BIS Innovation Hub

Project Meridian FX

Exploring synchronised settlement in FX

Bank of England · Banque de France Eurosystème · Banca d'Italia Eurosistema · Deutsche Bundesbank Eurosystem · European Central Bank Eurosystem

Contents

Glossary

batch settlement – the settlement of groups of payments, transfer instructions or other obligations together at one or more discrete, often pre-specified times during the processing day.

central bank money – a liability of a central bank, in this case in deposits held at the central bank, which can be used for settlement purposes.

cross-border payments – a payment in which the financial institutions of the payer and the payee are located in different jurisdictions.

delivery versus payment (DvP) – a securities settlement mechanism that links a securities transfer and a funds transfer in such a way as to ensure that delivery occurs if and only if the corresponding payment occurs.

digital assets – a digital representation in value that can be used for payment or investment purposes or to access a good or service.

distributed ledger technology (DLT) – protocols and supporting infrastructure that allow computers in different locations to propose and validate transactions and update records in a synchronised way across a network.

foreign exchange – the exchange of one country's currency for another country's currency.

interoperability – is the technical or legal compatibility that enables a system or mechanism to be used with other systems or mechanisms. It allows participants in different systems to conduct clear and settled payments or financial transactions across systems without participating in multiple systems.

netting – offsetting obligations between or among participants in the netting arrangement, thereby reducing the number and value of payments or deliveries needed to settle a set of transactions.

payment versus payment (PvP) – a settlement mechanism that ensures that the final transfer of a payment in one currency occurs if and only if the final transfer of a payment in another currency or currencies occurs.

List of abbreviations and acronyms

API	application programming interface
BIS	Bank for International Settlements
CAMT	cash management message
CHAPS	Clearing House Automated Payment System
DLT	distributed ledger technology
DL3S	Distributed Ledger for Securities Settlement System
DvP	delivery versus payment
EUR	Euro
EVM	Ethereum Virtual Machine
EXP	hypothetical currency of exchange partner
FR	fund reservation
FX	foreign exchange
GBP	British pound sterling
G20	Group of Twenty
ISO	International Organization for Standardization
LSA	liquidity saving approach
PACS	Payments Clearing and Settlement
PvP	payment versus payment
REST	Representational State Transfer
RTGS	Real-Time Gross Settlement
SO	synchronisation operator
TIPS	TARGET Instant Payment System
TMS	Transaction Management Simulator
TX	transaction
UI	user interface

Executive Summary

Project Meridian FX (Project MFX) is a joint project between the Bank for International Settlements (BIS) Innovation Hub (London and Eurosystem Centres), Bank of England, Bank of France, Bank of Italy, Deutsche Bundesbank and the European Central Bank. It explores how operators of wholesale payment infrastructures can enable interoperability with new technologies, such as distributed ledger technology (DLT), with a focus on foreign exchange (FX) transactions.

There is demand amongst market participants to settle digital asset financial transactions in central bank money atomically (Cipollone (2024)). Atomic settlement is the transfer of money or assets between parties that either succeeds or fails, eliminating the risk of partial or incomplete multi-party transactions. The ability to settle in central bank money enhances the efficiency and safety of financial market infrastructures. It provides the safest and most liquid settlement asset for market participants and serves as the cornerstone for financial stability. As technology continues to transform the provision of financial services, it is therefore important that central bank money remains compatible with new technologies.

One approach being explored to meet this demand is "synchronisation". It can potentially expand atomic settlement in central bank money to a broad spectrum of assets and funds. Synchronisation involves coordinating the movement of central bank money between accounts in wholesale payment infrastructures, including Real-Time Gross Settlement (RTGS) systems, transferring assets or funds across one or more other, external ledgers. Enhancing wholesale payment infrastructures to enable synchronisation will allow for central bank money to be settled atomically.

In 2023, Project Meridian explored the concept of synchronisation through the development of a synchronisation operator (SO) (BISIH (2023)). This prototype was designed to orchestrate the simultaneous transfer of digital assets and funds in a hypothetical UK housing transaction.

Project MFX subsequently builds on the concept of synchronisation to demonstrate its technical feasibility in a multicurrency FX transaction. Through the experiments conducted during the project, SOs enabled atomically settled FX transactions between different RTGS systems in various jurisdictions, as well as between an RTGS system and a DLT platform.

Project MFX provides important insights that can serve as building blocks for future innovations in wholesale payment infrastructures. Through its experiments, the project was able to demonstrate that a synchronisation model can:

- be agnostic to both the asset or fund of the transaction involved and the technology of the ledgers on which settlement is performed, highlighting its potential use in a range of markets.
- exchange via two wholesale payment infrastructures if either of them can earmark[1] assets or funds. This means that the model could potentially be applied to several different currencies and markets with different payment infrastructure functionality, while enabling the SO to handle the complexities of interoperability.
- conduct payment-versus-payment (PvP) FX settlements without taking on credit or liquidity risk, and without the SO needing an account in the respective RTGS systems. This limits the credit and liquidity risks SOs pose to the financial system.

1 In synchronisation, settlement is automated and atomic, reducing the need for arrangements like escrow. Escrow is replaced by earmarking the assets and funds for a short period of time before settlement, without moving them from their existing account.

- potentially enable additional functionalities to be provided to market participants to meet their needs through wholesale payment infrastructures: for example, allowing commercial banks to set approval limits on transactions flowing through their accounts.

- provide end users with greater choice and a way to balance the desires for immediacy and liquidity efficiency. Under the Meridian FX model, an SO could provide synchronised gross settlement for certain transactions (for example, those that end users might consider urgent) and synchronised gross settlement with offsetting for less urgent transactions where liquidity efficiencies can be realised.

Additionally, Project MFX demonstrated the various ways that synchronisation can be achieved based on the capabilities of the underlying wholesale payment infrastructure, as well as the potential enhancements that could further support atomic FX transactions.

Collectively, the findings establish synchronisation as a potentially transformative approach for modernising FX settlements and fostering interoperability and efficiency across diverse financial systems.

Background

The cross-border payments landscape

Since 2020, the G20 has prioritised enhancing cross-border payments, which could benefit citizens and economies worldwide (CPMI (2020)). To achieve these aims, they have developed a comprehensive roadmap to address the various frictions that underlie cross-border payments.

These frictions include delays in settling cross-border payments and the lack of interlinkages between cross-border settlement systems. The G20 is working to address these by, among other things, (i) facilitating the increased adoption of payment-versus-payment (PvP) mechanisms, and (ii) establishing more links between the payment infrastructures of different countries (CPMI (2023)).

First, PvP arrangements are specifically designed to mitigate foreign exchange (FX) settlement risk[2]. However, they are constrained by limited Real Time Gross Settlement (RTGS) settlement hours and differences in time zones, especially in cross-border transactions. Beyond reducing settlement risk, PvP mechanisms may also provide significant benefits if they can effectively provide netting, which reduces liquidity requirements for funding FX settlements (see the "Extension objectives" section below). Additionally, as markets increasingly shift towards shorter settlement, interest in enabling same-day PvP services that provide fast and reliable access to foreign currency liquidity has been growing.

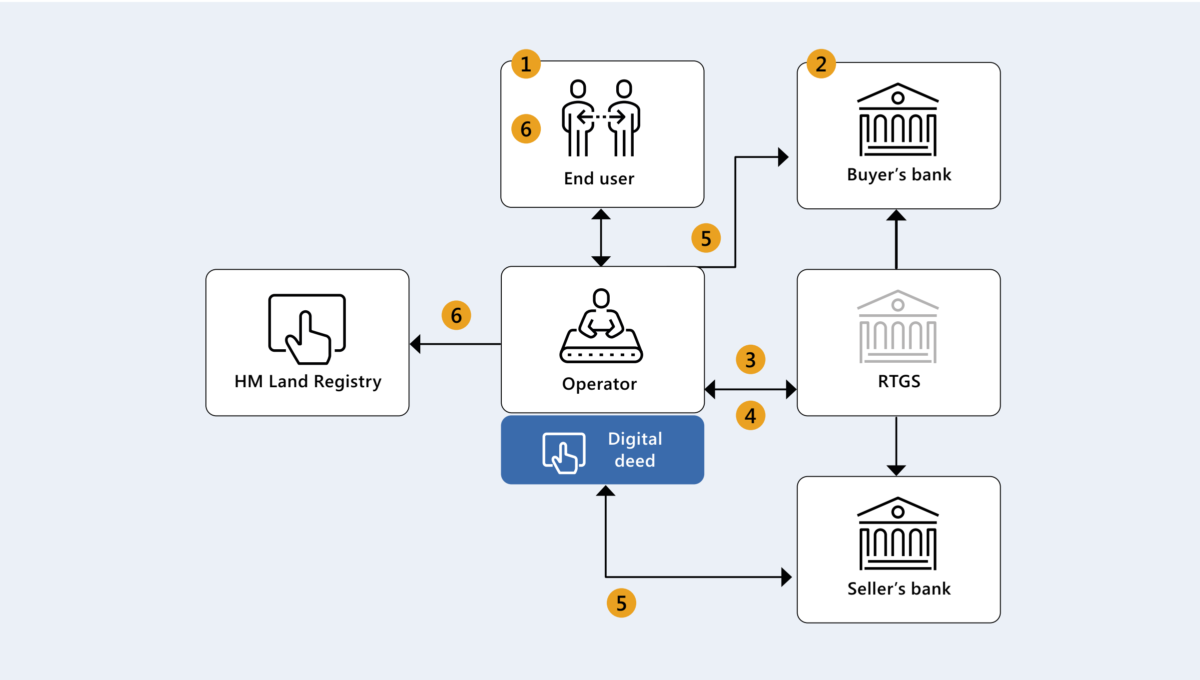
Second, establishing greater cross-border links between payment infrastructures enables more transactions to take place in central bank money – the safest and most liquid settlement asset. There is a growing demand from market participants to settle digital asset transactions using central bank money (Cipollone (2024)). Synchronisation provides an opportunity to build interoperability solutions with a cross-border capacity for these new technologies.

2 Settlement risk arises when one party in an FX transaction pays the currency it sold but does not receive the currency it bought. Such settlement failures can stem from default events, operational issues, or liquidity constraints in the market.

