

RTGS UDFS

Deviations from the URD

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.1.3	1.2.3.3	RTGS.TR.HVP.PAYT.030 RTGS.TR.HVP.PAYT.030.02 0 SHRD.UR.BDD.100	<p>The participants have the possibility to determine the execution time of their transactions, through From Time and either Till Time or Reject Time...</p> <p>RTGS shall ensure that a payment order can only be submitted to settlement if its Reject Time, if indicated, has not yet been reached. As soon as the Reject Time is reached and if the payment order has not been settled, the payment order will be rejected and a settlement failure notification will be sent out.</p> <p>If Till Time has been specified instead, the payment order shall not be rejected when this time is reached and the payment order has not been settled, and RTGS shall allow it to be submitted for settlement beyond this time. At 15 minutes before the indicated Reject Time / Till Time and if the payment order has not been settled, RTGS shall send out a warning notification to the party to be debited.</p>	<p>If TillTime and RejectTime are both provided in the payment then only the TillTime is considered. Therefore, it is strictly recommended to provide only one of the two possible "latest debit time indicators" in a payment. It is not possible to change the "earliest debit time indicator" of a payment which is already queued due to the fact that the original "earliest debit time indicator" had been reached and it was already tried to settle this payment.</p>	<p>As already defined in the usage guideline, it is foreseen that both (TillTime AND RejectTime) can be included in a message.</p>

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.1.6	1.7	RTGS.UR.HVP.LTRCV.040.0 10	In the case of liquidity transfers triggered by a lack of cash in CLM, they can be partially settled. And the remaining part should be queued (it will be decided during the realisation phase whether this queueing is done within RTGS or through the involvement of CLM with a second LT with a specific flag to indicate that it should be queued). In such a case, the processing will be similar to payment orders as described in the process "Payment Order Processing", considering that those LT are Highly Urgent. The queue will be common for payment orders and queued liquidity transfers.	Automated inter-service liquidity transfers to transfer liquidity from the RTGS component to CLM due to pending CBOs always have the highest priority (ie top of the urgent queue).	In order to ensure an efficient processing of payments in RTGS only CLM "pull" liquidity transfers due to pending CBOs shall be considered as "top priority" (to be partially queued). For the sake of consistency and transparency, it is advisable to update the URD accordingly. Moreover, please note that it is envisaged that the "pending LT" is generated in the RTGS component and in case CLM sends a new automatic LT due to pending CBOs, a cancel and replace logic shall apply in the RTGS component.

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.2.1	4.2.2	RTGS.UR.RTGS.UI.135	<p>RTGS shall provide the functionality to create a payment through U2A interface.</p> <p>Note: The ability to enter payments would be subject to necessary rights, allowing an organisation to control the use of this feature.</p>	<p>For the "standard RTGS participant" it is not foreseen to initiate payments via U2A. However, it is acknowledged that as part of the required "cost effective access" it might be possible to initiate payments in U2A. Owing to the fact that the details on the "cost effective access" are under elaboration, we refer to the "standard RTGS participant" in the UDFS only</p>	<p>In the current TARGET2 system only internet-based participants are allowed to create payments via U2A. We assume that a similar behaviour should be applied in the RTGS component. Consequently, the "standard RTGS participant" will not be allowed to create customer payments via U2A. Once the cost-effective solution is defined, it needs to be checked to which extent the UDFS needs to be updated. For the sake of transparency, we propose to clarify the URD accordingly.</p>
5.1.6	General	Priority types	<p>General: Terms used in the URD "highly-urgent", "urgent" and "normal"</p>	<p>Terms used in the UDFS "urgent", "high" and "normal"</p>	<p>As already communicated to the market, an URD CR will be raised to align the wording to "ISO terminology)</p>
General	General	Whitelist	<p>General: Whitelist concept envisaged</p>	<p>Whitelist concept no longer foreseen</p>	<p>As agreed in the TCCG, no whitelist is needed</p>

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.2.6	1.4	Payment order cancellation RTGS.UR.HVP.PAYC.030.01 0	<p>The payment order to be cancelled with the respective instruction has to be in an intermediate (i.e. not end) state to be eligible for cancellation (e.g. queued). Thus, cancellation of payment orders is not feasible if they are already in an end state (settled, rejected or cancelled).</p> <p>RTGS must reject the cancellation of a payment order the Service has already rejected, settled or cancelled and to which the payment cancellation refers to.</p> <p>A payment order eligible for cancellation can either be a queued payment order, an order for which the From Time was not reached yet or a warehouse payment.</p> <p>Payment orders which are captured in an optimisation cycle must also be treated as "potentially settled" and are therefore not available to an immediate cancellation. The check for availability should also wait for a short period of time until a currently ongoing optimisation cycle is over, so that the payment orders not settled within this settlement attempt reached again an intermediate state.</p>	<p>It is envisaged that the cancellation request is forwarded to the receiver of the underlying payment in case the payment already settled on the account. However, this is not the case if the cancellation request refers to a direct debit.</p>	<p>According to our understanding the current URD version does not reflect the latest discussions on this issue. A URD CR is needed to align the documentation.</p>

