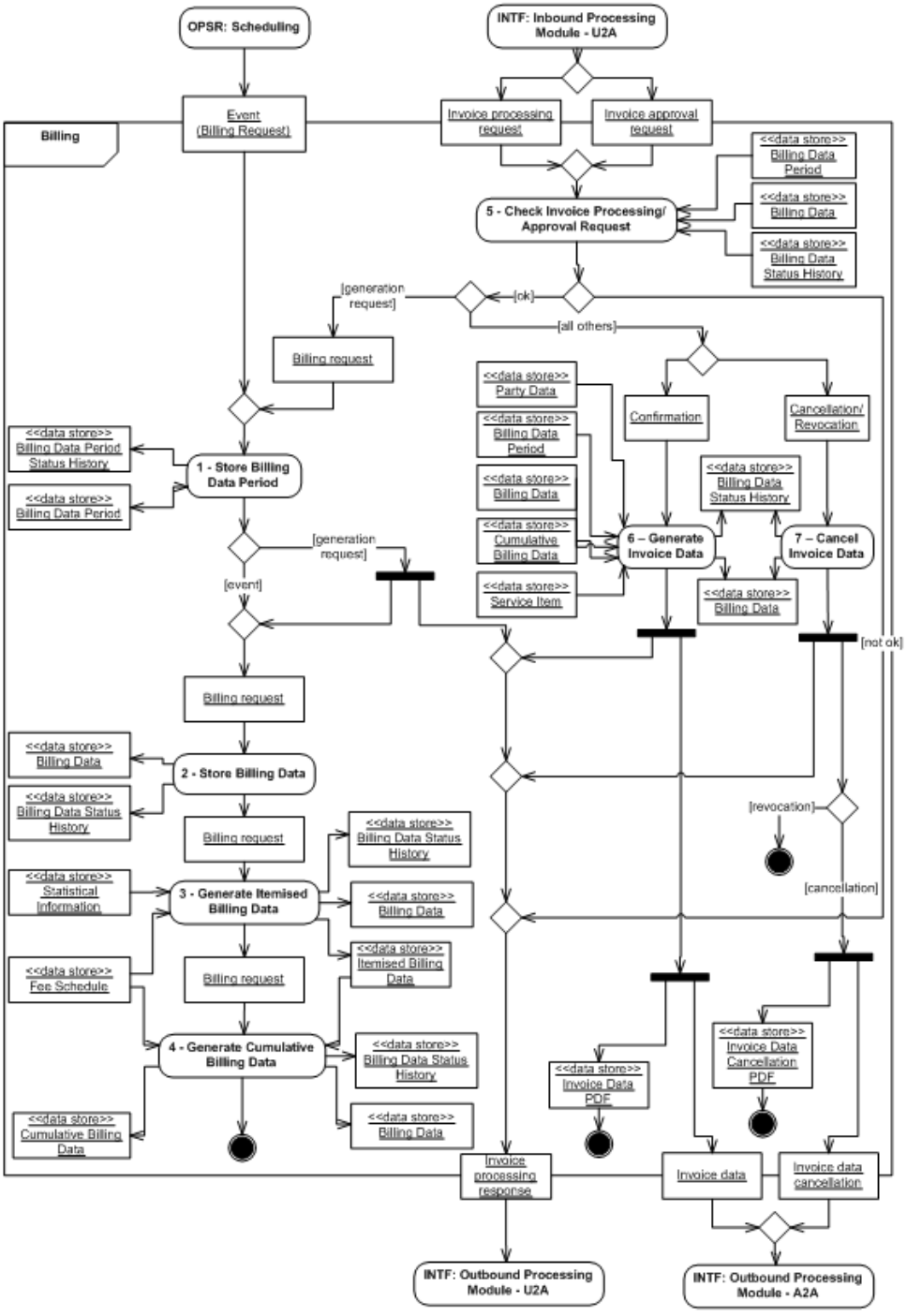
ATTRIBUTE	DESCRIPTION
Units	Sum of total number of units per service item.
Amount	Sum of Amount

8 – Itemised-Cumulative Billing Data Link

The entity contains the association between *Itemised Billing Data* and *Cumulative Billing Data* to document which *Itemised Billing Data* is included in which *Cumulative Billing Data*.

All rights reserved.

3.8.5.3 Diagram of the module



3.8.5.4 Description of the module

Introduction

The *Billing* module in T2S provides the functionality to generate invoices for CSDs and CBs. CSDs and CBs invoice their participants and customers respectively, based on the information that T2S provides and that CSDs and CBs may complement with additional data {**T2S.15.110**} {**T2S.15.050**}.

All rights reserved.

The *Billing* module distinguishes between:

- *Itemised Billing Data*: The *Itemised Billing Data* consists of the aggregated information on the level of a single securities, dedicated cash account or Party level within a given period for each service item.
- *Cumulative Billing Data*: The *Cumulative Billing Data* consists of the aggregated information on the level of a CSD or CB for a specified period for each service item.
- *Invoice*: The *Invoice* contains the *Cumulative Billing Data* and is the invoice according to legal aspects, which is sent to a CSD or CB.

The module generates the *Itemised Billing Data* and *Cumulative Billing Data* automatically on a monthly basis {**T2S.15.120}**. The Billing Event is sent at the first day after the billing period (i.e. the first day of the next month),by the *Scheduling* module which triggers this process {**T2S.15.100}**.

In exceptional circumstances, a privileged T2S Actor can initiate the generation of *Itemised Billing Data* and *Cumulative Billing Data* for CSDs or CBs on an ad-hoc basis.

The *Itemised Billing Data* and *Cumulative Billing Data* are based on the information in the *Statistical Information* module, which aggregates the service items on a daily basis**.**

After the generation of the *Itemised Billing Data* and *Cumulative Billing Data* a privileged T2S Actor can:

- Confirm for generation and sending of the invoice;
- Cancel a single invoice or all invoices in the case of errors (amendments to already generated invoices are not possible);

This module does not provide any functionality for account receivables management.

Delivery of the Invoice

T2S supports the delivery of the invoice:

- Via A2A as an XML message in the push mode;
- In paper-based format or as PDF via E-mail, depending on the legal requirements of the CSD or CB (created by the privileged T2S Actor user based on a request of the CSD or CB).

Invoice Query

The *Query Management* module allows CSDs and CBs to query their *Itemised Billing Data* and the *Cumulative Billing Data* {**T2S.13.290} {T2S.15.130}**.

Basis for Invoice Calculation

The fee schedule, a data store managed in the *Common Reference Data Management component*, defines the tariff structure and unit price for each service item and the billing frequency providing the basis for the invoice calculation {**T2S.15.140}**.T2S allows authorised T2S System Users to change the fee schedule at any time, but such updates takes effect only at the beginning of the next billing period.

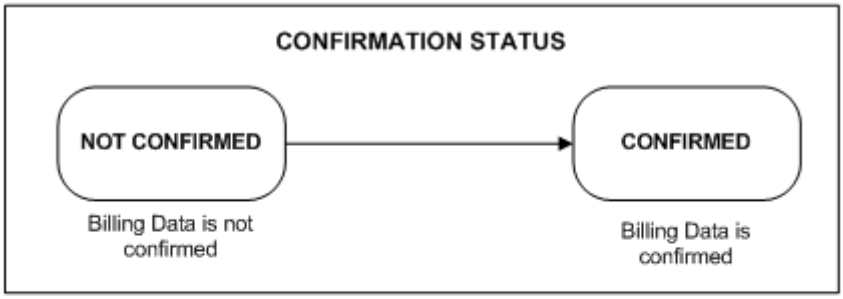
Status transition diagram

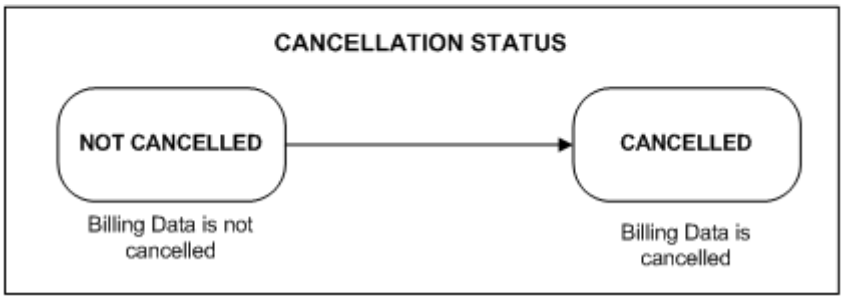
Within the *Billing* module the following statuses are managed for the *Itemised Billing Data* and *Cumulative Billing Data* (the status of the *Itemised Billing Data* and *Cumulative Billing Data* can not be different):

