

PONTES PILOT

Service Description

V1.0

DISCLAIMER

This document is a DRAFT.

The API Documentation (Appendix 6.1) containing the OpenAPI Document (swagger file) and the Graphical User Interface (Appendix 6.2) containing the User Handbook are separate documents.

The document is subject to the MIB approval.

Table of Content

TABLE OF CONTENT.....	3
TABLE OF ILLUSTRATIONS	6
TABLE OF TABLES	7
GLOSSARY	8
1. INTRODUCTION	10
1.1 CONTEXT.....	10
1.2 PURPOSE OF THIS DOCUMENT.....	10
1.3 OVERVIEW	11
2. ACTORS, ROLES AND USER PROFILES.....	13
2.1 ACTORS.....	13
2.1.1. European Central Bank (ECB)	13
2.1.2. National Central Banks (NCBs).....	13
2.1.3. Market Participants.....	14
2.1.4. Service Providers	14
2.1.5. Market DLT Operators.....	14
2.2 ROLES.....	15
2.2.1. Cash Token Issuer	15
2.2.2. Dedicated Cash Wallet Manager.....	15
2.2.3. Dedicated Cash Wallet Owner	16
2.2.4. Cash Token Supervisor.....	16
2.2.5. National Cash Token Supervisor.....	17
2.2.6. Operator	17
2.2.7. External entity instructing on behalf.....	17
2.3 RELATIONSHIP BETWEEN ACTORS	17
2.4 USER PROFILES	23
2.4.1. Administrator (Admin User)	24
2.4.2. Referential Users.....	24
2.4.3. Pilot Users	24
2.4.4. Operational Users	25
2.4.5. External Users.....	25
3. BUSINESS FUNCTIONALITIES	25
3.1 HIGH LEVEL FUNCTIONAL DESCRIPTION	25
3.2 U2A/A2A GENERAL PROCESS DESCRIPTION	26
3.2.1. Authentication & Authorisation	26

3.2.2. Communication channels & User profiles.....	26
3.3 NON-REPUDIATION OF ORIGIN (NRO)	27
3.4 LOCAL REFERENCE DATA MANAGEMENT	30
3.4.1. National Central Banks and ECB	31
3.4.2. Market Participants.....	32
3.4.3. T2 account reference	33
3.4.4. Dedicated Cash Wallets	34
3.4.5. Market DLT platforms.....	36
3.4.6. Market DLT Operators.....	36
3.4.7. Whitelists	37
3.4.8. "Instruct on behalf" configuration	37
3.4.9. Market Participant Blocking / Unblocking	38
3.4.10. Market DLT Operator Blocking / Unblocking	39
3.4.11. Users.....	39
3.5 SETTLEMENT TRANSACTIONS	40
3.5.1. Funding Requests	42
3.5.2. Defunding Requests.....	43
3.5.3. Wallet to wallet cash token transfers	45
3.5.4. Payments	46
3.5.5. Payment Free of Delivery	49
3.5.6. Delivery versus Payment & Payment versus Payment.....	51
3.5.7. Unhappy Paths & Error codes	78
3.6 QUERIES/MONITORING & REPORTING	82
3.6.1. Queries.....	82
3.6.2. Reporting and data extraction	91
3.7 A2A 1-STEP/2-STEPS ILLUSTRATION	98
4. OPERATIONAL BUSINESS DAY	100
4.1 START OF DAY (SOD) (8:00 - 9:00)	100
4.2 OPEN FOR ALL (9:00 - 16:00)	100
4.3 END OF DAY (EOD) (16:00 - 18:00)	100
4.4 CLOSED (18:00 - BD+1 8:00).....	101
5. HIGH LEVEL TECHNICAL DESCRIPTION	102
5.1 HIGH-LEVEL ARCHITECTURE OF THE SOLUTION	102
5.1.1. Extended Interoperability Interface.....	102
5.1.2. Eurosystem Distributed Ledger Technology.....	102
5.1.3. T2 Interface	103
5.2 HOSTING & RESILIENCE	103
5.3 INTEROPERABILITY MECHANISM.....	104
6. APPENDIX	106

6.1	API DOCUMENTATION • SWAGGER.....	106
6.2	GRAPHIC USER INTERFACE • USER HANDBOOK.....	108
6.3	PREPARING YOUR INTEGRATION • CONNECTIVITY GUIDE.....	110
6.3.1.	Global Connectivity Overview	110
6.3.2.	Authentication.....	110
6.3.3.	A2A channel	110
6.3.4.	U2A channel.....	111
6.3.5.	Lifecycle of Digital Certificates.....	112
6.3.6.	Connectivity testing	119
6.4	ONBOARDING PROCESSES	121
6.4.1.	Market DLT Operator	121
6.4.2.	Market Participant	122

Table of Illustrations

fig. i.	High-Level Architecture of Pontes Pilot Solution	11
fig. ii.	Two access models: U2A and A2A	12
fig. i.	Market Participant operating in two different jurisdictions	18
fig. i.	Interaction Modes	19
fig. i.	Relationship between Market DLT Platform and Market DLT Operator	20
fig. i.	Whitelisting.....	21
fig. i.	Data Scope visibility and Responsible Actor for Reference data	21
fig. i.	Example of Relationship between actors - Prerequisites	22
fig. i.	Example of Relationship between actors – Actual configuration.....	23
fig. ii.	Non-Repudiation of Origin mechanism.....	28
fig. iii.	High-level T2 / Pontes Pilot architecture and links between T2 RTGS Accounts and Wallets	31
fig. iv.	Funding request workflow in U2A mode.....	42
fig. v.	Defunding request workflow in U2A mode.....	44
fig. vi.	Wallet to wallet cash tokens transfer high level workflow in U2A mode.....	45
fig. vii.	Cash tokens payment high level workflow in U2A mode	46
fig. viii.	Payment in Direct T2 RTGS Settlement in U2A 4-eyes.....	48
fig. ix.	Payment Free of Delivery – DELI high level workflow in A2A mode	49
fig. x.	Payment Free of Delivery- RECE high level workflow in A2A mode	50
fig. i.	State Diagram of Payment status	52
fig. ii.	Relationship between Timeout and Payment	52
fig. iv.	XvP initialisation high level workflow in A2A (by Seller)	55
fig. i.	XvP initialisation query high level workflow in A2A (Buyer retrieves hash-lock).....	57
fig. i.	XvP settled in cash tokens high level workflow in A2A.....	59
fig. i.	XvP Direct Settlement high level workflow in A2A.....	60
fig. i.	Get Key & payment high level workflow in A2A.....	63
fig. ii.	XvP initialisation on a Market DLT platform	65
fig. iii.	XvP initialisation high level workflow in A2A.....	66
fig. iv.	Payment settled situation high level workflow in A2A	67
fig. v.	Cooperative execution high level workflow in A2A	68
fig. vi.	Forced execution 1 high level workflow in A2A	68
fig. vii.	Forced execution 2 high level workflow in A2A	69
fig. viii.	Unsettled situation high level workflow in A2A.....	70
fig. ix.	Cooperative cancellation high level workflow in A2A	70
fig. x.	Forced cancellation high level workflow in A2A	71
fig. xi.	Market DLT Operator: Actor of the Market DLT platform.....	71
fig. xii.	Market DLT Operator can act on behalf of Market Participants	72
fig. xiii.	XvP initialisation high level workflow in A2A – via Market DLT Operator.....	72
fig. xiv.	Payment settled situation high level workflow in A2A – via Market DLT Operator.....	73
fig. xv.	Cooperative execution high level workflow in A2A – via Market DLT Operator.....	73
fig. xvi.	Forced execution high level workflow in A2A – via Market DLT Operator	74
fig. xvii.	Payment unsettled situation high level workflow in A2A – via Market DLT Operator	75
fig. xviii.	Cooperative cancellation high level workflow in A2A – via Market DLT Operator.....	75
fig. xix.	Forced Cancellation high level workflow in A2A mode via Market DLT Operator	76
fig. xx.	Queries high level workflow	83
fig. xxi.	First step of A2A 2-steps Payment request	98
fig. xxii.	Second step of A2A 2-steps Payment request	99
fig. xxiii.	A2A 1-step Payment request	99

fig. xxiv. Business Day Schedule
101

Table of Tables

Table A.	Eurosystem DLT actors' roles depending on business typology	15
Table B.	User Profiles and categories of actions	23
Table C.	Actors and allowed User Profiles	24
Table D.	A2A/U2A Communication channels	26
Table E.	User Profiles and access channels	27
Table F.	List of reference data features in U2A and A2A	30
Table G.	Data Model of National Central Banks	32
Table H.	Data Model of Market Participant	33
Table I.	Data Model for the link between Entities and T2 accounts	34
Table J.	Data Model of Dedicated Cash Wallets	35
Table K.	Data Model of Market DLT platforms	36
Table L.	Data Model of Market DLT Operator	36
Table M.	Data Model of whitelist	37
Table N.	Data Model of Instruct of behalf for Cash Tokens	38
Table O.	Data Model of Instruct of behalf for Direct T2 RTGS	38
Table P.	Data Model of Users reference data	40
Table Q.	List of features available to Market Participants	41
Table R.	List of features available to market DLT Operators	41
Table S.	Queries accessible to NCBs and Market Participants in U2A and A2A ..	82
Table T.	Queries accessible to Market DLT Operators in A2A	83
Table U.	Client ID to be used for A2A authentication per user profile	110

Glossary

A2A	Application-to-Application
Burning tokens	Process of redeeming tokens
“BURNED” status	Forced status for a payment in the ESY DLT which avoids the payment of the cash leg of a DvP / PVP to be instructed once timeout is reached
Boolean	A boolean is a data type with only two possible values: true or false
CA	Certificate Authority
Cash Token	Proxy of euro denominated CeBM, representing a claim against the token issuer (i.e., the ECB) to transfer the respective amount of CeBM in T2 RTGS to the T2 RTGS DCA of respective market participant that owns such tokens.
CoBM	Commercial Bank Money
CeBM	Central Bank Money
CMU	Capital Markets Union
CSD	Central Securities Depository
CSDR	Central Securities Depository Regulation
CSR	Certificate Signing Request
Defunding	Defunding requests enable a Market Participant to defund a Dedicated Cash Wallet within the Pilot. The corresponding amount is debited (meaning burnt) the Dedicated Cash Wallet and credited to the Market Participant T2 RTGS Account
DCA	Dedicated Cash Account
DCW	Dedicated Cash Wallet
DLT	Distributed Ledger Technology
DLTSS	DLT Securities Settlement System
DTSS	DLT Trading and Securities Settlement
DSS	Market DLT Settlement System
DvP	Delivery versus Payment
ESY DLT	Eurosystem DLT
Funding	Funding requests enable a Market Participant to fund a Dedicated Cash Wallet within the Pilot. The corresponding amount is debited from the Market Participant T2 RTGS Account and credited (meaning minting) the Dedicated Cash Wallet.
Main Flag	Configuration setting within the reference data that designates the main Dedicated Cash Wallet to be used automatically for XvP transactions submitted without a wallet ID
Minting	Process of issuing tokens
NCB	National Central Bank
NSD	National Service Desk

On-Chain Submission	On-chain submission refers to the process of recording data or executing an operation directly on the blockchain ledger, where the transaction becomes permanently stored, validated, and immutable once included in a block.
PEM	Privacy-Enhanced Mail is a text-based container format for digital certificates and keys containing Base64-encoded data
PFoD	Payment Free Of Delivery
Payment Instruction	Object created in the processing of the Funding/Defunding, Payment and DvP/PvP with Direct T2 RTGS Settlement then translated by the T2 Interface into ISO 20022 pacs messages and communicated to T2 RTGS for settlement
PI	Payment Instruction
Private key	A cryptographic component which unlocks the right for its owner to control and use the associated crypto assets
Public key	A cryptographic component comparable to an address, publicly visible to all users on the network where crypto assets can be transferred or deposited for the benefit of a private key owner
PvP	Payment versus Payment
RTGS	Realtime Gross Settlement
SHA256 algorithm	A cryptographic hash function that converts input data into a unique alphanumeric string with fixed length of 256-bit (32byte)
SPs	Service Providers are The Banque de France, the Banca d'Italia, the Deutsche Bundesbank and the Banco de España (collectively referred to as the '4CB').
U2A	User-to-Application
UID v4	Universally Unique Identifier, Version 4 is a 128-bit identifier. It is represented as 32 hexadecimal digits separated by hyphens (e.g. b4490db4-b5c8-4180-99b7-e9f3601bc531) process for realising the technical participation of given financial institutions on a DLT
Whitelisting	Whitelisting is a reference-data configuration that establishes a formal relationship between two actors of the Pilot solution, enabling interaction between the two parties within the solution
Pilot BIC	BIC 11 referring to the Account BIC (Authorised Account User BIC, AAU= direct) of the T2 RTGS account used in pilot. The recommendation is the use of the CRDM Party BIC as Account BIC to facilitate consolidation.

1. Introduction

1.1 Context

The Eurosystem has been exploring Distributed Ledger Technologies (DLT) and Tokenised Wholesale Central Bank Money (CeBM) for several years. In parallel, and with significant synergies, the DLT Pilot Regime was established as a regulatory framework for trading and settling tokenised or native financial instruments.

Based on the success of this Exploratory Work¹ and on Market's feedback, the Governing Council of the European Central Bank (ECB) decided to expand its initiative² to settle transactions recorded on DLT in central bank money. The chosen approach for this initiative follows a two-track approach. The first track "Pontes" provides a short-term offering to the market and the second track "Appia" focuses on a potential long-term solution³.

Pontes aims to provide a DLT-based settlement solution for wholesale financial transactions, including wholesale Payments, Cash tokens transfers, Payment Free of Deliveries (PFoDs), Delivery-versus-Payments (DvPs) and Payment-versus-Payments (PvPs). DvP and PvP can be achieved through an interoperability mechanism with Market DLT platforms. With Pontes, the Eurosystem will continue to:

- Maintain central bank money as the anchor of a two-tier monetary system,
- Achieve strategic autonomy and increased resilience for European payments,
- Foster an integrated, competitive and innovative payments ecosystem,
- Support the international role of the euro,
- Support innovation, without compromising on safety and efficiency in financial market infrastructures.

After validation of technical feasibility within the Eurosystem exploratory work, the Pontes Pilot (in the following referred to as Pilot) represents the next step towards achieving the overall Eurosystem's strategy for DLT-based settlement in CeBM. The Pilot is designed to rapidly deliver a market-ready solution, rather than serving solely as a test, thereby driving market innovation and developments. It embeds feedback loops into Pontes' development to capture lessons learned and aims to further implement and validate operational processes and legal compliance, ensuring alignment with the standards of TARGET services. Beyond providing an immediate response to the market needs, Pontes Pilot lays the foundation for a scalable and future-proof architecture.

The Pilot ensures that the Eurosystem can deliver a robust, compliant, and interoperable solution that meets market expectations while paving the way for the long-term Eurosystem's strategy.

1.2 Purpose of this document

The purpose of this document is to complement the [Business Description Document \(BDD\)](#) by providing necessary information to National Central Banks (NCBs, see section 2.1.2), Market Participants (see

¹ Eurosystem Exploratory Work

² Eurosystem expands initiative to settle DLT-based transactions in central bank money

³ ECB commits to distributed ledger technology settlement plans with dual-track strategy

section 2.1.3) and Market DLT Operators (section 2.1.5) in order to connect to the solution and use the provided functionalities.

This document provides a detailed overview of the Pontes Pilot Solution key features in terms of organisation and services. It includes:

- a description of the flows and interactions with the solution for the provided features,
- a connectivity guide,
- the technical details of the onboarding process,
- the technical specifications for the Application Programming Interface (API) as a separated Annex,
- the User Handbook explaining how to use the Graphical User Interface (GUI) as a separated Annex,

1.3 Overview

The Pilot combines the most effective elements of the three interoperability options assessed during the 2024 Eurosystem exploratory work. It introduces a single Eurosystem solution that supports two settlement models: cash tokens settlement and direct settlement in T2 RTGS.

Finality of the cash leg is achieved as stated in the BDD (4.5 Finality and irrevocability of settlements “PILOT.SET.FIN.010”). The settlement finality will apply as specified in Article 18 Annex 1 Part I of the TARGET Guideline⁴.

“All-or-none” settlement is enabled for DvP and PvP transactions through the Hash-Link protocol.

The Pilot solution enhances automation, allowing seamless end-to-end processing with T2 RTGS. It operates outside the technical perimeter of TARGET Services and requires no changes to their existing legal, regulatory, functional, or operational framework.

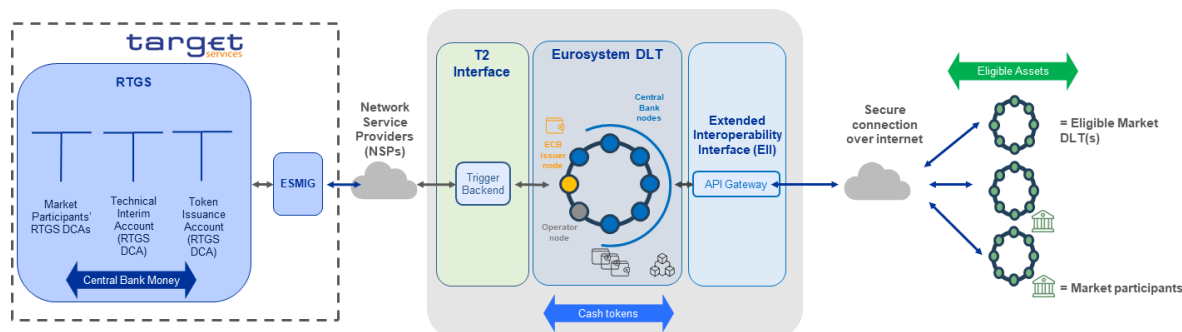


fig. i. High-Level Architecture of Pontes Pilot Solution

The Pilot solution is built around three main components:

- **Extended Interoperability Interface (EII):** Single access point to the Eurosystem DLT via APIs and/or GUI, routing requests to the correct node. It supports standard operations (queries, reference data, payments) and all-or-none settlements (DvP and PvP) using the Hash-Link Protocol.

⁴ Guideline (EU) 2022/912 of the European Central Bank of 24 February 2022 on a new-generation Trans-European Automated Realtime Gross Settlement Express Transfer system (TARGET) and repealing Guideline ECB/2012/27 (ECB/2022/8).

- The first option to access and interact with the Eurosystem DLT is **User-to-Application (U2A)**, which enables manual interaction through a graphical user interface (GUI) provided by the Eurosystem through the EII. This access mode can be used by Market Participants, NCBs and ECB, who need to enter transactions manually, monitor activity, or perform contingency operations. U2A is intended to lower the barrier to entry, as those actors do not need to develop their own application nor connect via APIs. Only a set of functions (namely payments, liquidity transfers and funding/defunding, reference data management and queries) are possible in U2A mode. DvP, PVP, and PFoD transactions are only available in A2A. Connectivity is established through a secure internet connection, and the GUI offers intuitive navigation and real-time visibility of transactions.
- The second option is **Application-to-Application (A2A)**, which allows automated interaction through application programming interfaces (APIs). Like U2A, A2A connectivity is provided via the EII over a secure internet connection, ensuring compliance with Eurosystem security standards. The API-based integration enables straight-through processing, reducing operational risk and improving efficiency.

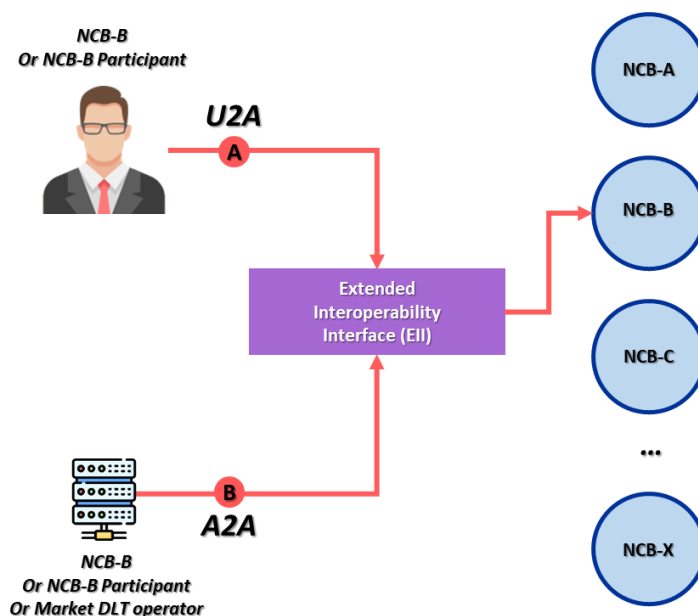


fig. ii. Two access models: U2A and A2A

- **Eurosystem DLT Platform (also referred as ESY DLT):** Private permissioned infrastructure managing the cash leg of DLT-based transactions, supporting settlement via cash tokens and payment instructions for direct settlement in T2 RTGS. It consists of central bank nodes, an operator node, and an ECB node, with Dedicated Cash Wallets and a Token Issuance Wallet. The balances on the Token Issuance Wallet on the Eurosystem DLT and the Token Issuance Account mirror each other, representing the total amount of issued cash tokens.
- **T2 Interface:** Single entry point to TARGET Services via ESMIG. The T2 Interface handles the communication between the ESY DLT and T2 RTGS for funding and defunding processes as well as direct T2 RTGS settlement. Besides to the Participants' T2 RTGS DCAs, two T2 RTGS DCAs are used: a Technical Interim Account for temporary funds holding and a Token Issuance Account for issuing and redeeming cash tokens, both under the legal responsibility of the ECB and operated by the Service Providers (SPs). (See section [3.5.1](#) and [3.5.2](#) for further details).

For this described purpose, the node of Deutsche Bundesbank serves a dual role as the T2 Interface as well as the NCB node. The graphical illustrations found in the **Error! Reference source not found.** section describing the BBk node are simplified to enhance the clarity of the business logic. Despite this dual role the responsibilities are clearly separated between Bundesbank acting as SP and as NCB, since the functionality handling the connection to T2 RTGS is implemented in the Trigger Backend which is connected to the BBk node.

2. Actors, roles and user profiles

2.1 Actors

An actor is defined as an entity allowed to interact with the Pilot solution.

2.1.1. European Central Bank (ECB)

The ECB owns the Issuer node in the ESY DLT that is responsible of the execution of issuance and redemption operations related to funding and defunding requests.

The ECB is the owner of the Technical Interim Account (used for any processes involving settlement in T2 RTGS) and the Token Issuance Account in T2 RTGS as well as the Token Issuance Wallet in the ESY DLT, all involved for funding and defunding processes.

2.1.2. National Central Banks (NCBs)

The Pilot solution is accessible to all NCBs of the Eurosystem that are onboarded by the SPs.

NCBs must own a node hosted in the infrastructure provided by the SPs. Owning a node does not come with any technical obligations; those are all covered by the SPs.

They are in charge of:

- Onboarding Market Participants of the NCB's jurisdiction,
- Onboarding Market DLT Operators responsible of the assets and/or instructing on behalf of at least one Market Participant under their jurisdiction,
- Creating, monitoring, amending and closing Dedicated Cash Wallets of their Market Participants as well as for themselves,
- Monitoring all activity of their Market Participants,
- Initiating transactions on behalf of their Market Participants in case of contingency⁵;
- Blocking/unblocking of their Market Participant,
- Setting up the "Instruction on Behalf" functionality for their Market Participants.
- Monitoring all testing activities conducted by Market Participants (MPs), Market DLT Operators (MDLTos) and certifying them for production

⁵ List of transactions that the NCB can instruct on behalf of its participant (see section [3.5](#))

2.1.3. Market Participants

Market Participants are institutions according to Article 4 Annex I Part I of TARGET Guideline holding a T2 RTGS account which allows them to own a Dedicated Cash Wallet in the Pilot solution. Despite not having a node, Market Participants have access to the ESY DLT through their NCBs nodes accessible through the EII.

Market Participants must hold a valid BIC within T2 and especially own a T2 RTGS account. Pilot doesn't provide technical access to indirect T2 RTGS participation schemes (e.g. addressable BIC or multi-addressee). Nevertheless, Market Participant or Market DLT Operator can instruct transactions on behalf⁶ of indirect participants.

To allow the Market DLT Operator to instruct transactions (cash token and/or Direct T2 RTGS settlement) on behalf of a the Market Participant, a specific setup must be put in place (see section [2.3](#)).

2.1.4. Service Providers

The 4CB serve as the Service Provides (SPs, also referred as the Operator), acting as the technical providers and operators of the Pilot solution.

They are in charge of:

- Managing the solution configuration and the operational Business Day
- Operating the Technical Interim Account owned by the ECB
- Operating the Token Issuance Account and its related Token Issuance Wallet owned by the ECB
- Hosting and operating the nodes on behalf of the NCBs

2.1.5. Market DLT Operators

Market DLT Operators offer a DLT service for the delivery, clearing and/or settlement of eligible assets or financial instruments on one (or more) Market DLT platform(s). In the Pilot, Market DLT Operators are configured as external participants.

They don't own a Dedicated Cash Wallet (and possibly also not a T2 RTGS account) but they are able to instruct payments, PFoDs, DvPs and PVPs on behalf of Market Participants.

Formally, a Market DLT Operator means:

- a. Central securities depositories (CSDs) authorised under the CSDR⁷ and NCB-run CSDs exempted from the CSDR authorisation procedure operating a Securities Settlement System based on DLT (including T2S CSDs subject to a derogation of the contractual and regulatory framework of T2S⁸).
- b. Operators of a DLT settlement system (DSS) or a DLT trading and settlement system (DTSS) as authorised under the DLT Pilot Regime Regulation⁹;
- c. Operators of a payment system established in the EU or the EEA that is subject to oversight by a competent authority.

⁶ The relationship between the Market Participants/Market DLT Operator and the indirect participant is managed outside the Pilot solution.

⁷ Regulation (EU) No 909/2014 of the European Parliament and of the Council of 23 July 2014 on improving securities settlement in the European union and on central securities depositories

⁸ A derogation to the T2S Framework Agreement would be required to enable T2S CSDs to settle DvP transactions in CeBM via the Pilot

⁹ Regulation (EU) 2022/858 of the European Parliament and of the Council of 30 May 2022 on a pilot regime for market infrastructures based on distributed ledger technology, and amending Regulations (EU) No 600/2014 and (EU) No 909/2014 and Directive 2014/65/EU

- d. Central counterparties (CCPs) authorised under EMIR¹⁰;
- e. Credit institutions, investment firms, market operators, and other licensed financial institutions, authorised under CRD, MiFID II and/or other relevant national legal frameworks in the EEA which is (i) subject to supervision or oversight specific for financial market infrastructures by a competent authority or (ii) following a case-by-case assessment conducted by the applicant's NCB based on agreed Eurosystem criteria to ensure proper risk management¹¹.

2.2 Roles

To access the Pilot solution and use the provided functionalities, each actor is granted with one or several roles according to its business typology.

The different role types detailed in this section are assigned to the entities (SPs, NCBs, Market Participants, and Market DLT Operators) depending on their business typology as shown in table A.

	Cash Token Issuer	DCW manager	DCW Owner	Cash Token Supervisor	National Cash Token supervisor	Operator	External entity instructing on behalf ¹²
Service Providers						✓	
National Central Banks	✓	✓	✓		✓		
ECB	✓	✓	✓	✓			
Market Participants			✓				
Market DLT Operators							✓

Table A. Eurosystem DLT actors' roles depending on business typology

2.2.1. Cash Token Issuer

This role is granted to the ECB and every NCB.

It allows requesting the issuance of cash tokens from the Token Issuance Wallet owned by the ECB.

2.2.2. Dedicated Cash Wallet Manager

This role is granted to the ECB and every NCB.

It allows:

- Reference data management for themselves and for Market Participants of their jurisdiction
- The creation of DCWs for themselves and for their Market Participants.
- The instruction of all transactions related to managed DCWs on behalf of their Market Participants (funding & defunding requests, cash token transfers, payments, issuances, redemptions) in case of contingency
- Visibility on balances and transactions history related to managed DCWs.

¹⁰ Regulation (EU) No 648/2012 of the European Parliament and of the Council of 4 July 2012 on OTC derivatives, central counterparties and trade repositories

¹¹ Harmonised assessment criteria shall be defined and documented in the Legal Agreement.

¹² Referring to Market DLT Operators

2.2.3. Dedicated Cash Wallet Owner

This role is granted to all T2 RTGS Account Holders (ECB, NCBs and Market Participants). Despite the role name, having an actual DCW is optional. This role is granted to Market Participants not only using the Cash Token Settlement for which a DCW is necessary, but also Market Participants using Direct T2 RTGS Settlement endpoints. In the latter case, creating an actual DCW is not required; the role, however, must be in place in both cases.

This role allows:

- The instruction of all the transactions on owned DCWs (funding & defunding requests, wallet-to-wallet transfers, wallet-to-wallet payments, PFOs, DvPs, PVPs)¹³
- Visibility on balances and transactions history related to owned wallets
- The Instruction of all Direct Settlement operations (Payments, DvP, PVP)
- Visibility on all reference data related to owned DCWs such as the T2 account references details, DCWs details and the whitelists

2.2.4. Cash Token Supervisor

This role is exclusively granted to ECB.

It allows visibility on:

- An aggregated view of settlement activity across the Euro area, as well as a breakdown of settlement activity by NCB
- All activities related to the Token Issuance Wallet

As a Cash Token Supervisor, the following values are reflected through different Dashboards:

- Dashboard - Global View: reflecting activity involving all wallets in EUR for the current Business Date with the following counters:
 - **Outstanding:** Total amount of all balances available on all DCWs in EUR
 - **Minted:** Total number and total amount of settled issuance transactions on all DCWs in EUR
 - **Burnt:** Total number and total amount of settled redemption transactions on all DCWs in EUR
 - **Funded:** Total number and total amount of settled funding requests on all DCWs in EUR
 - **Defunded:** Total number and amount of settled defunding requests on all DCWs in EUR
 - **Settled:** Total number and total amount of settled transactions impacting all DCWs in EUR, except issuance and redemption transactions
 - **Unsettled:** Total number and total amount of unsettled transactions involving all DCWs
- Dashboard - NCBs view: reflecting activity involving all wallets per jurisdiction (per NCB) for all participating NCBs for the current Business Date
 - **Outstanding:** Total amount of all balances available on all DCWs under the NCB's jurisdiction
 - **Minted:** Total number and total amount of settled issuance transactions on all DCWs under the NCB's jurisdiction
 - **Burnt:** Total number and total amount of settled redemption transactions on all DCWs under the NCB's jurisdiction
 - **Funded:** Total number and total amount of settled funding requests on all DCWs under the NCB's jurisdiction

¹³ Access mode to the functionalities is explained in 3.5

- **Defunded:** Total number and total amount of settled defunding requests on all DCWs under the NCB's jurisdiction
- **Settled:** Total number and total amount of settled transactions impacting all DCWs under the NCB's jurisdiction, except issuance and redemption transactions
- **Unsettled:** Total number and total amount of unsettled transactions involving all DCWs under the NCB's jurisdiction

Dashboard is updated in real time, considering all transactions involving wallets in the scope of each NCB. The counters are automatically refreshed whenever a transaction is completed, so the data always reflects the most recent activity.

This allows to monitor settlement activity as it evolves throughout the business day.

2.2.5. National Cash Token Supervisor

This role is granted to every NCB.

It allows visibility on all activities related to Market Participants under its jurisdiction:

- DCW balances,
- Transactions history such as funding and defunding requests, cash token issuances, cash token redemptions, cash token transfers, payments, DvPs, PVPs and PFoDs,
- All reference data such as the T2 account references details, DCWs details and the whitelisted Market DLT Operators.

2.2.6. Operator

This role is granted to the SPs.

It allows:

- the central configuration of the platform, including managing nodes, the list of participating NCBs and roles definitions, network parameters and other general solution configuration,
- control of the Business Day processes (End of Day, Full Redemption, Start of Day) and updates the time slots of the various windows (opening and closing times, etc.), and opening dates,
- management of global reference data (e.g. business date and list of Market DLT platforms).

2.2.7. External entity instructing on behalf

Market DLT Operators have the role of external entity instructing on behalf of Market Participants. They are allowed to Instruct payments, PFoDs, DvPs and PVPs via A2A¹⁴.

2.3 Relationship between actors

NCB stands for any National Central Bank of the Eurosystem owning a node in the Pilot Solution. Its reference data is managed by the Service Provider. This reference data is configured by an Operator User with an Operator role.

¹⁴ For the Pontes Pilot, the Market DLT Operators do not have access to the GUI. The queries capabilities of Market DLT Operators are limited.

Market Participants reference data are managed by the responsible NCB to which they refer to. The list of Market Participants is visible by all NCBs but their related reference data (T2 account references, DCWs, whitelists, etc.) are visible only within the data scope of the responsible NCB.

A Market Participant operating in two different jurisdictions (i.e. an institution holding a T2 RTGS account in the books of different NCBs) must be configured by the responsible NCB of each specific jurisdiction in the reference data of its node as detailed in section [3.4.2](#).