Project Meridian FX

Project Meridian, concluded by the BIS Innovation Hub (London Centre) and the Bank of England in 2023, successfully synchronised a delivery-versus-payment (DvP) transaction for a hypothetical housing purchase in the United Kingdom (see Figure 1). In doing so, it provided an important foundational step for further potential applications of synchronisation explored through Project Meridian FX (Project MFX).

Figure 1: The synchronisation model developed during Project Meridian (2023)



1 **Request settlement service**

To start a transaction, counterparties appoint a synchronisation operator.

2 **Buyer commits funds**

Reservation of the buyer's funds for the transaction by applying a hold is instructed.

3 **Funds reserved**

To prevent the funds or assets being used in other transactions, an earmark is placed. This can last forminutes, and approval is needed before they are placed.

4 **Settlement**

Final settlement is achieved when the synchronisation operator instructs funds to move from the buyer's bank to the sellers in the RTGS system. Ownership of the asset changes.

5 **Balances updated**

The buyer's and seller's bank account balances are updated after settlement finality is achieved.

6 **Confirmation of settlement**

End users receive confirmation that the transaction has been completed, and the digital deed is sent to HM Land Registry. Remaining steps in the house purchase process can now occur.

Project Meridian's key innovation was to build a prototype for a synchronisation operator (SO) and introduce it to the transaction. The SO could orchestrate atomic settlement across two platforms, transferring the asset (the housing deed) if the funds had been received. The project demonstrated the potential for synchronisation to speed up the transaction's settlement time, increase certainty for the seller and buyer, and reduce associated costs and risks.

The synchronisation concept was based on the assumption that it could be agnostic to both the asset being settled and the underlying technology of the ledgers involved. Such features could grant the SO the flexibility to be used for a range of financial transactions, connecting traditional and emerging infrastructures. With its DvP use case, Project Meridian was a crucial step in exploring the possibilities of synchronisation and payment interoperability.

The FX use case

Synchronisation is an asset-agnostic concept, and the Project Meridian prototype was developed to be potentially adapted to many asset classes. Settling an FX transaction using synchronisation has the potential to address several of the frictions in cross-border payments outlined above.

Synchronisation could be used in FX markets where PvP is not currently adopted, and settlement risk could be significantly mitigated by conditionalising the exchange of currencies. In addition, faster settlement could lower liquidity costs for counterparties by removing the need to set aside funds for an extended period to settle transactions.

Project Meridian FX Overview

Objectives

Project MFX builds upon the foundational work of Project Meridian to test and validate the technology enabling the atomic settlement of FX transactions across ledgers based in different jurisdictions and built on different technologies. It expands the scope of synchronisation from RTGS systems to include fast payment systems and distributed ledger technology (DLT) platforms.

To do this, the project identified two core objectives:

- To demonstrate that an SO can orchestrate synchronous settlement of an FX transaction. That is, the SO can enable the atomic settlement of an FX transaction involving payment systems in different jurisdictions.

- To demonstrate that an SO can orchestrate a synchronous settlement of FX transactions involving ledgers based on different technologies, for example, enabling the atomic settlement of an FX transaction between a DLT-based platform and non-DLT systems.

To meet its objectives, the project required the involvement of multiple central banks to demonstrate the ability of an SO to "connect" different wholesale payment infrastructures. Consistent with the BIS Innovation Hub's mandate, the project did not look at the commercial proposition of introducing SOs into the FX market or the potential regulatory or legal frameworks for their introduction.

Participation of central banks in Project MFX

While Project Meridian was a joint project between the Bank of England and the BIS Innovation Hub (London Centre), the potential cross-border application of synchronisation is relevant to the interests of many central banks.

The Bank of England's "Future Roadmap for RTGS" sets out a forward-looking approach to continuously evolving its RTGS service (after the new core ledger and settlement engine goes live) in response to the industry's changing needs (BoE (2024)). A core functionality of this roadmap is the development of a synchronisation interface, which would extend settlement capabilities in central bank money by incorporating a broader range of assets. Synchronisation would allow RTGS systems to interoperate with other external ledgers and technologies.

A major priority for the Bank of England is to co-create with prospective SOs, RTGS account holders and other relevant stakeholders, leveraging that collaboration to shape the design of the new service in a way that maximises value for the industry. The Bank of England has been working closely with the industry as part of the co-creation to assess business cases, refine high-level design and prepare for delivery.

The Eurosystem has also been exploring possible interoperability solutions amid the growing market interest in using DLT to settle wholesale financial transactions. Between May and November 2024, the Eurosystem conducted exploratory work on new technologies for wholesale central bank money settlement to investigate how new technologies, such as DLT, could interact with traditional payment systems (see Box A for more detail).

The Eurosystem's objectives for the exploratory work were (i) to consolidate and further develop the ongoing work of Eurosystem central banks in this area, and (ii) to gain insights into how different solutions could facilitate interaction between TARGET Services and DLT platforms. The Eurosystem's participation in Project MFX contributed to these learning objectives, providing insights on the interlinking approach to facilitate cross-border payments consistent with the G20 priorities (CPMI (2020)).

Box A: The Eurosystem's exploratory work on new wholesale central bank money settlement technologies

Between May and November 2024, the Eurosystem invited financial market stakeholders to participate in trials and experiments to explore new technologies for wholesale central bank money settlement. This work involved 60 market participants conducting 28 trials (settlements in central bank money) and 20 experiments (mock settlement). These trials and experiments covered a broad range of use cases in the wholesale financial market, from the issuance, life cycle management, and redemption of securities to automated wholesale payments and foreign exchange (FX) payments.

The Eurosystem provided three interoperability-type solutions for the exploratory work:

- the Trigger Solution – consisting of a distributed ledger technology (DLT) infrastructure that acts as technical bridge between T2 and market DLT platforms
- a full-DLT interoperability solution, DL3S – allowing the settlement of wholesale financial transactions in tokenised central bank money handled in wallets held on a DLT platform provided by the Eurosystem
- TARGET Instant Payment Settlement (TIPS) Hash-Link – enabling settlement of wholesale financial transactions in central bank money in accounts on a TIPS-like platform set up for the Eurosystem.

The results of the trials and experiments were assessed using a set of key performance indicators, including settlement performance and efficiency, liquidity management, and integration with the new environment (market DLT). The Eurosystem will publish the findings of the exploratory work in a forthcoming report.

In February 2025, the Governing Council of the ECB decided to expand the Eurosystem initiative to settle transactions recorded on DLT by following a two-track approach:

- On track 1, the Eurosystem will develop and implement a solution through an interoperability link with the existing TARGET Services in a timely fashion.
- On track 2, the Eurosystem will develop a long-term vision for a more integrated solution to settle DLT-based transactions in central bank money, which will include an international dimension.

For both tracks, the lessons learned from the Project MFX project will feed into the analysis.

Project Meridian FX's interoperability solutions

To test the synchronous atomic settlement of FX transactions, Project MFX connected an emulated version of the UK RTGS service, via an SO, to the three Eurosystem interoperability solutions: the Trigger Solution (developed by the Deutsche Bundesbank), the DL3S solution (developed by the Bank of France), and the TIPS Hash-Link solution (developed by the Bank of Italy).

These component pieces of the Project are described below.

RTGS emulator

Project MFX used an RTGS emulator to simulate the GBP leg of the FX transaction between the Bank of England and the three Eurosystem interoperability solutions, which simulated the EUR leg. While the UK RTGS emulator does not reflect all features of the current UK RTGS service, it served as a model to provide preliminary insights into the types of enhancements that might be necessary for RTGS services to integrate the SO's services across various financial applications in the future.

The emulator implemented instructions from the hypothetical direct participants in RTGS or the SO. It enforced a rule set on the messages received, validating the transactions and updating the emulated accounts used for the project. The emulator enabled multiple ISO 20022-like application programming interface (API) calls to be processed, emulating message traffic that would have been received from a global financial messaging network. These API calls were used mainly to support the reservation of funds and the eventual settlement of those reserved funds from bank to bank. (See the "Supported messages" section in the technical annex for a full list of messages supported by the emulator.)

Trigger Solution

The Trigger Solution (Deutsche Bundesbank) facilitates wholesale settlement by linking an external platform with T2 through a "trigger" mechanism (payment instruction smart contract). This mechanism enables the coordination of asset or fund transfers by providing blocking functionality through an interim account in T2. This setup enables atomic settlement by – depending on the chosen interoperability mechanism – first escrowing funds and settling payments in central bank money only upon confirmation that the asset transfer conditions on the external platform are met. The solution is designed to be ledger-agnostic, allowing integration with various platforms while fully leveraging existing RTGS infrastructures for settlement. (See Figure 8 in the technical annex for an illustrative diagram of the Trigger Solution.)

Full-DLT Interoperability Solution (DL3S)

The Full-DLT Interoperability Solution (DL3S) (Bank of France) technically allows wholesale central bank money to be issued and settled directly on a DLT platform, whereby tokens are issued on the Bank of France's DL3S DLT. Under this model, an SO orchestrates settlement by routing information between a foreign currency system and DL3S network participants.

It starts with instructing the PvP payment legs of each counterparty on its respective systems. Then each payment instruction is sent to the counterparty's system. Both legs are then paired in DL3S. The settlement is then triggered by each system at the intended settlement date (value date). Once both systems have locked the respective amounts, the SO orders their release, enabling atomic settlement of the transactions involving tokenised and/or fiat moneys across different external platforms.

In the Project MFX experiment context, DL3S uses the connectivity provided by SWIFT's Transaction Management Simulator (TMS). TMS provides secure communication and transaction synchronisation, enabling the RTGS emulator to interoperate with DL3S. (See Figure 10 in the technical annex for an illustrative diagram of DL3S.)

TIPS Hash Link Solution

The TIPS Hash-Link Solution (Bank of Italy) utilises a ledger-agnostic API interoperability model, employing cryptographic hash functions to create a secure link between an external DLT platform and a payment system that supports ISO 20022 messages. In the context of the Project MFX experiment, a local instance of the TARGET Instant Payment Settlement (TIPS) system was used as the payment system.