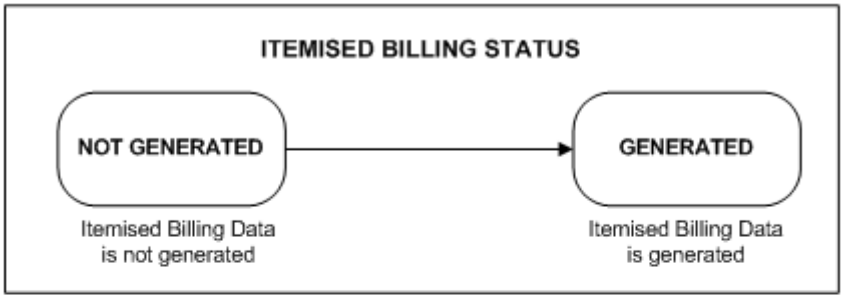
All rights reserved.

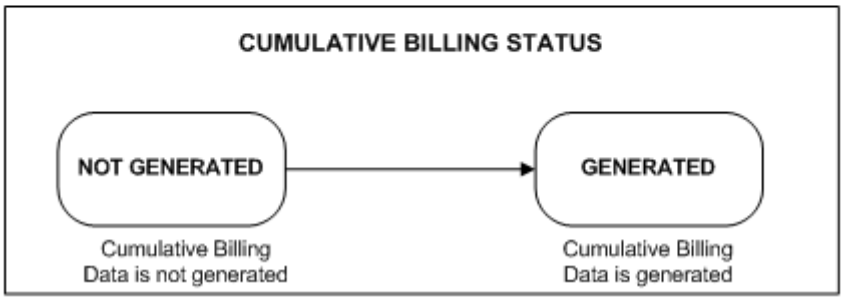
- Confirmation status specifies whether a privileged T2S Actor has confirmed the *Itemised Billing Data* and *Cumulative Billing Data* so that the *Billing* module can send them as Invoice to a CSD or CB;
- Cancellation status specifies whether a privileged T2S Actor has cancelled the *Itemised Billing Data* and *Cumulative Billing Data*;
- Itemised billing status specifies whether Billing has generated the *Itemised Billing Data* for a specified billing period and billable CB or CSD;
- Cumulative billing status specifies whether Billing has generated the *Cumulative Billing Data* for a specified billing period and billable CB or CSD;
- Approval status keeps track related to the Four-Eyes principle (please see chapter 3.2.6.3 "Four-Eyes principle" for further information).

All rights reserved.









CONFIRMATION STATUS	DESCRIPTION
"Not confirmed"	Billing Data is not confirmed
"Confirmed"	Billing Data is confirmed

CANCELLATION STATUS	DESCRIPTION
"Not cancelled"	Billing Data is not cancelled
"Cancelled"	Billing Data is cancelled

All rights reserved.

ITEMISED BILLING STATUS	DESCRIPTION
"Not generated"	Itemised Billing Data is not generated
"Generated"	Itemised Billing Data is generated

CUMULATIVE BILLING STATUS	DESCRIPTION
"Not generated"	Cumulative Billing Data is not generated
"Generated"	Cumulative Billing Data is generated

3.8.5.5 Description of the functions of the module

1 – Store Billing Data Period

Reference Id	***OPSR.BIL.SBP.1.1***

This function receives either an Event (*Billing Request*) flow from *Scheduling* module or *Billing request* flow from *Check Invoice Processing/Approval Request* function stemming from Invoice processing request (generation).

The function:

- Stores the Billing Data Period data (Billing Period From and Billing Period To) in the *Billing Data Period* data store and sets in the *Billing Data Status History* and in the *Billing Data* data store the approval status ;
- Generates an *Invoice processing response* flow to inform the T2S Actor that the actor's request has been accepted only when the flow stems from an Invoice processing request (generation);
- Forwards the *Billing request* flow to the *Store Billing Data* function.

2 – Store Billing Data

Reference Id	***OPSR.BIL.SBD.1.1***

This function receives a *Billing request* flow from *Store Billing Data Period* function.

The function:

- Stores for each CSD or CB the Billing Data (i.e. the Billing Reference which is an automatically generated unique reference) in the *Billing Data* data store;
- Sets in the *Billing Data Status History* and in the Billing Data data store the confirmation status to "*not confirmed*" and the cancellation status to "*not cancelled*";
- Sets in the *Billing Data Status History* and in the Billing Data data store the itemised billing status and cumulative billing status to "*not generated*";
- Forwards the *Billing Request* flow to the *Generate Itemised Billing Data* function.

3 – Generate Itemised Billing Data

Reference Id	***OPSR.BIL.GBD.1.1***

All rights reserved.

This function receives a *Billing Request* flow from *Store Billing Data* function.

The function:

- Calculates the total number of units per service item for each securities account or dedicated cash account and on Party level, based on the *Statistical Information Data* data store related to the requested billing period **{T2S.15.060} {T2S.15.070}** {**T2S.15.080}** {**T2S.15.090}**;
- Calculates the amount based on the total number of units per service item and the relevant data from the *Fee Schedule* data store;
- Stores the result of the calculation in the *Itemised Billing Data* data store;
- Sets in the *Billing Data Status History* and in the Billing Data data store the Itemised Billing Status to "*Generated*";
- Forwards the *Billing request* flow to the *Generate Cumulative Billing Data* function.

4 – Generate Cumulative Billing Data

Reference Id	***OPSR.BIL.GCD.1.1***

This function receives a *Billing Request* flow from *Generate Itemised Billing Data* function.

The function:

- Sums up the total number of units per service item for each CSD or CB based on the *Itemised Billing Data* data store;
- Sums up the Amount based on the total number of units per service item for each CSD or CB based on the *Itemised Billing Data* data store;
- Sets in the *Billing Data Status History* and in the Billing Data data store the Cumulative Billing Status to "*generated*";
- Stores the result of the calculation in the *Cumulative Billing Data* data store {**T2S.15.130}**.

At this point the generation of the billing data ends. The T2S Actor proceeds with entering an Invoice processing request.

5 – Check Invoice Processing/Approval Request

Reference Id	***OPSR.BIL.CIP.1.1***

This function receives the following data flows:

- *Invoice processing request* for:
  - Generation
  - Confirmation
  - Cancellation
- *Invoice approval request* for:
  - Approval
  - Revocation

All rights reserved.

– (according to chapter 3.2.6.3 "Four-Eyes principle")

The function checks the incoming request. The check is based on the *Billing Data Period* data store, *Billing Data* data store and the *Billing Data Status History* data store.

When the check has

- a positive result:
  - The *Billing request* flow is forwarded to the *Store Billing Data Period* function;
  - The *Confirmation* flow is forwarded to the *Generate Invoice Data* function;
  - The *Cancellation/Revocation* flow is forwarded to the *Cancel Invoice Data* function;
- a negative result *an Invoice processing response flow is sent to the Outbound Processing module – U2A.*

6 – Generate Invoice Data

Reference Id	***OPSR.BIL.GID.1.1***

This function receives a *Confirmation* flow from *Check Invoice Processing/Approval Request* function.

The function consolidates the data for the invoice for each CSD or CB. The consolidation consists of the following data:

- General information related to issuer of the invoice from the list of configuration parameters stored as attribute domains:
  - e.g. contact details and account information for remittance of CSD or CB;
- *Party data of the* CSD or CB which is the recipient of the invoice (from the *Party Data* data store):
  - e.g. Party Long Name;
- General Billing Data from the *Billing Data Period* data store and the *Billing Data* data store:
  - e.g. Billing period (Billing Date From and Billing Date To), Billing reference;
- Invoice details (from the *Cumulative Billing Data* data store enriched with data from the *Service Item* data store):
  - The invoice details are categorised by *Service Item Group*;
  - For each *Service Item Group* the amount is summed up;
  - At the end the all the Service Item Group totals are summed are summed up as invoice totals.