For example:

- Market Participant PBX (for “Payment Bank X”) operates both in Country 1 (C1) and Country 2 (C2)
- Must be configured as:
 - PBX-C1 by NCB1
 - PBX-C2 by NCB2

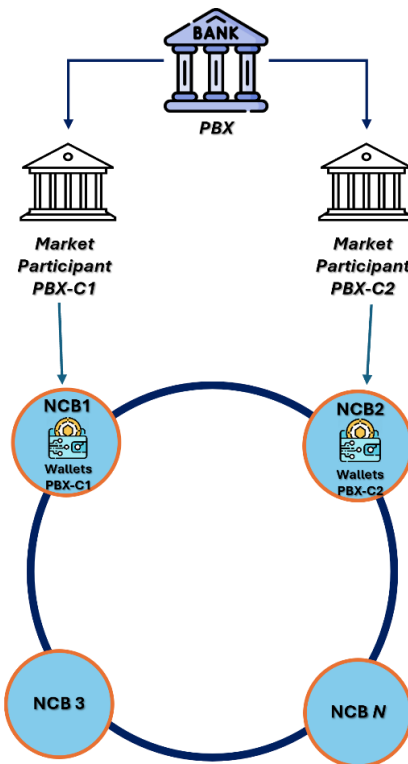


fig. i. Market Participant operating in two different jurisdictions

Interoperability functionalities and a subset of cash-related functionalities can be managed in two different modes:

- [1]** Market Participants ensure the link between the Pilot solution and the Market DLT platform. On one hand they interact directly with the Pilot solution to call cash-related and interoperability functionalities. On the other hand, they interact with a Market DLT Operator to manage assets on a Market DLT platform.
- [2]** Market Participants interact through a Market DLT Operator for:

- i. calling XvP¹⁵ and wholesale payment functionalities of the Pilot solution
- ii. and the assets management on a Market DLT platform.

If the Market Participant opts for mode [2], the Pilot solution verifies whether the Market DLT Operator is duly authorised by the Market Participant act on its behalf for using DvP and PvP, PFoD and payments functionalities.

If the Market Participant opts for mode [1], the Pilot solution performs validation and enforcement on the Market DLT Operator declared as responsible for the asset leg of the DvP / PvP. However, Pontes Pilot does not technically prevent a Market DLT Operator (with a user configured for this jurisdiction) to instruct on behalf of the Market Participant when a whitelisting between the two was configured, even if the Market DLT Operator is only intended to be used to reference the entity legally responsible for the settlement of the asset leg.

In the two previously mentioned scenarios, reference data modelling actors related to the Market DLT ecosystem and their relationship with Market Participants must be set up. These reference data are Market DLT platform, Market DLT Operator and Whitelisting.

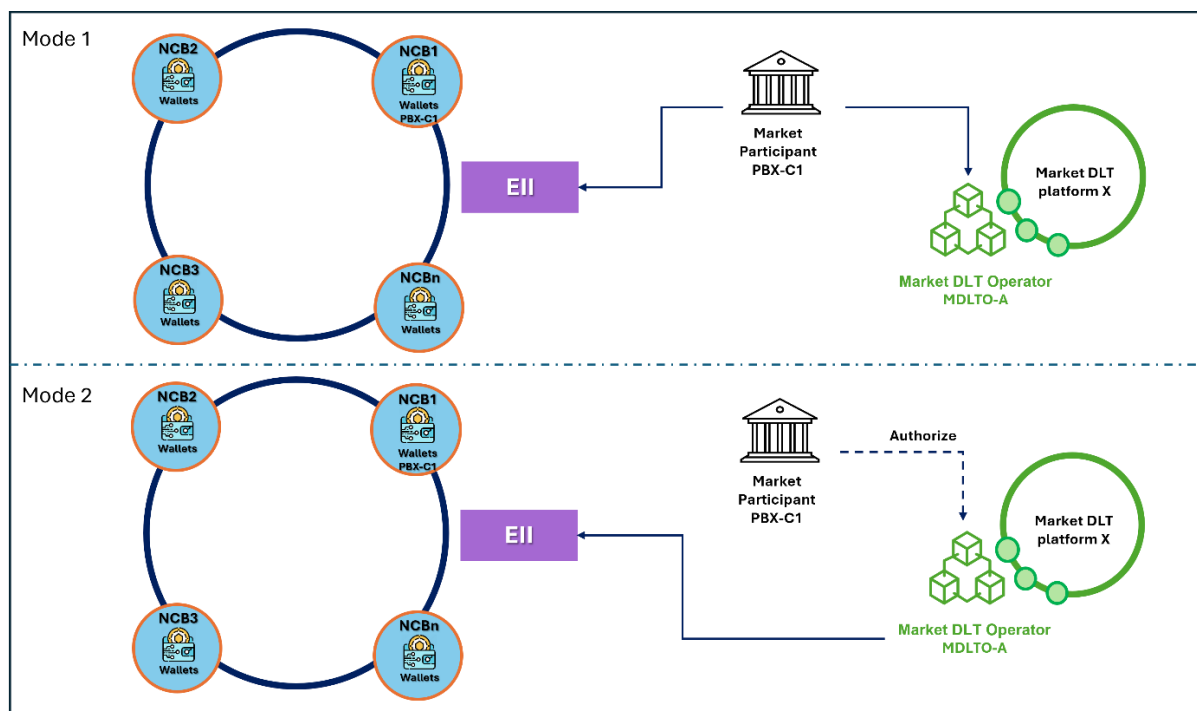


fig. i. Interaction Modes

Market DLT platform is a technical entity not representing any actor of the Pilot solution, but it needs to be configured and onboarded into the system to model the external system where assets other than CeBM (e.g. Securities, CoBM) are stored and managed.

The 2024 Eurosystem exploratory work¹⁶ showed that there can be different types of relationships between the Market DLT platforms storing the assets and Market DLT Operators managing the assets:

- 1) One Market DLT Operator operating on a private/permissioned Market DLT platform.
- 2) Multiple Market DLT Operators operating on a private/permissioned Market DLT platform.

¹⁵ Stands for DvP and PvP

¹⁶ https://www.ecb.europa.eu/pub/pdf/other/ecb_exploratoryworknewtechnologies202506_annex02.en.pdf

- 3) One or multiple Market DLT Operators operating on a public/permissionless Market DLT platform.

Because of scenarios 2) and 3), on the same Market DLT platform there could be more than one Market DLT Operator. For this reason, the Market DLT platform reference data is kept different from the Market DLT Operator. Moreover, the Market DLT platform reference data (detailed in [3.4.5](#)) is not managed by a single NCB but by the SPs and remains visible under the data scope of all NCBs.

The **Market DLT Operator** represents an actor as defined in chapter [2.1.5](#). It operates on a Market DLT platform and can process transactions on behalf of Market Participants who choose to rely on it to interact with the Pilot solution.

A Market DLT Operator must be configured in the Pilot solution by the NCB responsible for its eligibility assessment, with the reference data detailed in [3.4.6](#). Once configured, the Market DLT Operator becomes visible within the data scope of all NCBs.

The Market DLT Operator must be configured as an operator participant of an already configured Market DLT platform. While a Market DLT platform may have multiple Market DLT Operators operating on it, the same Market DLT Operator can't be configured in the Pilot solution as operating on different Market DLT Platforms. In this latter scenario, the Market DLT Operator must be configured as different reference-data occurrences.

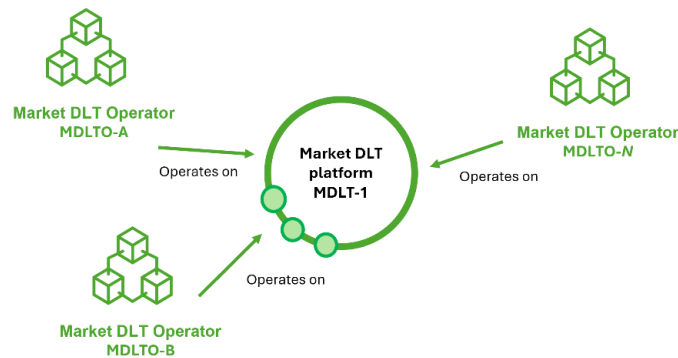


fig. i. Relationship between Market DLT Platform and Market DLT Operator

Whitelisting establishes a relationship between a Market Participant and a Market DLT Operator that enables the following:

- I. Allowing a Market DLT Operator to instruct on behalf of a Market Participant via A2A access.
- II. Designating the Market DLT Operator as the actor responsible for the asset leg of PvP and DvP transactions involving the Market Participant.

Both elements apply regardless of whether the Market Participant envisages to instruct on its own (mode 1) or whether the Market DLT Operator is supposed to instruct on behalf (mode 2) because the technical configuration in the Pilot solution does not differentiate respectively.

This reference data is managed by the NCB of the Market Participant upon Market Participant's request and is visible and valid only within that NCB data scope (see section [3.4.7](#)).

If a Market DLT Operator is delegated to operate for several Market Participants referring to different NCBs, the NCB responsible for each participant must set up the Whitelisting between the Market Participant under its responsibility and the Market DLT Operator.



fig. i. *Whitelisting*

Wrapping up the reference data, their responsible actor and their data-scope visibility in the Pilot solution:

Reference Data	Responsible Actor	Data-scope Visibility
 NCB	 Service Provider	 Pilot
 Market Participant	 NCB	 Pilot
 Market DLT Operator	 NCB	 Pilot
 Market DLT platform	 Service Provider	 Pilot
 Whitelisting	 NCB	 NCB

fig. i. *Data Scope visibility and Responsible Actor for Reference data*

Example:

- Market Participant PBX-C1 operates in Country 1 (C1)
- Market Participant PBY-C2 operates in Country 2 (C2)
- Market DLT Operator MDLTO-A is authorised under NCB3 of Country 3 (C3)
- Market DLT Operator MDLTO-A operates on Market DLT platform MDLT-1
- PBX-C1 and PBY-C2 both authorise MDLTO-A to operate on their behalf

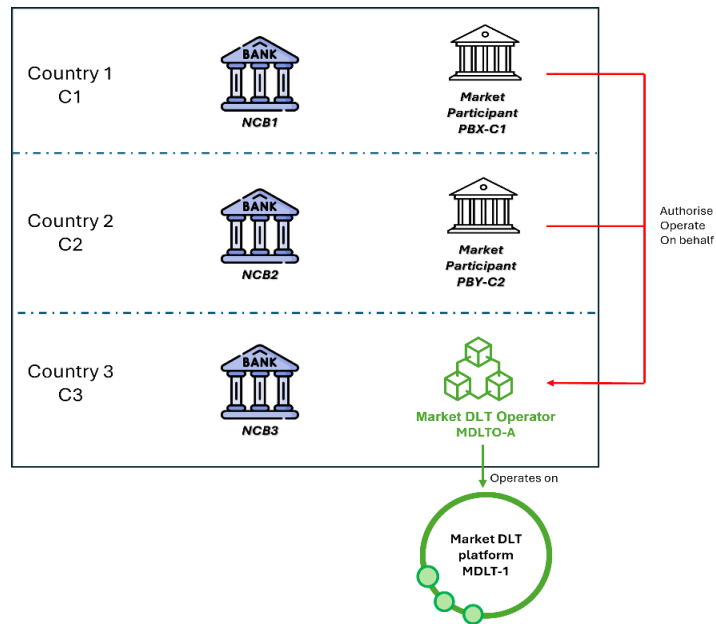


fig. i. Example of Relationship between actors - Prerequisites

The needed configuration steps are:

1. NCB1 configures Market Participant PBX-C1
2. NCB2 configures Market Participant PBY-C2
3. SPs configures MDLT-1
4. NCB3 configures MDLTO-A as participant of MDLT-1
5. NCB1 configures a Whitelist W1 to allow MDLTO-A to operate on behalf of PBX-C1
6. NCB1 configures a user MDLTOExtUsr1 with an "External User" profile allowing MDLTO-A to operate on behalf of PBX-C1 via A2A mode
7. NCB2 configures a Whitelist W2 to allow MDLTO-A to operate on behalf of PBY-C2
8. NCB2 configures a user MDLTOExtUsr2 with an "External User" profile allowing MDLTO-A to operate on behalf of PBY-C2 via A2A mode

The resulting configuration is:

- PBX-C1 is valid and visible for all NCBs (except for T2 account references, DCWs, whitelists which are only visible at the responsible NCB data scope)
- PBY-C2 is valid and visible for all NCBs (except for T2 account references, DCWs, whitelists which are only visible at the responsible NCB data scope)
- MDLT-1 is valid and visible for all NCBs
- MDLTO-A is valid and visible for all NCBs
- W1 is valid and visible only within NCB1's Node
- W2 is valid and visible only within NCB2's Node
- MDLTO-A can operate on behalf of PBX-C1 via A2A with user MDLTOExtUsr1
- MDLTO-A can operate on behalf of PBY-C2 via A2A with user MDLTOExtUsr2

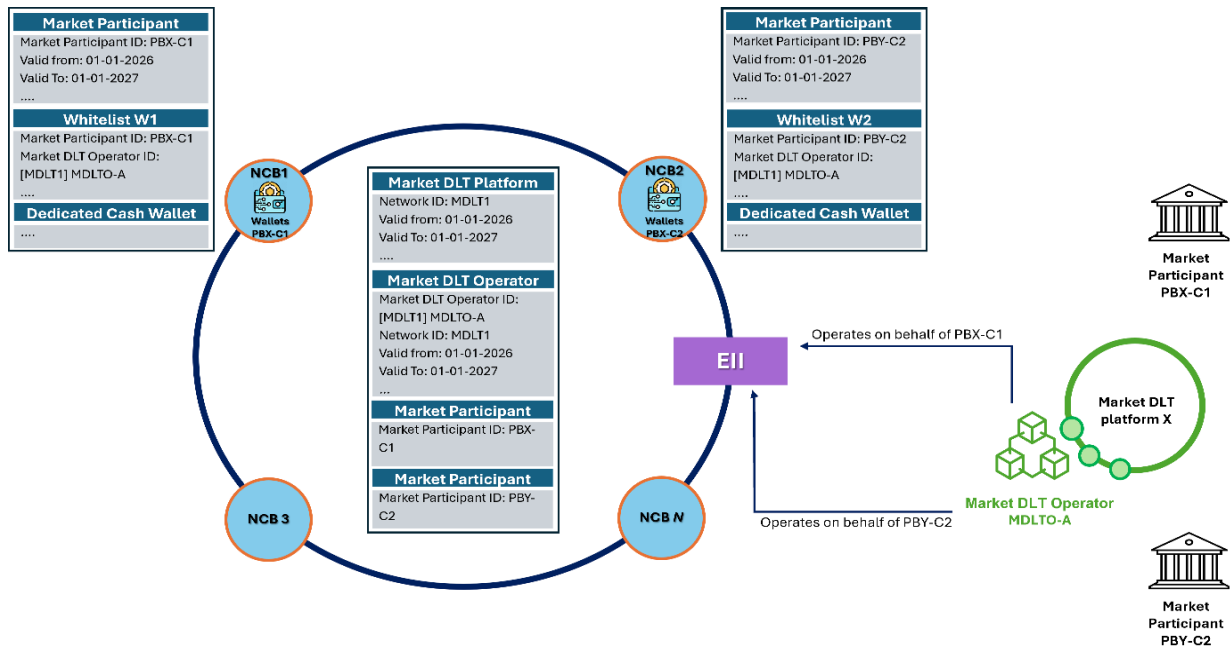


fig. i. Example of Relationship between actors – Actual configuration

2.4 User profiles

Within each actor (NCBs, Market Participants, SPs, Market DLT Operators), users are segregated according to their profiles to ensure a clear separation of duties.

Users may be granted access to multiple profiles, with exception of the administrator profile and the external user profile, which remain strictly restricted.

User profile	User Management	Reference Data	Transactions
Administrator	Read/ Create/ Update	No Access	No Access
Referential Read/Write	No Access	Read/ Create/ Validate/ Update	No Access
Referential Read-Only	No Access	Read only	No Access
Pilot Read/Write	No Access	Read only	Read/ Create/ Validate/ Update
Pilot Read-Only	No Access	Read only	Read only
Operational Read/Write	No Access	Read/ Create/Validate/Update	Read only
External User	No Access	No Access	Create

Table B. User Profiles and categories of actions

Actors/ Profiles	Administrator	Referential Read/Write	Referential Read-Only	Pilot Read/Write	Pilot Read-Only	Operational Read/Write	External User
Service Providers	✓	✗	✗	✗	✗	✓	✗
NCBs	✓	✓	✓	✓	✓	✗	✓
ECB	✓	✓	✓	✓	✓	✗	✓
Market Participants	✓	✓	✓	✓	✓	✗	✓
Market DLT Operators	✗	✗	✗	✗	✗	✗	✓

Table C. Actors and allowed User Profiles

2.4.1. Administrator (Admin User)

The administrator is exclusively responsible for the creation and management of users within their scope.

Following the onboarding forms, SPs are in charge of creating Admin users for NCBs and SPs

Admin User of an NCB/ECB can create non-admin users for the NCB/ECB itself and Admin users for its Market Participants (NCBs are not allowed to create non admin users for their Market Participants).

Admin User of a Market Participant can only create non-admin users within the entity.

Admin user of SPs can only create non-admin users within the entity.

This profile can only access user management functionalities and cannot view or modify any information related to reference data or transactions.

This profile can be assigned only to human..

2.4.2. Referential Users

Referential users are responsible for the management of reference data (e.g. T2 account references, Dedicated Cash Wallets...). They ensure alignment of data with business rules.

Two levels of access are defined for this profile:

- Referential Read/Write¹⁷: Allows viewing and full management (creation, validation, update) of reference data.
- Referential Read-Only: Allows viewing of reference data without modification rights.

This profile can be assigned to human or application user.

2.4.3. Pilot Users

Pilot users are responsible for managing transactions and they have read-only access rights to reference data, allowing them viewing only and not modifying any reference data information required for transaction processing.

Two levels of access with regards to transactions are also defined for this profile:

¹⁷ In order to be able to manage reference data, a user must have the "Referential Read Write" Profile and belong to an entity having the role of "DCW Manager" (only NCBs have this role). This prevents Market Participants from managing reference data.

- Pilot Read/Write: allowing viewing and full management (creation, validation, update) of transactions.
- Pilot Read-Only: allowing viewing transactions only, without the ability to modify data.

This profile can be assigned to human or application user.

2.4.4. Operational Users

Operational users are only granted to SPs users; this profile allows to perform all the functionalities available for Operator.

This profile can be assigned to human or application user.

2.4.5. External Users

This profile is dedicated to application users (accessing the solution through A2A only). They have access to a subset of functionalities:

- Wallet-to-wallet payment creation,
- Direct T2 RTGS payment creation,
- PFoDs, DvPs and PVPs creation.

The list of features is detailed in section [3.5](#) (see table P).

3. Business functionalities

3.1 High level functional description

The Pilot solution provides a set of business functionalities available via U2A and/or A2A. The following sections detail:

- the access channel (U2A/A2A) and mode (2-eyes/4-eyes, 1-step/2-steps) for each functionality according to the user profile,
- the signing process required for some functionalities in order to fulfil Non-Repudiation of Origin,
- the needed reference data to be configured in the Pilot solution,
- the business workflow for each settlement functionality,
- the queries and reporting available via the Pilot solution,
- the API to be used in A2A for each functionality.

It is to be noted that:

- In Pontes Pilot, the U2A functionalities are either available in mandatory 2-eyes or 4-eyes and not in both modes at the same time,
- some A2A functionalities require 2-steps process with an initialisation step and an approval one (see section [3.7](#) for more details),
- detailed documentation of the APIs can be found in the appendix [6.1](#),
- a user handbook guiding the user through the GUI in order to use the functionality in U2A can be found in the appendix [6.2](#).

3.2 U2A/A2A General Process Description

NCBs, ECB and Market Participants may access to the Pilot solution through User to Application (U2A) or Application to Application (A2A) channels. Market DLT Operators have access only via A2A.

Independently of the access channel, each actor's user will be issued a certificate for authentication before accessing the solution. The user uses its certificate to authenticate¹⁸ into the solution, once recognised by the Identity and Access Management component (IAM), signed token called Json Web Token (JWT¹⁹) is issued (See more details on connectivity in section [6.3](#)).

- In U2A, the user will then have access to the functionalities according to his access rights.
- In A2A, he will use the issued JWT for all subsequent A2A interactions.

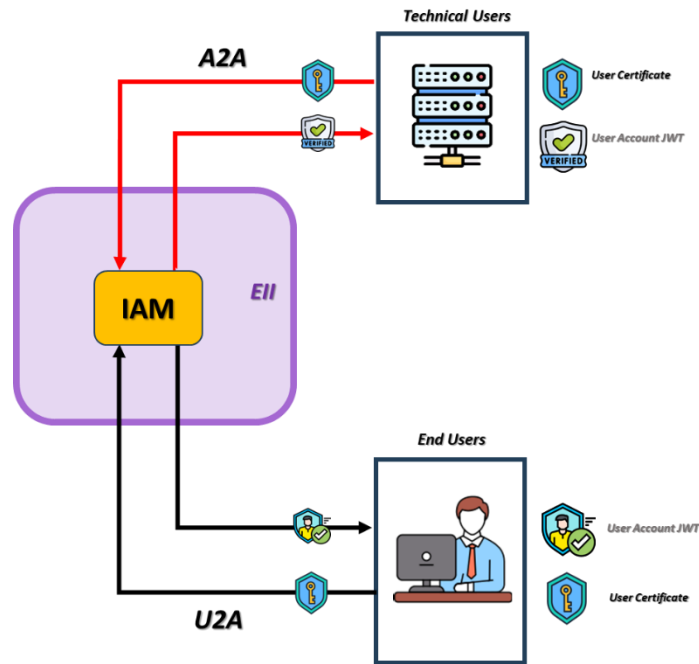


Table D. A2A/U2A Communication channels

3.2.1. Authentication & Authorisation

Each user is authenticated via its certificate which allows him to get an access control token (JWT) from the IAM. This token is used then to call the authorised Pilot APIs.

For each call, the solution checks the information provided in the JWT such as the user's profile and related entity. Then, depending on these checks the requested action is authorised or rejected.

3.2.2. Communication channels & User profiles

The table below lists the user profiles and the communication channels they are allowed to use. It is to be noted that users with profiles Administrator, Referential Read/Write, Referential Read-Only, Pilot Read/Write and Pilot Read-Only and accessing the solution through A2A will need to follow a 2-steps approach for some functionalities available in Pilot with mandatory 4-eyes validation principle (list of functionalities detailed in sections [3.4](#), [3.5](#) and [3.6](#))

¹⁸ The way to authenticate in U2A is explained in the User Handbook and in A2A in the connectivity guide and the API documentation

¹⁹ **JWT**: JSON Web Token. It is used as an authentication token, compliant with RFC 7519 standard. It represents information in a secured manner and becomes invalid if modified in anyway. It is composed of three parts: a Header with cryptographic references, a Payload with relevant meta-data, and a Signature ensuring integrity.

User profile	U2A	A2A
Administrator	Yes	No
Referential Read/Write	Yes	Yes
Referential Read-Only	Yes	Yes
Pilot Read/Write	Yes	Yes
Pilot Read-Only	Yes	Yes
Operational Read/Write	Yes	Yes
External User	No	Yes

Table E. User Profiles and access channels

3.3 Non-Repudiation of Origin (NRO)

The Non-Repudiation of Origin mechanism is designed to ensure legal certainty and accountability in transactions. It provides a way to verify the authenticity and integrity of a transaction, ensuring that the originator of a specific instruction cannot deny having sent it. Therefore, NRO helps to establish a clear chain of responsibilities and prevents disputes over the legitimacy of transactions.

For Pilot, the described NRO mechanism is implemented only for settlement transactions involving interactions with T2 RTGS through the T2 Interface since the standard NRO scheme of Target Services has no view on the data in Pilot:

- Funding and manual Defunding
- Payment with Direct Settlement in RTGS
- DvP/PvP payments with Direct Settlement in RTGS

User Certificate²⁰ and Signature:

To approve a settlement transaction, the user uses a signature. The signature embeds a certificate, a signature string and an ApprovalTimeOut.

NRO is embedded in the signatures used to approve transactions utilising the certificates of individual users. The signatures leverage asymmetric encryption to ensure that when users sign the listed settlement transactions, they are actively authorising its execution. The NRO logic is seamlessly integrated into direct settlement and funding processes. For the manual defunding (see section **Error! Reference source not found.**) an additional NRO flag is implemented in the signature, as the debited RTGS DCA is in the ownership of the ECB and not in the ownership of the Market Participant. Nonetheless the same NRO logic applies to this specific business case.

The Pilot Solution has implemented a functionality to approve (i.e. digitally sign) payment instructions in 4-eyes and 2-eyes mode. Each user's certificate must contain information about the privilege level of that user to sign the payment instructions (either 4-eyes or 2-eyes mode). During the signing process, the system will check the privilege of a user via a specific attribute in the certificate.

The certificate of a user is generated by creating a key pair (Private key and public key).

During Onboarding the privilege level and Organisation ID gets checked and the Certificate Authority (CA) attests the ownership of the public key as well as the information described above assigned to a user and sends back a verified certificate. This is part of the Connectivity Guide see section [6.3](#).

²⁰ The same certificate is used for authentication as well as signing payment /DvP/PvP instructions with direct settlement in RTGS and funding/defunding requests

Each Market Participant, NCB/ECB or market DLT Operator is responsible for securely managing the private key of their users, as it is a critical component of the signature process. If a private key is compromised, it is mandatory to invalidate the corresponding key pair. This process is managed by the Solution Operator; the certificate is revoked by including it in the revocation list available via API to all nodes and components of Pilot.

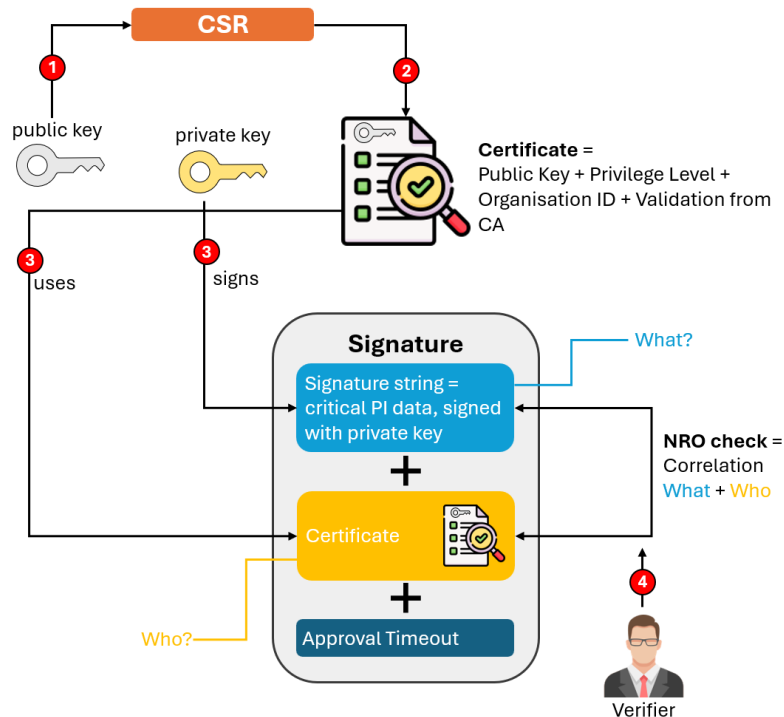


fig. ii. Non-Repudiation of Origin mechanism

Prerequisite of the steps explained in the graphic the user is identified in the IAM as a member of their organisation with their privilege level.

1. The user generates a keypair. They use the public key to send a Certificate Signing Request.
2. The information about the organisation and privilege level is checked against the IAM and embedded in the certificate together with the public key. This certificate is signed by the Certificate Authority and then sent back to the user.
3. A Signature of a transaction contains the signature string including the critical settlement transaction data, the Certificate and Approval Timeout. The user signs the signature string using their private key and using their certificate to validate the transaction.
4. The system verifies the non-repudiation of origin by checking the linkage between the signed critical settlement transaction data (What?) and the provided certificate (Who?). This is secured by leveraging asymmetric encryption, verifying that the public key provided in the certificate can "encrypt" the signed signature string.

It is to be noted that transactions with settlement in cash tokens (Transfer, Payment, DvP/PvP, PFoD) rely on the irrevocability of data recorded in the DLT which ensure the chain of responsibilities and mitigate the risk of disputes. Within the Pilot solution, each user is identified in the solution by a Unique Universal Identifier (UUID) linked to their username. All transactions instructed by the user are recorded on the ESY DLT with the corresponding UUID, making them immutable and irrevocable.

Signing process in U2A:

The description of the signing process through the GUI is described in the User Handbook.

Signing process in A2A:

For each settlement transaction, the API documentation provides detailed description on how to generate the signature and use it in the corresponding endpoint. Below a sample related to the Funding request.

Signing Rules for Funding request transactions:

- The transaction fields to be used for the signature are the following ones and they shall be concatenated in the listed specific order as a string:
 1. "techFundRequestID"
 2. "amount"
 3. "creditedCashWalletOwnerID"
 4. "debitedCashWalletOwnerID"
- No separator, delimiter, or any additional character shall be inserted between concatenated values
- The SHA256 hash shall be computed on the resulting concatenated string
- The resulting hash shall then be signed using the user private key
- The generated signature shall be provided in the "signature" field of the request payload

Step-by-Step Example:

```
- "techFundRequestID": "e3c8671d-44d7-4da1-b240-4a1b1e4e47e7"
- "amount": "10000.00"
- "creditedCashWalletOwnerID": "MP01FRAAXXX"
- "debitedCashWalletOwnerID": "ECBFDEFFXXX"
```

Step 1 - Concatenate the fields:

The concatenated string to be hashed is:

```
e3c8671d-44d7-4da1-b240-4a1b1e4e47e710000.00MP01FRAAXXXECBFDEFFXXX
```

Step 2 - Generate the SHA256 hash:

The generated hash to be signed is:

```
1a1e54c848894747b1b64ee6b3275b4c31fa516e72befa65efba16704a6fc59e
```

Step 3 - Sign the hash:

The SHA256 hash shall then be signed using the relevant user private key and using SHA256 with ECDSA algorithm.

The signing operation can be performed using different technical means, including command lines (e.g. OpenSSL), application code (e.g. NodeJS), or scripts (e.g. bash).

For instance, using openssl where privkey.pem is user's private key:

```
Openssl dgst -sha256 -binary <<
"/1a1e54c848894747b1b64ee6b3275b4c31fa516e72bfa65efba16704a6fc59e"> hash.bin
```

```
Openssl ecdsa -in privkey.pem -sing hash.bin
```

The generated signature is:

```
MEUCIBro3GEyw+e1x+oM3/ljaZt8mCobmmoptP2UOR7rsVznAiEA3YDip/rVwSAbth57kSD/7w5u
pLqszHWGPABctW8LRfE=
```

Step 4 - Populate the payload:

The generated signature shall be inserted into the "signature" of the request payload:

```
"signature":
"MEUCIBro3GEyw+e1x+oM3/ljaZt8mCobmmoptP2UOR7rsVznAiEA3YDip/rVwSAbth57kSD/7w5u
pLqszHWGPABctW8LRfE="
```

3.4 Local Reference Data Management

Each NCB node will securely store data in a Local Reference Data Management (LRDM) component to allow for the setup of its Market Participants and related wallets.