Under this model, a hash value is generated and shared between the external DLT platform and TIPS-like platform. The completion of a transaction on the external DLT platform is contingent upon revealing the secret that generates the hash, which is disclosed only after the corresponding payment in the TIPS-like platform is irrevocably settled. This mechanism ensures atomic settlement. The TIPS Hash-Link Solution supports continuous processing on a 24/7 basis and is designed to integrate seamlessly with multiple external platforms without requiring new developments for each integration. (See Figure 9 in the technical annex for an illustrative diagram of the TIPS Hash-Link Solution.)

The Project Meridian FX's synchronisation operator prototype

Function

The key component of Project MFX is the design and build of the SO prototype. It is based on DLT, which allows the operator to be either a single entity or distinct operators in different jurisdictions that run as nodes on a shared network.

The SO orchestrates FX transactions by tracking the overall status of a chain, where a chain is a set of transactions that should be settled together. Each transaction in the chain follows a defined life cycle: (i) collect and match data, (ii) earmark funds, and (iii) initiate settlement. The SO tracks each transaction's status within the chain and moves to the next life-cycle stage when all transactions are ready.

This workflow guides the SO through a series of states, triggering API calls to the relevant external systems at each step. The SO awaits responses from these systems before progressing to the next state. As each system has differences in how it interfaces (for example, unique messaging requirements, different workflows), an integration layer within the SO handles the unique characteristics of each external system. The integration layer implements different adapters to communicate with external systems. These adapters are responsible for constructing the message payload, handling the API security and parsing the responses based on the capabilities provided by the external system.

The standard data messages between the SOs and underlying settlement infrastructures follow the ISO 20022 payment messaging standards where possible.[3] This is because SOs are envisioned to interact with both RTGS systems and other entities that provide interoperability with the RTGS systems. As these other entities may interact via API, the SOs must be configured accordingly.

Additionally, the SOs can harmonise data structures of multiple ledgers to a common structure by facilitating information exchange between these ledgers. Changes on external ledgers are triggered by the SOs. The earmarking of funds ensures those funds can only be used for that transaction while all conditions for settlement are met. Once all conditions for settlement are met, this triggers settlement.

3 The value of the ISO 20022 messaging standard comes from providing consistency of data in payment messages.

Process

The following is the process by which the SO achieves atomic FX settlement (also see Figure 2):

Step 1: Outside the scope of the SO and the payment infrastructures, two direct participants of the infrastructure agree upon an FX transaction.

Step 2: The debtor entity in each jurisdiction sends a request to their respective SO node. Following this, the two requests are matched,[4] validating that they are related to the same overall transaction. Depending on the priority of the transaction, the SO will either queue the transaction into a liquidity queue or process it immediately.

Steps 3 and 4 have two variants, one of which will be chosen depending on the underlying payment infrastructures used in the transaction.

Step 3:

- In variant 1, where both underlying infrastructures are configured in the SO) to support reservations, each SO node must request a funds reservation from its respective payment infrastructure ("earmark"). Following this, each SO node awaits a successful response from its respective payment system.
- In variant 2, where only one of the two underlying infrastructures supports reservations, the SO node configured to use reservations will request a funds reservation and await a successful response from the payment system.

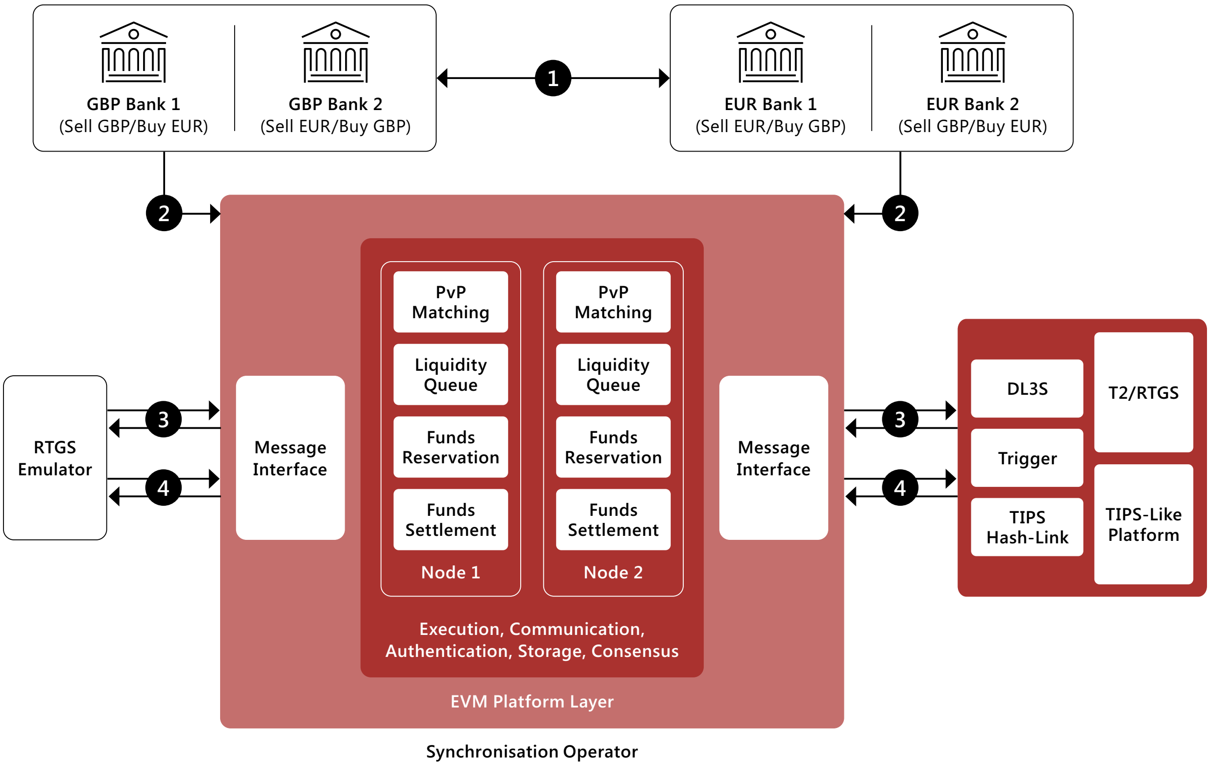
Step 4:

- In variant 1, each SO node will initiate a funds transfer in its respective wholesale payment infrastructure ("release" of earmark). Following this, each SO node will await a successful response from its respective payment system.
- In variant 2, the SO node configured not to use a reservation will initiate a funds transfer. Once that transaction has completed, the second node holding a reservation will initiate its funds transfer and await a successful response from the wholesale payment infrastructure.

Step 5 (not shown in Figure 2): The SO nodes will update the transaction status to complete.

4 Either in the SO or in one of the two wholesale payment infrastructures, depending on the configured setup.

Figure 2: MFX SO Model



Architecture

SOs could use centralised or DLT to deliver their functions. Project MFX chose to prototype a distributed platform architecture, as this approach allows the exploration of additional complexities. These include data sharing for transactions and privacy concerns and a multiplicity of SOs meeting requirements specific to different jurisdictions.

The Project MFX SO platform architecture is a DLT that uses the Ethereum Virtual Machine (EVM) and was designed to support multiple independent SOs.

This architecture enables the following:

- **Data locality**: Data for UK RTGS settlements and EU TARGET Services and interoperability solutions can be stored locally in each jurisdiction. This may be an important characteristic for jurisdictions with strict data locality regulations. It also supports a heterogeneous operating model, whereby a trusted service provider delivers SO node services for themselves or on behalf of other banks in one or more jurisdictions. A central service provider can also provide services to participants in different jurisdictions.
- **Data privacy**: Full settlement instruction data are persisted locally and hashed globally for matching and synchronisation. This ensures that the core synchronisation functionality is robust and does not leak information to the operators of synchronisation nodes. These could be other non-counterparty commercial banks with whom participants may not want to share commercially sensitive information, such as FX settlement instructions.
- **Immutability**: Once a transaction's state is recorded on the blockchain, it cannot be altered or tampered with. This supports data integrity.

Outcome of experiments

The Project MFX's SO demonstrated that PvP for FX transactions involving GBP and EUR currencies are technically feasible. Each Eurosystem solution successfully performed multiple PvP transactions with the UK RTGS emulator. Experiments were conducted with both successful transactions and planned failed transactions. Failure cases covered transactions with insufficient liquidity, invalid accounts, payment systems closure, and failed reservations.

Through its experiments, the project was able to demonstrate that a synchronisation model can:

- be agnostic to both the asset or fund of the transaction involved and the technology of the ledgers on which settlement is performed, highlighting its potential use in a range of markets.

- exchange via two wholesale payment infrastructures if either of them can earmark assets or funds. This means that the model could potentially be applied to several different currencies and markets with different payment infrastructure functionality, with the SO handling the complexities of interoperability.

- conduct PvP FX settlements without taking on credit or liquidity risk and without the SO needing an account in the respective RTGS systems. This limits the credit and liquidity risk posed by SOs to the financial system.

Additionally, the project demonstrated the various ways to achieve synchronisation based on the capabilities of the underlying payment infrastructure, as well as potential enhancements that could further support atomic FX transactions.

Extension objectives

Beyond its core objectives, Project MFX also explored additional functionalities that could be provided to market participants through wholesale payment infrastructures via an SO. Specifically, the project explored:

- how an SO could handle a chain of transactions to enable liquidity savings for participants
- how an SO could enable the use of gross settlement of batches of transactions to reduce the liquidity needs of participants
- how an SO could enable greater control of funds by participants (via approval limits) using the SO solution
- how to extend the basic PvP workflow to enable a chain of two FX transactions to settle atomically – referred to in this report as (PvP)+(PvP).