Furthermore the function:

- Stores the Invoice Data as PDF in *Invoice Data PDF* data store;
- Forwards the *Invoice data* flow to the *Outbound Processing module – A2A*;
- Forwards the *Invoice processing response* flow to the *Outbound Processing module – U2A*;
- Sets in the *Billing Data Status History* and in the Billing Data data store the confirmation status to "*confirmed*" (the cancellation status is still "*not cancelled*").

The generation of the Invoice follows the same principles as used in *Report Management* module.

All rights reserved.

7 – Cancel Invoice Data

Reference Id	***OPSR.BIL.CID.1.1***

This function receives a *Cancellation/Revocation* flow from *Check Invoice Processing/Approval Request* function.

In case the *Cancellation/Revocation* flow is a cancellation, the function:

- Stores the Invoice Data Cancellation as PDF in *Invoice Data Cancellation PDF data* store;
- Sets in the *Billing Data Status History* and in the Billing Data data store the cancellation status to "*cancelled*";
- Forwards the *Invoice data cancellation* flow to the *Outbound Processing module – A2A*;
- Forwards the *Invoice processing response* flow to the *Outbound Processing module – U2A*.

In case the *Cancellation/Revocation* flow is a revocation, the function:

- Sets in the *Billing Data Status History* and in the Billing Data data store the approval status (according to chapter 3.2.6.3 "Four-Eyes principle");
- Forwards the *Invoice processing response* flow to the *Outbound Processing module – U2A.*
- At this point the revocation ends.

Description of the Input/Output of the module

FLOW	IN/OUT	DESCRIPTION	FROM	TO
Event (billing request)	IN	Event triggering the invoice creation automatically	OPSR: Scheduling	
Invoice processing request	IN	Request sent by privileged T2S system user to Generate Itemised Billing Data and Cumulative Billing Data Confirm Itemised Billing Data and Cumulative Billing Data Cancel Itemised Billing Data and Cumulative Billing Data	INTF: Inbound Processing Module – U2A	
Invoice approval request	IN	Second user action in Four-Eyes Principle sent by privileged T2S		INTF: Inbound Processing Module – U2A

All rights reserved.

FLOW	IN/OUT	DESCRIPTION	FROM	TO
		system user to confirm or revoke the initial request for Generate Itemised Billing Data and Cumulative Billing Data Confirm Itemised Billing Data and Cumulative Billing Data Cancel Itemised Billing Data and Cumulative Billing Data		
Invoice data	OUT	Data for Invoice(s) in order to create XML message in the Interface domain		INTF: Outbound Processing Module – A2A
Invoice processing response	OUT	Result of initial request and second user action for Generate Itemised Billing Data and Cumulative Billing Data Confirm Itemised Billing Data and Cumulative Billing Data Cancel Itemised Billing Data and Cumulative Billing Data		INTF: Outbound Processing Module – U2A
Invoice data cancellation	OUT	Data for cancelled Invoice(s) in order to create XML message in the Interface domain		INTF: Outbound Processing Module – A2A

Data accessed by the module

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
Statistical Information		Read	All data relevant for billing
Billing Data Period		Read/Write	

All rights reserved.

DATA	DATA ENTITIES	ACCESS MODE	COMMENTS
Billing Data Period Status History		Read/Write	
Billing Data		Read/Write	
Billing Data Status History		Read/Write	
Itemised Billing Data		Read/Write	
Cumulative Billing Data		Read/Write	
Fee Schedule		Read	
Party Data		Read	
Service Item		Read	
Invoice Data PDF		Write	
Invoice Data Cancellation PDF		Write	

3.8.5.6 Billing (BI) use cases

Scope

This category of use cases describes the possible use of the Billing module:

- Automatically generation triggered by event (on a monthly basis);
- Generation triggered by T2S Actor (on an ad hoc basis);
- Confirmation for generation and sending of the Invoice;
- Cancellation of an Invoice.

Criteria

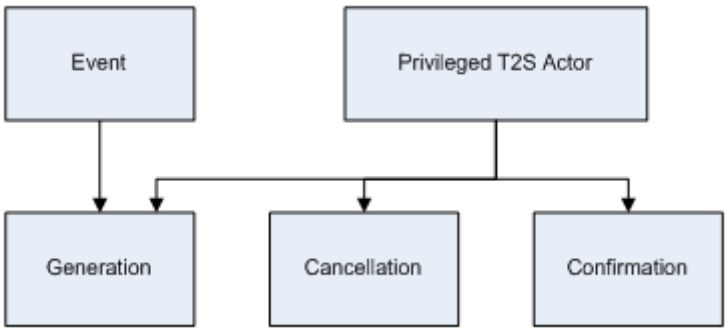
The criteria used to identify the use cases are the activation request and the action type to be performed. The possible values for these criteria (exhaustive list) are:

CRITERIA	POSSIBLE VALUES	COMMENT
Activation request	Event, privileged T2S Actor	Trigger of the possible action types
Action type	Generation, Cancellation, Confirmation	Type of request to be processed

List of Use Cases

The criteria described above are reported in the following tree:

All rights reserved.



The complete list of use cases is presented in the appendix.

3.8.5.7 Processing of Billing use cases

Selection of representative BI use cases

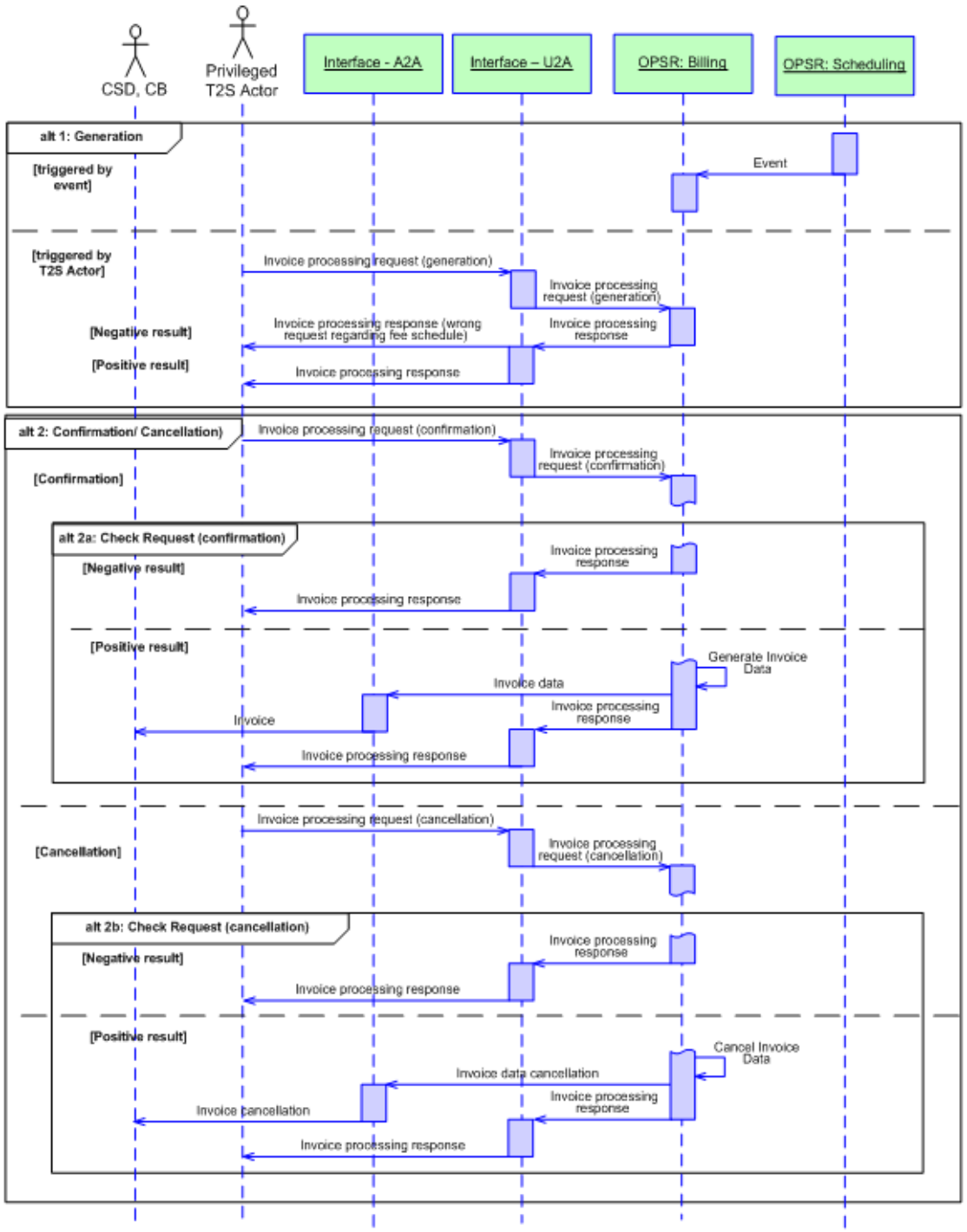
The processing of the following representative use case is described:

- UC-BI-1: Generation, Confirmation/ Cancellation of billing data and send Invoice

Processing of UC-BI-1: Confirmation of generated billing data

All rights reserved.

DIAGRAM 48 - GENERATION, CONFIRMATION/CANCELLATION OF BILLING DATA AND SEND INVOICE



Business assumption

Please note that all communication between the *Interface* domain and T2S Actors should be understood as:

- "Encrypted inbound communication" or "Encrypted outbound communication" for A2A mode;
- "XHTML request via HTTPS" or "XHTML response via HTTPS" for U2A mode.

All rights reserved.

as described in the *Interface* chapter. For the sake of readability, the diagram above uses the more specific terms for these flows.

Processing

Generation

Either:

- An event at the end of the billing period from the Scheduling module;
- A privileged T2S Actor

triggers the generation of Itemised and Cumulative Billing Data through an Invoice Processing Request (generation).

After successful technical validation of the Invoice processing request (generation) in the *Interface* Domain, the Billing Module generates and stores the Itemised and Cumulative Billing Data and forwards the Invoice processing response to the *Interface* Domain.

Confirmation

After the generation of the *Itemised Billing Data* and *Cumulative Billing Data*, a privileged T2S Actor can confirm the generation and sending of the Invoice through an *Invoice Processing Request* (confirmation).

After successful technical validations of the *Invoice processing request* (confirmation) in the *Interface* Domain, the *Billing* Module checks the request.

In case of negative result the *Billing* Module forwards an Invoice processing response to the *Interface* Domain.

In case of positive result the *Billing* Module generates the Invoice data, stores the Invoice Data as PDF and forwards the Invoice data and an Invoice processing response to the *Interface* Domain.

The *Interface* Domain forwards:

- The Invoice to the CSD or CB;
- The Invoice processing response to the T2S Actor.

Cancellation

After the generation of the Itemised and Cumulative Billing Data, a privileged T2S Actor can cancel the invoice through an *Invoice Processing Request* (cancellation).

After successful technical validations of the *Invoice processing request* (cancellation) in the *Interface* Domain, the *Billing* Module checks the request:

- In case of negative result the *Billing* Module forwards an Invoice processing response to the *Interface* Domain;
- In case of positive result the *Billing* Module cancels the Invoice data and forwards:
  - the Invoice data cancellation and;
  - an Invoice processing response;

to the *Interface* Domain.

All rights reserved.

The *Interface* Domain forwards:

- the Invoice cancellation to the CSD or CB;
- the Invoice processing response to the T2S Actor.

All rights reserved.

4 Appendices

4.1 List of Use Cases

4.1.1 Presentation: devising the use cases

This appendix presents the list of the identified T2S Use Cases, being understood as "An interaction between a user and a system or a component within a system by defining the discrete goal that the user wants to achieve with the system, without the requirement to reveal or to specify the system's internal behaviour" (T2S URD Glossary).

The approach for devising these use cases is based on the definition of the following categories covering the main situations of interaction of a T2S System user with T2S:

- SI: Settlement Instructions;
- MI: Maintenance Instructions (Hold/Release, Amendment and Cancellation instructions);
- SM: Reference Data Management;
- QU: Queries;
- RE: Reports;
- LT: Liquidity Transfers;
- EV: Events;
- LA: Legal Archiving;
- IN: Interface.

The description of each category of use cases, provided into the relevant sections of the main document includes:

- A short explanation of the scope of the category;

- The criteria that characterise a T2S System user interaction with T2S in this category: for instance, the criteria that characterise a use case of the "Reports" category are the communication mode, the information basis, the report triggering, the report type and the report sub-type;
- The possible values for each criterion. For instance, the possible values for the criterion "Report Type" are "Securities instructions", "balance" and "reference data".

On this basis, a T2S use case within a given category is defined as a combination of criteria values relevant from a business perspective.

For instance, a use case into the category "Report" can be defined as follows:

COMMUNICATION MODE	INFORMATION BASIS	TRIGGERING	TYPE	SUB-TYPE
A2A	On individual account	Event	Securities instructions	Statement of pending instructions

Such use case covers the situation where a report listing the pending securities instructions on a single account is triggered by a business event and accessed by a T2S system user in A2A mode.

Within a given category, the exhaustive list of use cases is therefore constituted of all the combinations of criteria values relevant from a business point of view. These combinations are provided in two different ways:

- In the description of the category provided into the main document, a decisional tree shows graphically the relevant combinations of criteria values;
- In this appendix, an exhaustive list of all the use cases in each category is provided.

4.1.2 SI: Settlement Instruction

The following criteria are omitted from the table, since all their combinations of values are possible:

- CSD configuration (Intra CSD, Cross-CSD, In/Out T2S);
- Partial (Yes/No);
- Auto-collateralisation (Yes/No);
- Communication mode (A2A/U2A).

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
1	No	No	DVP/DWP	Already Matched	DWP (already matched)	UC-SI-1:Settlement of a standard settlement instruction UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-9:Settlement of a corporate action with associated liquidity transfer UC-SI-10: Cross CSD/In-out CSD Standard DVP
2					DVP (already matched)	
3					Plain vanilla: CSDs already matched	
4					Securities lending: CSD already matched	
5					Auto-collateralisation	
6					Buy-in/sell out CSD (already matched)	
7					Market claim transaction: CSD (already matched)	
8					Back-to-back: CSD (already matched)	
9					Securities issuance and redemption (CE) (already matched)	
10					Redemptions (CE) (already matched)	
11					Cash distribution with securities delivery (CE) (already matched)	
12	No	Yes	DVP/DWP	Already Matched	Delivery by value (DBV) (already matched)	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-10: Cross CSD/In-out CSD Standard DVP
13					DVP already matched with linked instruction	

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
14	Yes	Yes	DVP/DWP	Already Matched	DVP already matched with linked instruction, conditional	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-6: Conditional Securities Delivery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
15	Yes	No	DVP/DWP	Already Matched	DVP already matched, conditional	UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-6: Conditional Securities Delivery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
16	No	No	DVP/DWP	Yes	Standard DVP (with matching)	UC-SI-3: Intra CSD-Standard DVP with Matching UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP
17					Plain vanilla: CSD Participant	
18					Securities lending: Participant	
19					DWP (with matching)	

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
20					Buy-in sell out CSD (with matching)	with Auto-collateralisation UC-SI-10: Cross CSD/In-out CSD Standard DVP
21					Buy-in/sell out: Participant	
22					Market claim transaction: CSD	
23	No	Yes	DVP/DWP	Yes	DVP with linked instruction	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-3: Intra CSD-Standard DVP with Matching UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-10: Cross CSD/In-out CSD Standard DVP
24					Back-to-back CSD Participant	
25					Basket: CSD Participant	
26	Yes	No	DVP/DWP	Yes	DVP conditional	UC-SI-3: Intra CSD-Standard DVP with Matching UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-6: Conditional Securities Deliv-
27					Issuer CSD external to T2S	
28					Registered securities DVP	