Reference data configuration	U2A			A2A			NCB only
	Available in U2A	2-eyes	4-eyes	Available in A2A	1-step	2-steps	
Dedicated Cash Wallet - Edit reference data	✓	No	✓	✓	No	✓	✓
Participant (NCB and Market Participants) - Edit reference data	✓	No	✓	✓	No	✓	✓
Market DLT Operator - Edit reference data	✓	No	✓	✓	No	✓	✓
User - Edit reference data	✓	No	✓	No	No	No	No
Participant - Block	✓	✓	No	✓	✓	No	✓
Participant - Unblock	✓	✓	No	✓	✓	No	✓
Market DLT Operator - Block	✓	✓	No	✓	✓	No	✓
Market DLT Operator - Unblock	✓	✓	No	✓	✓	No	✓
Market DLT Operator - Whitelist	✓	No	✓	✓	No	✓	✓
Instruct on behalf - Configuration	✓	No	✓	✓	No	✓	✓

Table F. List of reference data features in U2A and A2A

The T2 RTGS accounts are linked to the Pilot DCW as follows:

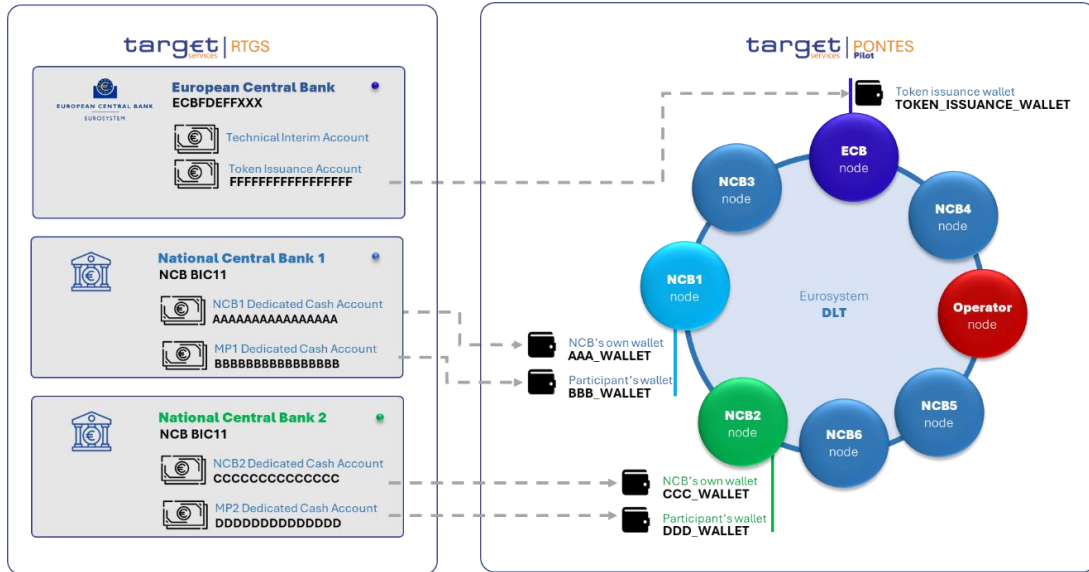


fig. iii. High-level T2 / Pontes Pilot architecture and links between T2 RTGS Accounts and Wallets

3.4.1. National Central Banks and ECB

NCBs and ECB are set up by the SPs as Operator of the solution.

Each actor is defined with the following information:

- A unique identifier:
 - BIC 11 referring to the Account BIC (Authorised Account User BIC) of the T2 RTGS account used in pilot. The recommendation is the use of the CRDM Party BIC as Account BIC to facilitate consolidation.
- Long Name: long name of the actor
- Short name: short name of the actor
- Central Bank Code
- Country code
- Country name
- Roles (see section 2.2)
- Valid from: Beginning of validity of the reference data
- Valid To: End of validity of the reference data

Field	Description	Type	Possible Values	Example	Unique	Mandatory
BIC	BIC11 of NCB/ECB	Alphanumeric (11)		NCBFRAAAXXX	✓	✓
SHORT NAME	Short name	Alphanumeric (∞)		NCB A		✓
LONG NAME	Long name	Alphanumeric (∞)		National Central Bank A		
CENTRAL BANK CODE	NCB's code	Alphanumeric (4)		FRCB	✓	✓
COUNTRY CODE	Country Code	Alphanumeric (2)	cf. ISO 3166	FR		✓
COUNTRY NAME	Country Name			FRANCE		
ROLES	Roles of the NCB		› National Cash Token Supervisor › Cash Token Supervisor › Cash Token Issuer › DCW Manager › DCW Owner			✓
VALID FROM	Beginning of validity (by default: current Business date)	Date		2027-01-01		✓
VALID TO	End of validity	Date		2027-01-31		

Table G. Data Model of National Central Banks

3.4.2. Market Participants

Market Participants are set up by their NCBs.

Each Market Participant is defined with the following information:

- A unique identifier (Pilot BIC):
 - BIC 11 referring to the Account BIC (Authorised Account User BIC, AAU= direct) of the T2 RTGS account used in pilot. The recommendation is the use of the CRDM Party BIC as Account BIC to facilitate consolidation.
- Long Name: long name of the Market Participant
- Short name: short name of the Market Participant
- Country code: Country code of the Market Participant, in the jurisdiction of the NCB.
- Country name: Country of the Market Participant, in the jurisdiction of the NCB.
- Roles (see section [2.2](#))
- Valid from: Beginning of validity of the reference data
- Valid To: End of validity of the reference data

Field	Description	Type	Possible Values	Example	Unique	Mandatory
PARTICIPANT BIC	Pilot BIC	Alphanumeric (11)		MP1ITABBXXX	✓	✓
SHORT NAME	Short name	Alphanumeric (∞)		MP A		✓
LONG NAME	Long name	Alphanumeric (∞)		Market Participant A		
COUNTRY CODE	Country Code	Alphanumeric (2)	cf. ISO 3166	IT		✓
COUNTRY NAME	Country Name			ITALY		
ROLES	Roles of the Market Participant		› DCW owner			✓
VALID FROM	Beginning of validity (by default: current Business date)	Date		2027-01-01		✓
VALID TO	End of validity	Date		2027-01-31		
BLOCKING STATUS	Blocking status of the participant	Boolean	› Blocked › Not blocked			

Table H. Data Model of Market Participant

Endpoint²¹ to create a new Market Participant entity:

Endpoint	/dlit/{ncb}/api/octopus/grs/entities
HTTP Method	POST
Actors	NCBs, ECB

3.4.3. T2 account reference

Having a T2 account number declared in the solution is a prerequisite for wallets creation. One cannot own a DCW without having a T2 RTGS account reference already declared in the solution. NCBs as DCW Managers are responsible of T2 account reference declaration in the solution for DCW Owners (including themselves).

Each T2 account reference is identified with the T2 RTGS Account number held by the Market Participant or the NCB/ECB in T2 RTGS. This reference is linked to a unique entity (Market Participant, NCB or ECB being the T2 RTGS Account holder in T2 RTGS).

The T2 account reference data is used only for manual reconciliation if needed.

Each T2 account reference is defined with the following information:

- T2 account reference: T2 RTGS account number
- Responsible NCB: the unique identifier of the NCB in charge of the T2 account in T2 RTGS
- Country code: Country code of the NCB.
- Linked Entity: the unique identifier of the entity owning the T2 account (NCBs, ECB or Market Participant)

²¹ All endpoints are exposed by the Pilot solution via EII

Field	Description	Type	Possible Values	Example	Unique	Mandatory
T2 account reference	DCA number in T2 RTGS	Alphanumeric (34)		RDEEURAAAADEBBXXX001	✓	✓
Responsible NCB	Responsible NCB's code	Alpha (4)		DECB		✓
Country Code	Country Code of NCB	Alpha (2)	cf. ISO 3166	DE		✓
Linked entity	T2 account Owner Identifier ²²	Alphanumeric (11)		AAAADEBBXXX		✓

Table I. Data Model for the link between Entities and T2 accounts

Endpoint to create a new T2 account reference:

Endpoint	/dlt/{ncb}/api/octopus/ams/t2accounts
HTTP Method	POST
Actors	NCBs, ECB

3.4.4. Dedicated Cash Wallets

Every DCW Owner (T2 RTGS account holder) can own one or several wallets in the solution linked to the same T2 account reference.

There are two types of Dedicated Cash wallets: Issuance wallets and Standard wallets. Each Dedicated Cash Wallet is defined with the following information:

Field	Description	Type	Possible Values	Example	Unique	Mandatory
WALLET ID	Wallet Identifier	Alphanumeric (18,35)		WCCAAABBB BBBBBBBBBxx xxxx W: Fixed value for Wallet CC: Country Code AAA: Currency Code BBBBBBBBBB B: Owner BIC of the Wallet xxxxxx: Alphanumeric: free text (max 17 characters)	✓	✓
OWNER	Wallet Owner identifier	Alphanumeric (11)		AAAAFRBBXXX		✓
MANAGER	Wallet Manager identifier	Alphanumeric (11)		CCCFRCBBXX X		✓
CURRENCY	Always EUR	Alpha (3)	EUR (ISO 4217)			✓
T2 ACCOUNT REFERENCE	Linked DCA Number	Alphanumeric ()		RFREURAAAA FRBBXXX001		✓
COUNTRY CODE	Country Code	Alpha (2)	cf. ISO 3166	FR		✓
MAIN WALLET FLAG	Main Wallet Flag	Boolean	› True › False			
BLOCKING STATUS	Blocking status	Boolean	› Blocked › Not blocked			

²² BIC 11 referring to the Account BIC (Authorised Account User BIC) of the T2 RTGS account used in Pilot. The recommendation is the use of the CRDM Party BIC as Account BIC to facilitate consolidation.

POA TO²³	GIVEN	BIC of Holder	PoA	Alphanumeric ()	DDDFRDBBXX X
POA AMOUNT	MAX	Maximum amount per transaction for PoA Holder		Numeric	10000
POA FROM	VALID	Beginning of validity (by default: current Business date)		Date	2027-01-01
POA VALID TO		End of validity		Date	2027-01-31
INITIATOR		Username of DCW's creator		Alphanumeric ()	JohnDoe
APPROVER		Username of DCW's validator		Alphanumeric ()	JaneDoe
VALID FROM		Beginning of validity (by default: current Business Date)		Date	2027-01-01
VALID TO		End of validity		Date	2027-01-31
CREATION DATE		DCW's creation date		Date	Value generated automatically 2026-01-01
LAST MODIFIED		Last modification timestamp		Timestamp	Value generated automatically 2026-02-18 10:47:08

Table J. Data Model of Dedicated Cash Wallets

a. Issuance wallet

For Pilot, a unique Token Issuance Wallet is set up by default and is owned by the ECB and operated by the SPs.

Token Issuance Wallet may have a zero or negative balance as it reflects the T2 RTGS payments resulting from funding and defunding requests.

b. Standard wallets

Standard DCWs are created by the responsible NCB as a DCW Manager. They should have either a zero or a positive balance.

Each DCW is identified by a wallet ID, is denominated in a specific currency “EUR” and is linked to a T2 account reference.

Both Market Participants and NCBs can own one or several standard Dedicated Cash Wallets. For each DCW Owner, all owned wallets are linked to the same T2 account reference.

For each participant (Market participant or NCB/ECB), a unique wallet can be flagged as “Main Wallet” and is used by default when processing XvPs with no Wallet ID referenced in the request.

Standard wallets are used for the settlement of cash tokens in the DLT.

Endpoint to create a Standard wallet:

Endpoint	/dlit/{ncb}/api/octopus/ams/wallets
HTTP Method	POST
Actors	NCBs, ECB

²³ A DCW can have several POAs but just one active POA for a given Business Date. Detailed information is provided for each POA active and inactive.

3.4.5. Market DLT platforms

Market DLT platform is a DLT service platform handling eligible assets and financial instruments. it is declared in the solution by the SPs.

Each Market DLT platform is defined with the following information:

Field	Description	Type	Possible Values	Example	Unique	Mandatory
MARKET DLT PLATFORM ID	Market Platform ID	Alphanumeric (∞)		MDLT1	✓	✓
MARKET DLT PLATFORM NAME	Market Platform Name	Alphanumeric (∞)		Market DLT 1		✓
OWNER NAME	Market Platform Owner Name	Alphanumeric (∞)		MDLT Owner1		✓
VALID FROM	Beginning of validity (by default: current Business Date)	Date		2027-01-01		✓
VALID TO	End of validity	Date		2027-01-31		

Table K. Data Model of Market DLT platforms

3.4.6. Market DLT Operators

Market DLT Operators are participants of the Market DLT Platform responsible for managing the asset leg. He is in relationship with the Market Participant for the asset leg processing in the Market DLT Platform and may also act on behalf of the Market Participant for the cash leg if this later delegate it to him.

Market DLT Operator is onboarded in the solution by the NCB responsible of its eligibility. Once configured, the Market DLT Operator is visible for all NCBs.

Field	Description	Type	Possible Values	Example	Unique	Mandatory
MARKET DLT OPERATOR ID	Unique identifier	Alphanumeric (∞)	ID prefixed with MDLT Platform ID MDLTPID-OpID	MDLT1-Operator1	✓	✓
RESPONSIBLE NCB	Responsible NCB code	Alpha (4)		ESCB		✓
MARKET DLT PLATFORM ID	Identifier of related Market DLT Platform	Alphanumeric (∞)		MDLT1		
BLOCKING STATUS	Blocking status of MDLT Operator	Boolean	› Blocked › Not blocked			
VALID FROM	Beginning of validity (by default: current Business Date)	Date		2027-01-01		✓
VALID TO	End of validity	Date		2027-01-31		

Table L. Data Model of Market DLT Operator

Endpoint to create a new Market DLT Operator:

Endpoint	/dlt/{ncb}/api/octopus/grs/mdlt-operators
HTTP Method	POST
Actors	NCBs, ECB

3.4.7. Whitelists

The **Whitelist** represents the relationship between a Market Participant and a Market DLT Operator (see section 2.3).

This reference data is managed by the NCB of the Market Participant upon Market Participant's request and is only visible in the data scope of the NCB.

Concretely, it is a link parametrised by the NCB between the Market Participant and the Market DLT Operator.

Field	Description	Type	Possible Values	Example	Unique	Mandatory
MARKET DLT OPERATOR ID	Market DLT Operator ID	Alphanumeric (∞)		MDLT1-Operator1	✓	✓
RESPONSIBLE NCB	Responsible NCB code	Alpha (4)		ESCB		✓
MARKET DLT PLATFORM ID	Identifier of the MDLT Platform	Alphanumeric (∞)		MDLT1		
PARTICIPANT BIC	Pilot BIC of whitelisted Participant	Alphanumeric (11)			✓	✓
VALID FROM	Beginning of validity (by default: current Business Date)	Date		2027-01-01		✓
VALID TO	End of validity	Date		2027-01-31		

Table M. Data Model of whitelist

Endpoint to whitelist a Market DLT Operator for a Market Participant:

Endpoint	/dlt/{ncb}/api/octopus/grs/mdlt-operators-whitelists-drafts
HTTP Method	POST
Actors	NCBs, ECB

3.4.8. “Instruct on behalf” configuration

a. Instruct on behalf for cash tokens configuration

Each Market Participant or NCB/ECB that has the “DCW owner” role can grant rights on his Dedicated Cash Wallet to another DCW owner which allows the latter to instruct payments debiting his wallet²⁴. The set up of the “instruct on behalf” in the Pilot solution is performed at DCW level by the responsible NCB following the Market Participant's request. The instruct on behalf authorization is valid within a fixed time period for a maximum amount per transaction. DCW can have several “instruct on behalf” setup but just one can be active for a given business date.

²⁴ Only Payments in Cash tokens are allowed (no XvPs nor PFDs, see section 3.5.4.a for more details on the functionality)

Field	Description	Type	Possible Values	Example	Unique	Mandatory
PARTICIPANT BIC	Identifier of the owner of the wallet	Alphanumeric (11)			✓	✓
WALLET ID	Wallet Identifier	Alphanumeric (19,34)		WEUFRRAAAAFR BBXXX1	✓	✓
POA ENTITY ID	Pilot BIC of the PoA Holder	Alphanumeric (11)		MP1AAAFRXXX	✓	✓
MAXIMUM AMOUNT	Maximum amount allowed for PoA Holder per transaction	Amount		10 000		
VALID FROM	Beginning of PoA Validity	Date		2027-01-01		✓
VALID TO	End of PoA validity	Date		2027-01-31		

Table N. Data Model of Instruct of behalf for Cash Tokens

Endpoint to set up an “Instruct on behalf”:

Endpoint	/dlt/{ncb}/api/octopus/ams/poa-drafts
HTTP Method	POST
Actors	NCBs, ECB

b. Instruct on behalf for direct RTGS configuration

Each Market Participant or NCB/ECB that has the “DCW owner” and or “DCW Manager” role can grant rights to another DCW owner/DCW manager which allows the latter to instruct payments in direct T2 RTGS on his behalf²⁵. The set up of the “instruct on behalf” in the DLT is performed by the responsible NCB. The authorization can be revoked at any time and is immediately effective.

Field	Description	Type	Possible Values	Example	Unique	Mandatory
GRANTOR PARTICIPANT BIC	Pilot BIC of the Participant granting the rights	Alphanumeric (11)		MP01ESAAXXX		✓
AUTHORIZED PARTICIPANT BIC	Pilot BIC of the Participant receiving the rights	Alphanumeric (11)		MP02ESAAXXX		✓
STATUS	Status of authorization	Alphanumeric (∞)	- Active - Revoked			

Table O. Data Model of Instruct of behalf for Direct T2 RTGS

Endpoint to set up an “Instruct on behalf”:

Endpoint	/dlt/{ncb}/api/octopus/tms/instruct-on-behalf
HTTP Method	POST
Actors	NCBs, ECB

3.4.9. Market Participant Blocking / Unblocking

Market Participant Blocking/Unblocking feature is only available for NCBs/ECB and the SPs as Operators of the Pilot Solution. A Market DLT Operator can only be blocked or unblocked by its responsible NCB.

²⁵ Only Payments in direct RTGS settlement are allowed (no XvPs nor PFDs, see section 3.5.4.a for more details on the functionality)

The blocking/unblocking are immediately effective and are not submitted to 4-eyes.

If a Market Participant is blocked:

- All his Dedicated Cash Wallets are blocked which prevents him from debiting and crediting any cash tokens from his wallets.
- This also prevents settlement of any new Direct Settlement transactions involving the blocked participant.
- Its wallets can only be debited through manual defunding instructed by its responsible NCB and Cash Token Sweep.

Blocking status is reflected both in the participants list and the wallets list (see sections [3.4.2](#) and [3.4.4](#)).

Endpoint to Block/unblock a Market Participant:

Endpoint	/dlt/{ncb}/api/octopus/grs/entities/{entityid}
HTTP Method	PATCH
Actors	NCBs, ECB, SPs (Operator)

3.4.10. Market DLT Operator Blocking / Unblocking

Market DLT Operator Blocking/Unblocking feature is available for NCBs. Only the NCB who onboarded the Market DLT Operator can block/unblock it, while every NCB can see its status (blocked or not).

The blocking/unblocking are immediately effective and are not submitted to 4-eyes.

If a Market DLT Operator is blocked:

- Payment with cash token, Payment with Direct T2 RTGS settlement, Payment Free of Delivery and DvP /PvP instructed by this market DLT Operator will be rejected.
- DvP and PvP instructed by a Market Participant with this market DLT Operator referenced as responsible of the asset part will be rejected

Blocking status is reflected in the Market DLT Operator list (see sections [3.4.6](#)).

Endpoint to Block/unblock a Market DLT Operator:

Endpoint	/dlt/{ncb}/api/octopus/grs/mdlt-operators/{mdltOperatorID}
HTTP Method	PATCH
Actors	NCBs, ECB

3.4.11. Users

Users with profile admin user can add users in the Pontes Pilot reference data and are responsible for their administration.

Once created, the following user-related information may be updated:

- First name
- Last name
- Profiles: it's possible to add or remove a profile (Exception: Admin and external users cannot be modified nor combined with other profiles)
- Status: Admin user can disable/enable an already created user, and the action is immediately effective

Field	Description	Type	Possible Values	Example	Unique	Mandatory
Username	Username of the user	Alphanumeric (∞)		For Admin user: AFR MP01FRAAXXTT For other users: PFR MP01FRAAXXTT	✓	✓
Email	Email of the user	Alphanumeric (∞)				✓
First name	First name of the user	Alphanumeric (∞)				
Last name	Last name of the user	Alphanumeric (∞)				
Entity ID	Identifier of the actor to which belongs the user	Alphanumeric (11)	Pilot BIC or Market DLT Operator ID	MP01FRAAXXX		✓
Profiles	Profile(s) assigned to the user		› Admin User › Referential Read/Write › Referential Read-Only › Pilot Read/Write › Pilot Read-Only › External User			✓
Email verified	Indicates if the user has already validated its email	Boolean	› Yes › No			
Status	Indicates whether the user is enabled or disabled	Boolean	› Enabled › Disabled			✓

Table P. Data Model of Users reference data

3.5 Settlement transactions

List of functionalities available on the Pilot solution for both NCBs/ECB and their Market Participants with the access mode:

	U2A			A2A		
	available in U2A	2-eyes	4-eyes	available in A2A	1-step	2-steps
Settlement transactions						
Funding Dedicated Cash Wallet - Instruct	✓	No	✓	✓	No	✓
Defunding Dedicated Cash Wallet - Instruct	✓	No	✓	✓	No	✓
Wallet to Wallet Cash Token Transfer - Instruct	✓	No	✓	✓	No	✓
Wallet to Wallet Payment - Instruct	✓	No	✓	✓	✓	✓
Payment Free of Delivery (PFoD) - Initialisation	No	No	No	✓	✓	No
Payment Free of Delivery (PFoD) - Settlement	No	No	No	✓	✓	No
DvP and PvP transaction - Initialisation	No	No	No	✓	✓	No
DvP and PvP transaction - Payment on Eurosystem DLT	No	No	No	✓	✓	No
DvP and PvP transaction - Get Key and payment info (Execution or Cancellation)	No	No	No	✓	✓	No
DvP and PvP transaction (Direct Settlement in T2 RTGS) - Initialisation	No	No	No	✓	✓	No
DvP and PvP transaction (Direct Settlement in T2 RTGS) - Payment in T2 RTGS	No	No	No	✓	✓	No
DvP and PvP transaction (Direct Settlement in T2 RTGS) - Get Key and payment info (Execution or Cancellation)	No	No	No	✓	✓	No
Payment (Direct Settlement in T2) - Instruct	✓	No	✓	✓	✓	✓

Table Q. List of features available to Market Participants

List of functionalities available in the Pilot solution for market DLT Operators “acting on behalf” with the access mode:

Market DLT Operators' functionalities	A2A
	Available in A2A (1-step only)
Wallet to Wallet Payment - Instruct	✓
Payment Free of Delivery (PFoD) - Initialisation	✓
Payment Free of Delivery (PFoD) - Settlement	✓
DvP and PvP transaction - Initialisation	✓
DvP and PvP transaction - Payment on Eurosystem DLT	✓
DvP and PvP transaction - Get Key and payment info (Execution or Cancellation)	✓
DvP and PvP transaction (direct settlement in T2 RTGS) - Initialisation	✓
DvP and PvP transaction (direct settlement in T2 RTGS) - Payment in T2 RTGS	✓
DvP and PvP transaction (direct settlement in T2 RTGS) - Get Key and payment info (Execution or Cancellation)	✓
Payment (direct settlement in T2 RTGS) - Instruct	✓

Table R. List of features available to market DLT Operators

3.5.1. Funding Requests

In order to fund a Dedicated Cash Wallet in the Pilot solution, Market Participants can instruct funding requests to the ESY DLT which will debit their account in T2 RTGS.

NCBs, as DCW managers, can also instruct funding request on behalf of their Market Participants (only in case of contingency situations.).

Once the funding request is submitted, a payment instruction is created and processed in T2 RTGS debiting the Market Participant's T2 RTGS account and crediting the ECB's Token issuance account through the ECB's Technical interim account.

Once the Payment is successful, an issuance request is generated and processed in the ESY DLT debiting the ECB's Token Issuance Wallet and crediting the Market Participant's wallet.

Funding requests are submitted following a 4-eyes principle.

Steps from 5 to 14 are automated and do not need any manual intervention.

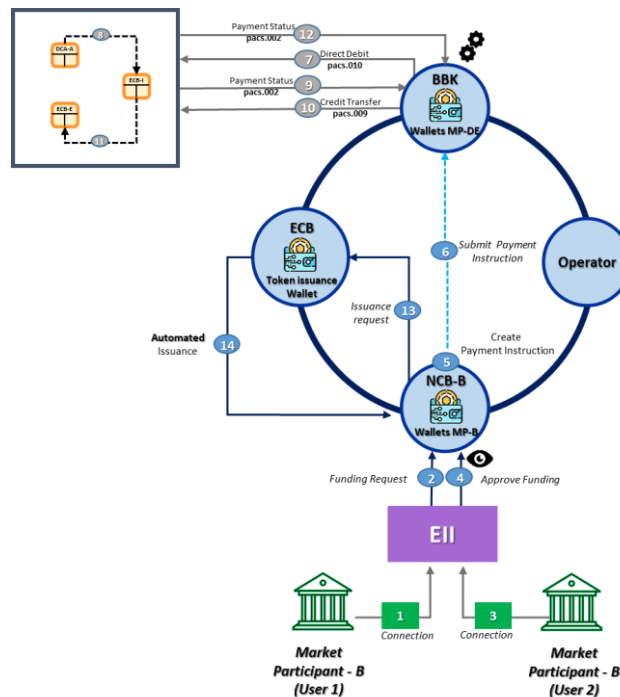


fig. iv. Funding request workflow in U2A mode

1&3. Connection:

User authenticates via the agreed security protocols, establishing a trusted session with the ESY DLT

2. Funding Request:

Market Participant sends a funding request to NCB Manager node, specifying the DCW, amount, and currency.

Funding request endpoint:

Endpoint	/dlt/{ncb}/api/octopus/tms/funding-request
HTTP Method	POST
Actors	Market Participant, NCB on behalf of the Market Participant

4. 4-eyes Approval:

User 2 Approves Funding request

Second step approval for funding request endpoint:

Endpoint	/dlit/{ncb}/api/octopus/tms /funding-request-draft/{id}/{status}
HTTP Method	PUT
Actors	Market Participant, NCB

5. Create Payment Instruction:

Upon approval of the request, NCB Manager node generates automatically the payment instruction, mapping the Account BIC with the Owner BIC of the DCW, doing the necessary reference data checks.

6. On-Chain Submission:

NCB Manager node submits the payment instruction to the ledger which is captured by BBK via Trigger.

7-12. T2 RTGS Settlement:

T2 Interface translates the on-chain payment instruction into ISO 20022 pacs messages, debits the Market Participant's T2 RTGS account, (subject to the existence of a direct debit mandate), credits the Interim T2 RTGS account, moves the funds into "Token Issuance Account" and receives T2 RTGS settlement confirmations

13. Issuance Request:

Upon successful status, the NCB Manager node backend sends automatically an issuance request to ECB Issuer node

14. Automatic Issuance:

ECB Issuer node debits the "Token Issuance Wallet" and credits automatically the requested amount to the Market Participant's DCW after confirming Payment Status

3.5.2. Defunding Requests

To defund a Dedicated Cash Wallet and receive the liquidity on their T2 RTGS account in the Pilot solution, Market Participants can instruct defunding requests from their DCW in the ESY DLT.

NCBs, as DCW managers, can also instruct defunding requests on behalf of their Market Participants (only in case of contingency situations.).

Once the defunding request is submitted, a redemption operation is processed in the ESY DLT debiting the Market Participant's wallet and crediting the ECB's Token Issuance Wallet.

Once the redemption is successful, a payment instruction is created and processed in T2 RTGS debiting the ECB's Token issuance account and crediting the Market Participant's account through the ECB's Technical interim account.

Defunding requests are submitted following a 4-eyes principle.

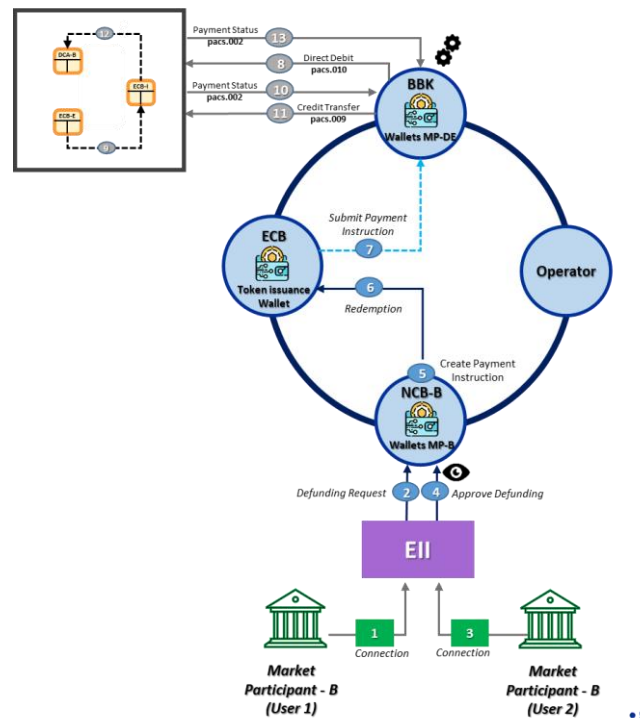


fig. v. Defunding request workflow in U2A mode

1&3. Connection:

User authenticates via the agreed security protocols, establishing a trusted session with the ESY DLT.

2. Defunding Request

Market Participant sends a defunding request to NCB Manager node, specifying the DCW, amount, and currency.

Defunding request endpoint:

Endpoint	/dlt/{ncb}/api/octopus/tms /defunding-request
HTTP Method	POST
Actors	Market Participant, NCB

4. 4-eyes Approval

User 2 Approves defunding request.

Second step approval for defunding request endpoint:

Endpoint	/dlt/{ncb}/api/octopus/tms /defunding-request-draft/{id}/{status}
HTTP Method	PUT
Actors	Market Participant, NCB

5. Create Payment Instruction:

NCB Manager node generates the payment instruction, mapping the Account BIC with the Owner BIC of the DCW and doing the necessary reference data checks

6. Redemption:

Upon approval by User 2, NCB Manager node backend instructs a redemption toward ECB Issuer node which debits the PB-B Wallet and credits the “Token Issuance Wallet”

7. Payment On-Chain Submission:

ECB Issuer node submits the Payment instruction to the ledger

8-13. T2 RTGS Settlement:

T2 Interface turns the on-chain payment instruction into ISO 20022 pacs messages, debits the ECB Token Issuance Account, credits the ECB Interim Account, moves the funds into Market Participant's T2 RTGS account and receives T2 RTGS settlement confirmations

3.5.3. Wallet to wallet cash token transfers

Cash tokens transfer is a movement of cash tokens between two wallets belonging to the same Market Participant for liquidity management and distribution purposes.

Transfers can be either instructed by the Market Participants itself or their NCBs on their behalf (in contingency situations) and are submitted following a 4-eyes principle.

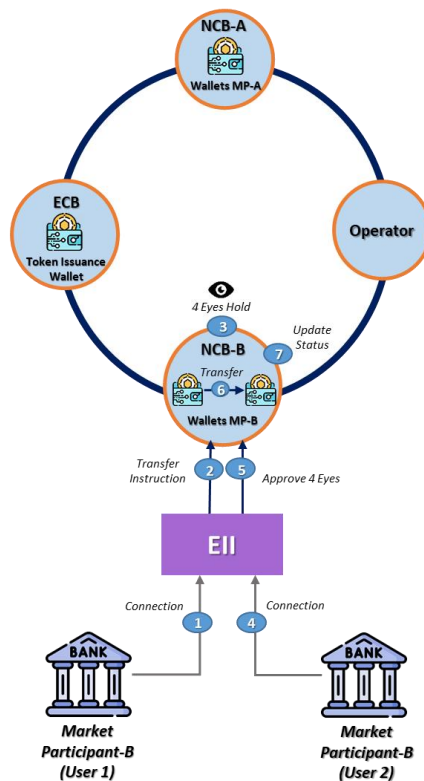


fig. vi. Wallet to wallet cash tokens transfer high level workflow in U2A mode

1. & 4. Connection

User authenticates via the agreed security protocols, establishing a trusted session with the ESY DLT.

2. Transfer request

Market Participant sends a transfer request to NCB Manager node, specifying the debited and credited DCWs and amount.

Wallet to wallet transfer endpoint:

Endpoint	/dlt/{ncb}/api/octopus/rvs/transactions-request
HTTP Method	POST
Actors	Market Participant, NCB

3. 4-eyes Hold

Transfer request enters a 4-eyes state, awaiting validation from a second user.

5. 4-eyes Approval

User 2 Approves transfer request.

Second step approval for wallet-to-wallet transfer endpoint:

Endpoint	/dlt/{ncb}/api/octopus/rvs/transactions-draft/{id}/{status}
HTTP Method	PUT
Actors	Market Participant, NCB

6 & 7. Transfer settlement & status update

Transfer is settled, MP-B DCW1 is debited and MP-B DCW2 is credited.

Status of the transfer is updated.

3.5.4. Payments

a. Payment with cash tokens

A payment in cash tokens is a transfer of cash between two wallets executed in exchange of an asset, usually carried out to discharge a debt or obligation.

Payments can be either instructed by:

- the Market Participant itself in U2A (4-eyes) or A2A (1-step and/or 2-steps),
- its NCB on its behalf in case of contingency in U2A (4-eyes) or A2A (1-step and/or 2-steps),
- another Market Participant in case of an “Instruct on behalf” set up in U2A (4-eyes) or A2A (2-steps),
- a Market DLT Operator in case of whitelisting set up in A2A (1-step).

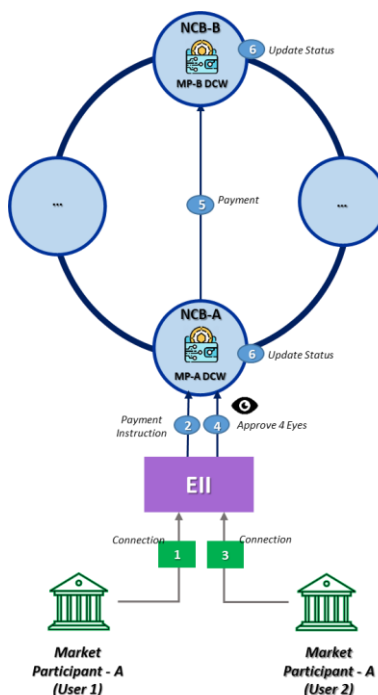


fig. vii. Cash tokens payment high level workflow in U2A mode

1&3. Connection

Users of Market Participant A authenticate via the agreed security protocols, establishing a trusted session with the solution.