Enabling liquidity savings

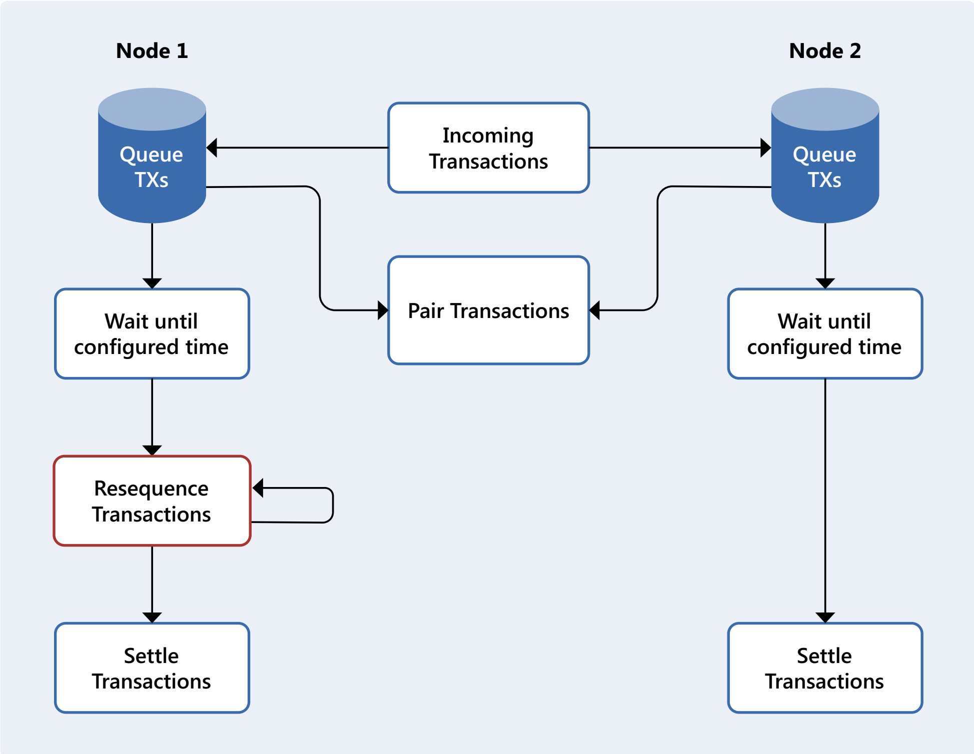
Synchronisation has the potential to provide liquidity efficiencies by coordinating the timing of payments between banks. Synchronised systems can match incoming and outgoing money flows so that payments happen at the same time.

Because the scale of liquidity efficiencies will depend on specific design choices made by future SOs, the project did not assume a single solution design or seek to quantify liquidity savings. Rather, it focused on identifying the RTGS enhancements necessary to enable future SOs to offer services providing liquidity efficiency to their customers.

To demonstrate scope for different solutions, the project explored two different mechanisms through which an SO could provide liquidity efficiencies: (i) optimising the order in which the transactions are settled, and (ii) gross settlement with offsetting.

Liquidity savings approach 1: gross settlement with queuing

Figure 3: Liquidity savings approach 1



The "gross settlement with queuing" approach (see Figure 3) is designed for interoperability mechanisms as they exist today, that is, ones that do not support net settlement. The liquidity savings approach queues all normal priority transactions received into an SO over a configured period (assumed within this project to be 30 minutes).

This liquidity savings approach begins a settlement cycle at the end of this "queuing period". Within this settlement cycle, the SO will attempt to find the optimal ordering of this set of transactions. As illustrated above, two SOs function in parallel to settle the transactions that are common between them. Importantly, under this arrangement, only one settlement system has an improved liquidity performance, that is, the optimal ordering arrangement is performed on only one system.

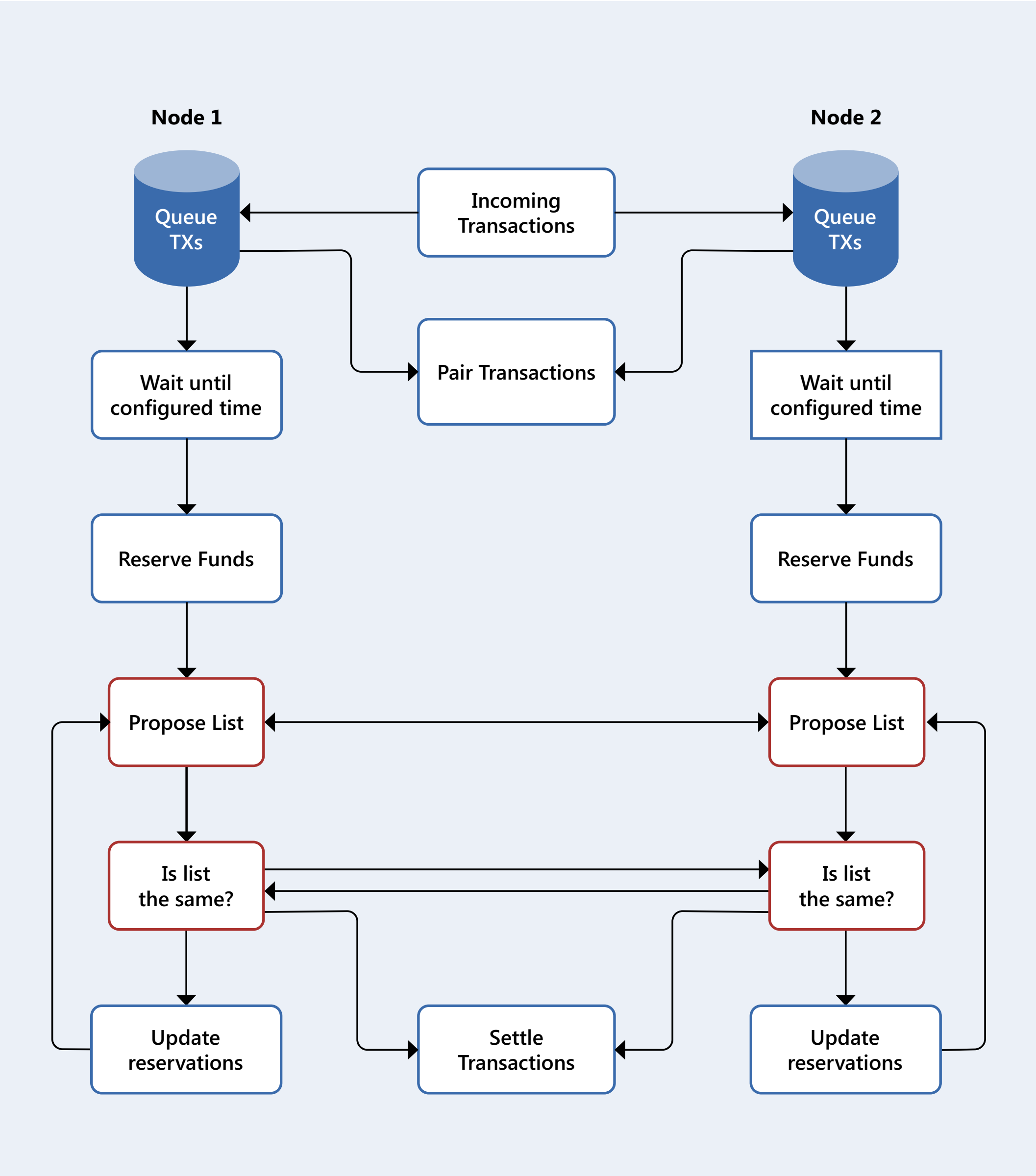
In essence, banks in the transaction set that do not have incoming payments and will never have an improved liquidity position will have their reservation and settlement stages occur first. Once those payments have completed successfully or failed, the SO will proceed to the next set of transactions. Under this methodology, banks receiving incoming payments may use those payments to make future outgoing payments, reducing the system's required liquidity.

Liquidity savings approach 2: gross settlement with offsetting

While the above liquidity savings approach could operate within existing interoperability mechanisms, the "gross settlement with offsetting" approach would require settlement infrastructures to support net settlement transactions (see Figure 4).

This liquidity savings approach follows the settlement cycle methodologies described in the queuing approach. At the end of every "queuing period", the liquidity savings approach will determine the net settlement obligations of each bank in the network. Earmarks are placed for all accounts with a negative settlement obligation; and using a PACS.029 message, all net settlement obligations are settled simultaneously in the respective RTGS service.

Figure 4: Liquidity savings approach 2



This approach has additional complexities. As transactions under a gross settlement with offsetting arrangement become dependent upon each other, a negotiation between the SOs must take place prior to settlement. Each SO must publish the set of transactions for which they have successfully placed a reservation onto the distributed ledger, and the corresponding SO (for the other leg of the transaction) must agree with the broadcasted list prior to settlement.

This liquidity savings approach has several potential advantages:

- It provides significantly better liquidity savings than the "queuing" approach.
- It is more certain in terms of results, as each bank's positions are known before settlement.
- Because it has only one settlement operation that includes all participants' transactions, this approach takes less time to settle than the gross settlement with queuing approach.

As noted above, this project did not attempt to benchmark the percentage of liquidity savings obtained by these two approaches. Given that percentage reductions are a function of the transactions within the system, accurate benchmarking is impossible without live transaction data.

Control Points

Control of funds is essential to a bank's liquidity management operations. Within the liquidity savings mechanism, Project MFX explored allowing banks to set approval limits on transactions settled through their accounts. Under this model, if that approval limit is breached within a liquidity queue, the bank must perform a manual approval stage to allow the transaction to be processed.

Atomic settlement of a chain of two FX transactions – (PvP)+(PvP)

The basic PvP workflow could be extended to allow a chain of two FX transactions (involving three payment legs, currencies and parties) to settle atomically, referred to here as (PvP) + (PvP).

In our experiment, two hypothetical currencies (EXP1, EXP2) were transacted using GBP as the bridge currency. EXP1 and EXP2, the "outside legs" of the transactions, were reserved. Once reserved, the GBP transactions, or "inside legs", were settled via a PACS.029. Finally, once the inside legs were completed, the outside legs of the transaction were settled. This flow enabled the bridging currency to utilise netting, reducing the liquidity in the bridge currency that would have been required when using separate transactions. (See Figure 13 in technical annex.)

Insights, lessons learned and areas for future exploration

Project MFX successfully demonstrated that PvP for FX transactions involving GBP and EUR currencies, using an SO, is technically feasible. It showed that synchronisation can achieve interoperability and implement new functionalities based on the capabilities of the underlying wholesale payment infrastructures. This paves the way for innovation in settlement services across a broad spectrum of asset classes.

The project also illustrated the flexibility of synchronised settlement models. They can enable new choices and flexibility for end users who want to realise overall liquidity efficiencies while treating certain transactions as urgent.

Regarding the G20 targets to enhance cross-border payments, Project MFX made significant progress towards Focus Area C: "Improve existing payment infrastructures and arrangements to support the requirements of the cross-border payments market" (CPMI (2020)). Specifically, the project has demonstrated the technical feasibility to reduce FX settlement risk via facilitating PvP settlements and has established realistic interoperability links between the wholesale payment infrastructures of different countries.