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
						ery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
29	Yes	Yes	DVP/DWP	Yes	DVP with linked instruction, conditional	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-3: Intra CSD-Standard DVP with Matching UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-6: Conditional Securities Delivery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
30	No	No	DVP	Already Matched	Monetary policy operations (REPO) (already matched)	UC-SI-1:Settlement of a standard settlement instruction UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-9:Settlement of a corporate action with associated liquidity trans-

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
						fer UC-SI-10: Cross CSD/In-out CSD Standard DVP
31	No	No	FOP	Already Matched	Securities distribution, e.g. Rights distribution (CE)	UC-SI-1:Settlement of a standard settlement instruction UC-SI-9:Settlement of a corporate action with associated liquidity transfer UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-10: Cross CSD/In-out CSD Standard DVP
32					Book out (CE)	
33					Primary market and IPO (CE)	
34					UCITS increase/decrease	
35					FOP already matched	
36					Monetary policy operation pledge/collateral management (already matched)	
37					Standard FOP	
38					Transfer of securities Participant	
39	No	Yes	FOP	Already Matched	FOP already matched with linked instruction	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-10: Cross CSD/In-out CSD Standard DVP

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
40	Yes	No	FOP	Already Matched	FOP already matched conditional	UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-6: Conditional Securities Delivery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
41					Issuer CSD external to T2S	
42					Registered securities	
43					FOP Conditional	
44	Yes	Yes	FOP	Already Matched	FOP already matched with linked instruction, conditional	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-6: Conditional Securities Delivery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
45	Yes	Yes	FOP	Yes	FOP matching required with linked instruction, conditional	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-3: Intra CSD-Standard DVP with Matching

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
						UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-6: Conditional Securities Delivery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
46	No	Yes	FOP	Yes	FOP with linked instruction	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-3: Intra CSD-Standard DVP with Matching UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-10: Cross CSD/In-out CSD Standard DVP
47	Yes	No	FOP	Yes	FOP Conditional	UC-SI-3: Intra CSD-Standard DVP with Matching UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
						with Auto-collateralisation UC-SI-6: Conditional Securities Delivery instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
48	No	No	FOP	Yes	FOP for special purpose (same owner or CE)	UC-SI-3: Intra CSD-Standard DVP with Matching UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-10: Cross CSD/In-out CSD Standard DVP
49					Securities issuance and redemption (CE)	
50	No	No	RESA/UNRESA	N/A	Reservation of securities or cash	UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-8: Reservation instruction UC-SI-10: Cross CSD/In-out CSD Standard DVP
51	No	Y	RESA/UNRESA	N/A	Reservation of securities or cash with linked instructions	UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP

ID	CRITERIA				COMMENT	REPRESENTATIVE USE CASE
	COSD	LINKED	INSTR. TYPE	MATCHING		
						with Auto-collateralisation UC-SI-8: Reservation instruction UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-10: Cross CSD/In-out CSD Standard DVP
52	No	No	BLK/UNBLK	N/A	Block securities or cash position with linked instruction	UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-7b: Blocking UC-SI-10: Cross CSD/In-out CSD Standard DVP
53					No settlement involved CE (CE)	
54	No	Y	BLK/UNBLK	N/A	Block securities or cash position with linked instruction	UC-SI-2: Intra-CSD Standard Linked DVP Instructions UC-SI-4: Intra CSD-Standard DVP with Partial Settlement UC-SI-5: Intra CSD-Standard DVP with Auto-collateralisation UC-SI-7b: Blocking UC-SI-10: Cross CSD/In-out CSD Standard DVP

4.1.3 MI: Maintenance Instructions (Hold/Release, Amendment and Cancellation instructions)

ID	CRITERIA				REPRESENTATIVE USE CASE
	MAINTENANCE INSTRUCTION TYPE	FIELDS TO BE AMENDED	STATUS OF THE MAINTENANCE INSTRUCTION	STATUS OF THE ORIGINAL SETTLEMENT INSTRUCTION	
55	HOLD	N/A	N/A	Before Intended Settlement Date Any status but "Settled", "Partially settled", "Cancelled", "CoSD on Hold Yes", "Party Hold Yes", "CSD Hold "Yes"	UC-MI-2: Hold / Release Maintenance Instructions
56	HOLD	N/A	N/A	On or After Intended Settlement Date Any status but "Settled", "Partially settled", "Cancelled", "CoSD on Hold Yes", "Party Hold Yes", "CSD Hold "Yes"	
57	HOLD	N/A	N/A	"Settled", "Partially settled", "Cancelled", "CoSD on Hold Yes", "Party Hold Yes", "CSD Hold "Yes"	
58	RELEASE	N/A	N/A	Before Intended Settlement Date Any status but "Settled", "Partially settled", "Cancelled", "CoSD on Hold Yes", "Party Hold Yes", "CSD Hold "Yes"	
59	RELEASE	N/A	N/A	On or After Intended Settlement Date Any status but "Settled", "Partially settled", "Cancelled", "CoSD on Hold Yes", "Party Hold Yes", "CSD Hold "Yes"	
60	RELEASE	N/A	N/A	"Settled", "Partially settled", "Cancelled", "CoSD on Hold Yes", "Party Hold Yes", "CSD Hold "Yes"	

ID	CRITERIA				REPRESENTATIVE USE CASE
	MAINTENANCE INSTRUCTION TYPE	FIELDS TO BE AMENDED	STATUS OF THE MAINTENANCE INSTRUCTION	STATUS OF THE ORIGINAL SETTLEMENT INSTRUCTION	
61	AMEND	Matching Fields	N/A	Before Intended Settlement Date Any status but :"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	UC-MI-1: Amendement Maintenance Instruction
62	AMEND	Matching Fields	N/A	On or after Intended Settlement Date Any status but :"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
63	AMEND	Matching Fields	N/A	"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
64	AMEND	Additional Settlement Fields	N/A	Before Intended Settlement Date Any status but :"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
65	AMEND	Additional Settlement Fields	N/A	On or after Intended Settlement Date Any status but :"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
66	AMEND	Additional Settlement Fields	N/A	"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
67	AMEND	Other Fields	N/A	Before Intended Settlement Date	

ID	CRITERIA				REPRESENTATIVE USE CASE
	MAINTENANCE INSTRUCTION TYPE	FIELDS TO BE AMENDED	STATUS OF THE MAINTENANCE INSTRUCTION	STATUS OF THE ORIGINAL SETTLEMENT INSTRUCTION	
				Any status but :"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
68	AMEND	Other Fields	N/A	On or after Intended Settlement Date Any status but :"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
69	AMEND	Other Fields	N/A	"Settled", "Partially settled", "Cancelled, "CoSD Hold Yes"	
70	CANCEL	N/A	N/A	Unmatched and any status but : "Settled", "Partially settled", "Cancelled", "CoSD Hold Yes", "Not cancelled" (but cancellation by the same party already requested)	UC-MI-3: Cancellation Maintenance Instruction
71	CANCEL	N/A	Unmatched	Matched and any status but : "Settled", "Partially settled", "Cancelled", "CoSD Hold Yes", "Not cancelled" (but cancellation by the same party already requested)	
72	CANCEL	N/A	Matched	Before the Intended Settlement Date Matched and any status but : "Settled", "Partially settled", "Cancelled", "CoSD Hold Yes", "Not cancelled" (but cancellation by the same party already requested)	
73	CANCEL	N/A	Matched	On or after the Intended Settlement Date	

ID	CRITERIA				REPRESENTATIVE USE CASE
	MAINTENANCE INSTRUCTION TYPE	FIELDS TO BE AMENDED	STATUS OF THE MAINTENANCE INSTRUCTION	STATUS OF THE ORIGINAL SETTLEMENT INSTRUCTION	
				Matched and any status but : "Settled", "Partially settled", "Cancelled", "CoSD Hold Yes", "Not cancelled" (but cancellation by the same party already requested)	
74	CANCEL	N/A	N/A	"Settled", "Partially settled", "Cancelled", "CoSD Hold Yes", "Not cancelled" (but cancellation by the same party already requested)	