2. Payment Request

Market Participant A sends a payment request to its NCB Manager node, specifying the debited and credited DCWs and amount.

Wallet to wallet payment endpoint (2-steps) subject to second step approval:

Endpoint	/dlt/{ncb}/api/octopus/rvs/transactions-requests
HTTP Method	POST
Actors	Market Participant, NCB

Wallet to wallet payment endpoint (1-step):

Endpoint	/dlt/{ncb}/api/bridge/payments
HTTP Method	POST
Actors	Market Participant, NCB, Market DLT Operator on behalf of Market Participant

4. 4-eyes Approval

User 2 Approves payment request.

Second step approval for wallet-to-wallet payment endpoint:

Endpoint	/dlt/{ncb}/api/octopus/rvs/transactions-draft/{id}/{status}
HTTP Method	PUT
Actors	Market Participant, NCB

5&6. Payment settlement & status update

Payment is settled, MB-A DCW of Market Participant A is debited and MB_B DCW of Market Participant B is credited.

Status of the payment is updated on both debited and credited nodes.

b. Payment with direct T2 RTGS settlement

Payments that are instructed on the ESY-DLT can be settled directly in T2 RTGS leading to immediate finality. This functionality does not require Market participants to own a DCW.

Payments with Direct T2 RTGS settlement can be either instructed by:

- the Market Participant itself in U2A (4-eyes) or A2A (1-step and/or 2-steps),
- its NCB on its behalf in case of contingency in U2A (4-eyes) or A2A (1-step and/or 2-steps),
- another Market Participant or NCB in case of an "Instruct on behalf" set up in U2A (4-eyes) or A2A (2-steps),
- a Market DLT Operator in case of whitelisting set up in A2A (1-step).

The unhappy paths and error codes can be found in section [3.5.7](#).

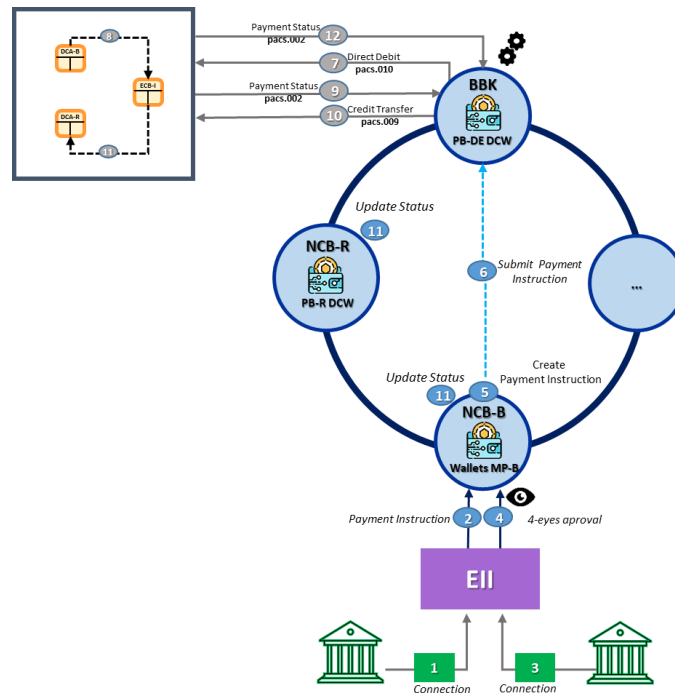


fig. viii. Payment in Direct T2 RTGS Settlement in U2A 4-eyes

1.-3. Connection: User authenticates via the agreed security protocols, The process of establishing a trusted session connection with the EII is explained in chapter 6.3.

2. Direct T2 RTGS Payment Initiation: The Payment with Direct T2 RTGS Settlement is initiated by a participant. The required fields to be completed are payer BIC, receiver BIC, amount and Correlation ID.

Direct T2 RTGS payment endpoint (2-steps) subject to second step approval:

Endpoint	/dlt/{ncb}/api/octopus/tms/direct-rtgs/payments
HTTP Method	POST
Actors	Market Participant, NCB

Direct T2 RTGS payment endpoint (1-step):

Endpoint	/dlt/{ncb}/api/bridge/direct-rtgs/payments
HTTP Method	POST
Actors	Market Participant, NCB, Market DLT Operator on behalf of Market Participant

After being authorised, the Market Participant or the Market DLT Operator acting on behalf of the Market Participant sends a request to Pilot solution for the Payment using the end points listed above.

4. 4-Eyes Approval: The second user 4-eyes approves the Payment with Direct T2 RTGS settlement

Second step approval for Direct T2 RTGS payment endpoint:

Endpoint	/dlt/{ncb}/api/octopus/tms/direct-rtgs/payments-drafts/{id}/{status}
HTTP Method	PUT
Actors	Market Participant, NCB

5. **Payment Instruction Generation on the Ledger:** The corresponding NCB Manager node creates the Payment Instruction (PI) on the ledger retrieving the necessary information from the reference data.
6. **On-Chain Approval and Submission:** The PI is approved and submitted on the ledger. The Bundesbank node (Trigger backend) receives the PI and forwards it to the T2 Interface connected to T2 RTGS.
- 7.-10. **T2 RTGS Settlement:** The T2 Interface converts the PI into ISO 20022 pacs messages, debits the participant's T2 RTGS account, credits the T2 RTGS DCA used as a technical interim account, transfers the funds to the receivers T2 RTGS account, and receives the T2 RTGS settlement confirmation.
11. **Broadcast and Update Payment Status:** The Bundesbank node receives the settlement confirmation, records it on the ledger, and forwards it to the involved participants.

3.5.5. Payment Free of Delivery

Payment Free of Delivery (PFoD) is a transaction where cash tokens are transferred without the simultaneous exchange of the related asset or security.

Example: A coupon payment on a bond is a typical PFoD, the bond issuer pays interest (the coupon) to the bond holder, but no security is delivered in return at that moment.

PFoD involves two separate legs: a delivering leg "DELI" and a receiving leg "RECE" both originating from Market DLT Operators or Market Participants and both processed by the ESY DLT. The Pilot ensures that the amount provided by both legs matches.

a. Payment Free of Delivery – DELI leg

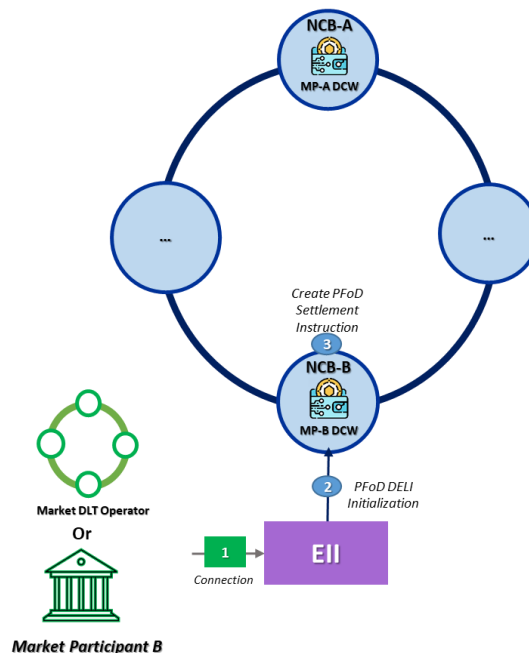


fig. ix. Payment Free of Delivery – DELI high level workflow in A2A mode

1. Connection:

Market Participant MP-B or Market DLT Operator authenticates via the agreed security protocols, establishing a trusted session.

2. PFoD DELI Initialisation:

The connected actor sends the PFoD DELI request to the corresponding NCB of the Market Participant.

PFoD DELI endpoint (1-step):

Endpoint	/dlt/{ncb}/api/bridge/initpfodeli
HTTP Method	POST
Actors	Market Participant, NCB, Market DLT Operator on behalf of Market Participant

3. Create PFoD Instruction:

Upon reception of PFoD request, NCB Manager node backend create a PFoD Settlement Instruction.

b. Payment Free of Delivery – RECE leg

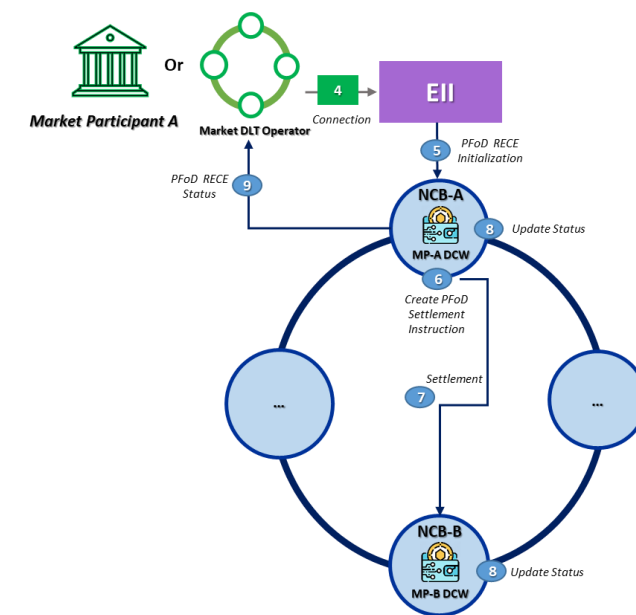


fig. x. Payment Free of Delivery- RECE high level workflow in A2A mode

4. Connection:

Market Participant MP-A or the Market DLT Operator authenticates via the agreed security protocols, establishing a trusted session.

5. PFoD RECE Initialisation:

The connected actor sends the PFoD RECE request to the corresponding NCB of the debited Market Participant.

PFoD RECE endpoint (1-step):

Endpoint	/dlt/{ncb}/api/bridge/initpfodrece
HTTP Method	POST
Actors	Market Participant, NCB, Market DLT Operator on behalf of Market Participant

6. Create PFoD Instruction:

Upon reception of PFoD settlement request, NCB Manager node backend creates a PFoD Settlement Instruction

7 & 8. Settlement and status update:

PFoD is settled with a cash transfer from the wallet of MP-A to the wallet of MP-B.

Status of the PFoD is updated on both debited and credited nodes.

9. PFoD RECE status update:

Status of the PFoD update is sent to Market DLT Operator.

3.5.6. Delivery versus Payment & Payment versus Payment

The DvP and PvP functionalities enable the settlement of transactions made up by two legs registered in different ledgers.

The first leg involves the delivery of an asset (Securities, Commercial Money, any other eligible asset defined in Business Description Document) and the second leg involves the payment of a cash amount in Central Bank Money (CeBM).

In the interoperability model the two legs involved in DvP and PvP reside in different ledgers managed by different systems, in our case the asset leg is managed in an external Market DLT platform, and the cash leg is managed in the Pilot solution.

The atomicity of the transaction among different ledgers and systems is guaranteed by an interoperability mechanism which manages the all-or-none settlement of the two legs of the transactions.

The interoperability mechanism used in the Pilot solution is the Hash Link Protocol (see details in section [5.3](#)) which is implemented in Pilot solution as follows.

The Seller of a DvP / PvP is modelled with a Market Participant which receives Euro on the Pilot solution while delivering assets (Securities, Commercial Money, any other eligible asset) on the Market DLT platform.

The Buyer of a DvP / PvP is modelled with a Market Participant which delivers Euro on the Pilot solution while receiving assets (Securities, Commercial Money, any other eligible asset) on the Market DLT platform.

The role of “Oracle of time” is implemented applying a timeout to each DvP / PvP during which is possible for the Buyer to pay for the cash leg. This timeout is equal to timestamp of creation of the DvP / PvP plus the system parameter defining the DvP / PvP timeframe. This system parameter is defined by the Operator, and it is equal to 30min for all DvP / PvP operations.

The ESY-DLT component stores payments related to the cash leg of the DvP / PvP and keeps track of the status of the payment which changes during the life of the DvP / PvP:

- When the Buyer submits a payment request for the DvP / PvP, the payment is stored in ESY-DLT with status “PENDING”. This status will evolve eventually in either “SETTLED” or “UNSETTLED” status.
- In case the payment for the cash leg of the DvP / PvP is correctly settled, it is stored in the ESY-DLT with status “SETTLED”.
- In case the payment for the cash leg of the DvP / PvP is not settled for missing funds, it is stored in the ESY-DLT with status “UNSETTLED”.
- In case the DvP / PvP is timed out and no payment has been created yet; the Pilot solution creates a payment for the DvP / PvP and stores it in the ESY-DLT with status “BURNED”

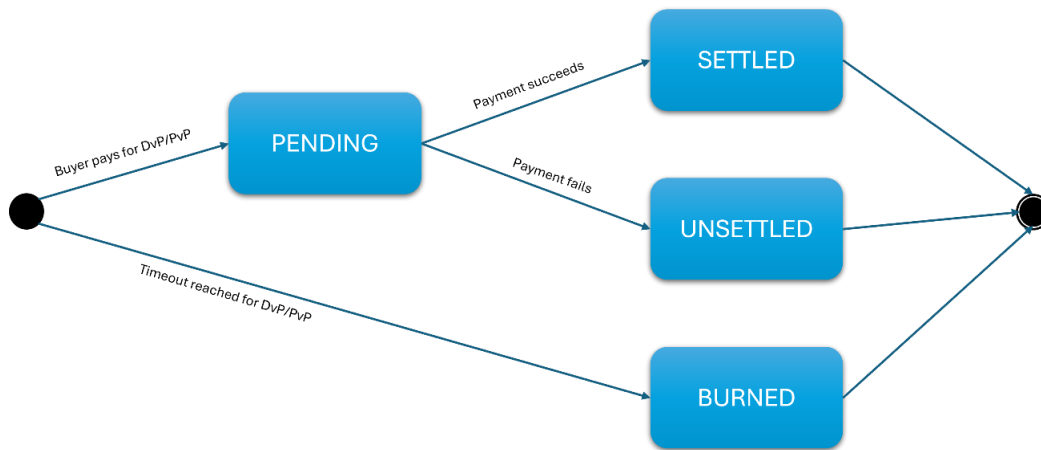


fig. i. State Diagram of Payment status

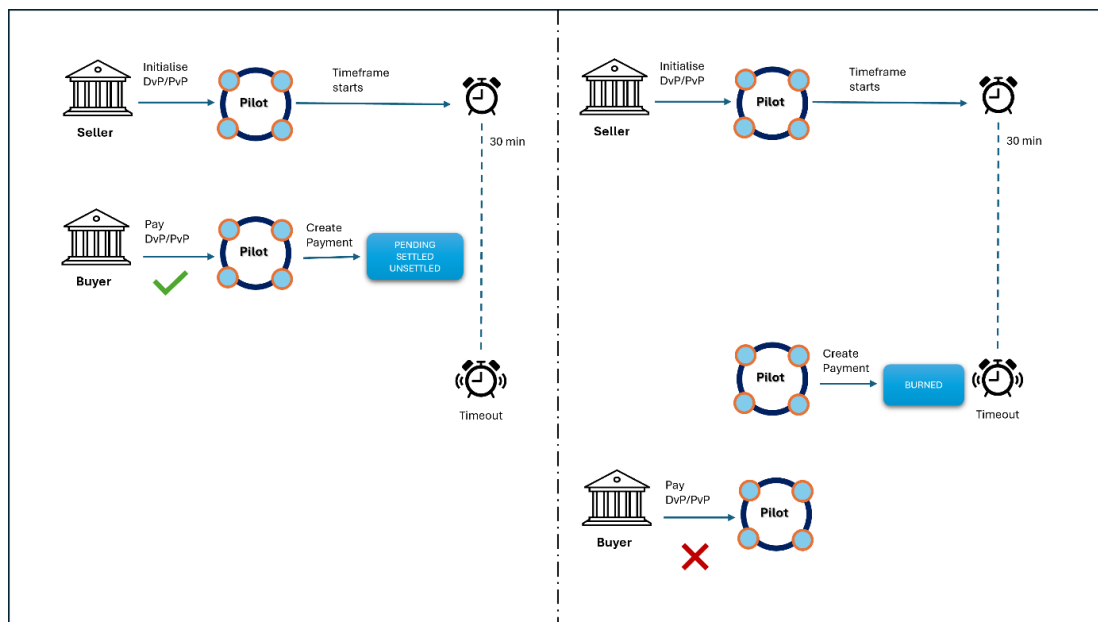


fig. ii. Relationship between Timeout and Payment

The role of trusted oracle for disputes is implemented by conditioning the release of the execution and cancellation keys to the payment of the cash leg of the DvP / PvP:

- In case there is no payment in ESY-DLT, neither execution nor cancellation key is released because it is still possible to pay for the DvP / PvP.
- In case payment is in status “PENDING”, neither execution nor cancellation key is released because even if it is not possible to pay anymore for the DvP / PvP, the status is not yet final.
- In case payment is in status “UNSETTLED” or “BURNED”, only the cancellation key is released to the Seller because it is not possible to pay anymore for the DvP / PvP and the status indicates that the Seller will never receive the funds agreed in the DvP / PvP.
- In case payment is in status “SETTLED”, only the execution key is released to the Buyer because the status indicates that the Seller has received the funds agreed in the DvP / PvP.

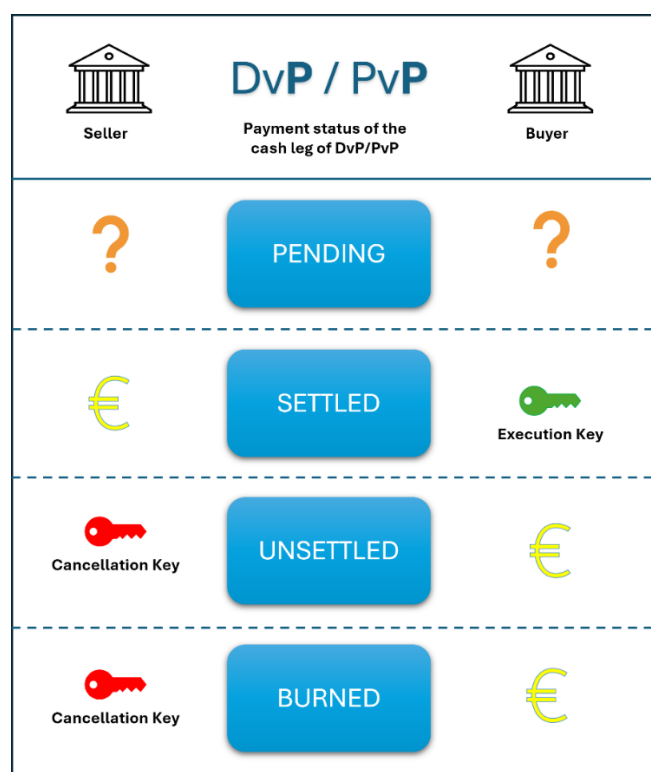


fig. iii. Relationship between payment status and keys disclosure

The cash leg of the DvP and PvP can be managed in the Pilot solution with two different settlement models:

- Settlement in Cash Token: cash token liquidity is settled immediately, i.e. DCWs are immediately credited/debited accordingly with the amounts mentioned in the DvP/PvP instruction; while finality in CeBM is reached (i.e. T2 RTGS accounts are credited/debited) at the end of the business day, (with an automatic liquidity sweep from the Pilot solution to T2 RTGS) or after a defunding request initiated by the Market Participant.
- Direct Settlement in T2 RTGS: finality in CeBM is reached immediately in T2 RTGS, i.e. T2 RTGS accounts are immediately credited/debited accordingly with the amounts mentioned in the DvP/PvP instruction.

As previously mentioned, the Pilot solution handles only the initialisation and the cash leg of DvP and PvP transactions via interoperability functionalities. It is not responsible for managing the asset leg on the Market DLT platform, which falls under the responsibility of the Market DLT Operator.

Interoperability functionalities can be managed in two different modes:

1. Market Participants interact directly with the Pilot solution to call interoperability functionalities and interact directly with a Market DLT Operator to manage assets on a Market DLT platform.
2. Market Participants don't interact directly with the Pilot solution but delegate to a Market DLT Operator both calling interoperability functionalities of Pilot and the managing of the assets on a Market DLT platform.

In both modes, the actor invoking Pilot functionalities for DvP and PvP must provide a reference to the Market DLT Operator responsible for the asset leg among submitted input data.

Pilot solution:

- i) performs enforcement and validation on this reference against Market DLT Operator reference data and its whitelisting with Market Participant involved in the operation
- ii) retains this reference for audit purposes, supporting investigations in case issues arise on the asset side.

The DvP and PvP functionality is made up by different sub-functionalities detailed as follow. For the sake of simplicity, from now on we will use only DvP in the following paragraphs because the concepts apply in the same way also to PvP.

a. Initialisation functionality

The Initialisation functionality allows the Seller creating a new DvP on the Pilot solution in a synchronous way.

Upon Seller request, the Pilot solution creates and stores a new DvP, returning as a response the set of information needed to initialise the Hash Link Contract in the Market DLT Platform and to operate according to the Hash Link protocol.

This functionality is available only in A2A mode and it is exposed with two REST endpoints for the two different settlement models.

The endpoint for each settlement model manages both DvP and PvP transactions.

When calling the Initialisation functionality, the Seller determines the settlement mode for the DvP / PvP by selecting the dedicated endpoint (see the subsequent tables). From this step on, the settlement mode is fixed for all the lifecycle of the DvP / PvP, and the endpoints of any other functionality must be called accordingly.

For example, it is not possible to invoke the direct settlement mode endpoint for payment for a DvP created with the cash token settlement mode endpoint for initialisation.

Initialisation functionality endpoint for DvP and PvP with Settlement in Cash Token:

Endpoint	/igw/{ncb}/v1/xvps
HTTP Method	POST
Actors	Market Participant (Seller), Market DLT Operator instructing on behalf of the Seller

Initialisation functionality endpoint for DvP and PvP with Direct Settlement in T2 RTGS:

Endpoint	/igw/{ncb}/v1/direct-rtgs/xvps
HTTP Method	POST
Actors	Market Participant (Seller), Market DLT Operator instructing on behalf of the Seller

High level workflow of the functionality

The high-level workflow of the functionality is the same for both settlement models.

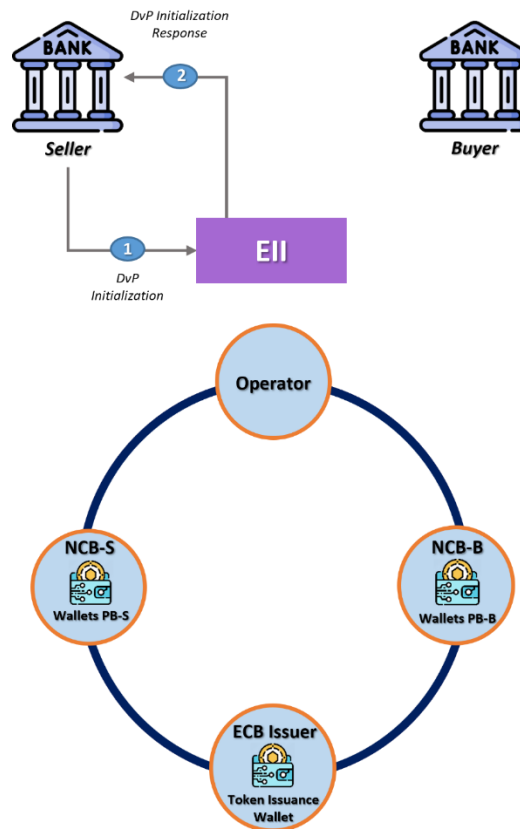


fig. iv. XvP initialisation high level workflow in A2A (by Seller)

1 – DvP Initialisation

After being authorised, the Actor sends a request to the Pilot solution for the creation of a new DvP.

The relevant information contained in the request is:

- BICs defined as Entity ID of the Market Participants involved as Seller and Buyer;
- ID of the Market DLT Operator responsible for the asset leg of the DvP from the Seller perspective.
- Information of the Payment to submit to settle the cash leg (amount, currency).

The endpoint for Settlement in Cash Token accepts additional optional information:

- DCW ID of the Dedicated Cash Wallet to credit in the Payment instead of the main Dedicated Cash Wallet of the Seller Market Participant.

The exhaustive list of input data is detailed in the API documentation (see Swagger).

2 – DvP Initialisation Response

The Pilot solution performs authorisation controls and checks the validity of input data.

If all checks and controls are successful, the Pilot solution creates the new DvP and returns its details to the Actor.

The relevant information contained in the response is:

- BICs defined as Entity ID of the Market Participants involved as Seller and Buyer.
- ID of the Market DLT Operator responsible for the asset leg of the DvP.
- Information of the Payment to submit to settle the cash leg (amount, currency).

- Unique ID of the newly created DvP transaction to use in all subsequent steps.
- Hashes of the execution and cancellation keys needed to create the HLC smart contract on the Market DLT platform.
- Timeout value for the DvP which indicates i) the moment until when it is possible for the Buyer to pay for the cash leg of the DvP and ii) the moment after which it is possible for the Seller to request the cancellation key in case the cash leg of the DvP is not settled.

The endpoint for Direct Settlement in Cash Token returns additional information if it's present:

- DCW ID of the Dedicated Cash Wallet to credit in the Payment instead of the main DCW of the Seller Market Participant.

The exhaustive list of output data is detailed in the API documentation.

b. Initialisation Query functionality

The Initialisation Query functionality allows the Seller and the Buyer to retrieve the details of an existing DvP from the Pilot solution in a synchronous way.

This functionality serves two use cases:

- In the principal use case, the Buyer calls the functionality to retrieve the data of the DvP agreed with the Seller and verify that the HLC smart contract has been created on the Market DLT platform in compliance with agreed terms before proceeding with the payment.
- In the marginal use case, the Seller calls the functionality to retrieve the data of a DvP previously created.

This functionality is available only in A2A mode and it is exposed with two REST endpoints for the two different settlement models.

The endpoint for each settlement model manages both DvP and PvP transactions.

Initialisation Query functionality endpoint for DvP and PvP with Settlement in Cash Token:

Endpoint	/igw/{ncb}/v1/xvps/{xvpTransactionId}
HTTP Method	GET
Actors	Market Participant (Buyer), Market DLT Operator instructing on behalf of the Buyer Market Participant (Seller), Market DLT Operator instructing on behalf of the Seller

Initialisation Query functionality endpoint for DvP and PvP with Direct Settlement in T2 RTGS:

Endpoint	/igw/{ncb}/v1/direct-rtgs/xvps/{xvpTransactionId}
HTTP Method	GET
Actors	Market Participant (Buyer), Market DLT Operator instructing on behalf of the Buyer Market Participant (Seller), Market DLT Operator instructing on behalf of the Seller

High level workflow of the functionality

The High-level workflow of the functionality is the same for both settlement models.

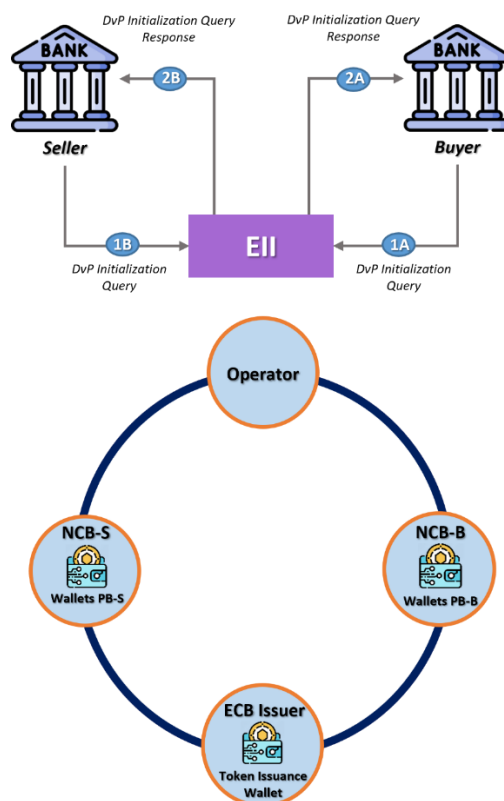


fig. i. XvP initialisation query high level workflow in A2A (Buyer retrieves hash-lock)

1a and 1b – DvP Initialisation Query

After being authorised, the Actor sends a request to the Pilot solution to retrieve the details of a DvP identified by a unique ID.

2a and 2b – DvP Initialisation Query Response

The Pilot solution performs authorisation controls to check if the DvP exists.

If all checks and controls are successful, the Pilot solution retrieves the requested DvP and returns its details to the Actor.

The relevant information contained in the response is:

- BICs defined as Entity ID of the Market Participants involved as Seller and Buyer.
- ID of the Market DLT Operator responsible for the asset leg of the DvP.
- Information of the Payment to submit to settle the cash leg (amount, currency).
- Unique ID of the created DvP transaction to use in all subsequent steps.
- Hashes of the execution and cancellation keys used to create the HLC smart contract on the Market DLT platform.
- Timeout value for the DvP which indicates i) the moment until when it is possible for the Buyer to pay for the cash leg of the DvP and ii) the moment after which it is possible for the Seller to request the cancellation key in case the cash leg of the DvP is not settled.

The endpoint for Settlement in Cash Token returns additional information if it's present:

- DCW ID of the Dedicated Cash Wallet to credit in the Payment instead of the main DCW of the Seller Market Participant.

The exhaustive list of output data is detailed in the API documentation.

c. Payment functionality

The Payment functionality allows the Buyer to pay for the cash leg of a DvP in a synchronous way.

Upon Buyer request to pay for a DvP, the Pilot solution returns as a response:

- Status of the Payment related to DvP
- Execution Key, if the Payment has been properly executed

This functionality is available only in A2A mode and it is exposed with two REST endpoints for the two different settlement models.

The endpoint for each settlement model manages both DvP and PvP transactions.

Payment functionality endpoint for DvP and PvP with Settlement in Cash Token:

Endpoint	/igw/{ncb}/v1/xvps/{xvpTransactionId}/payment
HTTP Method	POST
Actors	Market Participant (Buyer), Market DLT Operator instructing on behalf of the Buyer

Payment functionality endpoint for DvP and PvP with Direct Settlement in T2 RTGS:

Endpoint	/igw/{ncb}/v1/direct-rtgs/xvps/{xvpTransactionId}/payment
HTTP Method	POST
Actors	Market Participant (Buyer), Market DLT Operator instructing on behalf of the Buyer

Settlement in Cash Tokens

In the Settlement in Cash Tokens, the Payment is settled in cash token on ESY-DLT involving Seller and Buyer's wallets (represented as "PB S wallet" and "PB B wallet" in the graphs).

The Payment is settled in the form of tokenised cash on the Pilot solution, it will be final in CeBM only after mirroring it also in Seller and Buyer's T2 RTGS DCAs, i.e. after the automatic end of day cash sweep or upon a defunding request initiated by the Market Participant.

High level workflow of the functionality

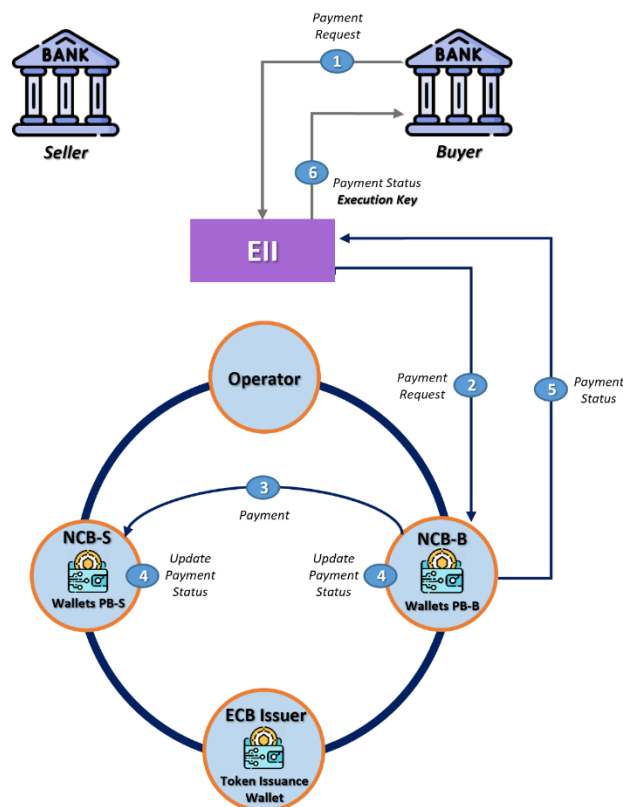


fig. i. XvP settled in cash tokens high level workflow in A2A

1 – Payment Request

After being authorised, the Actor sends a request to the Pilot solution for the Payment of the cash leg of an existing DvP.

The relevant information contained in the request is:

- Unique ID of the DvP transaction for which the payment is sent.
- BICs defined as Entity ID of the Market Participants involved as Seller and Buyer.
- ID of the Market DLT Operator responsible for the asset leg of the DvP from Buyer perspective.
- Information of the Payment to submit to settle the cash leg (amount, currency).