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.4.2.1	1.5.3.2	RTGS.UR.HVP.LIQT.020.02 5 Description in chapter 1.5.2	<p>RTGS shall check whether both accounts belong to the same participant, or to participants within the same Liquidity Transfer Group or not. If not, the order will be rejected.</p> <p>This check is not performed for CB Accounts.</p>	<p>From our point of view also LTs between RTGS DCA and the "linked sub-account" of the same participant as well as in case of liquidity transfers in the context of ASI 6 RT there is no need to check any LTG. Moreover, instead of "participant" we refer to "party" - in line with the CLM URD.</p>	<p>We propose to clarify the URD and clearly state in which scenarios the LTG needs to be checked. Unfortunately, currently only a generic reference to "accounts" is included.</p> <p>Also the generic description of the chapter might be updated accordingly.</p> <p>Moreover, in the presentation of the TCCG on 6 June it was stated "Intra-service liquidity transfers (i.e. between two RTGS DCAs or between two CLM MCAs) can only take place between accounts belonging to the same LTG". No reference was made what shall happen in case the RTGS DCAs belong to the same RTGS participant and the CLM URD refers to party. This should be aligned.</p>

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.4.2.2	1.5.3.6	RTGS.UR.HVP.LIQT.060.01 0	Check on Floor/Ceiling	In order to avoid a "ping-pong" of liquidity transfers between the RTGS component and CLM, only AS related payment instructions and payments are taken into account for floor/ceiling	We propose to delete the URD as it seems sub-optimal to start floor/ceiling processes due to settled liquidity transfers
5.4.2.3.1	1.5.2	Pre-conditions	Both RTGS DCA/sub accounts exist	Based on market comments received we have added that the account needs to exist and active	As we deem this deviation is for clarification only, no update seems necessary.
5.4.2.3.1	1.5.2	sucess notification	Where the intra-RTGS liquidity transfer (partly) succeeds, RTGS will transfer (part of) the amount requested and RTGS will send a (partly) transfer success notification to the participants involved (where the participant opted for it).	The sending of camt.025 is mandatory in case of a liquidity transfer	Actual design envisages the mandatory sending of camt.025
3.2	9.2 (SHRD)	SHRD.UR.BDD.080	Trigger: Either a time-based or event-based trigger that will initiate the Standing Order for Limit	An Attribute "Trigger" shall not be made available	Standing orders for limits will only be executed with the start of the new business day. Therefore neither a time nor an event based trigger is necessary.

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
3.2	9.2 (SHRD)	SHRD.UR.BDD.120	Partial execution: Flag identifying whether partial execution is possible or not	An Attribute "Partial execution" shall not be made available	All standing order liquidity transfers executed at the same time/event will be executed according to a pro rata rule. Therefore a partial execution in case of need is a mandatory feature.
3.2	9.2 (SHRD)	SHRD.UR.BDD.160	Trigger: An event-based trigger that will initiate the Standing Order to create a Reservation	An Attribute "Trigger" shall not be made available	Standing orders for reservations will only be executed with the start of the new business day. Therefore neither a time nor an event based trigger is necessary.
3.2	9.2 (SHRD)	SHRD.UR.BDD.160	Partial execution: Flag identifying whether partial execution is possible or not	An Attribute "Partial execution" shall not be made available	Reservations that can't be covered for the full amount will always create a pending reservation for the remaining amount

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.4.2.4.1	1.5.3.1 1.2.3.1	RTGS.UR.HVP.PAYT.010.02 0 RTGS.TR.HVP.LIQT.010	Duplicate check - technical validation	Duplicate check only during business validation is envisaged	Due to the fact that a duplicate check is also required on payment/liquidity transfer level we propose not to double this check. On the one hand the current implementation in TARGET2 is relying on the duplicate check on payment level which is perfectly fine and sufficient. Moreover doubling the checks would increase the costs.
5.4.3.1.1	1.9.3	RTGS.UR.HVP.LIQR.040.02 0	RTGS shall allow for interventions on pending reservation requests: New reservation requests related to the participant's RTGS account will either increase the pending amount, or decrease it.	Upon receipt of <ul style="list-style-type: none"> • End of Day notification, • a reservation revocation or • a new reservation order, the RTGS component stops processing of the original reservation order, ie the new reservation replaces the pending one.	The RTGS component shall always take into account the latest request. Therefore, it might be appropriate to clarify the URD.