However, it is important to note that, consistent with the mandate of the BIS Innovation Hub, the project did not assess the commercial proposition of introducing SOs into the FX market. Such an assessment could be undertaken as a potential next step by interested central banks, competition authorities, or existing or emergent market participants.

Interoperability

The experiments proved that a single synchronisation interface can serve as an integration point across diverse central bank solutions, whether based on DLT or traditional ledgers. This integration capability achieved fund reservation (when supported) and settlement finality across different settlement systems, highlighting synchronisation's adaptability and technology-neutral design.

Flexibility

The various solutions used in Project MFX showed the flexibility of RTGS services in connecting to different types of settlement orchestration through new infrastructures. This flexibility also demonstrates the potential for SOs to exchange via two wholesale payment infrastructures if either of them can earmark assets or funds. This means the model could be applied to several different currencies.

Alongside 2023's Project Meridian, Project MFX has also provided evidence that the SO model is agnostic to both the asset/transaction involved and the technology of the ledgers, highlighting its potential use in other markets.

Finally, through its experiments, the project also indicated that synchronisation can be supported via a range of wholesale payment infrastructures, such as those that:

- do not support the reservation of funds through an earmark or escrow-type functionality. For these infrastructures, PvP transactions are enabled on a conditional basis. First, the payment is made in the system that does not support reservations, and then, if successful, the reserved fund is released and transferred.
- support the reservation of funds directly related to the settlement of those reserved funds.
- support reservations as "buckets" of funds that may be used for any authorised settlement. Additionally, these systems support net settlement, enabling greater flexibility for the SO to optimise liquidity.

Functionality

Within Project MFX, the SO could conduct the transactions without taking on credit or liquidity risk and without needing an account in the respective RTGS systems.

Synchronisation could also occur even if one of the underlying wholesale payment infrastructures does not have the ability to use earmarks or escrow. This is potentially important for jurisdictions without interoperability solutions.

Project MFX also explored extra functionalities that a wholesale payment infrastructure could provide to market participants, for example, the prototype allows participants to set approval limits on transactions settled through their accounts and a degree of liquidity savings.

Liquidity savings

Project MFX underscored the potential of a synchronisation interface to enable an SO to offer a form of hybrid synchronised settlement that provides end users with greater choice and a way to balance desires for immediacy and liquidity efficiency.

Under the Project MFX model, an SO could provide synchronised gross settlement for certain transactions: for example, for those that end users consider urgent. And it could also provide synchronised gross settlement with offsetting for less urgent transactions, where liquidity efficiencies can be realised.

While the design of the interoperability mechanisms used in the project allowed for liquidity savings using resourcing algorithms, the level of actual liquidity savings realised depend greatly on transactions flowing through the system.

The (PvP) + (PvP) extension objective suggests that a liquidity savings mechanism can facilitate atomic settlement across three ledgers. This supports bridge currency settlement, which could facilitate FX transactions involving less liquid or widely traded currencies.

Although the SOs achieved liquidity savings in the current phase, the project's focus was not on maximising these savings. Instead, it aimed to identify the RTGS enhancements necessary to create a conducive environment for future SO services.

Future insights

Beyond the FX use case, Project MFX also highlighted the broader applicability of synchronised settlement, potentially transforming how cross-border transactions across asset classes are managed, settled and recorded.

In addition, it provides important insights which can serve as building blocks for future innovations in wholesale payment infrastructures. For example, while the UK RTGS emulator does not fully reflect the design considerations of the current UK RTGS service, it could serve as a model to inform on possible modifications required to create an environment conducive to support synchronisation operators' functionalities in the future.

The project also reinforced the value of collaboration, not only in advancing synchronisation but also in providing insights for future central bank innovations and the evolution of the global payment ecosystem. Looking ahead, sustaining progress in synchronisation innovation is vital to extending the reach of settlement in central bank money. Expanding its application to include digital assets and securities can foster a more inclusive and resilient financial ecosystem.

Future work

Bank of England

A focus area of the Bank of England's "Future Roadmap for RTGS" is to develop a synchronisation interface. Insights gained from Project MFX will play a pivotal role in shaping the design and implementation of this interface, particularly in refining the technical aspects of synchronisation and enhancing the Bank of England's understanding of interoperability within wholesale settlement environments.

In parallel, the Bank of England's wholesale experimentation programme will continue to explore the use cases for the synchronisation concept as compared with potential wholesale digital currency (CBDC) approaches. These experiments are a key pillar of the Bank of England's wholesale payment strategy, aimed at enhancing the role of the RTGS service as a platform for innovation in payments.

Eurosystem

The Eurosystem valued the highly successful collaboration with the BIS Innovation Hub and private and public entities during its exploratory work and is committed to supporting the use of innovative solutions in its market infrastructures while maintaining the safety and efficiency of TARGET Services. Through Project MFX, all three of the Eurosystem's interoperability solutions were successfully connected to the SO and demonstrated their adaptability to perform PvP FX settlement. The Eurosystem's involvement also enabled a comparison of the three interoperability solutions in terms of their technical merits for cross-border settlement, in a standardised and harmonised test scenario. The collaboration between the participating organisations also provided an opportunity for the Eurosystem to gain practical experience, forming an integral part of their exploratory work on new technologies for wholesale settlement.

The lessons learned from Project MFX will feed into the Eurosystem's follow-up work, which is organised in two tracks:

- On track 1, the Eurosystem will develop and implement a solution through an interoperability link with the existing TARGET Services.
- On track 2, the Eurosystem will develop a long-term vision for settling DLT-based transactions in central bank money, which will include an international dimension.

The Eurosystem will determine the project timeline in the coming months and will provide further details in due course (ECB (2025)).

Project participants and acknowledgements

Project sponsors

Raphael Auer, Eurosystem Centre Head

Francesca Hopwood Road, London Centre Head

BIS Innovation Hub

Rita Haddad, Adviser

Stephanie Haffner, Adviser

Thomas Schardt, Adviser

Micah Smith, Adviser and Technical Lead

John Yeo, Adviser

Bank of England

Yi Wei Chai, Lead Policy Analyst, Product Design Team, Payments

John Jackson, Head of Payment Strategy Division

Irina Mnohoghitnei, Head of FinTech Hub

Dovile Naktinyte, Senior Manager, Product Design Team, Payments

Rajan Patel, Policy Adviser, Product Design Team, Payments

Bank of France

Anais Bertrand, Digital Money Expert, New Technologies and Innovation for Market Infrastructures Department

Sana Ghariani, DL3S Product Manager, Directorate Urbanisation and National and European Information Systems

Victorien Goldscheider, Experimentation Manager, New Technologies and Innovation for Market Infrastructures Department

Matthieu Herbeau, Experimentation Manager, New Technologies and Innovation for Market Infrastructures Department

Corentin Simon, Technical Lead DL3S, Directorate General Information System

Bank of Italy

Pietro Corte, TARGET Services Expert, Payment Systems Directorate

Vitangelo Lasorella, Senior Payment Systems Expert, IT Directorate

Alessio Manzo, Payment Systems Expert, Payment Systems Directorate

Giuseppe Santangeli, Manager, Payment Systems Directorate

Deutsche Bundesbank

Nadine Knaust, Market infrastructure Senior Expert, Payment and Settlements Systems Department

Katharina Tobiasch, Head of Section T2 and T2S Strategy and Policy, Payments and Settlement Systems Department

European Central Bank

Matteo Breda, Market Innovation and Integration Division

Paul Gardin, Market Innovation and Integration Division

Holger Neuhaus, Market Innovation and Integration Division

Christian Wilk, Market Innovation and Integration Division

Acknowledgements

The authors are grateful to Cecilia Skingsley, Morten Bech, Ben Lee and Takeshi Shirakami for reviewing the report and providing extremely valuable feedback, and to the business and technical project teams within the participating central banks.

References

Bank of England (BoE) (2024): "Future roadmap for RTGS", webpage, www.bankofengland.co.uk/payment-and-settlement/rtgs-future-roadmap.

BIS Innovation Hub (BISIH) (2023): Project Meridian: simplifying transactions through innovation, www.bis.org/publ/othp63.pdf.

Cipollone, P (2024): "Towards a digital capital markets union", keynote speech at the Deutsche Bundesbank symposium on the Future of Payments, Frankfurt am Main, 7 October, www.bis.org/review/r241015f.pdf.

Committee on Payments and Market Infrastructures (CPMI) (2020): Enhancing cross-border payments: building blocks of a global roadmap, Bank for International Settlements, July www.bis.org/cpmi/publ/d193.pdf.

Committee on Payments and Market Infrastructures (CPMI) (2023): Final report: facilitating increased adoption of payment versus payment (PvP), Bank for International Settlements, March, www.bis.org/cpmi/publ/d216.pdf.

European Central Bank (ECB) (2025): "Eurosystem expands initiative to settle DLT-based transactions in central bank money", press release, 20 February, www.ecb.europa.eu/press/pr/date/2025/html/ecb.pr250220_1~ce3286f97b.en.html.