The endpoint for Settlement in Cash Token accepts additional optional information:

- DCW ID of the Dedicated Cash Wallet to debit in the Payment instead of the main DCW of the Buyer Market Participant.
- The exhaustive list of input data is detailed in the API documentation.

2 – Internal Payment Request

EII component performs authorisation controls and checks the validity of input data, then checks if DvP exists, and checks payment data submitted in input against the payment data stored for the DvP (e.g. Seller BIC and Buyer BIC in the request must be equal to Seller BIC and Buyer BIC of the DvP; amount and currency in the request must be equal to amount and currency of the DvP).

If any of this check fails, the request is rejected. The Actor can retry the payment of the DvP by submitting again the request with correct data.

Then EII checks if it is possible to pay for the DvP, namely if a payment already exists for the cash leg of the DvP or if the timeout for submitting the payment request for the DvP has been already reached.

If any of these checks fail, the request is rejected. In such case, the Actor is not permitted to retry the payment of the DvP.

If all checks are successful, EII it forwards the Payment request to the Eurosystem DLT node of the NCB of the buyer.

3 – Payment

The Eurosystem DLT node of the NCB of the buyer takes in charge the payment request and, together with the node of the NCB of the Seller, checks the payment data against reference data (e.g. involved DCWs are valid, whitelisting is in place, etc.)

If any of this check fails, the request is rejected. The Actor can retry the payment of the DvP by submitting again the request either with correct data or after amending the reference data.

If all checks are successful, the Eurosystem DLT node of the NCB of the Buyer and the node of the NCB of the Seller insert the Payment in the ledger with status “PENDING”.

4 – Update Payment Status

The Eurosystem DLT node of the NCB of the Buyer checks if it is possible to settle the payment (e.g. the Debited DCW has sufficient funds).

If this check fails, the Eurosystem DLT node of the NCB of the Buyer and the node of the NCB of the Seller update the Payment in the ledger setting the status “UNSETTLED”.

If the check is successful, the Eurosystem DLT node of the NCB of the Buyer and the node of the NCB of the Seller update the Payment in the ledger setting the status “SETTLED”.

5 – Internal Payment Response

Eurosystem DLT node of the NCB of the buyer returns the status of Payment to EII component.

6 –Payment Response

Depending on the status of the Payment, EII components checks if also Execution Key can be returned together with Payment status information.

The relevant information contained in the response is:

- Unique ID of the DvP transaction for which the payment is sent
- Information of the created Payment (ID, status, status reason).
- The Execution Key needed to call the forced execution of the HLC. This is returned only if Payment status is “SETTLED”.

The exhaustive list of output data is detailed in the API documentation.

Direct Settlement

The Direct Settlement functionality allows the settlement of the Payment in Central Bank Money in T2 RTGS involving the Seller and the Buyer's T2 RTGS accounts.

High level workflow of the Direct Settlement functionality

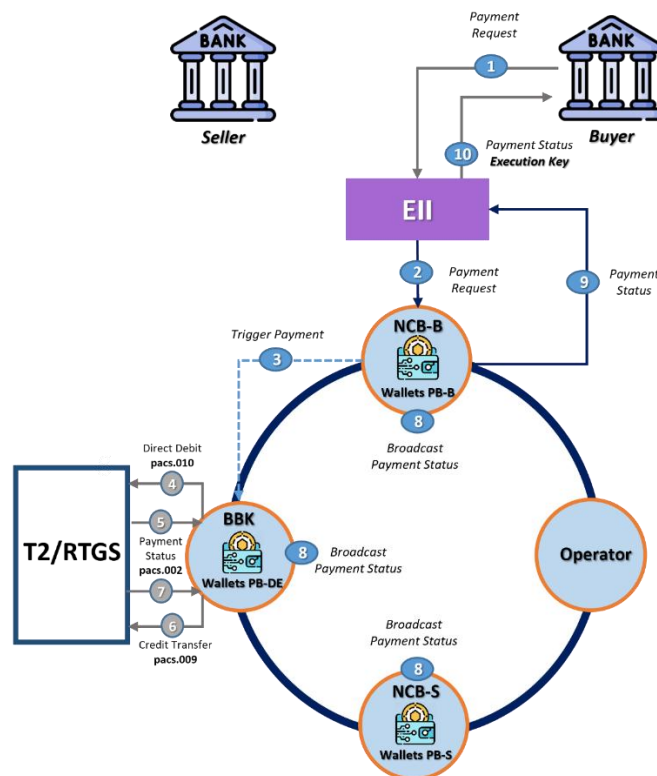


fig. i. XvP Direct Settlement high level workflow in A2A

1 – Payment Request

After being authorised, the Actor sends a request to the Pilot solution for the Payment of the cash leg of an existing DvP.

The relevant information contained in the request is:

- Unique ID of the DvP transaction for which the payment is sent.
- BICs defined as Entity ID of the Market Participants involved as Seller and Buyer.
- ID of the Market DLT Operator responsible for the asset leg of the DvP from the Buyer perspective.
- Information of the Payment to submit to settle the cash leg (amount, currency).

The endpoint for Direct Settlement requires additional mandatory information:

- Information required for NRO purposes, namely the signature of all the input data and the certificate used to sign.

The exhaustive list of input data is detailed in the API documentation.

2 – Internal Payment Request

EII component performs authorisation controls and checks the validity of input data, then checks if DvP exists, and checks payment data submitted in input against the payment data stored for the DvP (e.g. Seller BIC and Buyer BIC in the request must be equal to Seller BIC and Buyer BIC of the DvP; amount and currency in the request must be equal to amount and currency of the DvP).

If any of this check fails, the request is rejected. The Actor can retry the payment of the DvP by submitting again the request with correct data.

Then EII checks if it is possible to pay for the DvP, namely if a payment already exists for the cash leg of the DvP or if the timeout for submitting the payment request for the DvP has been already reached.

If any of this check fails, the request is rejected. In such case, the Actor is not permitted to retry the payment of the DvP.

If all checks are successful, EII forwards the Payment request to the Eurosystem DLT node of the NCB of the Buyer.

3 – Payment Request

The Eurosystem DLT node of the NCB of the buyer, after other checks, create a Payment Instruction with status “PENDING”, approves it and submits it to BBK DLT Node.

4, 5, 6, 7– T2 RTGS Settlement

On BBK DLT node, Trigger Backend component, acting as link to ESMIG, turns the payment instruction into ISO 20022 pacs messages.

It debits the Buyer T2 RTGS DCA using pacs.010, then gets confirmation of the outcome of the successful or unsuccessful debit of the buyers T2 RTGS DCA via the pacs.002 response and, in case of success, credits the T2 RTGS DCA of the Sellers DCA via pacs.009 and gets the pacs.002 response with the outcome of the crediting the Seller T2 RTGS DCA.

8 – Broadcast Payment Status

BBK node receives the T2 RTGS settlement outcome, records the success on-chain or the failure, and propagates the final payment status, “SETTLED” in case of success and “UNSETTLED” in case of failure to the involved NCB nodes.

9 –Internal Payment Response

Eurosystem DLT node of the NCB of the buyer returns the status of Payment to EII component.

10 –Payment Response

Depending on the status of the Payment, EII components checks if also Execution Key can be returned together with Payment status info.

The relevant information contained in the response is:

- Unique ID of the DvP transaction for which the payment is sent.
- Information of the created Payment (ID, status, status reason).
- The Execution Key needed to call the forced execution of the HLC. This is returned only if Payment status is "SETTLED".

The exhaustive list of output data is detailed in the API documentation.

d. Get Key and Payment Info functionality

The Get Key and Payment Info functionality allows the Seller and the Buyer to retrieve from the Solution in a synchronous way:

- Status of the Payment related to DvP
- Execution Key, if the calling actor is the Buyer and the Payment has been properly executed
- Cancellation Key, if the calling actor is the Seller and the Payment has not been properly executed

This functionality serves two use cases:

- In the main use case, the Seller calls the Pilot solution to retrieve the data of the Payment and the Cancellation Key if the DvP is failed , meaning that it is impossible to settle the cash leg because i) the DvP is timed out (the Payment of the DvP is in status "BURNED") or ii) the payment of the DvP was not successful (Payment of the DvP is in status "UNSETTLED").
- In the marginal use case, the Buyer calls the Pilot solution to retrieve the data of the Payment and the Execution Key if the DvP is executed, meaning that the payment of the DvP is in status "SETTLED". This scenario is marginal because the Buyer receives the Execution key in Payment functionality.

This functionality is available only in A2A mode and it is exposed with two REST endpoints for the two different settlement models.

The endpoint for each settlement model manages both DvP and PvP transactions.

Get Key and Payment Info functionality endpoint for DvP and PvP with Settlement in Cash Token:

Endpoint	/igw/{ncb}/v1/xvps/{xvpTransactionId}/payment?key={key}
HTTP Method	GET
Actors	Market Participant (Seller), Market DLT Operator instructing on behalf of the Seller Market Participant (Buyer), Market DLT Operator instructing on behalf of the Buyer

Get Key and Payment Info functionality endpoint for DvP and PvP with Direct Settlement in T2 RTGS:

Endpoint	/igw/{ncb}/v1/direct-rtgs/xvps/{xvpTransactionId}/payment?key={key}
HTTP Method	GET
Actors	Market Participant (Seller), Market DLT Operator instructing on behalf of the Seller Market Participant (Buyer), Market DLT Operator instructing on behalf of the Buyer

High level workflow of the functionality

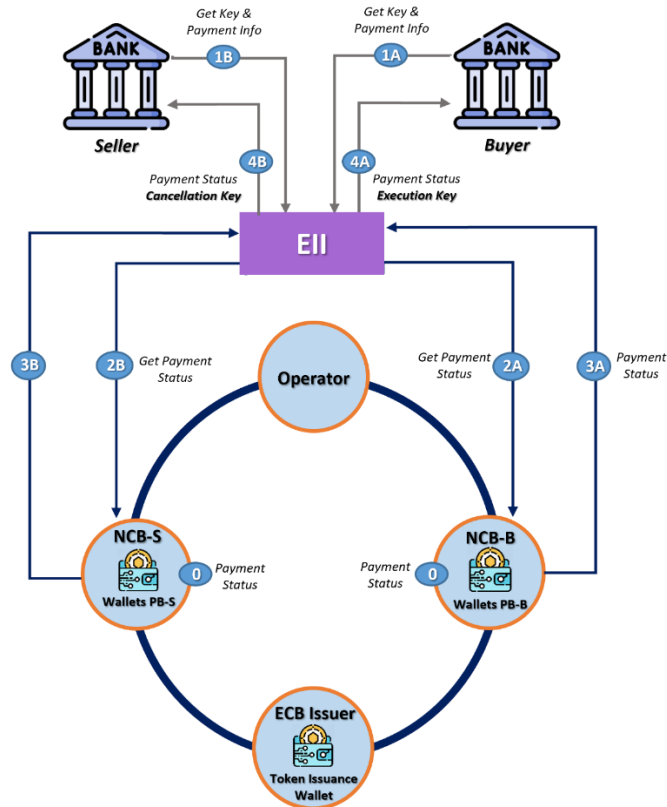


fig. i. Get Key & payment high level workflow in A2A

The High-level workflow of the functionality is the same for both settlement models.

The input and output data are the same for both endpoints.

1a and 1b – Get Key and Payment Info request

After being authorised, the Actor sends a request to the Pilot solution to retrieve the key (if available) and the details of the payment related to a DvP.

The relevant information contained in the request is:

- Unique ID of the DvP transaction for which the payment is sent
- Key type to retrieve depending on the calling Actor. Seller can only request “CANCELLATION” key while Buyer can only request “EXECUTION” key.

The exhaustive list of input data is detailed in the API documentation.

2a and 2b – Payment Status internal request

The request is first handled by EII component which performs authorisation controls and checks the validity of input data, then tries to identify the Payment related to the DvP.

If any of this check fails, the request is rejected. The Actor can retry to query the payment of the DvP by submitting again the request with correct data.

If all checks are successful, EII creates and forwards an internal request to the Eurosystem DLT node of the NCB of the Actor, to retrieve the status of the Payment.

3a and 3b – Payment Status internal response

The Eurosystem DLT node of the NCB of the Actor performs checks if the Payment exists

If the check fails, the request is rejected because there is any Payment related to the DvP.

If the check is successful, the Eurosystem DLT node of the NCB of the Actor retrieves the Payment from the ledger and returns its status to EII component.

4a and 4b – Get Key and Payment Info response

Depending on the status of the Payment, EII components checks if Cancellation Key or Execution Key can be returned together with Payment status info.

The relevant information contained in the response is:

- Unique ID of the DvP transaction for which the payment is sent;
- Information of the Payment (ID, status, status reason);
- The Execution Key needed to call the forced execution of the HLC. This is returned only if Payment status is “SETTLED” and the Actor is the Buyer;
- The Cancellation Key needed to call the forced cancellation of the HLC. This is returned only if Payment status is “UNSETTLED” or “BURNED” and the Actor is the Seller.

The exhaustive list of output data is detailed in the API documentation.

e. Initialisation on Market DLT platform

The Seller, after the Initialisation of the DvP on the Pilot solution, proceeds to initialise the Hash Link Contract in the Market DLT platform.

To operate accordingly to the Hash Link Protocol, the Hash Link Contract shall have these features:

- Act as Escrow locking the asset on the Market DLT platform.
- Transfer the asset to the Seller or the Buyer conditionally to either proving their identity or the provisioning of a secret (Cancellation or Execution key).

Aforementioned features can be implemented by a set of functions used in the key management scenarios:

- Cooperative Execution: transfer the asset to the Buyer, triggerable by the Seller upon proving its identity
- Forced Execution: transfer the asset to the Buyer, triggerable by the Buyer upon proving its identity and providing the Execution key
- Cooperative Cancellation: transfer the asset to the Seller, triggerable by the Buyer upon proving its identity
- Forced Cancellation: transfer the asset to the Seller, triggerable by the Seller upon proving its identity and providing Cancellation key

The Initialisation of the Hash Link contract is out of the perimeter of the Pilot solution, nevertheless, a possible interaction, to be intended only as a suggestion, could be:

- a) Create Hash Link Contract: The Seller creates HLC on Market DLT platform with DvP information
- b) Lock the asset on the Market DLT platform: The Seller locks the asset in HLC
- c) The Buyer becomes aware of HLC creation: The Buyer becomes aware of the transaction ID and the asset is locked on the Market DLT platform. This can happen in different ways: The Seller notifies off-chain the Buyer, the Buyer receives a notification from the DLT (e.g. event), the Buyer queries the contract on the DLT.

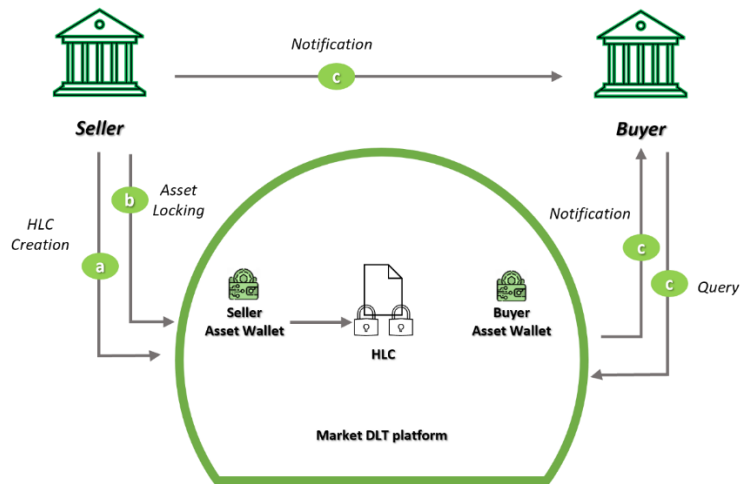


fig. ii. XvP initialisation on a Market DLT platform

f. Key Management flows for Market Participant

Hereafter are reported the interaction flows of a Market Participant with Hash Link functionalities describing the management of the secrets (Execution Key and Cancellation Key) needed to transfer the asset on Market DLT platform.

Anyway, the perimeter of the Pilot solution stops with the delivery of the keys and payment status to the actor while all the subsequent steps follow under the domain of Market DLT platform.

So, these flows must be considered as examples showing how Delivery versus Payment & Payment versus Payment functionalities can serve different scenarios in Market DLT platform and they must not be considered in any case binding for any policy or implementation on Market DLT platform.

The following flows apply in the same way both to DvP and PvP and both to settlement in cash token and Direct settlement mode. For the sake of simplicity, in the examples we will use only DvP with settlement in cash token.

Initialisation

In the Initialisation flow, the Seller calls the Initialisation function, the Pilot solution creates a new instance of a DvP and then returns back to Seller DvP details data.

Then the Seller, using DvP details data, initialises on Market DLT platform a new Hash Link Contract and locks the asset into it.

Outside the perimeter of the Pilot solution, the Buyer becomes aware of the creation of the Hash Link Contract on the Market DLT platform and then calls the Initialisation Query function to verify that the DvP created by the Seller on the Pilot solution respects the agreed terms.

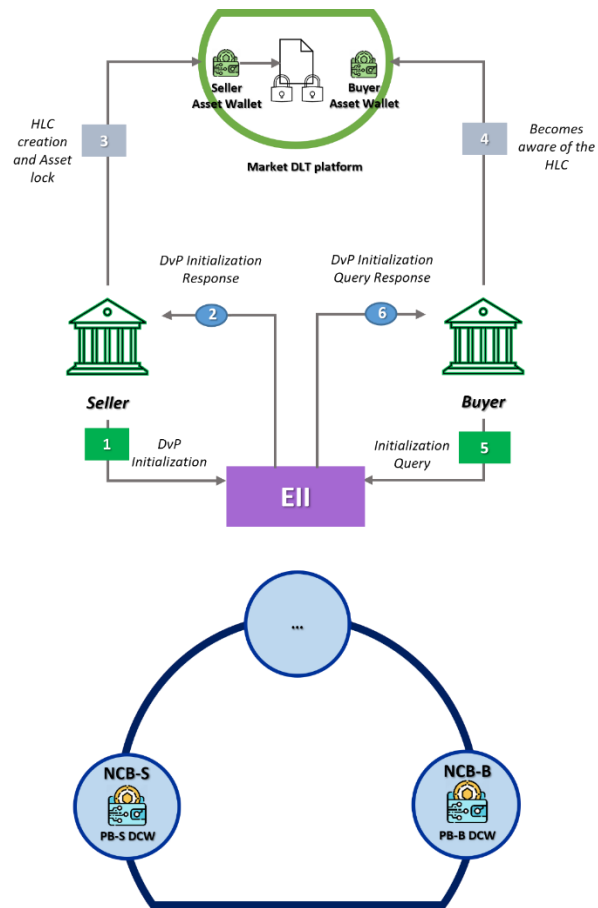


fig. iii. XvP initialisation high level workflow in A2A

Payment Settled

In the Payment Settled flow, the Buyer calls the Payment functionality, and the process ends with the related payment having the status settled. The Pilot solution returns to the Buyer the status of the Payment and the Execution Secret.

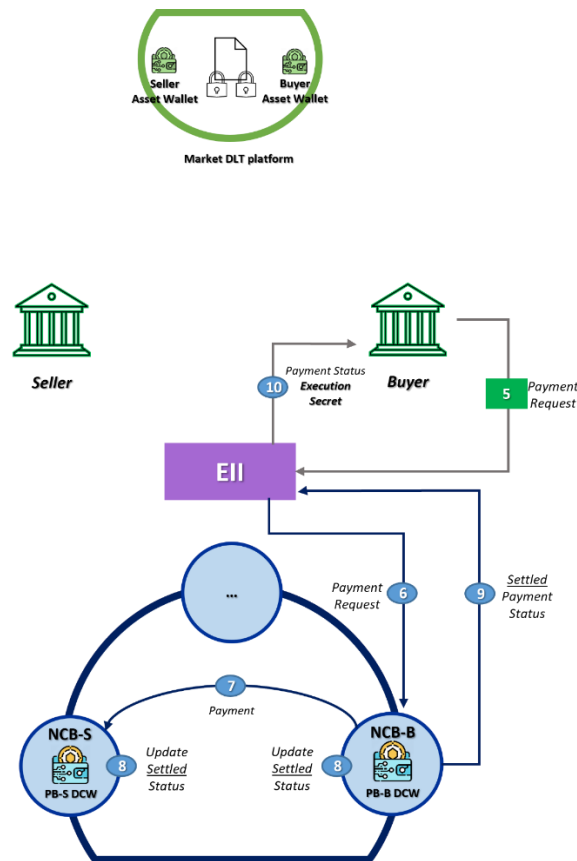


fig. iv. Payment settled situation high level workflow in A2A

The flow has three possible outcomes:

- 1) Cooperative execution: The Seller unlocks the asset in favour of the Buyer
- 2) Forced execution 1: The Buyer forces the unlock of the asset using Execution Key regardless of Seller's behaviour
- 3) Forced execution 2: The Seller does not unlock the asset in favour of the Buyer. The Buyer forces the unlock of the asset using Execution Key

Cooperative Execution

In the Cooperative Execution, the Seller checks the status of the Payment using the Get Key and Payment Info function which returns the Payment in status "Settled". So, the Seller becomes aware that the Buyer has performed the payment and consequently executes the transfer of the asset to the Buyer on Market DLT platform.

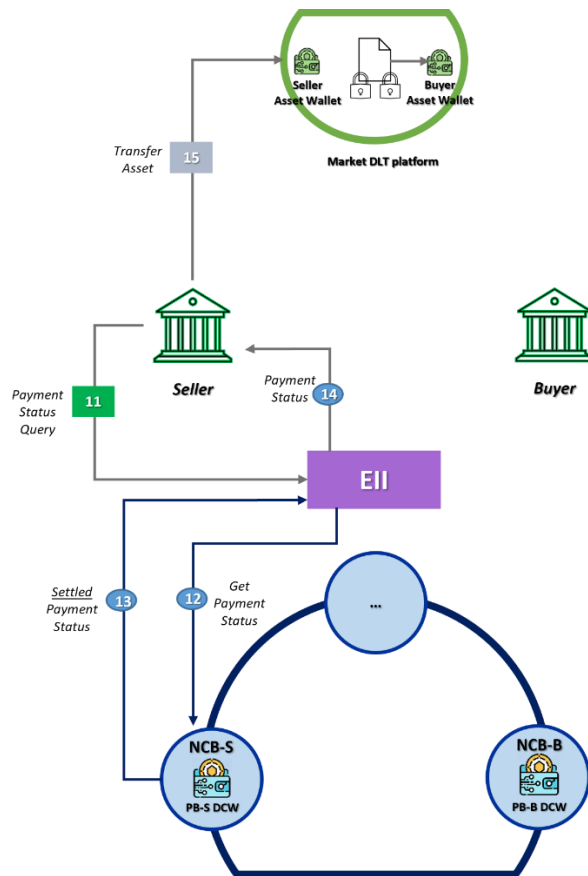


fig. v. Cooperative execution high level workflow in A2A

Forced Execution 1

In Forced Execution 1 scenario, the Buyer does not check the status of the asset on the Market DLT platform but directly forces the transfer of the asset to itself on the Market DLT platform using the Execution Secret it knows from the last step of the payment.

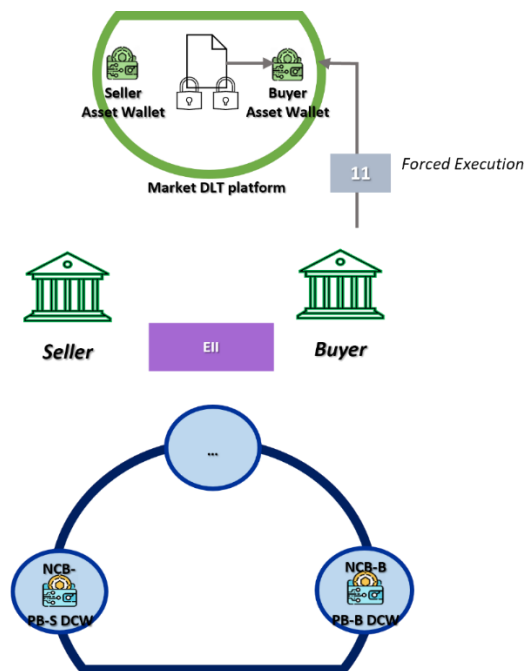


fig. vi. Forced execution 1 high level workflow in A2A

Forced Execution 2

In Forced Execution 2 scenario, the Buyer checks the status of the asset on the Market DLT platform and becomes aware that the Seller did not transfer the asset yet.

Then the Buyer directly forces the transfer of the asset to itself on the Market DLT platform using the Execution Secret it knows from the last step of the payment.

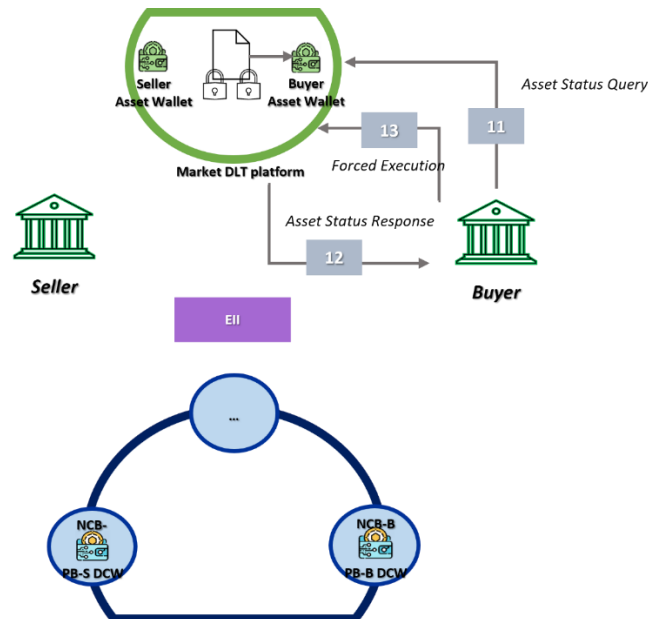


fig. vii. Forced execution 2 high level workflow in A2A

Payment Unsettled

In the Payment Unsettled flow, the Buyer calls the Payment functionality, and the process ends with the related payment having the status Unsettled meaning it is not executed on the Pilot solution (i.e. due to lack of cash).

This scenario also applies to Payment burned due to a DvP timeout.

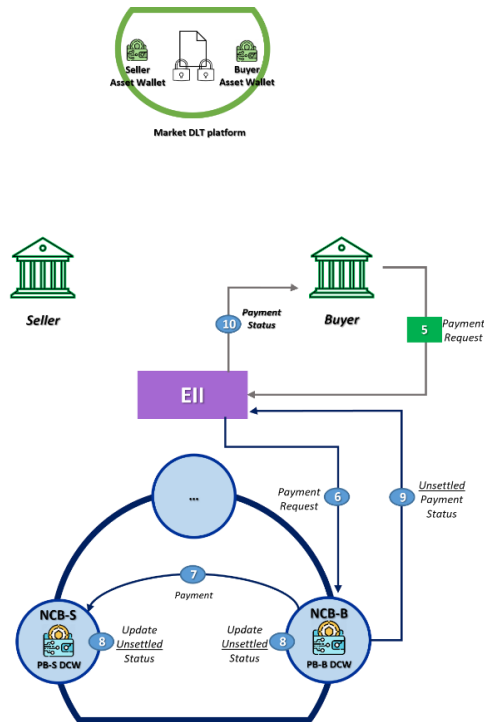


fig. viii. Unsettled situation high level workflow in A2A

The flow has two possible outcomes:

- 1) Cooperative cancellation: The Buyer unlocks the asset in favour of the Seller
- 2) Forced cancellation: The Seller forces the unlock of the asset using the Cancellation Key

Cooperative Cancellation

In the Cooperative Cancellation, the Buyer executes the transfer of the asset to the Seller on the Market DLT platform.

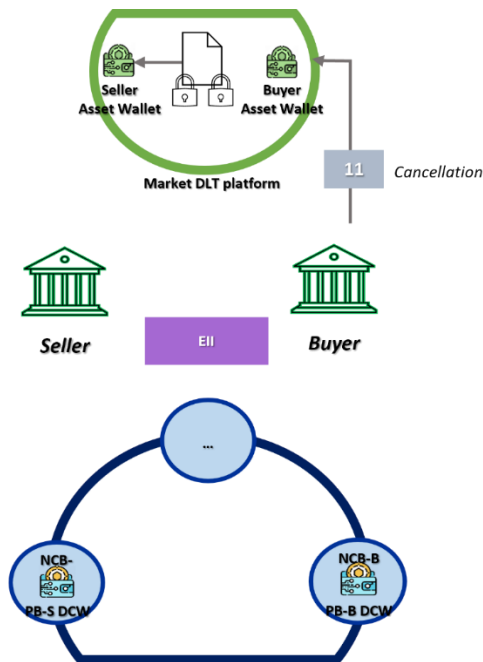


fig. ix. Cooperative cancellation high level workflow in A2A

Forced Cancellation

In the Forced Cancellation, the Seller retrieves the Cancellation Secret from the Pilot solution; then executes the transfer of the asset to itself on the Market DLT platform.

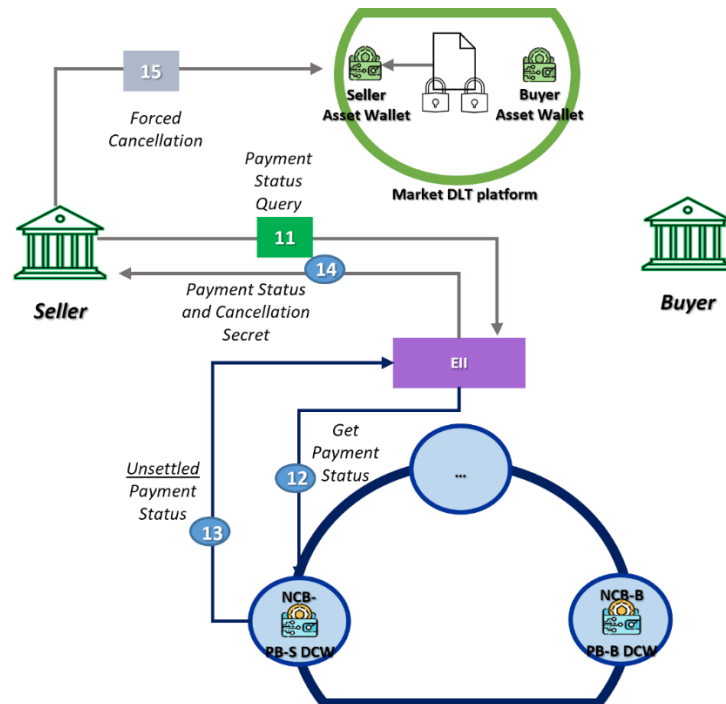


fig. x. Forced cancellation high level workflow in A2A

g. Key Management flows for Market DLT Operator

The Key Management flows change when a Market DLT Operator is involved.

Also in these flows, the perimeter of the Pilot solution stops with the delivery of the keys and payment status to the actor while all the subsequent steps follow under the domain of Market DLT platform.

So, these flows must be considered as examples showing how Delivery versus Payment & Payment versus Payment functionalities can serve different scenarios in Market DLT platform and they must not be considered in any case binding for any policy or implementation on Market DLT platform.

The following flows apply in the same way both to DvP and PvP and both to settlement in cash token and Direct Settlement mode. Anyway, for the sake of simplicity, in the examples we will use only DvP with settlement in cash token.



fig. xi. Market DLT Operator: Actor of the Market DLT platform

Having a third party acting on behalf of the Seller and/or the Buyer, reduces the number of interactions with the Pilot solution.

In the next sections we'll see how the flows change in the scenario of a Market DLT Operator acting on behalf of both the Seller and the Buyer.

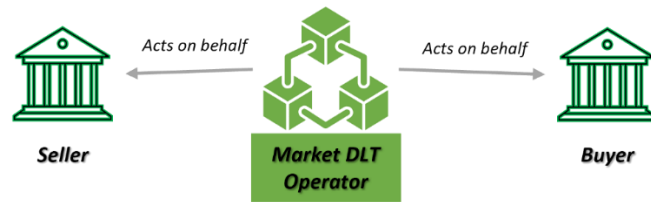


fig. xii. Market DLT Operator can act on behalf of Market Participants

Initialisation

In the Initialisation flow, the Market DLT Operator calls the Initialisation functionality on behalf of the Seller, but it does not call the Initialisation Query functionality on behalf of the Buyer because it does not need to verify DvP Initialisation data.