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.4.3.4	1.5.3.6	RTGS.UR.HVP.LIQT.060.01 0	Check on Floor/Ceiling Similar to RTGS.UR.HVP.PAYT.080.010	The floor / ceiling functionality itself will only be triggered after the settlement of a payment or a payment instruction stemming from the settlement of ancillary systems. It is not triggered for liquidity transfers.	In order to avoid a "ping pong" of liquidity transfers between CLM and the RTGS component, we propose to delete this URD.
5.3.2	4.2.2	RTGS.UR.RTGS.UI.160	Revoke an AS file	The RTGS service shall provide the functionality to revoke an AS file which has not reached a final status yet through U2A and A2A interface.	In TARGET2 revocation is only possible via U2A. It is unclear which message should be used to revoke an AS file, it seems no appropriate message exists to perform the revocation. In addition this new functionality would mean, that the AS business is changed strongly, whereas it was the initial plan to leave AS business untouched as most as possible. Maybe URD should be changed to allow only revocation via U2A.

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.2.5	1.3.3.2	RTGS.UR.HVP.PAYA.020.01 0	<p>RTGS shall ensure that an amendment of a payment order can be sent:</p> <ul style="list-style-type: none"> • By the participant owning the account to be debited or • By the respective CB acting on its behalf. <p>If the validation failed, a rejection notification with appropriate reason code shall be sent to the sender of the payment amendment instruction.</p>	As long as a payment is not settled (including warehoused payments), an authorised system user has the possibility to change the relevant parameters of this payment.	According to our understanding not only the participant (or the CB acting on ist behalf) can amend payments. We assume that this should be possible for any authorised system user.
5.4.3.1	1.9.2	Process Overview	Where the reservation operation (partly) succeeds, the Service will reserve (part of) the amount requested and the Service will send a (partial) reservation success notice to the sender of the request and to the account owner.	<p>In case of (partial) execution of the reservation a camt.025 is created and sent via ESMIG to RTGS participant A. ...</p> <p>...In case of an immediate reservation sent by an entity actor different from the account owner, also the sender of the camt.048 receives a camt.025</p>	Unfortunately, the URD for RTGS are not entirely clear how many notifications shall be sent and to whom. Moreover, currently we are deviating in the RTGS UDFS from the CLM behaviour as the URD for CLM are different. The topic needs to be clarified.

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.4.3.2	1.2.2	Process Overview		Limit management – positive validation Creation and forwarding of camt.025 by the RTGS component via ESMIG to RTGS participant A	The URD are not entirely clear when a notifications with regard to limit management shall be sent. Once the "notification issue as a whole" was discussed, it is clear how to update the URD.
5.2.7	1.2.3.3	RTGS.UR.HVP.PAYT.030.040	End of Day - revocation of queued orders Description RTGS shall ensure that a queued payment order can only be settled until the relevant cut-off time is reached, and the last optimisation algorithm has run (see SHRD.UR.BD.OPER.000.030 in chapter Business day in the URD for Shared Services). RTGS shall revoke:	Rejection	At the end of day the rejection is done by the RTGS component. Therefore, it should be called "rejection"
5.4.2.2	1.2.3.7.2	RTGS.UR.HVP.PAYT.080.010	If either is the case, then the second step is to check which action has been specified: • Notification to be sent in A2A and/or Notification to be sent as an alert in U2A	What is meant with "and/or"?	It is not foreseen to get an A2A notification AND an U2A alert at the same time.
5.5	4.2	RTGS.UR.RTGS.UI.110	...and the MCA number (Party BICs and MCA numbers in case of a Central Bank as a user)In addition the query shall allow the user to specify any combination of the following optional selection criteria.	There are not MCAs in the RTGS component which can be queried.	Wording of the URD should be updated

Section in UDFS	Section in URD	URD ID	URD Text	Deviation	Reason for deviation
5.3	2.1.5	Table 7	Settlement on dedicated Liquidity Account (so-called technical account for procedure 6) (real-time) 6 Real-Time Usual real-time gross mode settlement of bilateral high value payments. Settlement on dedicated Liquidity Account (so-called sub accounts) (interfaced) 6 Interfaced Usual real-time gross mode settlement of bilateral high value payments.	Dedicated AS procedures are foreseen for AS procedure 6 interfaced and real-time	The URD should be updated
5.3	2.1.5	Description	For procedures 4 and 5, AS transactions will be sent by files.		In order to avoid potential misunderstandings it might be good to clarify that these AS do not use a "file (with BFH)"