Technical annex

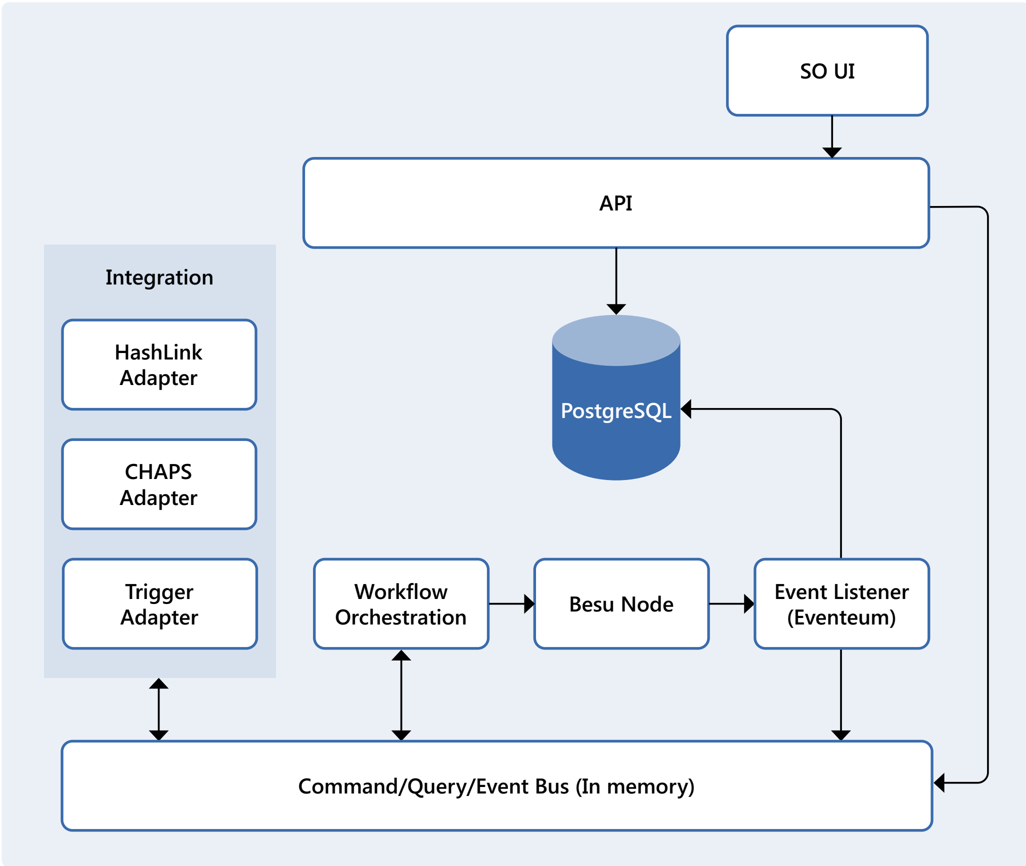
Project MFX synchronisation operator

Platform architecture

A single synchronisation operator (SO) node can be seen as a collection of the following components:

- The MFX backend consists of multiple SO components – such as application programming interface (API), integration layer, workflow orchestration, event listener and in-memory bus to synchronise across different components (see Figure 5).
  - API layer: The API layer exposes the SO's functionality to the user interface (UI) layer. The API can also be used for application-to-application integration.
  - Workflow orchestration: The SO uses state machines to execute various workflows. This component is responsible for tracking the current state and deciding the next action based on the workflow involved.
  - Integration layer: This module is responsible for integrating with external systems. The functionality includes generating message payloads, parsing the responses, handling API security, making API calls and in some instances digitally signing the payload.
  - Event listener: This component listens to events from the blockchain network and relays them to the other components.
  - Command/Query bus: This component is responsible for in-memory asynchronous message communication between the components.
- MFX database: A PostgreSQL database is used to store application and event-related information. All confidential payment data are stored on the SO's database and not shared with other SO nodes.

Figure 5: MFX Platform architecture



- MFX Besu node: The Hyperledger Besu node keeps a copy of the distributed ledger, enables the SO to participate in consensus and is used to share data to be appended to the ledger with other SOs in the network.
- MFX UI: The UI enables end users to submit payment transactions and display the status of those transactions.

Blockchain network

The Project MFX prototype deploys four SO nodes. Each SO node runs the same set of software; but through configuration, the SO node assumes the role of a different party on the network and integrates with the relevant payment system.

The architecture of the distributed ledger provides some fault tolerance. In the event of one SO node failure, data can still be appended to the ledger, and FX transactions that do not interact with the failed SO payment system can still complete. However, if two SO nodes fail, the entire system will stall; and if more than three SO nodes malfunction, immutability of appended data could be lost.

The blockchain signing keys are stored locally on the SO node. Payment details are sent to the SO nodes involved in the transaction. The SO node keeps the confidential payment details on its own database and transmits only the hashes of the key payment details to the blockchain for matching purposes.

The Project MFX prototype runs on a private/permissioned Hyperledger Besu network and utilises QBFT Byzantine-fault-tolerant (BFT) consensus algorithm. QBFT provides immediate data immutability and Byzantine resilience (the protocol can withstand up to (n – 1) / 3 validators malfunctioning or behaving maliciously).

API service

The API layer exposes the following API endpoints to the UI layer. The APIs can also be used for application-to-application integration.

Endpoint	Type	Description
api/v1/settlements	POST	Creates a settlement instruction
api/v1/settlements/highPriority	GET	Returns a list of high priority settlements
api/v1/settlements/normalPriority	GET	Returns a list of normal priority settlements
api/v1/settlements/unmatched	GET	Returns a list of unmatched settlements
api/v1/settlements/(PvP)+(PvP)	GET	Returns a list of (PvP)+(PvP) settlements
api/v1/settlements/{id}	GET	Returns the details of a given settlement
api/v1/settlements/approval/pending	GET	Returns a list of settlements pending for approval
api/v1/settlements/{id}/approve	POST	Approves a settlement with given identification
api/v1/lsa/type	GET	Returns the configured LSA type
api/v1/settlements/{settlementId}/graph	GET	Returns the liquidity savings graph for a given settlement

On-chain and off-chain data

The distributed ledger, as explained in the 'Contracts' section below, does not store actual information for the settlements to ensure privacy. Neither the smart contracts, administrator role nor individual SOs can see the leg information for settlements, since only the hash of settlement data is appended to the distributed ledger.

Other data, however, are stored on the distributed ledger, including the following:

Role registry

- list of registered clearing systems.
- list of clearing systems and whether they can perform fund reservations.

Match settlement

- list of hashes for all settlement legs.
- matching identification (ID) of each matched settlement.

Execute settlement

- list of settlements by matching ID, storing the status of the settlements and the participant clearing systems.
- batch ID and matching ID indexes for ID generation.
- negotiation slots for the liquidity saving approach (LSA) functionality, including what settlements are being negotiated between the different SOs.
- fund reservation data for negotiation, storing the matching IDs of the included settlements.
- (PvP)+(PvP) data for included settlements.

The rest of the data, as explained in the previous section, are stored only off-chain, as they are not needed within the contracts.

Contracts

Match Settlement Contract

The match settlement smart contract is designed to facilitate the confirmation and settlement of financial transactions between counterparties (see Figure 6). It integrates with an external role registry to verify roles and ensure proper permissions for the parties involved. The contract tracks transaction details, referred to as "leg1" and "leg2", using a mapping structure to manage counterparties and transaction hashes. The smart contract does not store detailed information about the individual legs of a settlement. Instead, it stores a cryptographic hash for each leg. This design ensures that sensitive data remain private while still enabling verification.

When a settlement is requested, the smart contract records the hashed values of each leg. If the counterparty submits a settlement with identical values, the smart contract compares the hashes. Upon a successful match, the contract emits an event containing the generated matching ID and transitions the settlement to the MATCHED state. Each participating node listens for this event and updates its internal state to mark the settlement as matched. This approach guarantees that no confidential settlement information is exposed on the smart contracts, maintaining privacy for participants not directly involved in the transaction.

Matching example

Given the following settlement with the data:

```
// Leg 1
{
  "clearingSystemName": "UK RTGS",
  "debtorAgentName": "Debtor Agent",
  "creditorAgentName": "Creditor Agent",
  "currencyName": "USD",
  "amount": 100,
  "dateTime": "2023-01-01T00:00:00.000Z",
  "priority": 1
}
// Leg 2
{
  "clearingSystemName": "EXP1",
  "debtorAgentName": "Debtor Agent 2",
  "creditorAgentName": "Creditor Agent 2",
  "currencyName": "EXP1",
  "amount": 80,
  "dateTime": "2023-01-01T00:00:00.000Z",
  "priority": 1
}
```

Each individual leg is then hashed using SHA-256, which produces:

```
leg1Hash:
16becfc0eaf1c6fdbe7ade6c5f7fdd2a511448aaff39450dfbab1a9577b3fc60
leg2
Hash: ffdd4fc8b3a219e8795c1754b7c7cdd33590bb08f5cc0d6a9238cc87e95bb063
```

When the counterparty submits a settlement with the same legs, but reversed:

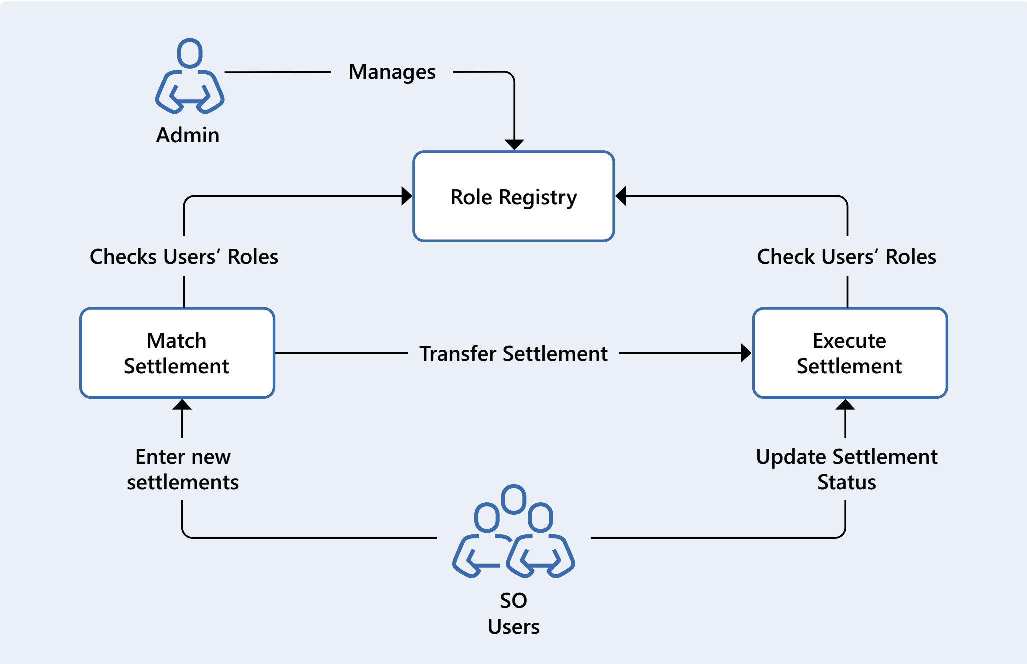
```
// Leg 1
{
  "clearingSystemName": "EXP1",
  "debtorAgentName": "Debtor Agent 2",
  "creditorAgentName": "Creditor Agent 2",
  "currencyName": "EXP1",
  "amount": 80,
  "dateTime": "2023-01-01T00:00:00.000Z",
  "priority": 1
}
// Leg 2
{
  "clearingSystemName": "UK RTGS",
  "debtorAgentName": "Debtor Agent",
  "creditorAgentName": "Creditor Agent",
  "currencyName": "USD",
  "amount": 100,
  "dateTime": "2023-01-01T00:00:00.000Z",
  "priority": 1
}
```

The smart contract will compare the hashes, and since they match, it creates a new matching ID (1) and saves the information. It then emits a settlement confirmed event with the hashes and the matching ID. This is saved by the affected nodes (UK RTGS and EXP1), and the state of the settlement is updated to MATCHED.