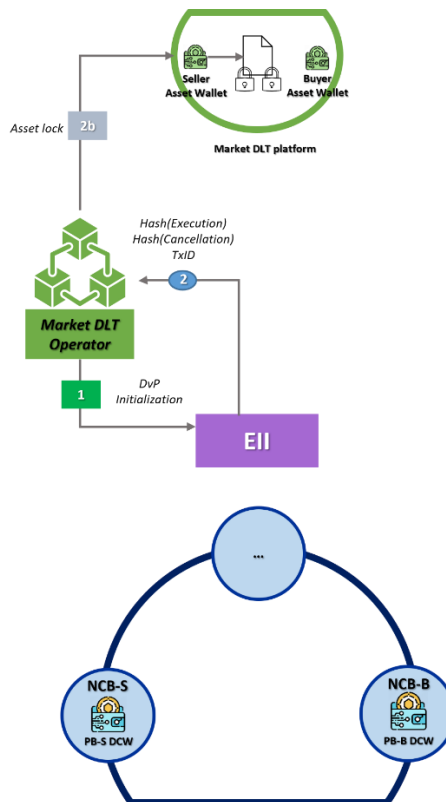


fig. xiii. XvP initialisation high level workflow in A2A – via Market DLT Operator

Payment Settled

In the Payment Settled flow, the Market DLT Operator calls the Payment feature on behalf of the Buyer and the process ends with the related payment having the status settled.

The Market DLT Operator, on behalf of the Buyer, receives the execution secret (and not the cancellation secret).

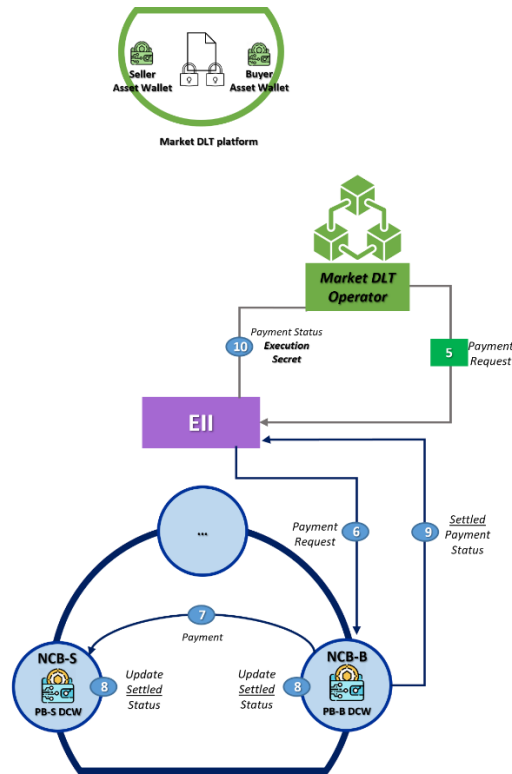


fig. xiv. Payment settled situation high level workflow in A2A – via Market DLT Operator

The flow has two possible outcomes:

- 1) Cooperative execution: Market DLT Operator executes the transfer of the asset to the Buyer
- 2) Forced execution: Market DLT Operator forces the transfer of the asset to the Buyer using Execution Key

Cooperative Execution

In the Cooperative Execution, the Market DLT Operator does not need to check the status of the Payment on behalf of the Seller because it already knows it from the last step of the Payment flow. So, it directly executes the transfer of the asset to the Buyer on Market DLT platform.

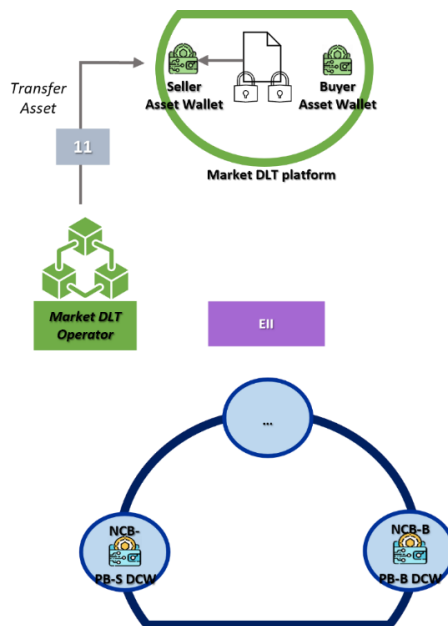


fig. xv. Cooperative execution high level workflow in A2A – via Market DLT Operator

Forced Execution

In Forced Execution scenario, Market DLT Operator does not need to check the status of the asset on the Market DLT platform because, directly operating on the Market DLT platform, it already knows it. So, it forces the transfer of the asset to the Buyer on the Market DLT platform using the Execution Secret it knows from the last step of the payment.

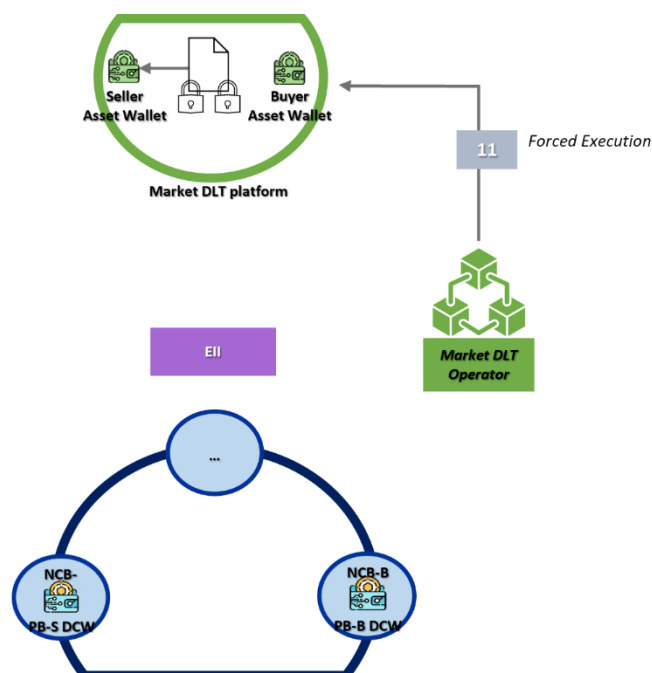


fig. xvi. Forced execution high level workflow in A2A – via Market DLT Operator

Payment Unsettled

In the Payment Unsettled flow, the Market DLT Operator calls the Payment feature on behalf of the Buyer and the related payment has status Unsettled meaning it is not executed on the Pilot solution (i.e. due to lack of cash).

This scenario also applies to Payment burned due to a DvP timeout.

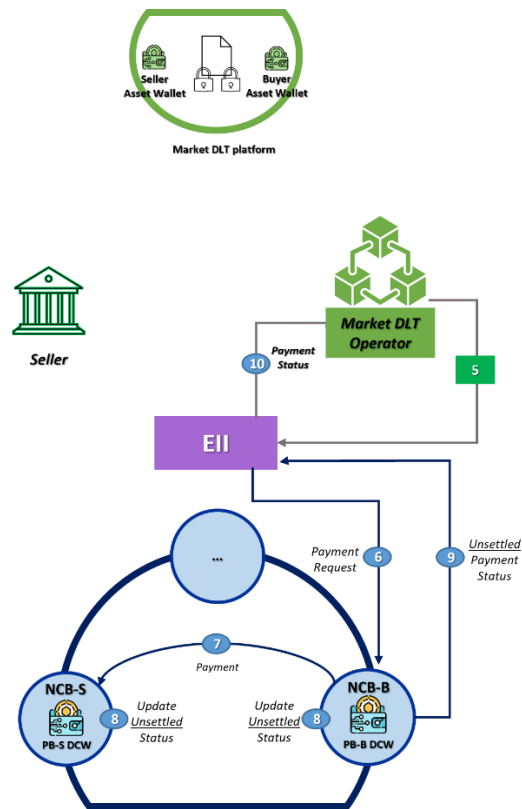


fig. xvii. Payment unsettled situation high level workflow in A2A – via Market DLT Operator

The flow has two possible outcomes:

- 1) Cooperative cancellation: Market DLT Operator executes the transfer of the asset in favour of the Seller
- 2) Forced cancellation: Market DLT Operator forces the transfer of the asset to the Seller using Cancellation Key

Cooperative Cancellation

In the Cooperative Cancellation, the Market DLT Operator executes the transfer of the asset to the Seller on the Market DLT platform. The flow is the same as for Market Participant.

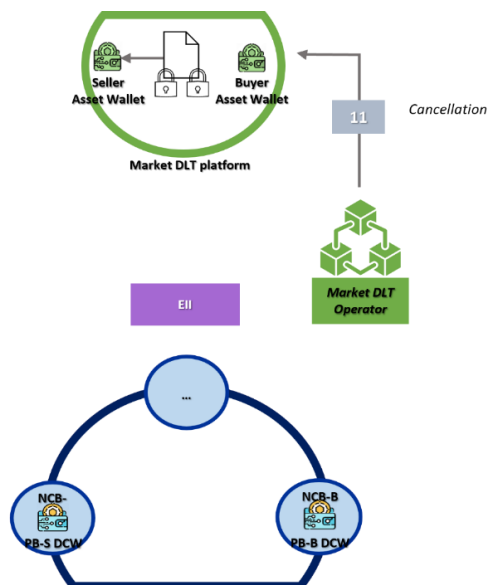


fig. xviii. Cooperative cancellation high level workflow in A2A – via Market DLT Operator

Forced Cancellation

In the Forced Cancellation, the Market DLT Operator retrieves the Cancellation Secret from the Pilot solution invoking Get Key and Payment Info functionality on behalf of the Seller; then executes the transfer of the asset to the Seller on Market DLT platform. The flow is the same as for Market Participant.

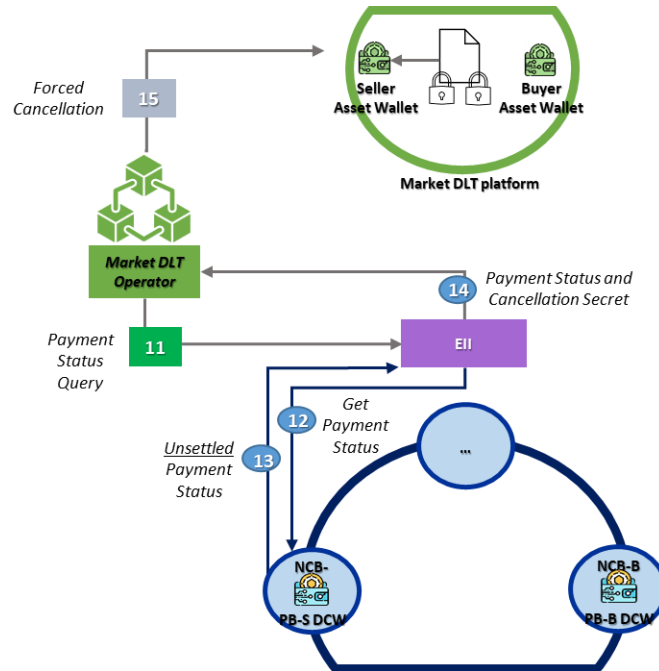


fig. xix. Forced Cancellation high level workflow in A2A mode via Market DLT Operator

h. Resilience measures in case of Actor unavailability

Delivery versus Payment and Payment versus Payment functionalities are exposed only in A2A mode and they are not available for NCB users. This means that if an Actor (Market Participant or Market DLT Operator) loses its A2A connection to Pilot solution, it is not possible for him nor to continue the DvP / PVP management flow via U2A nor to delegate its NCB to step in and act on its behalf.

Despite these limitations, there are several resilience measures in case of unavailability of the Actors involved in the DvP / PVP.

Native feature of the Hash Link interoperability mechanism for the following scenarios:

If the Buyer is unable to connect to the Pilot solution to send the payment for the cash leg of a DvP, the Hash Link allows the Seller to unlock the asset on the Market DLT platform using the cancellation key. This is done by calling the *Get Key and Payment Info* functionality once the timeout of the DvP has expired.

If the Seller is unable to connect to the Market DLT platform to unlock the asset in favour of the Buyer, the Hash Link protocol allows the Buyer to unlock the asset on Market DLT platform using the execution key by calling the *Get Key and Payment Info* functionality (same as Forced Execution scenario).

If the Buyer is unable to connect to the Market DLT platform to unlock the asset in favour of the Seller, the Hash Link protocol allows the Seller to unlock the asset on Market DLT platform using the cancellation key by calling the *Get Key and Payment Info* functionality (same as Forced Cancellation scenario).

Availability of Execution and Cancellation Key

In the scenario in which an Actor is unable to receive one of the keys due to temporary unavailability, it is possible for the Actor to call the *Get Key and Payment Info* functionality without constraints in terms of retention period or number of invocations.

Indeed, once the payment related to the cash leg of the DvP / PvP reaches a final status (“SETTLED”, “UNSETTLED”, “BURNED”), the Pilot solution always returns either cancellation or execution key to the entitled Actor.

For example:

- MPA, which is the Buyer for DvP1, sends a valid payment for the cash leg of DvP1
- The Pilot solution inserts a payment in status “SETTLED” for DvP1
- MPA loses its connectivity towards Pilot solution while receiving payment response. So, it does not get the execution key to unlock the assets on Market DLT platform.
- MPA resumes its connectivity towards Pilot solution.
- MPA retrieves execution key by calling Get Key and Payment Info functionality and unlocks the assets on Market DLT platform.

Market Participant steps in for unavailable Market DLT Operator

In the scenario in which the Market DLT Operator instructing a DvP / PvP on behalf of a Market Participant, if the Market DLT Operator becomes unavailable, it is possible for the Market Participant to step in.

For example:

- Market Participant MPA and Market DLT Operator MDLTOA are duly whitelisted and MDLTOA has an A2A user with an “external user” profile
- MPA is the Seller of DvP1
- MDLTOA initialises DvP1 acting on behalf of MPA
- The Buyer of DvP1 does not pay before the timeout of DvP1
- The Pilot solution inserts a payment in status “BURNED” for DvP1
- MDLTOA loses its connectivity toward the Pilot solution
- MPA steps in and retrieves cancellation key by calling Get Key and Payment Info functionality
- MPA unlocks the assets on Market DLT platform.

Market DLT Operator steps in for unavailable Market Participant

In the scenario in which the Market Participant instructing on its own a DvP / PvP, if the Market Participant becomes unavailable, it is possible for the Market DLT Operator acting on behalf of the Market Participant and involved in the DvP / PvP to step in.

For example:

- Market Participant MPA and Market DLT Operator MDLTOA are duly whitelisted
- MPA is the Buyer of DvP1
- MPA sends the payment for the cash leg of DvP1
- The Pilot solution inserts a payment in status “SETTLED” for DvP1
- MPA loses its connectivity towards the Pilot solution while receiving payment response. So, it does not get the execution key.
- MDLTOA steps in and retrieves execution key by calling Get Key and Payment Info functionality
- MDLTOA unlocks the assets on Market DLT platform.

Operational procedure

If previous measures don’t resolve the issue, it is always possible for Market Participants to ask for support to retrieve cancellation or execution key via operational procedure.

3.5.7. Unhappy Paths & Error codes

a. Funding Requests

Business Rule ID	Description	Error Text	HTTP Error Code
FR-DCW-001	Inactive or closed DCW	ERR-DCW-001 DCW is closed or ERR-DCW-011 DCW is inactive	400
FR-DCW-002	Inactive T2 account link	ERR-DCW-002 The T2 Account Link related to the considered DCW is inactive	400
FR-DCW-003	Blocked DCW (DCW Owner is blocked)	ERR-DCW-003 The owner of the DCW is blocked	400
FR-AMT-001	Wrong format of the amount	ERR-AMT-001 The format of the amount is not correct (more than 15 digits or too many decimals)	400
FR-AMT-002	Negative amount	ERR-AMT-002 Amount cannot be negative	400
FR-CUR-001	Wrong currency transmitted by the API	ERR-CUR-001 The transmitted currency is not Euro	400
FR-ID-001	ID already exists in trigger	ERR-DR-ID-001 ID in the funding request already exists in trigger	400
FR-SIGN-001	Invalid Signature	ERR-DR-SIGN-001 Invalid signature	400
FR-SIGN-002	Expired certificate	ERR-DR-SIGN-002 Certificate related to signature for funding purpose expired	400
TP-DCA-001	T2 DCA Account inactive or closed in T2	Error 'E007' Account number / Account BIC in indicated currency unknown in addressed settlement service	400
TP-DCA-002	T2 DCA blocked	If NCB denies the Payment: Error 'E023' At least one of the impacted parties or accounts is blocked. The earmarked cash transfer order has been disagreed by the respective CB/OT If message remains in the queue until reject time ²⁶ is reached Error 'E076' Reject Time reached	400
TP-AMT-001	Requested amount not available	Error 'E076' Reject Time reached	400
TP-PART-002	Blocked participant in T2	If NCB denies the Payment: Error 'E023' At least one of the impacted parties or accounts is blocked. The earmarked cash transfer order has been disagreed by the respective CB/OT If message remains in the queue until reject time is reached Error 'E076' Reject Time reached	400
TP-FR-ST-001	System time after reject time	Error E022 "Till time or reject time earlier than current system time"	400
TP-FR-MF-001	Admin triggers internal API to manually fail a Payment Instruction (Manual fail can only be performed before pacs.009 is sent)	Error MF Text (input by the admin performing the manual failure)	400
TP-FR-DD-001	Direct Debit Rejected – T2 RTGS rejects pacs.010 debit	If debit mandate does not exist: Error "E044": Instructing Agent not mandated to debit Instructed Agent	400

²⁶ Reject Time is a timeout parameter set in the T2 Interface to 5 minutes and corresponding to the maximum timeframe within which the Pilot solution expects a response from T2 RTGS

		If debit mandate is lower than payment instruction amount: Error “E045”: Direct debit amount exceeds the maximum amount per payment	
TP-FR-PI-001	Payment Instruction Submission – No execution start date – Value/Settlement date could not be calculated	“Execution start date could not be created”	500
TP-FR-PI-002	Pacs.002 not received after timeout of 15 minutes reached	Error MF	400
IR-DCW-001	Wallet is blocked due to Wallet Owner blocking between funding Request submission and issuance request creation	ERR-FR-ISS-PART-001 Funding request failed. Reason: Issuance failed due to DCW owner blocked	400
IR-PART-001	Wallet Owner is blocked between funding Request submission and issuance request creation	ERR-FR-ISS-001 Issuance failed. Reason: technical issue	400
IR-ISS-001	Issuance not executed by ECB Issuer (Technical issue on ECB node)		

b. Defunding Requests

Business Rule ID	Description	Error Text	HTTP Error Code
DR-DCW-001	Inactive or closed wallet	ERR-DCW-001 DCW is closed or ERR-DCW-011 DCW is inactive	400
DR-DCW-002	Inactive T2 account link	ERR-DCW-002 The T2 Account Link related to the considered DCW is inactive	400
DR-DCW-003	Blocked DCW (DCW Owner is blocked)	ERR-DCW-003 The owner of the DCW is blocked	400
DR-AMT-001	Wrong format of the amount	ERR-AMT-001 The format of the amount is not correct (more than 15 digits or too many decimals)	400
DR-AMT-002	Negative amount	ERR-AMT-002 Amount cannot be negative	400
DR-CUR-001	Wrong currency transmitted	ERR-CUR-001 The transmitted currency is not Euro	400
DR-ID-001	ID already exists in trigger	ERR-DR-ID-001 ID in the defunding request already exists in trigger	400
DR-SIGN-001	Invalid Signature	ERR-DR-SIGN-001 Invalid signature	400
DR-SIGN-002	Expired certificate	ERR-DR-SIGN-002 Certificate related to signature for defunding purpose expired	400
TP-DR-DCA-001	T2 RTGS account inactive or closed in T2	Error “E007” Account number / Account BIC in indicated currency unknown in addressed settlement service	400
TP-DR-DCA-002	T2 RTGS account blocked	If NCB denies the Payment: Error ‘E023’ At least one of the impacted parties or accounts is blocked. The earmarked cash transfer order has been disagreed by the respective CB/OT If message remains in the queue until reject time is reached Error ‘E076’ Reject Time reached	400
TP-DR-AMT-001	Requested amount not available	Error “E076” Reject Time reached	400
TP-DR-PART-002	Blocked participant in T2	If NCB denies the Payment: Error ‘E023’ At least one of the impacted parties or accounts is blocked. The earmarked cash transfer order has been disagreed by the respective CB/OT	400

		If message remains in the queue until reject time is reached Error 'E076' Reject Time reached	
TP-DR-ST-001	System time after reject time	Error 'E022' "Till time or reject time earlier than current system time"	400
TP-DR-MF-001	Admin triggers internal API to fail a Payment Instruction manually (Manual fail can only be performed before pacs.009 is sent)	Error MF Text (input by the admin performing the manual failure)	400
TP-DR-PI-001	Payment Instruction Submission: - No execution start date - Value/Settlement date could not be calculated	"Execution start date could not be created"	500
TP-DR-DD-001	Direct Debit Rejected – T2 RTGS rejects pacs.010 debit	If debit mandate does not exist: Error "E044": Instructing Agent not mandated to debit Instructed Agent If debit mandate is lower than payment instruction amount: Error "E045": Direct debit amount exceeds the maximum amount per payment	400

c. Direct T2 RTGS Payments

Business Rule ID	Description	Error Text	HTTP Error Code
TP-DCA-001	T2 DCA Account inactive or closed in T2	Error 'E007' Account number / Account BIC in indicated currency unknown in addressed settlement service	400
TP-DCA-002	T2 DCA blocked	If NCB denies the Payment: Error 'E023' At least one of the impacted parties or accounts is blocked. The earmarked cash transfer order has been disagreed by the respective CB/OT If message remains in the queue until reject time is reached Error 'E076' Reject Time reached	400
TP-AMT-001	Requested amount not available	Error 'E076' Reject Time reached	400
TP-PART-002	Blocked participant in T2	If NCB denies the Payment: Error 'E023' At least one of the impacted parties or accounts is blocked. The earmarked cash transfer order has been disagreed by the respective CB/OT If message remains in the queue until reject time is reached Error 'E076' Reject Time reached	400
TP-ST-001	System time after reject time	Error E022 "Till time or reject time earlier than current system time"	400
TP-MF-001	Admin triggers internal API to manually fail a Payment Instruction (Manual fail can only be performed before pacs.009 is sent)	Error MF Text (input by the admin performing the manual failure)	400
TP-DD-001	Direct Debit Rejected – T2 RTGS rejects pacs.010 debit	If debit mandate does not exist: Error "E044": Instructing Agent not mandated to debit Instructed Agent If debit mandate is lower than payment instruction amount:	400

		Error “E045”: Direct debit amount exceeds the maximum amount per payment	
TP-PI-001	Payment Instruction Submission – No execution start date – Value/Settlement date could not be calculated	“Execution start date could not be created”	500
TP-PI-002	Pacs.002 not received after timeout of 15 minutes reached	Error MF	400

d. Delivery versus Payment & Payment versus Payment

Business Rule ID	Description	Error Text	HTTP Error Code
HL-GER-001	Generic error occurred	Internal Error	500
HL-GER-002	Generic error occurred on ESY-DLT component	Internal Error	500
HL-GER-003	Generic error occurred while processing input data	Generic Error in input data	400
HL-GER-004	Generic error occurred during the communication with ESY-DLT component	Internal Error	500
HL-GER-005	Required operation is not available during current Business Day Window	System is not open for the required operation	500
HL-ATH-001	JWT of the HTTP Request is not valid if: - Not present - Expired - The signing certificate is revoked	JWT not valid	401
HL-ATH-002	Profile contained in the JWT is not valid	Profile not valid	401
HL-ATH-003	JWT of the HTTP Request is not valid for the ESY-DLT Component	JWT not valid	401
HL-PLD-001	One or more mandatory values are missing from input	Missing mandatory values \${list of missing values}	400
HL-PLD-002	One or more values are provided in the wrong format	Format not valid for input \${list of wrongly formatted values}	400
HL-PLD-003	Seller BIC value set in the payload is not valid for the requesting Market Participant	Seller BIC in the payload is not valid	400
HL-PLD-004	Buyer BIC value set in the payload is not valid for the requesting Market Participant	Buyer BIC in the payload is not valid	400
HL-PLD-005	Market DLT Operator value set in the payload is not valid for the requesting Market DLT Operator	Market DLT Operator in the payload is not valid	400
HL-PLD-006	There is no whitelisting in place between the requesting Seller Market Participant and the Market DLT Operator value set in the payload	The Market DLT Operator in the payload is not whitelisted for requesting Market Participant	400
HL-PLD-007	Inserted Payment data in the payload are not consistent with those related to the requested DvP/PvP	Payment fields in the payload are not valid \${list of wrong values}	400
HL-PLD-008	Inserted Payment data in the payload are not consistent with those related to the requested DvP/PvP for the ESY-DLT component	Payment fields in the payload are not valid \${list of wrong values}	400
HL-PLD-009	There is no whitelisting in place between the requesting Market DLT Operator and the Market Participant identified by the Seller BIC value set in the payload	The requesting Market DLT Operator is not whitelisted for Seller Market Participant	403
HL-DVP-001	The requested DvP/PvP is not found on the Pilot solution	DvP/PvP not found	404
HL-DVP-002	Requesting Market Participant is not authorised for the Buyer BIC value set in the DvP/PvP	Requesting actor is not authorised for Buyer BIC of the DvP/PvP	403
HL-DVP-003	There is no whitelisting in place between the requesting Market DLT Operator and the Market Participant identified by the Buyer BIC value set in the DvP/PvP	The requesting Market DLT Operator is not whitelisted for Buyer Market Participant	403
HL-DVP-004	Requesting Market Participant is not authorised for the Seller BIC value set in the DvP/PvP	Requesting actor is not authorised for Seller BIC of the DvP/PvP	403
HL-DVP-005	Requesting Market DLT Operator is not authorized since different Market DLT Operator	Requesting actor is not authorised for Seller BIC of the DvP/PvP	403

	has been configured for the Seller Market Participant for this DvP/PvP		
HL-DVP-007	DvP/PvP expired. Payment is not allowed	Payment not allowed for DvP/PvP	410
HL-DVP-008	ESY-DLT - Data Invalid for Get Payment	Internal Error	500
HL-PMT-001	It is not possible to pay the requested DvP/PvP	Payment not allowed for DvP/PvP	410
HL-PMT-005	Payment already existing for the requested DvP/PvP	Payment already exists for DvP/PvP	409
HL-PMT-006	Payment not found for the requested DvP/PvP	Payment not found for DvP/PvP	404
HL-PMT-008	There is no whitelisting in place between the requesting Market DLT Operator and the Market Participant identified by the Buyer BIC value set in the payload	Requesting actor is not authorised for Buyer BIC of the DvP/PvP	403

3.6 Queries/Monitoring & reporting

All queries and reports are available either through U2A or A2A for all Pilot users belonging to any entity in the solution (Market Participants, NCBs, ECB and SPs).

The reports files can be extracted at any time of the business day from SoD to EoD.

3.6.1. Queries

NCBs and Market Participants have access to the following list of queries:

Queries	U2A			A2A		
	Available in U2A	2-eyes	4-eyes	Available in A2A	1-step	2-steps
Dedicated Cash Wallet - Query reference data	✓	✓	No	✓	✓	No
Dedicated Cash Wallet - Query balance	✓	✓	No	✓	✓	No
Dedicated Cash Wallet - Query Statement of Wallet report	✓	✓	No	✓	✓	No
Participant (NCB and Market Participants) - Query reference data	✓	✓	No	✓	✓	No
User - Query	✓	✓	No	No	No	No
All transactions - Query	✓	✓	No	✓	✓	No
DvP and PvP transaction (cash token and direct RTGS) - Initialisation Query	No	No	No	✓	✓	No
DvP and PvP transaction (cash token and Direct RTGS) – Payment status and keys query	No	No	No	✓	✓	No
Business day status - Query	✓	✓	No	✓	✓	No
Market DLT Platform - Query	✓	✓	No	✓	✓	No
Market DLT Operator - Query reference data	✓	✓	No	✓	✓	No
Market DLT Operator whitelist - Query	✓	✓	No	✓	✓	No

Table S. Queries accessible to NCBs and Market Participants in U2A and A2A

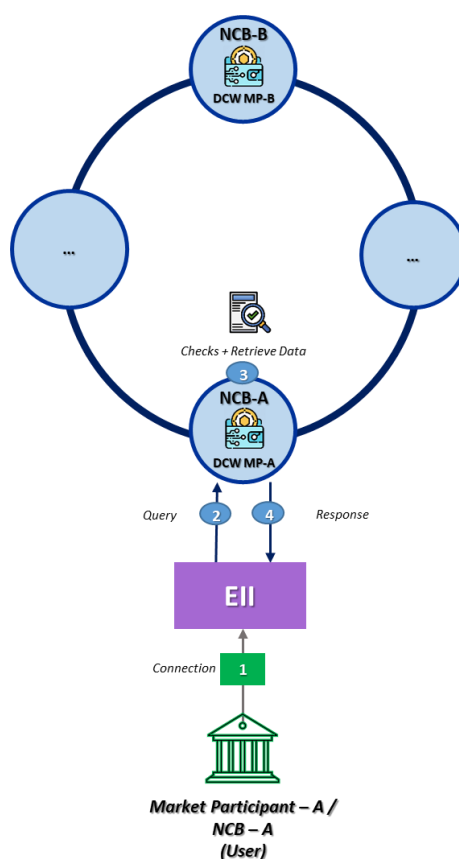


fig. xx. Queries high level workflow

Market DLT Operators have access to the following list of queries:

Queries	A2A		
	Available in A2A	1-step	2-steps
DvP and PvP transaction (cash token and direct RTGS) - Initialisation Query	✓	✓	No
DvP and PvP transaction (cash token and Direct RTGS) – Payment status and keys query	✓	✓	No
Current business window - Query	✓	✓	No
Market DLT Operator whitelist - Query	✓	✓	No

Table T. Queries accessible to Market DLT Operators in A2A

a. Dedicated Cash Wallets reference data query

The user can either extract all the details related to wallets in his scope or filter on specific fields and then export the file in order to have a customised report with the filters applied.

Filters can be applied on the following fields:

- Owner BIC
- Country Code
- Creation date (fixed date or range of dates)

The scope of the export depends on the user performing the action:

- If connected as a Market Participant, the file reports the details of their owned wallets.
- If connected as an NCB, the file reports the details of the wallets owned and managed (NCB's wallets if any and their Market Participants wallets).
- If connected as ECB, the file reports the details of all owned and managed wallets including Token Issuance wallet.
- If connected as SPs, the file reports the details of all the wallets in the DLT.

The export contains the following information:

Field	Description
Wallet ID	Unique Identifier of the Wallet
Owner	BIC of the DCW Owner
Manager	BIC of the DCW Manager
Currency	Currency of the DCW
Country Code	Country Code of the DCW Owner/Manager
Creation Date	Date of DCW creation
Last Modified	Timestamp of last modification performed on the DCW
Blocking status	If Participant is Blocked, DCW status is blocked, otherwise it's not blocked
Main Flag	Specifies the DCW to be used in case of DvP with Hash Link not mentioning the DCW ID
Valid from	Beginning of DCW Validity
Valid To	End of DCW Validity
PoA Given To	BIC of the Participant that can instruct on behalf on this DCW
PoA Validity Dates	Beginning and End of DCW Validity
PoA Max Amount	Maximum amount allowed per transaction for PoA Holder

Endpoints to retrieve Dedicated Cash Wallets reference data:

All wallets under the data scope of the requestor

Endpoint	/dlt/{ncb}/api/octopus/ams/wallets
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

A specific wallet per Wallet ID (=walias)

Endpoint	/dlt/{ncb}/api/octopus/ams/wallets/{walias}
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

b. Dedicated Cash Wallets balance query

The user can either extract all the balances related to wallets in his scope or filter on specific fields and then export the file in order to have a customised report with the filters applied.

Filters can be applied on the following fields:

- Owner BIC
- Country Code
- Creation date (fixed date or range of dates)

The scope of the export depends on the user performing the action:

- If connected as a Market Participant, the file reports the balances of their owned wallets.
- If connected as an NCB, the file reports the balances of the wallets owned and managed (NCB's wallets if any and their Market Participants wallets).
- If connected as ECB, the file reports the balances of all owned and managed wallets including Token Issuance wallet.
- If connected as SPs, the file reports the balances of all the wallets in the DLT.

The export contains the following information:

Field	Description
Wallet ID	Unique Identifier of the Wallet
Owner	BIC of the DCW Owner
Manager	BIC of the DCW Manager
Currency	Currency of the DCW
Available Balance	Amount available on the DCW
Country Code	Country Code of the DCW Owner/Manager
Creation Date	Date of DCW creation
Last Modified	Date and time of last update on the DCW
Blocking status	If Participant is Blocked, DCW status is blocked, otherwise it's not blocked
Main Flag	Specifies the DCW to be used in case of DvP with Hash Link not mentioning the DCW ID
Valid from	Beginning of DCW Validity
Valid To	End of DCW Validity
PoA Given To	BIC of the Participant that can instruct on behalf on this DCW
PoA Validity Dates	Beginning and End of DCW Validity
PoA Max Amount	Maximum amount allowed per transaction for PoA Holder

Endpoints to retrieve Dedicated Cash Wallets balance:

All wallets under the data scope of the requestor

Endpoint	/dlt/{ncb}/api/octopus/ams/wallets
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

A specific wallet per Wallet ID (=walias)

Endpoint	/dlt/{ncb}/api/octopus/ams/wallets/{walias}
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

c. Transactions for a Dedicated Cash Wallet query

The user can either extract all the transactions in cash tokens related to wallets in his scope or filter on specific fields and then export the file in order to have a customised report with the filters applied.