4.1.4 SM: Reference Data Management

ID	CRITERIA						REPRESENTATIVE USE CASE
	INSTRUCTING PARTY	ACCESS TYPE	MODE	FUNCTION TYPE	DATA TYPE	ACTION TYPE	
75	CB, CSD, T2S Operator	U2A	2 eyes	Create	Parties	Maintenance Request	UC-SM-1: Reference Data Maintenance
76	CSD	U2A	2 eyes	Create	Securities	Maintenance Request	
77	CSD	U2A	2 eyes	Create	Securities Account	Maintenance Request	
78	CB	U2A	2 eyes	Create	T2S Dedicated Cash Account	Maintenance Request	
79	CB, CSD, T2S Operator	U2A	2 eyes	Create	Rules	Maintenance Request	
80	CB, CSD, T2S Operator	U2A	2 eyes	Create	Parameters	Maintenance Request	

ID	CRITERIA						REPRESENTATIVE USE CASE
	INSTRUCTING PARTY	ACCESS TYPE	MODE	FUNCTION TYPE	DATA TYPE	ACTION TYPE	
81	CB, CSD, T2S Operator	U2A	2 eyes	Update	Parties	Maintenance Request	
82	CSD	U2A	2 eyes	Update	Securities	Maintenance Request	
83	CSD	U2A	2 eyes	Update	Securities Account	Maintenance Request	
84	CB	U2A	2 eyes	Update	T2S Dedicated Cash Account	Maintenance Request	
85	CB, CSD, T2S Operator	U2A	2 eyes	Update	Rules	Maintenance Request	
86	CB, CSD, T2S Operator	U2A	2 eyes	Update	Parameters	Maintenance Request	
87	CB, CSD, T2S Operator	U2A	2 eyes	Delete	Parties	Maintenance Request	
88	CSD	U2A	2 eyes	Delete	Securities	Maintenance Request	
89	CSD	U2A	2 eyes	Delete	Securities Account	Maintenance Request	
90	CB	U2A	2 eyes	Delete	T2S Dedicated Cash Account	Maintenance Request	
91	CB, CSD, T2S Operator	U2A	2 eyes	Delete	Rules	Maintenance Request	
92	CB, CSD, T2S Operator	U2A	2 eyes	Delete	Parameters	Maintenance Request	
93	All	U2A	2 eyes	Retrieve	Parties	Access Request	UC-SM-3: Reference Data Access
94	All	U2A	2 eyes	Retrieve	Securities	Access Request	
95	CSD, CSD Participant	U2A	2 eyes	Retrieve	Securities Account	Access Request	
96	CB, Payment Bank	U2A	2 eyes	Retrieve	T2S Dedicated Cash Account	Access Request	

ID	CRITERIA						REPRESENTATIVE USE CASE
	INSTRUCTING PARTY	ACCESS TYPE	MODE	FUNCTION TYPE	DATA TYPE	ACTION TYPE	
97	All	U2A	2 eyes	Retrieve	Rules	Access Request	
98	All	U2A	2 eyes	Retrieve	Parameters	Access Request	
99	CB, CSD, T2S Operator	U2A	4 eyes	Create	Parties	Maintenance Request	UC-SM-1: Reference Data Maintenance
100	CB	U2A	4 eyes	Create	Securities	Maintenance Request	
101	CSD	U2A	4 eyes	Create	Securities Account	Maintenance Request	
102	CB	U2A	4 eyes	Create	T2S Dedicated Cash Account	Maintenance Request	
103	CB, CSD, T2S Operator	U2A	4 eyes	Create	Rules	Maintenance Request	
104	CB, CSD, T2S Operator	U2A	4 eyes	Create	Parameters	Maintenance Request	
105	CB, CSD, T2S Operator	U2A	4 eyes	Update	Parties	Maintenance Request	
106	CSD	U2A	4 eyes	Update	Securities	Maintenance Request	
107	CSD	U2A	4 eyes	Update	Securities Account	Maintenance Request	
108	CB	U2A	4 eyes	Update	T2S Dedicated Cash Account	Maintenance Request	
109	CB, CSD, T2S Operator	U2A	4 eyes	Update	Rules	Maintenance Request	
110	CB, CSD, T2S Operator	U2A	4 eyes	Update	Parameters	Maintenance Request	
111	CB, CSD, T2S Operator	U2A	4 eyes	Create	Parties	Approval Request	UC-SM-2: Reference Data Approval
112	CSD	U2A	4 eyes	Create	Securities	Approval Request	
113	CSD	U2A	4 eyes	Create	Securities Account	Approval Request	

ID	CRITERIA						REPRESENTATIVE USE CASE
	INSTRUCTING PARTY	ACCESS TYPE	MODE	FUNCTION TYPE	DATA TYPE	ACTION TYPE	
114	CB	U2A	4 eyes	Create	T2S Dedicated Cash Account	Approval Request	
115	CB, CSD, T2S Operator	U2A	4 eyes	Create	Rules	Approval Request	
116	CB, CSD, T2S Operator	U2A	4 eyes	Create	Parameters	Approval Request	
117	CB, CSD, T2S Operator	U2A	4 eyes	Update	Parties	Approval Request	UC-SM-2: Reference Data Approval
118	CSD	U2A	4 eyes	Update	Securities	Approval Request	
119	CSD	U2A	4 eyes	Update	Securities Account	Approval Request	
120	CB	U2A	4 eyes	Update	T2S Dedicated Cash Account	Approval Request	
121	CB, CSD, T2S Operator	U2A	4 eyes	Update	Rules	Approval Request	
122	CB, CSD, T2S Operator	U2A	4 eyes	Update	Parameters	Approval Request	
123	CB, CSD, T2S Operator	U2A	4 eyes	Delete	Parties	Approval Request	UC-SM-2: Reference Data Approval
124	CSD	U2A	4 eyes	Delete	Securities	Approval Request	
125	CSD	U2A	4 eyes	Delete	Securities Account	Approval Request	
126	CB	U2A	4 eyes	Delete	T2S Dedicated Cash Account	Approval Request	
127	CB, CSD, T2S Operator	U2A	4 eyes	Delete	Rules	Approval Request	
128	CB, CSD, T2S Operator	U2A	4 eyes	Delete	Parameters	Approval Request	

ID	CRITERIA						REPRESENTATIVE USE CASE
	INSTRUCTING PARTY	ACCESS TYPE	MODE	FUNCTION TYPE	DATA TYPE	ACTION TYPE	
129	CB, CSD, T2S Operator	U2A	4 eyes	Delete	Parties	Maintenance Request	UC-SM-1: Reference Data Maintenance
130	CSD	U2A	4 eyes	Delete	Securities	Maintenance Request	
131	CSD	U2A	4 eyes	Delete	Securities Account	Maintenance Request	
132	CB	U2A	4 eyes	Delete	T2S Dedicated Cash Account	Maintenance Request	
133	CB, CSD, T2S Operator	U2A	4 eyes	Delete	Rules	Maintenance Request	
134	CB, CSD, T2S Operator	U2A	4 eyes	Delete	Parameters	Maintenance Request	
135	CB, CSD, T2S Operator	A2A	2 eyes	Create	Parties	Maintenance Request	
136	CSD	A2A	2 eyes	Create	Securities	Maintenance Request	
137	CSD	A2A	2 eyes	Create	Securities Account	Maintenance Request	
138	CB	A2A	2 eyes	Create	T2S Dedicated Cash Account	Maintenance Request	
141	CB, CSD, T2S Operator	A2A	2 eyes	Update	Parties	Maintenance Request	
142	CSD	A2A	2 eyes	Update	Securities	Maintenance Request	
143	CSD	A2A	2 eyes	Update	Securities Account	Maintenance Request	
144	CB	A2A	2 eyes	Update	T2S Dedicated Cash Account	Maintenance Request	