Execute settlement contract

The execute settlement contract manages the execution of settlements. It tracks settlement data, negotiation candidates for LSA and fund reservations, progressing settlements through the different statuses of the settlement life cycle. Integrating with role registry and match settlement, the execute settlement contract enforces strict checks on participant roles, IDs and permissions to ensure secure operations. The contract supports batch negotiation and fund reservation consensus for all participating nodes, allowing for batch processing and negotiation of settlements for the LSA functionalities.

Figure 6: Role registry



Role registry

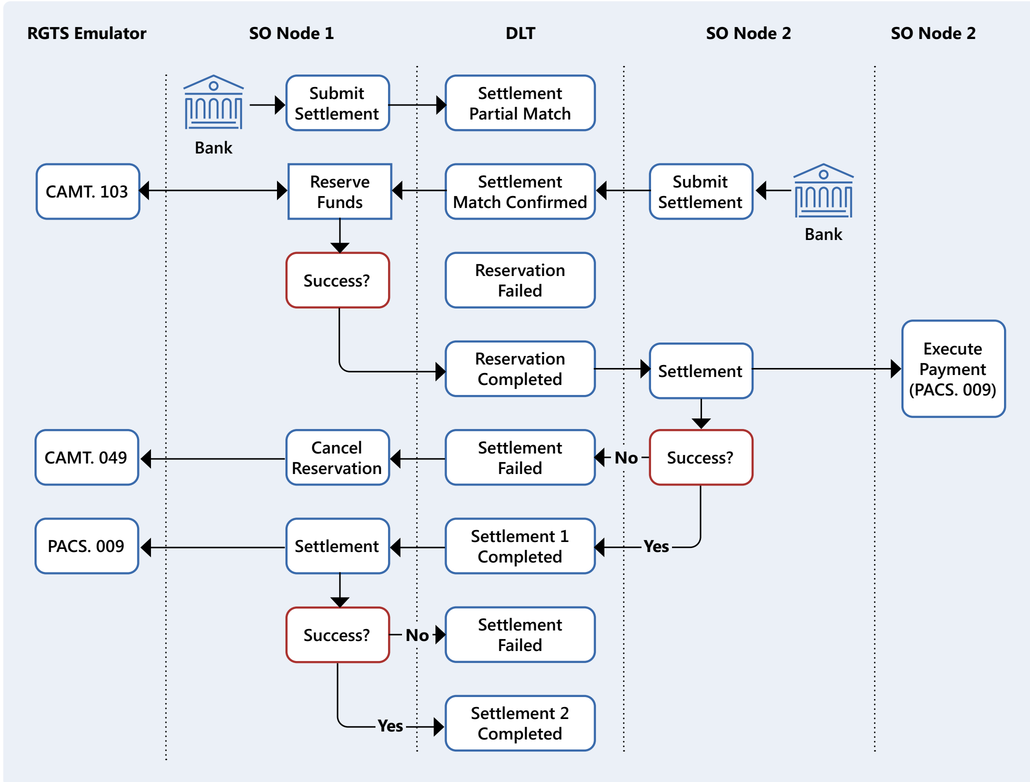
The role registry contract is designed to manage the role-based access control for clearing systems and handle fund reservation capabilities. Built on OpenZeppelin's AccessControlUpgradeable, it supports secure and upgradeable role management. During initialisation, it registers predefined roles such as UK RTGS_ROLE and TRIGGER_ROLE, while associating specific roles with fund reservation capabilities. The contract allows assigning accounts to clearing systems while ensuring that no account is registered more than once. It provides functionality to query the clearing systems associated with a specific account and verify the existence of roles. Additionally, it includes methods to check if an account holds the default admin role and to confirm whether a clearing system supports fund reservation.

Transit settlement events

For high priority settlements, the general flow for the status transitions for the settlement can be seen in Figure 7 and is described as follows.

The involved SO systems in the settlement will listen to the events emitted by the DLT, making status updates and saving the relevant information for each step. The matching ID is an integer value used to relate settlements across the different nodes.

Figure 7: Transit settlement events



1. Settlement Submission:

- The agent from one of the clearing systems submits a new settlement.
- The agent from the other clearing systems submits a settlement that matches the one submitted onto the other clearing system.

2. Funds reservation:

- The process starts with an attempt to reserve funds.
- A decision point checks if the funds reservation was successful.
  - **If NO,** the process terminates with a failure (reservation failed).
  - **If YES,** the process moves to DLT for settlement matching.

3. Settlement matching confirmed:

- If funds reservation is successful, the system confirms the settlement matching.
- Once confirmed, it transitions to settlement reservation complete.

4. Settlement reservation complete.

- The process updates to settlement reservation completed and moves to the settlement steps.

5. Settlement failure

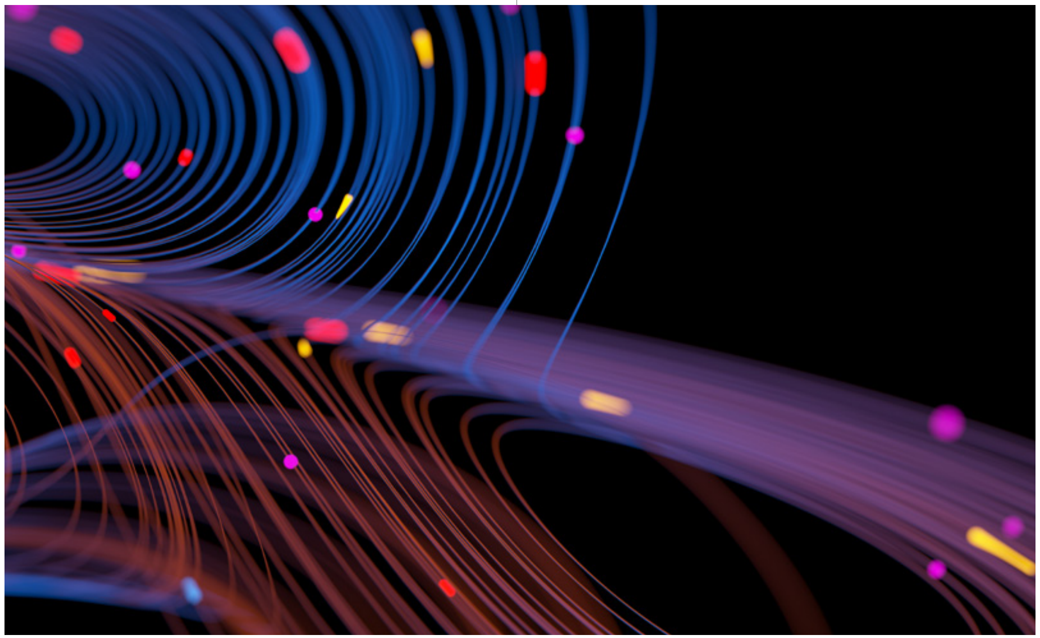
- Cancel the reservation due to errors in non-reserved settlement.

6. Process non-reserved settlement:

- A decision point checks:
  - **If the settlement is successful and not expired,** proceed to **non-reserved payment complete.**
  - **If the settlement is not successful,** the process reverts to handle transaction failure.

7. Process reserved settlement:

- After completing the non-reserved settlement, it moves to **reserved settlement**.
- A final decision checks if the reserved settlement was successful.
  - **If YES,** The process concludes with primary payment complete.
  - **If NO,** The system escalates to manual intervention for resolution.



Transit settlement events - LSA

Matching process

1. Settlement submission by agent:

- The process starts when an **agent** submits a settlement request to the system.
- The settlement is processed through the **UK RTGS SO Node**, and then it interacts with **DLT.**

2. Matching confirmation:

- If a settlement request achieves a **partial match**, it proceeds to further settlement matching in the **DLT** system.
- Once the **Settlement Matching Confirmed** status is reached, it is queued into a batch slot.

Payment process

3. Chain creation:

- For each chain in the payment process, chains are created and ordered to optimise liquidity and reduce settlement failures.

4. Funds reservation:

- The system attempts to reserve funds for the settlement.
- If the reservation fails due to insufficient liquidity, the system checks if the settlement can be queued into the next slot.
- If no slot is available, it will transition to insufficient liquidity and fail.
- If successful, the **settlement reservation complete** status is achieved, and the process moves forward.

5. Non-reserved settlement:

- The system handles **non-reserved settlement** requests:
  - If **successful and not expired**, the flow transitions to **non-reserved payment complete**.
  - If **unsuccessful**, the system cancels reservations (via **CAMT.049**) and can loop back for a requeue of the settlement process.

6. Reserved settlement:

- After completing the non-reserved settlement, the process advances to **reserved settlement**.
- A decision point checks if the reserved settlement is successful:
  - **If YES,** The flow ends with **reserved payment complete**.
  - **If NO,** It escalates to **manual intervention**, to undo the payment done in the secondary settlement.

Transit Settlement Events (LSA): gross settlement with offsetting

Matching Process

1. Settlement submission by agents:

- Two agents initiate the process by submitting a settlement request.
- These requests are queued into batch slots within their respective **SO Nodes**.

2. Settlement matching:

- When matched, the settlement matching is confirmed, and they are queued into the next batch slot.

Negotiation process

3. Negotiation proposal:

- The UK RTGS SO Node and the EXP1 SO Node submit **negotiation proposals** to the **DLT**.
- The **DLT** consolidates these proposals, and when ready, a **negotiation result** is returned to both nodes. This list will contain the union of the results.

4. Funds reservation:

- Once the negotiation process is complete, both nodes attempt a **funds reservation** using the **CAMT.103** message format.
- A reservation result is generated based on the availability of funds and is communicated back to the nodes.