Filters can be applied on the following fields:

- Transaction type with possible values:
 - Transfer
 - Issuance
 - Redemption
 - Payment
 - Payment-XvP
 - All
- Transaction status with possible values:
 - Accepted
 - Settled
 - Unsettled
 - Rejected
 - Pending settlement
 - All

The scope of the export depends on the user performing the action:

- If connected as a Market Participant, the file reports the transactions related to their owned wallets.
- If connected as an NCB, the file reports the transactions related to the wallets owned and managed (NCB's wallets if any and their Market Participants wallets).
- If connected as ECB, the file reports the transactions related to all owned and managed wallets including Token Issuance wallet.
- If connected as SPs, the file reports the transactions related to all the wallets in the DLT.

The export contains the following information:

Field	Description
Transaction ID	Unique Identifier of the transaction
Request ID	Identifier of the related Funding or Defunding Request
Type	Transaction type: • Transfer • Issuance • Redemption • Payment • Payment-XvP
Amount	Amount of the transaction
Currency	Currency
Status	Status of the transaction: • Accepted • Settled • Unsettled • Rejected • Pending Settlement
Debited Wallet ID	ID of the wallet to be debited by the transaction
Debited Owner Entity ID	BIC of the debited DCW Owner
Debited Manager Entity ID	BIC of the debited DCW Manager
Credited Wallet ID	ID of the wallet to be credited by the transaction
Credited Owner Entity ID	BIC of the credited DCW Owner
Credited Manager Entity ID	BIC of the credited DCW Manager
Acceptance Timestamp	Date and time of transaction validation
Settlement Timestamp	Date and time of transaction settlement
Settlement Business Date	Business Date on which the transaction was settled
Creation Date	Date of transaction creation
Intended Settlement Date	Business Date on which the transaction is intended to be settled
Payment Reference	Reference related to Payments only
Instructing entity ID	ID of the participant that has initiated the transaction
Correlation ID	End to End ID used for Funding/defunding requests as well as DvPs and Payments that links with T2 RTGS Payments
Last Modified	Date and time of last update on the transaction

Endpoint to retrieve Dedicated Cash Wallets settled transactions

Endpoint	/dlit/{ncb}/api/octopus/ams/wallets/{walias}/transactions
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

d. Participants²⁷ reference data query

The list provides all the Participants registered on the Pilot Solution and the data query contains the following information:

Field	Description
Participant BIC	Pilot BIC of the participant
Long Name	Long name of the participant
Short Name	Short Name of the participant
Country Code	Country Code of related jurisdiction
Roles	List of roles attributed to the Participant
Valid From	Beginning of Participant validity
Valid To	End of Participant validity
Blocking status	Status of the Participant if blocked or not

Endpoint to retrieve Participants reference data

Endpoint	/dlt/{ncb}/api/octopus/grs/entities
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

e. User query

The list of users provides all the users created by the Admin user of the Participant

Field	Description
Username	The username used to register the user in Pilot solution reference data
Email	Email address of the user
Last Name	Last name of the user
First Name	First name of the user
Entity	Unique identifier, in the Pilot solution, of the participant to which the user belongs
Status	Status of the user: • Enabled • Disabled
Profiles	Profile(s) assigned to the user
Email verified	Yes, if the user has verified its email, No otherwise

f. Business day status query

The query can be executed at any time of the business day from SoD to EoD and consists in listing the following information:

If the initiator of the query is an external user:

²⁷ All Participants including NCBs, ECB, SPs and market participants

Field	Description
Window Name	Current Business Window Name: • Start of day • Open for All • End of day • Closed
Start Time	Start time of the current Business window
End Time	End time of the current Business window

Endpoint to retrieve Business Day status

Endpoint	/dlt/{ncb}/api/bridge/current-business-window
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants Market DLT Operators

If the initiator of the query is not an external user:

Field	Description
Window Name	Current Business Window: • Start of day • Open for All • End of day • Closed
Start Time	Start time of the current Business window
End Time	End time of the current Business window
Next Window Name	Next Business Window Name: • Start of day • Open for All • End of day • Closed

Endpoint to retrieve Business Day status

Endpoint	/dlt/{ncb}/api/octopus/grs/business-windows
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

g. Market DLT Platform query

The list of Market DLT platforms contains the following information:

Field	Description
Market DLT Platform ID	Unique Identifier of the Market DLT Platform
Market DLT Platform Name	Name of the Market DLT Platform
Owner Name	Name of the Market DLT Platform Owner
Valid From	Beginning of validity
Valid To	End of validity

Endpoint to export Market DLT Platform details

Endpoint	/dlt/{ncb}/api/octopus/grs/mdlt-platforms-details
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants Market DLT Operators

h. Market DLT Operators query

The list of Market DLT operators contains the following information:

Field	Description
Market DLT Platform ID	Unique Identifier of the Market DLT Platform
Market DLT Platform Name	Name of the Market DLT Platform
Owner Name	Name of the Market DLT Platform Owner
Market DLT Operator ID	Unique Identifier of the Market DLT Operator
Responsible NCB	Central Bank Code of the Market DLT Operator's responsible NCB
Blocking Status	Blocking status of the Market DLT Operator

Endpoint to retrieve Market DLT Operators list

Endpoint	/dlt/{ncb}/api/octopus/grs/mdlt-platforms/{networked}/mdlt-operators
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants Market DLT Operators

i. Market DLT Operator whitelist query

The list of NCB/ECB and Market Participants whitelisted for each market DLT Operator contains the following information:

Field	Description
Market DLT Operator ID	Unique Identifier of the Market DLT Operator
Responsible NCB	Central Bank Code of the Market DLT Operator's responsible NCB
Market DLT Platform ID	Unique Identifier of the Market DLT Platform
Participant BIC	Pilot BIC of the whitelisted participant for the Market DLT Operator
Manager BIC	Pilot BIC of the whitelisted Participant's NCB
Valid From	Beginning of validity
Valid To	End of validity

Endpoint to retrieve Market DLT Operators whitelist:

Endpoint	/dlt/{ncb}/api/octopus/grs/mdlt-operators-whitelists
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants Market DLT Operators

Market DLT Operator can query the Pilot solution (in A2A only) to verify if its whitelisted for a Market Participant or not

Endpoint	/dlt/{ncb}/api/bridge/whitelist/verify
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants Market DLT Operators

j. DvPs and PvPs related queries

As described in section [3.5.6](#), Market Participant and Market DLT Operators can query the initialised transaction and the payment status and keys. This can be done only in A2A, and details can be found in the API documentation.

3.6.2. Reporting and data extraction

a. Funding and Defunding extraction

Funding and Defunding Requests:

The user can either extract all the transactions related to wallets in his scope or filter on specific fields and then extract the file in order to have a customised report with the filters applied.

Filters can be applied on the following fields:

- Request's type with possible values:
 - Funding
 - Defunding
 - All
- Request's status with possible values:
 - Initialised
 - Pending approval
 - Pending review
 - Accepted
 - Payment submitted
 - Payment settled
 - Issuance created
 - Settled
 - Unsettled
 - Rejected
 - Cancelled
 - All

The extraction contains the following information:

Field	Description
Request ID	Identifier of the related Funding or Defunding Request
Amount	Amount of the request
Currency	Currency
Status	Status of the Request
T2 Account Reference	DCA number linked to the DCW to be funded/defunded
Debited Wallet ID	ID of the wallet to be debited by the issuance/redemption transaction
Debited Owner Entity ID	BIC of the debited DCW Owner
Debited Manager Entity ID	BIC of the debited DCW Manager
Credited Wallet ID	ID of the wallet to be credited by the issuance/redemption transaction
Credited Owner Entity ID	BIC of the credited DCW Owner
Credited Manager Entity ID	BIC of the credited DCW Manager
Creation Date	Date of request creation
Acceptance Timestamp	Date and time of request validation
Payment Submission Timestamp	Date and time of Payment instruction submission
Payment Settlement Timestamp	Date and time of Payment instruction settlement
Issuance Creation Timestamp	Date and time of issuance transaction creation for funding requests
Issuance Settlement Timestamp	Date and time of issuance transaction settlement for funding requests
Redemption Creation Timestamp	Date and time of redemption transaction creation for defunding requests
Redemption Settlement Timestamp	Date and time of redemption transaction settlement for defunding requests
Settlement Timestamp	Date and time of request settlement
Settlement Business Date	Business Date on which the request was settled
Initiator	Username of the request creator
Approver	Username of the request validator

T2 RTGS payments triggered within a funding or defunding request:

The user can either extract all the transactions related to wallets in his scope or filter on specific fields and then export the file to have a customised report with the filters applied.

Filters can be applied on the following fields:

- Payment's status with possible values:
 - Prepare
 - Initially Approved
 - Approved
 - Ready
 - Triggered
 - Cancelled
 - Failed

- Success

The extract contains the following information:

- Payment ID
- Correlation ID
- Payer Bank
- Receiver Bank
- Currency
- Amount
- Submission Timestamp
- Last Updated Timestamp
- Status

Field	Description
Payment ID	Identifier of the Payment
Correlation ID	End to End ID that links with T2 RTGS Payments
Status	Status of the payment
Currency	Currency
Amount	Amount of the Payment
Payer bank	BIC of the Payer Bank (Account BIC of Debited DCA)
Receiver Bank	BIC of the Payer Bank (Account BIC of Credited DCA)
Creation Date	Date of Payment instruction creation
Last Updated Timestamp	Date and Time of last update on Payment instruction
Initiator	Username of the request creator
Approver	Username of the request validator

Endpoint to extract funding and defunding requests:

Endpoint	/dlt/{ncb}/api/octopus/tms/funding-defunding-requests/extract
HTTP Method	POST
Actors	NCBs, ECB, SPs, Market Participants

Endpoint to extract Payments related to funding and defunding requests:

Endpoint	/dlt/{ncb}/api/octopus/tms/funding-defunding-requests/payment-instructions/extract
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

b. Cash tokens transactions extraction

The user can either extract all the transactions in cash tokens related to wallets in his scope or filter on specific fields and then extract the file in order to have a customised report with the filters applied.

Filters can be applied on the following fields:

- Transaction Type with possible values:

- Transfer
 - Payment
 - Payment-XvP
 - Redemption
 - Issuance
 - Reverse redemption
- Status with possible values:
 - Accepted
 - Settled
 - Unsettled
 - Rejected
 - Pending Settlement
 - Creation Date: to extract operations created on a specific Business Date or for a range of several days such as a week, a month ...
 - Settlement Business Date: to extract operations settled on a specific Business Date or for a range of several days such as a week, a month ...
 - Debited and Credited Wallet Owners and Managers Pilot BICs
 - T2 account reference
 - Country Code
 - Debited and Credited Wallets

The extraction contains the following information:

Field	Description
Operation ID	Unique Identifier of the transaction
Type	Transaction type • Transfer • Payment • Payment-XvP • Redemption • Issuance • Reverse redemption
Amount	Amount of the transaction
Currency	Currency
Status	Status of the transaction • Accepted • Settled • Unsettled • Rejected • Pending Settlement
Debited Wallet ID	ID of the wallet to be debited by the transaction
Debited Owner Entity ID	BIC of the debited DCW Owner
Debited Manager Entity ID	BIC of the debited DCW Manager
Debited Country Code	Country Code of the debited DCW
Debited T2 account reference	T2 account reference linked to the debited DCW
Credited Wallet ID	ID of the wallet to be credited by the transaction
Credited Owner Entity ID	BIC of the credited DCW Owner

Credited Manager Entity ID	BIC of the credited DCW Manager
Credited Country Code	Country Code of the credited DCW
Credited T2 account reference	T2 account reference linked to the credited DCW
Last Status Business Date	Business date of transaction last status
Last Status Timestamp	Date and time of transaction last status
Settlement Business Date	Business Date on which the transaction was settled

Endpoint to extract cash Token transactions:

Endpoint	/dlt/{ncb}/api/octopus/ims/transactions
HTTP Method	POST
Actors	NCBs ECB SPs

c. Cash Postings history extraction

The user can either extract all the cash postings (credits and debits related to settled cash tokens transactions) related to wallets in his scope or filter on specific fields and then extract the file in order to have a customised report with the filters applied.

Filters can be applied on the following fields:

Transaction Type with possible values:

- Transfer
- Redemption
- Issuance
- Reverse redemption
- Payment
- Payment-XvP
- DvP
- PFoD
- Settlement Business Date: to extract operations settled on a specific Business Date or for a range of several days such as a week, a month ...
- Wallet Owner and Manager Pilot BICs
- T2 account reference
- Country Code
- Wallet ID

The extraction contains the following information:

Field	Description
Transaction ID	Unique Identifier of the transaction
Movement Type	Specifies whether the cash posting result from a credit or a debit
Transaction Type	Type of the transaction that generated the Cash posting
Wallet ID	ID of the wallet for which the extract is generated
DCW Owner BIC	Owner BIC of the wallet for which the extract is generated
DCW Manager BIC	Manager BIC of the wallet for which the extract is generated
Country Code	Country Code of the wallet for which the extract is generated
T2 account reference	T2 account reference of the wallet for which the extract is generated
Counterpart DCW ID	ID of the wallet to be credited by the transaction
Counterpart DCW Owner BIC	BIC of the counterpart DCW Owner
Counterpart DCW Manager BIC	BIC of the counterpart DCW Manager
Settled Amount	Credited/Debited amount
Currency	Currency
Status	Status of the transaction: always Settled for cash postings
Settlement Timestamp	Date and time of transaction settlement
Intended Settlement Date	Business Date on which the transaction was intended to be settled
Payment reference	Reference related to Payments only
Correlation ID	End to End ID used for Funding/defunding requests as well as DvPs and Payments that links with T2 RTGS Payments

Endpoint to extract cash postings history:

Endpoint	/dlt/{ncb}/api/octopus/ams/wallets/transactions/extract
HTTP Method	GET
Actors	NCBs ECB SPs

d. Direct T2 RTGS Payments extraction

The user can either extract all the Direct T2 RTGS payments related to wallets in his scope or filter on specific fields and then extract the file in order to have a customised report with the filters applied.

Filters can be applied on the following fields:

- Type with possible values:
 - Direct RTGS Payment
 - Direct RTGS Payment-XvP
- Status with possible values:
 - Initialized

- Accepted
- Payment submitted
- Completed
- Failed
- Cancelled
- All

The extraction contains the following information:

Field	Description
Payment ID	Identifier of the Payment
Correlation ID	End to End ID that links with T2 RTGS Payments
Type	Specifies type of the Payment: • Direct RTGS Payment • Direct RTGS Payment-XvP
Status	Status of the payment
Currency	Currency
Amount	Amount of the Payment
Payer bank	BIC of the Payer Bank (Account BIC of Debited DCA)
Receiver Bank	BIC of the Payer Bank (Account BIC of Credited DCA)
Creation Date	Date of Payment instruction creation
Settlement Timestamp	Date and time of settlement
Last Updated Timestamp	Date and time of last update on Payment instruction
Instructing entity ID	ID of the participant that has initiated the transaction

Endpoint to extract Direct T2 RTGS Payments

Endpoint	/dlt/{ncb}/api/octopus/tms/direct-rtgs/payments/extract
HTTP Method	GET
Actors	NCBs ECB SPs Market Participants

3.7 A2A 1-step/2-steps illustration

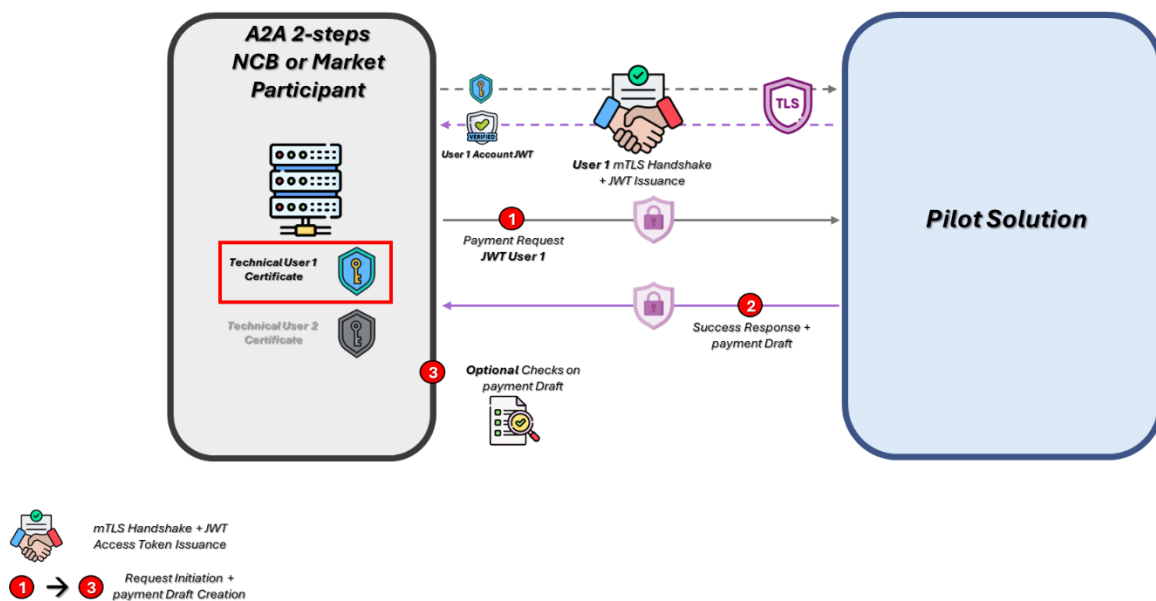
The aim of this section is to illustrate the process of the functionalities available in A2A 2-steps compared to 1-step.

Payment functionality is taken as an example; same principles apply for all the functionalities.

a. Payment in A2A 2-steps

NCB/ECB or Market Participants using this functionality will need two application users.

The first application user (technical user 1) will connect to the solution and send the payment request (1). The Participant's application will get an answer with the payment Draft created in the solution (2). Checks can be automated in the Participant's application allowing to ensure that the payment draft created with the data sent by the first user are correct (3); this is an optional step.



28

fig. xxi. First step of A2A 2-steps Payment request

A second connection will then need to be established with a second application user (technical user 2) and used to approve the previously created payment draft by calling the appropriate API (4).

Same principle applies for all functionalities available in A2A 2-steps (funding/defunding, reference data management, wallet-to-wallet cash token transfer)

²⁸ MTLS: Mutual Transport Layer Security is a protocol ensuring a secure communication with the solution. It is an extension of the TLS protocol in which **both the client and the server authenticate each other** using X.509 certificates.

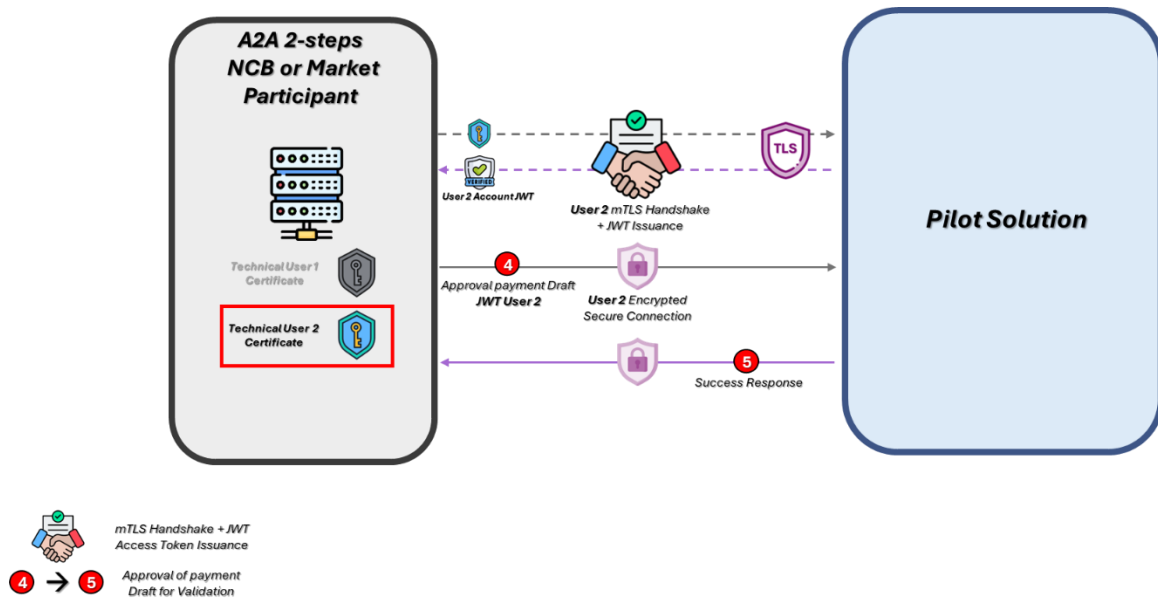


fig. xxii. Second step of A2A 2-steps Payment request

b. Payment in A2A 1-steps

NCB/ECB, Market Participants or Market DLT Operators using this functionality will need one application user (technical user).

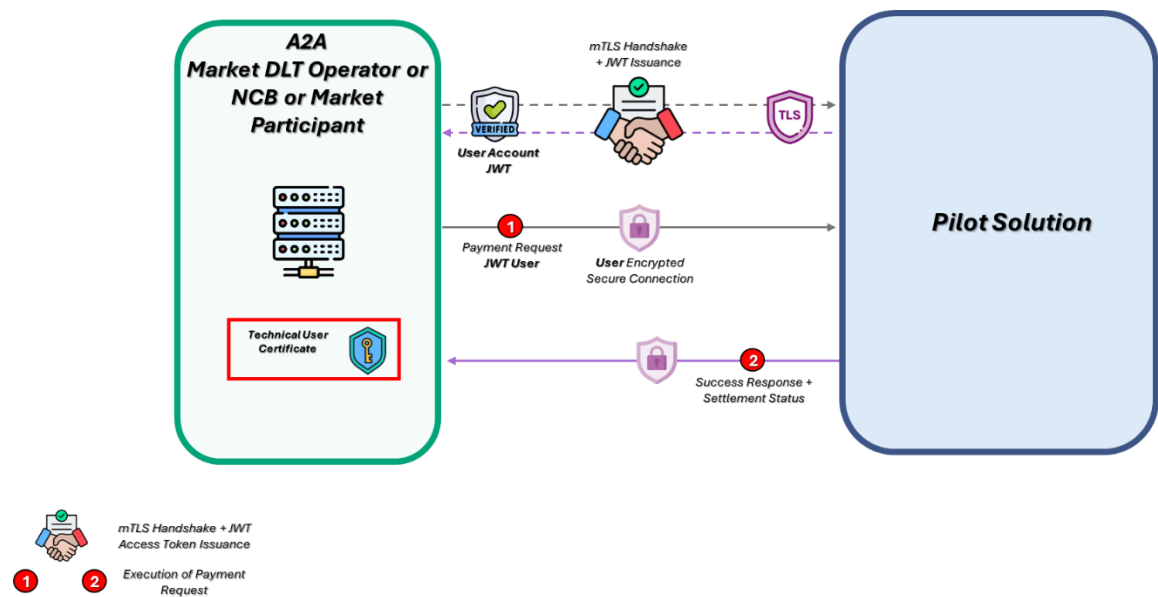


fig. xxiii. A2A 1-step Payment request

4. Operational Business Day

An “Operational Business Day” defines days where on-chain trading is allowed as well as the hours of a day where activities can take place. It is divided by a set of events with underlying technical processes that trigger business functions and allow or disallow inputting transactions. This notion differs from the traditional “Calendar Day”.

The operational business day is split in different Business windows and is managed by the SPs as Operators of the solution. Each business window has a start and end time and includes specific actions and processes as detailed below.

4.1 Start of Day (SoD) (8:00 - 9:00)

The SoD period starts after the successful completion of the previous Business Day.

It includes the change of the business date, which is followed by the processing of funding requests and cash token transfers.

All NCBs and Market Participants as well as SPs have access to the Pilot solution.

SPs and NCBs/ECB are allowed to create or update any reference data.

4.2 Open for all (9:00 - 16:00)

In the Open for all window, in addition to the funding requests and cash tokens transfers, all transactions can be processed including payments, DvPs, PVPs, PFOs and on-demand defunding requests.

SPs and NCBs are allowed to create or update any reference data.

All NCBs and Market Participants as well as SPs have access to the Pilot solution.

4.3 End of Day (EoD) (16:00 - 18:00)

EoD window starts with a 10 to 15-minute cool-down period, allowing the finalisation of on-going transactions in ESY DLT (until they reach a final status). At 16:15, all transactions pending 4-eyes validation are cancelled (Transaction cleanup process).

Just after transaction cleanup processed, the automated EoD Cash Token Sweep starts to empty all DCWs balances.

In case of technical issues or balances not redeemed respectively:

1. EoD Cash Token Sweep can be manually processed again by SPs,
2. On-demand Defunding Requests can be instructed by NCBs.

During this window:

- All actors have access to the Pilot solution but are not allowed to instruct any transaction nor create/update any reference data,
- They can access queries and reporting functionalities they are allowed to,
- NCBs can instruct on-demand defunding requests or redemptions in case of contingency.

Exceptionally in case of a major issue in the Pilot solution or in T2, the solution accepts Overnight Balances with the option to force the completion of EoD business window.

4.4 Closed (18:00 - BD+1 8:00)

The Pilot solution is only accessible to the SPs as Operators of the platform.

Below a graphical recap of the Business Day Schedule without overnight balances/contingency procedures which will be described in the Term Of Reference document (ToR).

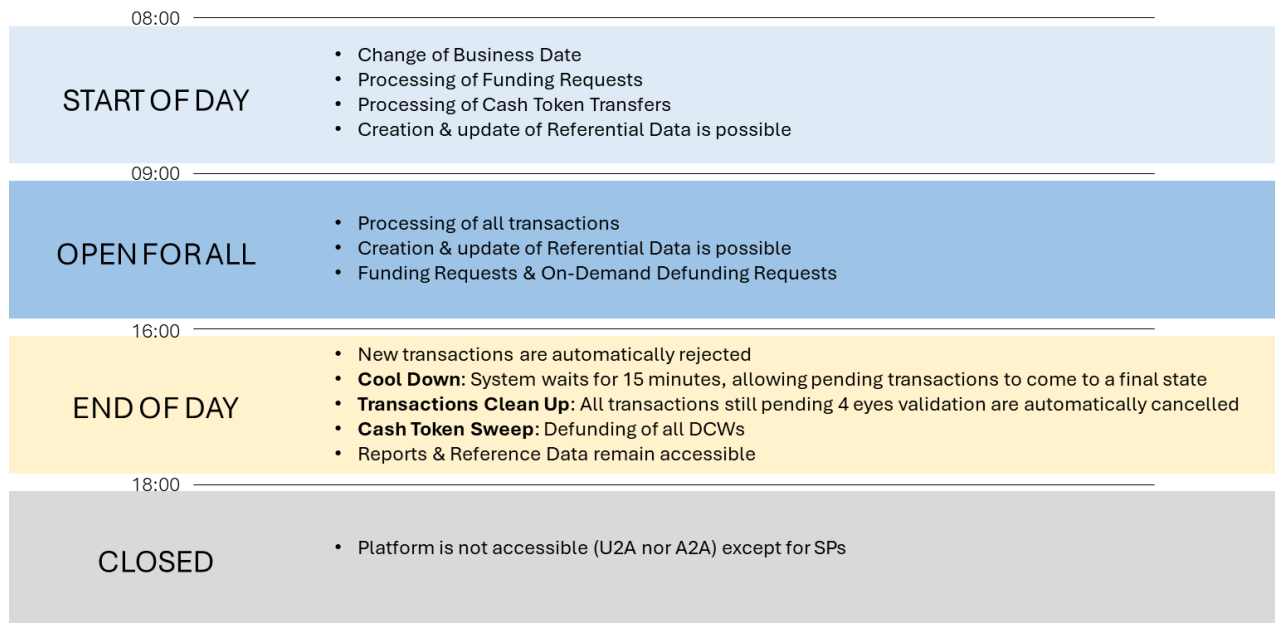


fig. xxiv. Business Day Schedule

5. High level technical description

5.1 High-level architecture of the solution

5.1.1. Extended Interoperability Interface

The Extended Interoperability Interface (EII) serves as the primary “gateway” between the Eurosystem DLT and the external ecosystem, ensuring secure, efficient, and standardised interactions. Its design addresses the need for a single, controlled access point to the internal infrastructure, while enabling advanced integration capabilities (i.e., atomic settlement) for complex financial use-cases. The key business features of EII are outlined below:

- **Single Access Point:** EII provides a unified entry point for all external interactions, supporting both User-to-Application (U2A) and Application-to-Application (A2A) interfaces. This centralised approach simplifies connectivity for Pontes Pilot Actors and reduces operational complexity by consolidating communication channels into a single, secure endpoint.
- **Authentication and Authorisation:** To safeguard the integrity of the entire system, EII implements robust mechanisms for authentication and authorisation of incoming requests. These controls ensure that only verified entities can initiate transactions or access system resources, thereby mitigating risks associated with unauthorised access and enhancing compliance with security standards.
- **Request Routing:** EII is responsible for routing incoming requests to the appropriate Eurosystem DLT node within the payment system.
- **Integration for Complex Use Cases:** Beyond basic connectivity, EII, leveraging the Interoperability Gateway (IGW) and the Hash-Link protocol implementation, enables seamless integration with external DLT-based systems (Market DLT platforms) to support advanced financial scenarios. Examples include Delivery versus Payment (DvP) and Payment versus Payment (PvP) workflows, which require coordinated and atomic execution across multiple platforms. By facilitating these interactions, EII plays a critical role in enabling interoperability and supporting multi-party settlement processes.

5.1.2. Eurosystem Distributed Ledger Technology

The **ESY DLT** is a private permissioned blockchain built on a Hyperledger Fabric-based infrastructure to ensure confidentiality, traceability, and auditability of transactions. Above the DLT layer, the application layer orchestrates business workflows through modular services. ESY DLT handles the cash leg of DLT-based transactions, settled either through cash tokens or directly in T2 RTGS.

Cash tokens act as digital proxies of euro central bank money (CeBM), representing a contractual claim for conversion into CeBM, without legal tender status.

The platform includes:

- Central Bank Nodes (one per NCB) – validate, sign, and store transactions and manage participant data.
- Operator Node – manages network parameters and NCB/ECB and Market Participants list.
- ECB Issuer Node – issues and redeems cash tokens.

It supports besides the Dedicated Cash Wallet for NCB/ECB or Market Participants' token balances and transactions also the ECB owned Token Issuance Wallet, mirroring the Token Issuance Account in T2, cleared to zero daily.

5.1.3. T2 Interface

The Single Automated T2 Interface provides a unified entry point to the TARGET Services through the Eurosystem Single Market Infrastructure Gateway (ESMIG). It serves as the technical orchestration layer between the T2 RTGS environment and the Eurosystem DLT platform. This ensures seamless interaction between traditional payment systems and DLT-based components.

The interface is responsible for managing and initiating all liquidity movements required for settlement, including both funding and defunding operations related to tokenised liquidity. By using existing TARGET Services infrastructure, it enables automated and secure processing of settlement instructions coming from the Market DLT platform.

Within this architecture, the T2 Interface acts as the operational bridge between T2 RTGS and the ESY DLT platform. It translates DLT-based payment instructions into standardised T2 RTGS settlement messages. The T2 Interface supports both tokenised settlements (via Funding/Defunding ESY DLT wallets) and direct T2 RTGS transactions, eliminating the need for additional effort in liquidity management.

The communication between the T2 Interface and the T2 RTGS system relies on ISO 20022 compliant pacs messages (pacs.010, pacs.009, pacs.002, pacs.004) to ensure interoperability with existing TARGET Services using the established financial messaging standards.

The message transfers are routed through a message broker system integrated with a Network Service Provider (NSP) as is standard to guarantee reliable and secure data exchange via ESMIG.

5.2 Hosting & Resilience

Topology

Pilot Pontes infrastructure is designed by following and supporting the service features and capabilities; to test, to take advantage of the DLT decentralised design. At the infrastructure level different regions are envisaged relying on the Service Provider's infrastructure.

Each region provides a common, shared interface layer based on a containerisation platform in order to allow deployment strategies to be unified and respond to the same Service Level Agreement (SLA)

Connectivity

The independent regions are fully segregated and isolated from each other. However, a distributed application is made of several internal components that communicate and interact one with another. The network used for communication flow is the public internet. Though, in order to keep the highest security standard, the regions are connected in a secure way with a site-to-site VPN solution.

The EII module allows connections from NCB/ECB, Market Participants and Market DLT Operators. It is secured by dedicated firewall rules, WAF and DDoS protection, IP whitelisting. Moreover, the access of the users is clearly disintermediated by an authentication and authorisation layer to block and avoid unauthorised connections. .

Environments

Pilot Pontes has an external testing environment and production environment, physically segregated.