ID	CRITERIA						REPRESENTATIVE USE CASE
	INSTRUCTING PARTY	ACCESS TYPE	MODE	FUNCTION TYPE	DATA TYPE	ACTION TYPE	
147	CB, CSD, T2S Operator	A2A	2 eyes	Delete	Parties	Maintenance Request	
148	CSD	A2A	2 eyes	Delete	Securities	Maintenance Request	
149	CSD	A2A	2 eyes	Delete	Securities Account	Maintenance Request	
150	CB	A2A	2 eyes	Delete	T2S Dedicated Cash Account	Maintenance Request	
153	All	A2A	2 eyes	Retrieve	Parties	Access Request	UC-SM-3: Reference Data Access
154	All	A2A	2 eyes	Retrieve	Securities	Access Request	
155	CSD, CSD Participants	A2A	2 eyes	Retrieve	Securities Account	Access Request	
156	CB, Payment Banks	A2A	2 eyes	Retrieve	T2S Dedicated Cash Account	Access Request	

4.1.5 QU: Queries

ID	CRITERIA			REPRESENTATIVE USE CASE
	COMMUNICATION MODE	QUERY CATEGORY	QUERY TYPE	
159	U2A	Instructions	Settlement Instruction Query	UC-QU-1: Query on Securities Settlement Instructions
160	U2A	Instructions	Settlement Instruction Current Status Query	
161	U2A	Instructions	Settlement Instruction Status Audit Trail Query	
162	U2A	Instructions	Settlement Instruction Audit Trail Query	
163	U2A	Security Accounts	Securities Account Position Query	
164	U2A	Security Accounts	Securities Account Position History Query	
165	U2A	Cash Accounts	T2S Dedicated Cash Account Balance Query	
166	U2A	Cash Accounts	T2S Dedicated Cash Account Posting Query	
167	U2A	Cash Accounts	Outstanding Auto-Collateralisation Credit Query	
168	U2A	Cash Accounts	T2S Overall Liquidity Query	
169	U2A	Cash Accounts	Cash Forecast Query	

ID	CRITERIA			REPRESENTATIVE USE CASE
	COMMUNICATION MODE	QUERY CATEGORY	QUERY TYPE	
171	U2A	Cash Accounts	Limit Utilisation Journal Query	
172	U2A	Cash Accounts	Limit Utilisation Query	
173	U2A	Cash Accounts	Total collateral value per T2S Dedicated Cash Account Query	
174	U2A	Cash Accounts	Collateral Value per T2S Dedicated Cash Account Query	
175	U2A	Cash Accounts	Collateral Value of a Security Query	
176	U2A	Cash Accounts	Immediate Liquidity Transfer Order Detail Query	
177	U2A	Cash Accounts	Immediate Liquidity Transfer Order List Query	
180	U2A	Reference Data	Data Changes Queries	
193	U2A	Reference Data	Total amount of standing and predefined orders Query	
208	U2A	Dynamic Queries	Data Changes Queries	
209	U2A	Dynamic Queries	Report Query	

ID	CRITERIA			REPRESENTATIVE USE CASE
	COMMUNICATION MODE	QUERY CATEGORY	QUERY TYPE	
211f	U2A	Penalty	Monthly Net Penalty Amount Query	
211g	U2A	Penalty	Penalty Details Query	
211h	U2A	Penalty	Penalty List Query	
212	A2A	Instructions	Settlement Instruction Query	
213	A2A	Instructions	Settlement Instruction Current Status Query	
214	A2A	Instructions	Settlement Instruction Status Audit Trail Query	
215	A2A	Instructions	Settlement Instruction Audit Trail Query	
216	A2A	Security Accounts	Securities Account Position Query	
217	A2A	Security Accounts	Securities Account Position History Query	
218	A2A	Cash Accounts	T2S Dedicated Cash Account Balance Query	
219	A2A	Cash Accounts	T2S Dedicated Cash Account Posting Query	

ID	CRITERIA			REPRESENTATIVE USE CASE
	COMMUNICA-TION MODE	QUERY CATEGORY	QUERY TYPE	
220	A2A	Cash Accounts	Outstanding Auto-Collateral-isation Credit Query	
221	A2A	Cash Accounts	T2S Overall Liquidity Query	
222	A2A	Cash Accounts	Cash Forecast Query	
222a	A2A	Cash Accounts	Liquidity Information Query	
224	A2A	Cash Accounts	Limit Utilisation Journal Query	
225	A2A	Cash Accounts	Limit Utilisation Query	
226	A2A	Cash Accounts	Total collateral value per T2S Dedicated Cash Account Query	
227	A2A	Cash Accounts	Collateral Value per T2S Ded-icated Cash Account Query	
228	A2A	Cash Accounts	Collateral Value of a Security Query	
229	A2A	Cash Accounts	Immediate Liquidity Transfer Order Detail Query	
230	A2A	Cash Accounts	Immediate Liquidity Transfer Order List Query	
233	A2A	Reference Data	Data Changes Queries	

ID	CRITERIA			REPRESENTATIVE USE CASE
	COMMUNICA-TION MODE	QUERY CATEGORY	QUERY TYPE	
246	A2A	Reference Data	Total amount of standing and predefined orders Query	
256	A2A	Dynamic Queries	Data Changes Queries	
257	A2A	Dynamic Queries	Report Query	
258	A2A	Dynamic Queries	Invoice Query	
259a	A2A	Dynamic Queries	Cumulative Invoice Query	
259b	A2A	Dynamic Queries	Itemised Billing Data Query	

4.1.6 RE: Reports

ID	CRITERIA						REPRESETATIVE USE CASE
	COMMUNICA-TION MODE	INFORMATION BASIS	TRIGGERING	CLASSIFICATION	CATEGORY	TYPE	
260	A2A	Party	Fixed time	Complete Report	Current Settlement Day Cash In-formation Report	Balance	UC-RE-1: Current Settlement Day Cash Information Report
261	A2A	Party	Fixed time	Complete Report	Following Settlement Day Cash Forecast Report	Balance	
262	A2A	Party	Fixed time	Complete Report	Statement of allegements	Securities in-structions	

ID	CRITERIA						REPRESETATIVE USE CASE
	COMMUNICA-TION MODE	INFORMATION BASIS	TRIGGERING	CLASSIFICATION	CATEGORY	TYPE	
263	A2A	Party	Fixed time	Complete Report	Statement of pending instructions	Securities in-structions	
264	A2A	Party	Fixed time	Complete Report	Statement of holdings	Balance	
265	A2A	Party	Fixed time	Complete Report	Statement of transactions	Securities in-structions	
266	A2A	Party	Fixed time	Complete Report	Statement of static data	Static data	
267	A2A	Party	Fixed time	Complete Report	Statement of accounts	Balance	
268	A2A	Party	Fixed time	Delta Report	Statement of allegements	Securities in-structions	
269	A2A	Party	Fixed time	Delta Report	Statement of pending instructions	Securities in-structions	
270	A2A	Party	Fixed time	Delta Report	Statement of holdings	Balance	
271	A2A	Party	Fixed time	Delta Report	Statement of transactions	Securities in-structions	
272	A2A	Party	Business Event	Complete Report	Current Settlement Day Cash In-formation Report	Balance	
273	A2A	Party	Business Event	Complete Report	Following Settlement Day Cash Forecast Report	Balance	
274	A2A	Party	Business Event	Complete Report	Statement of allegements	Securities in-structions	

ID	CRITERIA						REPRESETATIVE USE CASE
	COMMUNICA-TION MODE	INFORMATION BASIS	TRIGGERING	CLASSIFICATION	CATEGORY	TYPE	
275	A2A	Party	Business Event	Complete Report	Statement of pending instructions	Securities in-structions	
276	A2A	Party	Business Event	Complete Report	Statement of holdings	Balance	
277	A2A	Party	Business Event	Complete Report	Statement of transactions	Securities in-structions	
279	A2A	Party	Business Event	Complete Report	Statement of accounts	Balance	
280	A2A	Party	Business Event	Delta Report	Statement of allegements	Securities in-structions	
281	A2A	Party	Business Event	Delta Report	Statement of pending instructions	Securities in-structions	
282	A2A	Party	Business Event	Delta Report	Statement of holdings	Balance	
283	A2A	Party	Business Event	Delta Report	Statement of transactions	Securities in-structions	
283a	A2A	Penalty	Business Event	Complete Report	Daily Penalty List	Penalty	
283b	A2A	Penalty	Business Event	Complete Report	List of Modified Penalties	Penalty	
283c	A2A	Penalty	Business Event	Complete Report	Monthly Aggregated Amounts	Penalty	

4.1.7 LT: Liquidity Transfers

ID	CRITERIA							REPRESENTATIVE USE CASE
	COMMU-NICATION MODE	OWN ACT ON BEHALF	DIRECTION	ORDER TYPE	EXECU-TION	CONCRETE TRANSFER TYPE	ACTION	
284	U2A	Own	Within T2S	Immediate LT	Real-time	Exe Specific Amount cu-tion	Execution	UC-LT-1: Execution of Immediate liquidity transfers within T2S
285	U2A	Own	T2S to RTGS	Immediate LT	Real-time	Exe Specific Amount cu-tion	Execution	UC-LT-2: Execution of Immediate liquidity transfers from T2S to RTGS
286	U2A	On behalf	Within T2S	Immediate LT	Real-time	Execut Spe-cific Amount ion	Execution	UC-LT-1: Execution of Immediate liquidity transfers within T2S
287	U2A	On behalf	Within T2S	Immediate LT	Real-time	Partial	Execution	
288	U2A	On behalf	T2S to RTGS	Immediate LT	Real-time	Specific Amount	Execution	UC-LT-2: Execution of Immediate liquidity transfers from T2S to RTGS
289	U2A	On behalf	T2S to RTGS	Immediate LT	Real-time	Partial	Execution	
290	A2A	Own	Within T2S	Immediate LT	Real-time	Specific Amount	Execution	UC-LT-1: Execution of Immediate liquidity transfers within T2S

ID	CRITERIA							REPRESENTATIVE USE CASE
	COMMU-NICATION MODE	OWN ACT ON BEHALF	DIRECTION	ORDER TYPE	EXECU-TION	CONCRETE TRANSFER TYPE	ACTION	
291	A2A	Own	T2S to RTGS	Immediate LT	Real-time	Specific Amount	Execution	UC-LT-2: Execution of Immediate liquidity transfers from T2S to RTGS
292	A2A	On behalf	Within T2S	Immediate LT	Real-time	Specific Amount	Execution	UC-LT-1: Execution of Immediate liquidity transfers within T2S
293	A2A	On behalf	Within T2S	Immediate LT	Real-time	Partial	Execution	
294	A2A	On behalf	T2S to RTGS	Immediate LT	Real-time	Specific Amount	Execution	UC-LT-2: Execution of Immediate liquidity transfers from T2S to RTGS
295	A2A	On behalf	T2S to RTGS	Immediate LT	Real-time	Partial	Execution	
296	A2A	N/A	RTGS to T2S	Inbound LT	Real-time	Specific Amount	Execution	UC-LT-3: Execution of liquidity transfer from RTGS to T2S
297	N/A	N/A	T2S to RTGS	Standing LTO	Selected Event	Specific Amount	Execution	UC-LT-4: Execution of Standing and Predefined liquidity transfer orders
298	N/A	N/A	T2S to RTGS	Standing LTO	Selected Event	Partial	Execution	

ID	CRITERIA							REPRESENTATIVE USE CASE
	COMMU-NICATION MODE	OWN ACT ON BEHALF	DIRECTION	ORDER TYPE	EXECU-TION	CONCRETE TRANSFER TYPE	ACTION	
299	N/A	N/A	T2S to RTGS	Standing LTO	Selected Event	Reimburse-ment mul-tiple liquidity providers	Execution	from T2S to RTGS
300	N/A	N/A	T2S to RTGS	Standing LTO	Defined Time	Specific Amount	Execution	
301	N/A	N/A	T2S to RTGS	Standing LTO	Defined Time	Partial	Execution	
302	N/A	N/A	T2S to RTGS	Standing LTO	At different points in time	Specific Amount	Execution	
303	N/A	N/A	T2S to RTGS	Standing LTO	At different points in time	Partial	Execution	
304	N/A	N/A	T2S to RTGS	Predefined LTO	Selected Event	Specific Amount	Execution	
305	N/A	N/A	T2S to RTGS	Predefined LTO	Selected Event	Partial	Execution	
306	N/A	N/A	T2S to RTGS	Predefined LTO	Defined Time	Specific Amount	Execution	

ID	CRITERIA							REPRESENTATIVE USE CASE
	COMMU-NICATION MODE	OWN ACT ON BEHALF	DIRECTION	ORDER TYPE	EXECU-TION	CONCRETE TRANSFER TYPE	ACTION	
307	N/A	N/A	T2S to RTGS	Predefined LTO	Defined Time	Partial	Execution	
308	N/A	N/A	T2S to RTGS	Predefined LTO	At different points in time	Specific Amount	Execution	
309	N/A	N/A	T2S to RTGS	Predefined LTO	At different points in time	Partial	Execution	
310	N/A	N/A	T2S to RTGS	Automated LTO	Selected Event	EOD	Execution	UC-LT-5: End-of-Day liquidity transfer

4.1.8 EV: Events

ID	CRITERIA		REPRESENTATIVE USE CASE
	ACTION TYPE	ACTION TYPE	
311	Operating Day Management	Insert	UC-EV-1: Operating Day Management
312	Operating Day Management	Update	
313	Operating Day Management	Complete	
314	Operating Day Management	Execute	
315	Daily Scheduler	Insert	UC-EV-2: Daily Scheduler

ID	CRITERIA		REPRESENTATIVE USE CASE
	ACTION TYPE	ACTION TYPE	
316	Daily Scheduler	Update	
317	Daily Scheduler	Complete	
318	Daily Scheduler	Execute	

4.1.9 LA: Legal Archiving

ID	CRITERIA		COMMENT	REPRESENTATIVE USE CASE
	INTERFACE USED	REQUESTER ENTITY TYPE		
319	U2A	CSD	Request from a CSD via the U2A interface	UC-LA-1 Answer processing to a request for archived data
320	U2A	CB	Request from a CB via the U2A interface	
321	U2A	T2S Operator	Request from a T2S Operator via the U2A interface	
322	A2A	CSD	Request from a CSD via the A2A interface	
323	A2A	CB	Request from a CB via the A2A interface	
324	A2A	T2S Operator	Request from a T2S Operator via the A2A interface	

4.1.10 IN: Interface

ID	CRITERIA		REPRESENTATIVE USE CASE
	COMMUNICATION MODE	RESEND MESSAGE	
325	A2A – pull	No	UC-INTF-1: Interface via A2A-pull
326	A2A - pull	Yes	UC-INTF-2: Interface via A2A – pull - Resend
327	A2A - push	NO	UC-INTF-3: IN Terface via A2A - push
328	U2A	No	UC-INTF-4: Interface via U2A
329	U2A	Yes	UC-INTF-5: Interface via U2A - Resend

4.2 Data Model

The complete T2S logical data model is provided in HTML format, allowing an easier browsing through the data model entities (T2S_DataModel_HTML_V5.0.zip).

4.3 UR Index

The index of UR references is provided as a separate file.

4.4 Most frequently used acronyms

ACRONYM	STANDS FOR
BATM	Bilaterally agreed treasury management instructions
BIC	Bank identifier code
CARL	Corporate action rebalancing liquidity
CB	Central Bank
CBO	Central bank operations
CLP	Common limit and provision checking
CMB	Credit memorandum balance
CMS	Collateral management system
CNCB	Central bank collateral operation
CoSD	Conditional securities delivery
CSD	Central Securities Depository
DVP	Delivery versus payment
DWP	Delivery with payment
EOD	End of day
EUR	Euro
FIFO	First in first out
FOP	Free of payment
ICC	Intraday credit capacity
ISD	Intended settlement date
ISIN	International securities identification number
ISO	International organisation for standardisation
LT/LQT	Liquidity transfer
NCB	National Central Bank
PFOD	Payment free of delivery
RTGS	Real time gross settlement system

All rights reserved.

ACRONYM	STANDS FOR
SI	Settlement instruction
SOD	Start of day
SR	Settlement restriction
T2S DCA	T2S dedicated cash account

All rights reserved.