Modules hosting

As better described in the previous chapters, the Pilot Pontes service rely on three main modules:

- EII: the single access point for any interaction with the ESY DLT.
- ESY DLT: the blockchain itself.

- T2 Interface: the component in charge of the interactions with T2 liquidity management.

While the ESY DLT is distributed across all regions, the other modules are hosted in a single region.

Resilience

Nodes are NCB-specific, because they self-contain private and transactional data of an NCB community; in this scenario, a distributed global ledger is still used, allowing each node to continue working independently from other nodes failures; however, the NCB node must be kept up and running to allow a given community to correctly operate with the DLT service. The high availability of ESY DLT relies on the resilience of the overlying infrastructure. In that sense, the target infrastructure will automatically recover the service, even in full availability zone (AZ) failure, by exploiting low-level data synchronisation, resource redundancy and highly available infrastructure technologies. The infrastructure, in each region is tolerant to a full AZ/site disaster; and thus, this is valid for any node and module depending on the infrastructure which they are hosted in.

5.3 Interoperability mechanism

The interoperability mechanism used in the Pilot solution to guarantee the all-or-none settlement of DvP and PVP transactions between a Seller and a Buyer in a “all-or-none” mode is the “API-based DvP/PvP with Hash Link Contract” referred to as “Hash Link Protocol”.

The Hash Link Protocol is built on Hash-Time Locked Contracts (HTLC) overcoming some limitations in terms of safety of DvP/PvP transactions.

The protocol main features are:

- DLT agnostic
- API based
- A trusted oracle to resolve potential disputes between the Buyer and the Seller (like orchestrated models)

The involved systems and actors are:

- Payment System which guarantees the delivery of correct status about the payment.
- Market DLT platforms, where tokenised assets are issued and transferred (It can be any DLT/blockchain that is capable of basic programmability properties).
- Trusted oracle which acts as
 - o Generator and the keeper of secrets used to manage the DvP/PvP transaction lifecycle.
 - o “Oracle of Time”, unambiguously verifying that a payment has (or has not) been executed within a pre-agreed timeframe.
- The Buyer, which agrees to buy a certain amount of asset for a certain amount of funds within a predefined timeframe.
- The Seller, which agrees to sell the afore mentioned asset for the agreed funds, within the agreed timeframe.

Within the Pilot Solution design, the EII acts as the trusted DvP/PvP oracle.

On the Market DLT platform side, the Hash Link protocol entails a Hash-Link Contract (HLC), which locks assets until specific conditions are satisfied. This HLC needs to be deployed on the Market DLT platform.

From a business standpoint, these conditions are:

- The Buyer agrees to return the assets to the Seller.
- The Seller agrees to transfer the assets to the Buyer.
- A dispute arises between Buyer and Seller, requiring oracle intervention.

The oracle's decision is revealed by the disclosure of secrets, which it exclusively controls: the execution key and the cancellation key:

- If the oracle rules are in favour of the Buyer (e.g., because payment has been confirmed), it reveals the execution key, allowing the Buyer to claim the assets from the HLC.
- If the oracle rules are in favour of the Seller (e.g., because payment was not completed within the agreed timeout), it reveals the cancellation key, enabling the Seller to reclaim the assets from the HLC.

The inclusion of a trusted oracle ensures transaction safety, even in cases where one of the parties (including the API gateway) is temporarily unavailable.

6. Appendix

6.1 API Documentation • Swagger

The API documentation containing the OpenAPI Document (swagger file) is provided in a separate document.

APPENDIX 2 • PONTES PILOT
User Handbook

6.2 Graphic User Interface • User Handbook

The User Handbook is provided in a separate document.

APPENDIX 3 • PONTES PILOT
Connectivity Guide

6.3 Preparing your integration • Connectivity Guide

6.3.1. Global Connectivity Overview

The Pilot solution relies on Public Internet for the communication with the Participants²⁹. The Pilot solution is reachable from a public Internet URL that route the requests to the EII module.

The EII is the common entry point for all interactions with the Pilot solution, it manages the access, authentication, authorisation and internal routing to the ESYDLT behind.

6.3.2. Authentication

The Authentication and Authorisation protocols of the Platform are based on the standard OAuth2.0 and OIDC.

6.3.3. A2A channel

The A2A channel between the User and the Pilot solution is based on a standard designed according to REST principles and exposed via the HTTPS protocol following authentication and authorisation principles.

The User authentication will be performed with an asymmetric-key approach.

The protocol Oauth2.0 is used to ensure the application is duly authorized to access Pilot solution resources.

To perform the first step of the protocol and obtain the JWT, the client application must send a POST request to the token endpoint (detailed in the API documentation) using:

- **Certificate** issued to the A2A user to set up the mTLS connection
- **grant_type** request parameter equal to “password”
- **client_id** request parameter depending on profiles granted to A2A user.

The value of the client_id parameter depends on the profiles granted to the A2A user:

User Profile	Client ID
Pilot Read-Only	esydlit-web-app
Pilot Read/Write	esydlit-web-app
Referential Read-Only	esydlit-web-app
Referential Read/Write	esydlit-web-app
External User	esydlit-backend-service

Table U. Client ID to be used for A2A authentication per user profile

For example, a request sent by A2A user identified by its private key KEY1 and certificate CERT1.pem on the node CBX and granted with profile the following profiles , will be:

²⁹ All Participants including NCBs, ECB, SPs, Market Participants and Market DLT Operators

Pilot Read-Only, Pilot Read/Write, Referential Read-Only or Referential Read/Write:

```
curl
--key KEY1.key --cert CERT1.pem
--request POST --url "$URL_PLACEHOLDER$/iam/realms/CBX/protocol/openid-connect/token"
--data "client_id= esydlt-web-app"
--data "grant_type=password"
```

External User:

```
curl
--key KEY1.key --cert CERT1.pem
--request POST --url "$URL_PLACEHOLDER$/iam/realms/CBX/protocol/openid-connect/token"
--data "client_id= esydlt-backend-service"
--data "client_secret= esydlt-backend-service"
--data "grant_type=password"
```

Client_id and client_secret are technical parameters provided with fixed values to be used by each A2A user according to its profile. On their own they are not sufficient for user authentication. To successfully authenticate, users must also use valid certificates and access the system from whitelisted IP addresses.

See section [6.3.6](#) for a sample response for the authentication request.

6.3.4. U2A channel

The U2A interface between the User and the Pilot solution is based on the standardised HTTPS protocol. The Pilot solution U2A will be reachable via the Internet from a specific URL for any possible services provided by the Pilot solution.

The User must use a valid individual certificate for U2A communication.

a. URLs

The generic Pilot URLs, reachable through the Public Internet, are:

TEST: utest.pontes-pilot.target-ssp.eu

PROD: pontes-pilot.target-ssp.eu

Node-specific URLs are available, composed as follows:

[https://\(utest.\)pontes-pilot.target-ssp.eu/dlt/<ORG>/webapp/](https://(utest.)pontes-pilot.target-ssp.eu/dlt/<ORG>/webapp/)

Here's the exhaustive ORG list:

ORG id	Country	Central bank name
ECB	European Union	European Central Bank
OENB	Austria	Oesterreichische Nationalbank
NBB	Belgium	Banque nationale de Belgique / Nationale Bank van België
BNB	Bulgaria	Bulgarian National Bank
CNB	Croatia	Hrvatska narodna banka
CBC	Cyprus	Central Bank of Cyprus
EP	Estonia	Eesti Pank
BOF	Finland	Suomen Pankki / Finlands Bank
BDF	France	Banque de France
BBK	Germany	Deutsche Bundesbank
BOG	Greece	Bank of Greece
CBI	Ireland	Central Bank of Ireland
BDI	Italy	Banca d'Italia
LB	Latvia	Latvijas Banka
BOL	Lithuania	Lietuvos bankas
BCL	Luxembourg	Banque centrale du Luxembourg
CBM	Malta	Central Bank of Malta
DNB	Netherlands	De Nederlandsche Bank
BDP	Portugal	Banco de Portugal
NBS	Slovakia	Národná banka Slovenska
BSI	Slovenia	Banka Slovenije
BDE	Spain	Banco de España

b. Qualified configurations

The recommended (“qualified”) technical configuration is outlined below:

OS: Windows 11

Browser: Microsoft Edge 140 // Chrome 140 // Firefox 142

6.3.5. Lifecycle of Digital Certificates³⁰

Digital certificates for the authentication of U2A and A2A users are issued by the TARGET Service Desk upon reception of a signed request by the end user or another authorised member of their organisation.

The workflow relies on a dedicated Certification Authority, held and operated by Deutsche Bundesbank as Service Provider.

The following process covers the lifecycle of digital certificates for A2A and U2A users irrespective of the type of participation (i.e. National Central Bank, Market participant, or Market DLT Operator). The communication flow is as follows:

- Central Banks send their requests directly to the TARGET Service Desk via their National Service Desk (NSD) via ESCBMail. The TARGET Service Desk delivers the certificates back to the NSD by storing them in a password-protected format on a dedicated repository, segregated per CB.
- Market Participants send their issuance/revocation/renewal requests to the responsible Central Bank. The National Service Desk validates and routes the requests to the TARGET Service

³⁰ The workflow described in this chapter is a manual procedure relying on the delivery of different pieces of information via email or upload on a secure repository. An automated procedure directly available to the end users and relying on a dedicated UI is currently under study.

Desk via ESCBMail, who delivers the certificates back to the NSD by storing them in a password-protected format on a dedicated repository, segregated per CB.

- **Market DLT Operators'** users must be registered separately, obtaining separate certificates for each node they expect to operate on. Issuances/revocations/renewal requests are collected by each Market Participant that the MDLO services and delivered from the MP to the relevant Central Bank NSD. The NSD validates and routes the requests to the TARGET Service Desk, who delivers the certificates back to the NSD by storing them in a password-protected format on a dedicated repository, segregated per CB.

a. Issuance and activation of a certificate

This workflow envisages the following steps:

- 1) CSR generation

The user generates an ECDSA key pair through a dedicated GUI and stores the private key and the Certificate Signing Request (CSR) locally³¹. This task can also be executed manually via open-source tools such as OpenSSL (see detailed user instructions in [CSR generation – user guide](#)).
- 2) Issuance form submission
 - a. The user fills up and signs the relevant form, which contains the necessary information both for the registration of the user in the Pilot reference data and for the issuance of the certificate³². Mandatory info for the issuance of the certificate is:
 - i. Username (matching the Common Name included in the CSR). The username must be unique in the system; therefore, it must follow a pre-agreed naming convention³³.
 - ii. Email address of the user
 - iii. Actor/Participant organisation ID (BIC if available, or LEI). This information has to match the BIC identifying the MP/NCB/ECB (Pilot Participant BIC), or the BIC/LEI defined as Market DLT Operator ID in the Pilot solution
 - iv. Privilege (2-Eyes or 4-Eyes)³⁴
 - v. Profiles to be assigned (Administrator, Referential Read/Write, Referential Read/Only, Pilot Read/Write, Pilot Read/Only, External User)³⁵
 - vi. Requestor's IP address(es)³⁶
 - b. The user submits the form and the previously generated CSR to the responsible Central Bank via a secure delivery method.
 - c. The responsible Central Bank delivers the signed form and CSR to the TARGET Service Desk, certifying that the request comes from a legitimate source and that the identity of the users has been verified, and requesting the generation of the certificate. Delivery can take place via email or upload on a secure repository.
- 3) Registration in the Pilot reference data

³¹ The private key must be stored and handled safely. See *0 Security recommendations*.

³² The form will contain some additional fields that are only useful for the identification of human users, but do not need to be shared with the TARGET Service Desk as they do not have a bearing on the certificate issuance process.

³³ For admin users, alphanumeric string composed of: "A" + country code + DLT BIC + free text. For any other users: alphanumeric string composed of: "P" + country code + DLT BIC/ LEI + free text.

³⁴ Human users (operating via U2A) can only request 4E privileges; application users (operating via A2A) can only request 2E privileges.

³⁵ See section [2.4](#) for details on user profiles

³⁶ For IP whitelisting. Can also be subnets / IP ranges.

With the information included in the form, the responsible entity³⁷ registers the user in the Pilot reference data. For further details see section 4.1.1 of the User Handbook – Annex 6.2 of the Service Description Document).

4) Issuance of the certificate

The TARGET Service Desk submits the request to the Certification Authority. The Common Name will match the user name included in the form and registered in the Pilot reference data.

5) Certificate distribution and activation

- a. The certificate is distributed by the TARGET Service Desk to the responsible NSD via email/ link to a secure repository. The NSD delivers it to the original requestor.
- b. The certificate is active immediately.

6) Certificate installation

The user installs the certificate on the operating system. Connectivity tests can afterwards be executed.

The certificate installation can be performed in several ways. As an example, you can produce the PFX file from the PEM file received at step 5 and the private key generated at the step 1, with the following command:

```
openssl pkcs12 -export -out <NEW_CERT>.pfx -inkey privatekey.key -in certificate.pem
```

Once the PFX file is generated, you can double-click on the file and the installation wizard is launched.

After the completion of the installation process you can confirm the certificate is actually installed by going to the Cert Manager applet in windows (WIN+R → “certmgr.msc” → Personal → Certificates) and look for the newly added certificate;

the fourth column of the table shows the scope of the certificate, it should be “all”. It may be possible that the certificate can’t be used/installed due some organization security policies configured in Windows. In this case, the local IT support should be asked to overcome the issue.

b. Revocation of a certificate

This workflow envisages the following steps:

1) Revocation form submission

The end user fills up and signs the relevant form to request the revocation of their certificate and, optionally, the deletion of the relevant user in the Pilot reference data³⁸.

2) The Service Desk acknowledges and revokes the certificate. The relevant serial number is added to the Certificate Revocation List (CRL). Revocation is irreversible.

- a. Since the CRL is downloaded by the EII once a day, the maximum delay between the revocation of a certificate and it being effective in the Pilot system is 24 hours.
- b. Deletion of the user is effective immediately and is recommended in order to disable access for the unauthorized user as soon as possible.

³⁷ The NCB Admin user creates the Market Participant Admin User only, and then the Market Participant Admin User creates all other Market participant users. The NCB Admin user creates the MDO users in their node.

³⁸ In case the revocation is requested due to a security breach (i.e. the certificate was compromised, but the user is still operational), the revocation/renewal of the certificate can be executed without deleting the IAM user.

c. Renewal of a certificate

Certificates are active for 24 months from the issuance date.

When a certificate approaches its expiry date, but ahead of its expiration, the end user triggers the “Issuance and activation of a certificate” workflow, thus requesting the generation of a new certificate.

Since the user configuration in the Pilot reference data does not expire with the certificate, step 3 of the workflow will be skipped.

d. CSR generation – user guide

As a part of onboarding, users generate a private key and a CSR (certificate signing request) with their username included in the certificate. The private key must be stored safely and not shared with other users. Secure storage and handling of the private key is a responsibility of the user and of their organization.

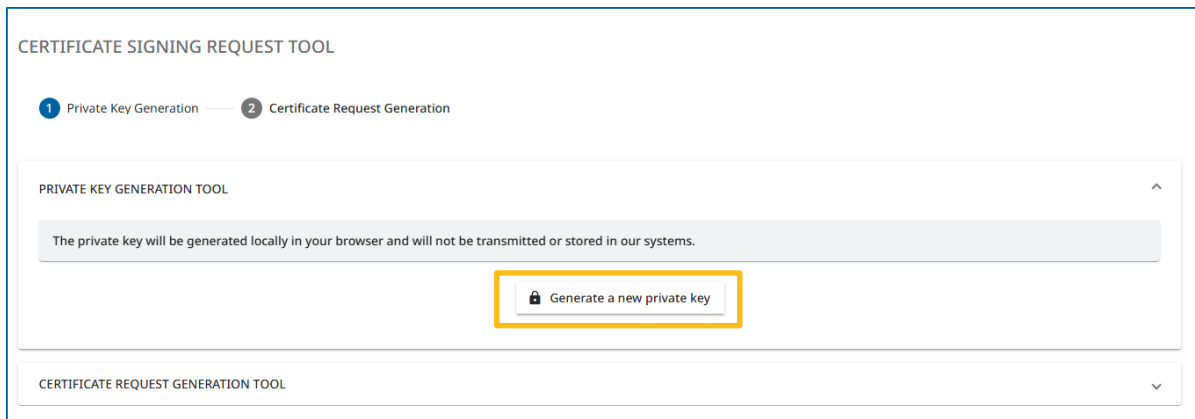
Automated generation

This section describes how to generate a private key and create the CSR request from it by means of a dedicated GUI, publicly available on the Internet and not restricted behind IP whitelisting. No information is stored by the application that generates the files, the key pair generation happens on client’s side using the local browser.

The URL of the page will be provided as part of the registration forms.

The screenshots below are purely informational; the final look-and-feel of the page may differ slightly.

- 1) Access the “Certificate Management” screen;
- 2) Click on “Generate a new private key”.



The screenshot displays a web interface titled "CERTIFICATE SIGNING REQUEST TOOL". At the top, there are two steps in a progress bar: "1 Private Key Generation" (active) and "2 Certificate Request Generation". Below this, the "PRIVATE KEY GENERATION TOOL" section is visible. It contains a message: "The private key will be generated locally in your browser and will not be transmitted or stored in our systems." Below the message is a button labeled "Generate a new private key" with a lock icon, which is highlighted by a yellow rectangle. At the bottom, the "CERTIFICATE REQUEST GENERATION TOOL" section is partially visible.

- 3) Download and save the generated private key.

CERTIFICATE SIGNING REQUEST TOOL

1 Private Key Generation

2 Certificate Request Generation

PRIVATE KEY GENERATION TOOL

The private key will be generated locally in your browser and will not be transmitted or stored in our systems.

Generate a new private key

user-pkey.pem



.....



Please ensure to download and securely store your private key.

It is crucial to maintain its confidentiality and not share it with anyone under any circumstances.

CERTIFICATE REQUEST GENERATION TOOL

This step is important!

The downloaded and saved private key is needed for the NRO process of any Funding/Defunding, Payment and DvP/PvP with Direct T2 RTGS settlement requests.

- 4) Expand the section “Certificate Request Generation Tool”.

CERTIFICATE SIGNING REQUEST TOOL

1 Private Key Generation

2 Certificate Request Generation

PRIVATE KEY GENERATION TOOL

The private key will be generated locally in your browser and will not be transmitted or stored in our systems.

Generate a new private key

user-pkey.pem



.....



Please ensure to download and securely store your private key.

It is crucial to maintain its confidentiality and not share it with anyone under any circumstances.

CERTIFICATE REQUEST GENERATION TOOL

- 5) Enter the mandatory fields and push the button “Generate a Certificate Signing Request”.

117 • 124

7) Provide the CSR (not the private key) to the responsible entity as per the Certificate Issuance and Activation workflow.

As an alternate option, the CSR can be generated manually by the user without relying on the dedicated GUI, but rather using open sources tools such as OpenSSL.

N.B.: The platform uses ECDSA (Elliptic Curve Digital Signature Algorithm) with the P-256 curve. Only ECDSA-based CSRs are accepted. RSA-based CSRs will be rejected.

1) Generate your ECDSA private key

Launch the following command:

```
openssl ecparam -genkey -name prime256v1 -noout -out my_private_key.pem39
```

This creates a 256-bit private key on the P-256 curve (also known as prime256v1 or secp256r1).

2) Generate the CSR from your private key

```
openssl req -new -key my_private_key.pem -out my_csr.pem -sha25640
```

When prompted, fill in the fields as follows:

Common Name (CN): Your exact ESY DLT IAM username. This is the most critical field.

Organisation (O): Your organisation name as registered in ESY DLT.

Country (C): Your two-letter country code (DE, FR, IT, ES, etc).

Other fields: May be left blank or filled as appropriate.

Additional mandatory fields such as the Organization ID and Privilege will be added by the TARGET Service Desk at the moment of the generation of the certificate (based on the information included in the registration form).

3) Verify your CSR before submitting

```
openssl req -in my_csr.pem -noout -text
```

Check the output carefully:

- the Subject line must show the correct CN matching the reference data user name;
- the Public Key Algorithm must show id-ecPublicKey with a 256-bit key. If you see rsaEncryption anywhere, you have generated the wrong key type and must start over from Step 1.

Security recommendations

The private key must remain exclusively under the user or their IT administrator's control at all times. It should not be shared with other users nor included with the CSR submission, emailed, uploaded to shared storage, or stored in an unprotected location.

Only the CSR file is submitted. The CSR contains the user's public key and identity information. The private key stays within the organization.

If the private key is compromised at any point, the certificate must be revoked immediately, and the issuance process repeated. To protect the key, it is recommended to:

- a) **Restrict file permissions** so that only the intended user account can read it.

³⁹ The private key can be generated in either .key or .pem format

⁴⁰ The CSR can be generated in either .csr or .pem format.

- b) **Use a Hardware Security Module (HSM) or secure keystore.** If the user's organisation provides an HSM or secure keystore, the key should be stored there.
- c) **Consider encrypting the key file with a passphrase** for additional protection.

6.3.6. Connectivity testing

Connectivity with the Pontes Pilot Platform will be tested by each connected actor, irrespective of the connection mode. Separate success conditions are envisaged for A2A and U2A users.

The tests at establishing that the dialogue between the user and the Pilot solution is successfully established, verifying both "pure" connectivity and the correct user setup in the IAM. The tests are considered successful upon materialisation of the outcome(s) described below.

a. A2A test

Execution:

The participant authenticates using a digital certificate (previously stored on the systems initiating the connection) via mutual TLS; once recognised by the IAM, a signed JWT is issued and used for all subsequent A2A interactions.

Outcome:

If a JWT is successfully issued, the test demonstrates that:

- network connectivity is in place;
- the certificate is valid;
- the user is correctly registered in the reference data;
- the requestor is inside the platform security perimeter.

If the request cannot reach the authentication endpoint or the connection is rejected before the TLS handshake completes, this indicates that:

- the source IP address is not authorised (not whitelisted); or
- a general network or perimeter connectivity issue is present.

If the TLS connection is successfully established but the JWT is not issued, this indicates that:

- technical connectivity is still proven;
- the certificate is valid at transport level;
- the user is not yet registered or fully configured in the Pilot reference data.

Evidence of a successful test: A2A response containing the JWT.

Sample A2A request and response:

curl

```
--key <KEY.key> --cert <CERT.pem>
```

```
--request POST --url "https://(utest.)pontes-pilot.target-ssp.eu/iam/realms/<ORG>/protocol/openid-connect/token"
```

```
--data "client_id=esydlit-web-app"
```

```
--data "grant_type=password"
```

```
{
  "access_token":"eyJhbGciOiJSUz....",
  "expires_in":300,
  "refresh_expires_in":0,
  "token_type":"Bearer",
  "id_token":"eyJhbGciOiJSUz....",
  "not-before-policy":0,
  "scope":"..."
}
```

The JWT token is contained in the `access_token` key.

b. U2A test

Execution:

Users access the platform through a web interface using their digital certificate, previously installed on their browser.

Successful login confirms that the user is correctly registered in the Pilot reference data and is authorised to access the platform, thereby proving correct connectivity and perimeter access.

Outcome:

If the user can successfully log in and access the GUI, the test demonstrates that:

- network connectivity is in place;
- the request originates from an authorised (whitelisted) network;
- the certificate is valid;
- the user is correctly registered in the Pilot reference data;
- the user is inside the platform security perimeter.

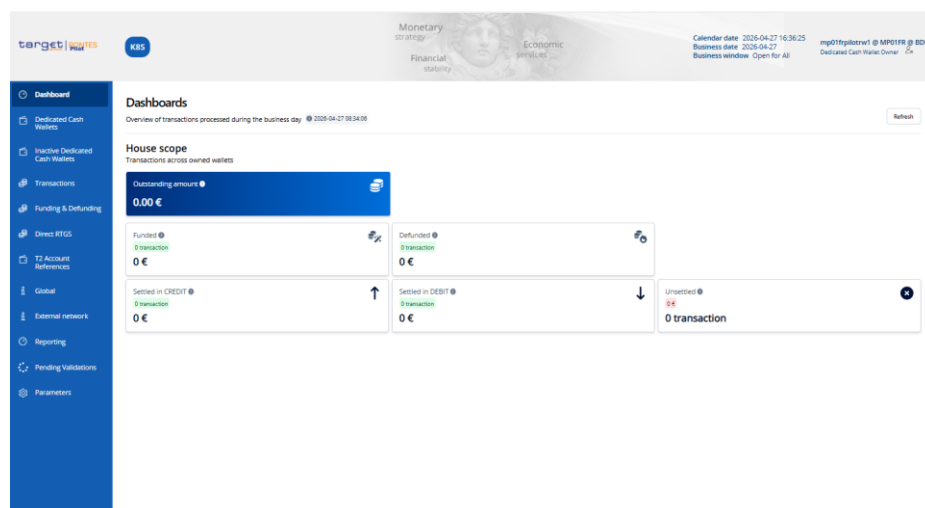
If the user cannot reach the login page or the connection is rejected before certificate selection, this indicates that:

- the source IP address is not authorised (not whitelisted); or
- a general network or perimeter connectivity issue is present.

If the login fails after certificate selection, this indicates that:

- the user is not yet registered or fully configured in the Pilot reference data;
- or the certificate is not correctly associated with the user (e.g. Common Name $\neq$ Pilot reference data username).

Evidence of a successful test: Sample screen for a Market Participant after a successful U2A authentication



Dedicated URLs can be used for troubleshooting by triggering specific connectivity checks:

IP Whitelisting: [https://\(utest.\)pontes-pilot.target-ssp.eu/check/ip](https://(utest.)pontes-pilot.target-ssp.eu/check/ip)

MTLS: [https://\(utest.\)pontes-pilot.target-ssp.eu/check/mtls](https://(utest.)pontes-pilot.target-ssp.eu/check/mtls)

DLT node healthcheck: [https://\(utest.\)pontes-pilot.target-ssp.eu/dlt/<ORG>/api/octopus/health](https://(utest.)pontes-pilot.target-ssp.eu/dlt/<ORG>/api/octopus/health)

6.4 Onboarding processes

6.4.1. Market DLT Operator

Data to gather from the institution to be onboarded:

DATA	FORMAT	DESCRIPTION
Market DLT Operator ID	Alphanumeric (∞)	ID of the Market DLT Operator (recommended BIC or LEI)
Market DLT platform Name	Alphanumeric (∞)	Name of the Market DLT platform
Market DLT platform Owner Name	Alphanumeric (∞)	Name of the Market DLT platform's Owner
Email address	Email address	Email address associated to the Market DLT Operator User A2A

Actors

NCB responsible of the eligibility of the Market DLT Operator

NCB User 1: NCB User granted with Profile Referential Read/Write on Pilot

NCB User 2: NCB User granted with Profile Administrator on Pilot

Service Provider Operator User with Profile Operational Read/Write on Pilot

Pilot Configuration – Step 1

NCB User 1 checks if Market DLT platform identified by **Market DLT Platform Name** is already registered on the Pilot solution

If Market DLT platform is not present:

NCB User 1 asks to Service Provider Operator User to register the Market DLT platform on the Pilot solution

Service Provider Operator User creates a new Market DLT platform with:

- Market DLT platform ID
- **Market DLT platform Name**
- **Market DLT platform Owner Name**
- Valid From set with proper value
- Valid To set with proper value

Pilot Configuration – Step 2

NCB User 1 checks if Market DLT Operator identified by **Market DLT Operator ID** is already registered on the Pilot solution as participant of Market DLT platform identified by **Market DLT Platform Name**.

If Market DLT Operator is not present:

NCB User 1 adds a new Market DLT Operator to the Market DLT platform identified by **Market DLT Platform Name** with:

- **Market DLT Operator ID**

The actual ID assigned to the newly created Market DLT Operator by the Pilot solution is equal to the Market DLT Operator ID prefixed with the Market DLT platform ID in the format:

[Market DLT platform ID] Market DLT Operator ID

Pilot Configuration – Step 3

NCB User 2 creates a new user allowed to instruct via A2A with:

- User Profile «External User»
- **Email address**
- Username set with proper value
- Related entity to which the user belongs (**Market DLT Operator ID**)

Once the user profile has been created, an email will be sent to the user to enable the configuration of credentials, the updating of personal information and the acceptance of the GDPR clause.

6.4.2. Market Participant

Data to gather from the institution to onboard

DATA	FORMAT	DESCRIPTON
Party BIC	BIC11	Party BIC used in TARGET Services
Pilot Participant BIC	BIC11	Authorised Account User BIC of the T2 RTGS account used as unique identifier in Pilot
Short Name	Alphanumeric (∞)	Short Name of the Market Participant
Long Name	Alphanumeric (∞)	Long Name of the Market Participant
T2 Account Number	Alphanumeric (34)	T2 RTGS DCA to be used for funding/defunding and Direct Settlement functionalities
Settlement in cash token	Flag [Y;N]	If Y, enables the settlement in cash token with the creation of a Dedicated Cash Wallet. If N, only Direct Settlement is enabled.
DvP / PvP	Flag [Y;N]	If Y, the Market Participant must provide also data of at least one Market DLT Operator.
Self-Instructing DvP / PvP	Flag [Y;N]	Mandatory if flag DvP / PvP = Y If Y, the Market Participant wants to instruct DvP / PvP by itself, A2A User Email is mandatory

Administrator Email	Email address	Email address of the Market Participant's Administrator User
A2A User Email	Email address	Email address of the A2A User for instructing DvP / PvP/Payments/PFoDs

Actors

NCB responsible of the eligibility of the Market Participant

NCB User 1: NCB User granted with privileges to operate on CRDM

NCB User 2: NCB User granted with Profile Referential Read/Write on Pilot

NCB User 3: NCB User granted with Profile Administrator on Pilot

CRDM configuration

NCB User 1 checks on CRDM that:

- **Party BIC** is the same Party BIC defined in CRDM for the Market Participant
- **T2 Account Number** is an existing and valid T2 RTGS DCA owned by the Market Participant
- An AAU with type Direct is in place between **T2 Account Number** and **Pilot Participant BIC**

NCB User 1 sets up on CRDM a Direct Debit Mandate allowing the debiting of **T2 Account Number** and the crediting of Pilot Party as Payee

Pilot Configuration – Step 1

NCB User 2 creates a new Market Participant with:

- **Pilot Participant BIC** as unique identifier
- **Short Name**
- **Long Name**
- Valid From set with proper value
- Valid To set with proper value
- Role equal to «DCW Owner»

Pilot Configuration – Step 2

NCB User 2 creates a new T2 Account Reference with:

- **T2 Account Number** as T2 Account Reference
- **Pilot Participant BIC** as linked entity to the T2 account reference

This step is only needed for retro compatibility with legacy software components stemming from Exploratory Works. T2 Account number is a needed information, but it is not actually used by the Pilot solution.

Pilot Configuration – Step 3

NCB User 3 creates a new Administrator User for the Market Participant with:

- User Profile «Administrator»
- **Email address**
- Username set with proper value

Once the user profile has been created, an email will be sent to the user to enable configuration of credentials, the updating of personal information and the acceptance of the terms and conditions.

Pilot Configuration – Step 4

If the flag Settlement in Cash Token is equal to 'Y':

NCB User 2 creates a new Dedicated Cash Wallet with:

- Dedicated Cash Wallet ID set with proper value as a unique identifier of the wallet
- Short Name set with proper value referring to the entity owning the wallet (Market Participant short name)
- T2 Account Reference equal to **T2 Account Number**
- Valid From set with proper value
- Valid To set with proper value
- Main Flag equal to «Y»

Pilot Configuration – Step 5

If the flag DvP / PvP is equal to 'Y':

Market Participant must provide also data of at least one Market DLT Operator as described in 6.4.1.

NCB User 2 checks that Market DLT Operator ID provided by Market Participant is registered in the platform.

If not, NCB responsible of the eligibility of the Market Participant must deal with the NCB responsible of the eligibility of the Market DLT Operator, asking for the onboarding of Market DLT Operator following the process described in 6.4.1.

If the Market DLT Operator is registered, NCB User 2 configures the Whitelist between the Market Participant and the Market DLT Operator:

- Select the list of Whitelisted NCB/ECB and Market Participants of the Market DLT Operator
- Add the Market Participant to the whitelist with following information:
 - o **Pilot Participant BIC**
 - o Whitelist Valid From set with proper value
 - o Whitelist Valid To set with proper value

Pilot Configuration – Step 6

If the flag Self Instructing DvP / PvP is equal to 'Y' the Market Participant will send instructions directly to the Pilot Solution.

NCB User 3 creates a new user to instruct via A2A:

- User Profile «External User»
- **Email address**
- Username set with proper value
- Related entity to which the user belongs (**Pilot Participant BIC**)

Once the user has been created, an email will be sent to the user to enable configuration of credentials, the updating of personal information and the acceptance of the GDPR clause .

As noted in Section [2.3](#), after Step 5 configuration, if the Market DLT Operator has a user set up for the Market Participant's jurisdiction, the operator is not technically prevented from sending instructions on behalf of the Market Participant, even when the Self-Instructing DvP/PvP flag is set to "Y".