5. Consensus check:

- The system checks if there is **consensus** between both nodes regarding maximum fund reserved:
  - **If NO**, the system issues a **Cancel/Modify Reservation** request (via **CAMT.049**), by using the removal algorithm, and submits a new result to retry.
  - **If YES**, the process proceeds to the **Payment Process**.

Payment process

6. Execute Payment:

- Once consensus is achieved, the systems proceed to payment execution using the PACS.029 message format.
- Payments are processed in two stages:
  - Batch Secondary Complete: The secondary settlement payment of the batch is finalised.
  - Batch Primary Complete: The primary settlement payment of the batch is finalised.

7. Final settlement:

- The settlement concludes when payments are fully executed on both sides, marking the end of the dual LSA process.

RTGS emulator

The RTGS emulator was built to enable rapid prototyping and adjustments throughout the life cycle of the project. The emulator, built on the python flask framework, accepts API calls from the SO representing reservation and payment instructions. At its core, the emulator provides support for tracking balances, enforcing a rule set on messages received, and implementing instructions received from banks or an SO.

The emulator enabled multiple ISO, such as API calls, to be processed. Initially, it was envisioned to support basic reservations and payment instructions. As the project evolved, it was realised that enhancements to these basic functions would enable greater functionality within the SO.

Supported Messages

CAMT.103

A CAMT.103 message enables the placement of a reservation or 'earmark' onto an account.

A CAMT.103 message will do the following:

- Check the validity of the message fields.
- Validate that the accounts exist.
- Validate the message_sender_bic is authorised as an instructing party for the account_iban.
- Check the start_date_time, which denotes the time an earmark request should start. Past dates will be rejected via a camt.047 message with status code RJCT, dates for the same day will be processed immediately, and future dated items will be rejected.
- If a reservation_id is provided, the system will use it. If not, the system will assign a uuid4 string to the reservations.
- The system then validates the funding position of account_iban and places an earmark on the account. An earmark reduces the available liquidity of the account.
- Note: If a request for an earmark exceeding the accounts balance is made, the system will return a partial earmark. It will send an acceptance message, with status PARTIAL_RESERVATION and an amount representing the value that was reserved.
- Finally, the system will send confirmation of earmark messages using a CAMT.047 message.

CAMT.048

A CAMT.048 message enables the modification of the amount of an existing reservation. It will overwrite the reservation value and reset the start time. The reservation ID and account_iban must match an existing valid reservation (that is, not used or timed out). A CAMT.048 will trigger a CAMT.047 confirmation message as a reply.

CAMT.049

A CAMT.049 message enables the cancelation of an earmark that has been placed.

PACS.009

A PACS.009 message enables the movement of funds from one account to another.

A PACS.009 message will do the following:

- Check the validity of the message fields.
- Validate that the accounts exist.
- Validate that the message_sender_bic is authorised as an instructing party for the debtor_agent_bic.
- Check the settlement date, Past settlement dates will be rejected via a PAIN.002 message with the status code RJCT. Settlement dates for the same day will be processed immediately, and settlement dates for a future day will be reflected via a PAIN.002 message.
- If "end_to_end_id" is populated, the system will follow the steps below, if not it will simply transfer funds between the accounts.
  - Validate the end_to_end_id is matched to an existing reservation ID that is of status new.
  - Validate the amount is less than or equal to the reserved amount, and that the senders BIC matches the reservation's debtor agent.
  - Return a PAIN.002 message with either an ACCP or RJCT message (RJCT messages will contain a reason for rejection).

PACS.029

PACS.029 enables net settlement of transactions. It requires that any bank with a negative net position has a valid reservation placed prior to the receipt of the pacs.029.

To use this message type, a message sender must:

- calculate the reservation needs for each bank.
- for any bank that needs a reservation, place only one for the entire amount. (Banks with a positive net position do not need reservations.)
- send a pacs.029. All outgoing transactions for a bank should reference the reservation ID in "reference". If the same bank has multiple reservation IDs, the request will be rejected.

The system will:

- validate that the same reference is used for each settlement_request made for a settlement_agent.
- calculate the net position for each settlement_agent (sender) and participant (receiver).
- for any settlement_agent (sender) in the array of settlement_requests, if their net position is negative:
  - check that the net position is less than or equal to the reservation amount denoted by reference.
  - update the net transfers for all banks.
  - reduce the reserved amount of any net senders.

Data structures

Each SO uses three different databases, all using PostgreSQL:

- **blockchain** – stores the addresses and keys for the interactions with the smart contracts.
- **Eventeum** – stores information for the filters, blocks processed, and transaction monitoring used by Eventeum. (https://github.com/eventeum/eventeum).
- **Meridianfx** – stores all information needed for the SO to function and keep track of state changes, for example, on settlements, agents, clearing systems and process data.

Some of the most relevant tables used in the meridianfx database include:

- agent – stores agent (bank) data.
- clearing_system – stores clearing system information.
- dlt_account – stores the DLT account information for the clearing system.
- leg_settlement – stores leg data for settlements.
- settlement – stores settlement information.

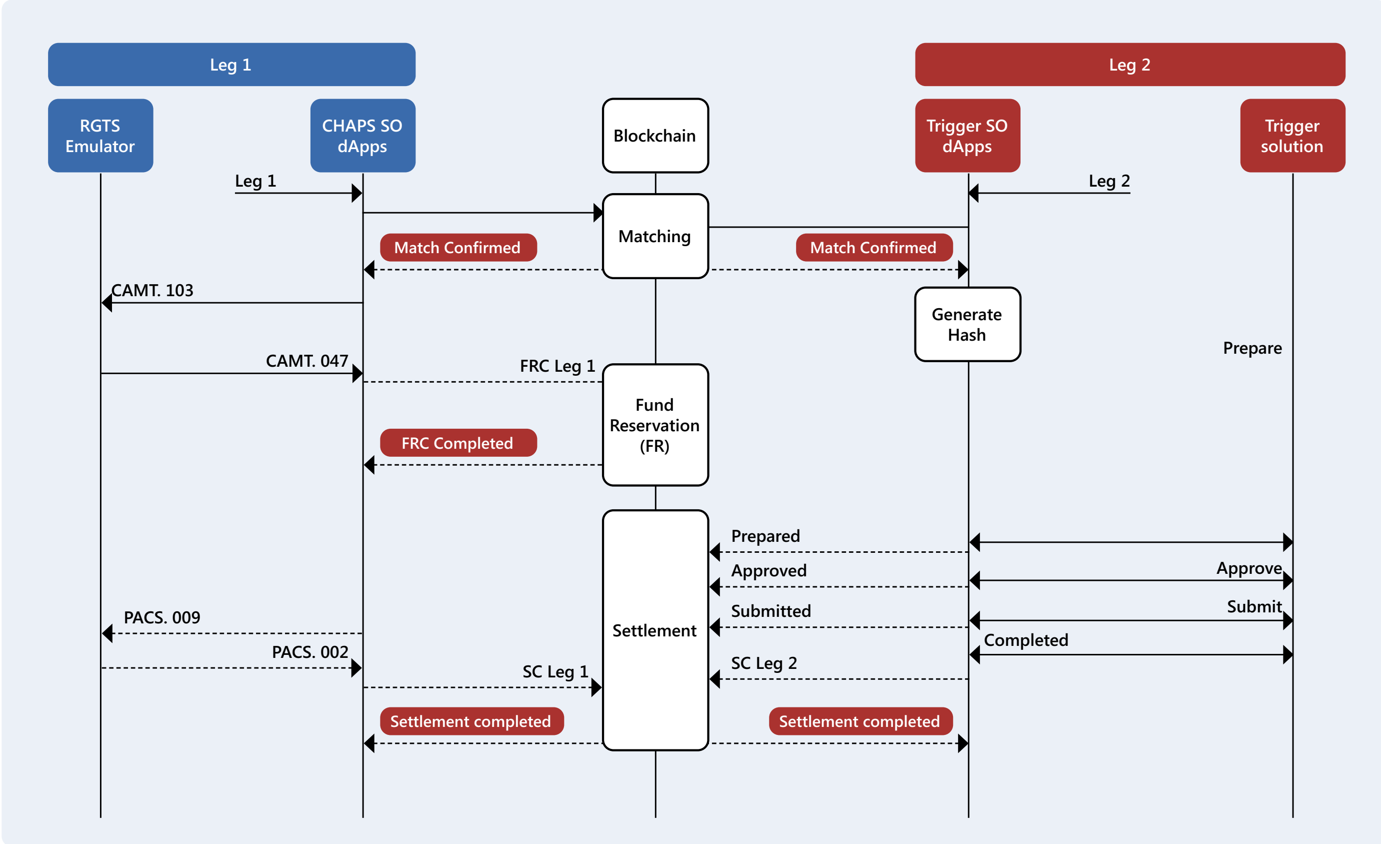
Integration

The SO orchestrates payments across different external systems, ensuring that transactions involving two or more parties are completed consistently. These external systems vary in their integration capabilities and transaction life cycles.

To manage this complexity, the SO employs a state machine–driven workflow. This workflow guides the SO through a series of states, triggering API calls to the relevant external systems at each step. The SO awaits responses from these systems before progressing to the next state.

An integration layer within the SO handles the unique characteristics of each external system. The integration layer implements different adapters to communicate with external systems. These adapters are responsible for constructing the message payload, handling the API security and parsing the responses based on the capabilities provided by the external system.

Figure 8: Trigger Integration



RTGS emulator integration

As described in this document, the RTGS emulator exposes several API endpoints to perform various operations, such as funds reservation and settlement. The SO node integrates with the RTGS emulator by invoking those APIs. Like the other integrations, message payload formatting, generation and response parsing are handled by the RTGS adapter on the integration layer.

Trigger integration

The SO integrates with the Trigger Solution using the REST APIs offered by Deutsche Bundesbank. The Trigger Solution enables APIs to create and interact with payment instructions as well as to query the statuses. The SO creates a payment instruction, follows its life-cycle process and periodically calls the trigger to query the result.

Upon the receipt of a success or failure response, the workflow moves to the next state and the next action is processed. The Trigger Solution has an approve function that basically translates the approval of the payment by the payer bank into its literal meaning.

The SO calls the approve function on behalf of the payer bank and digitally signs the request.

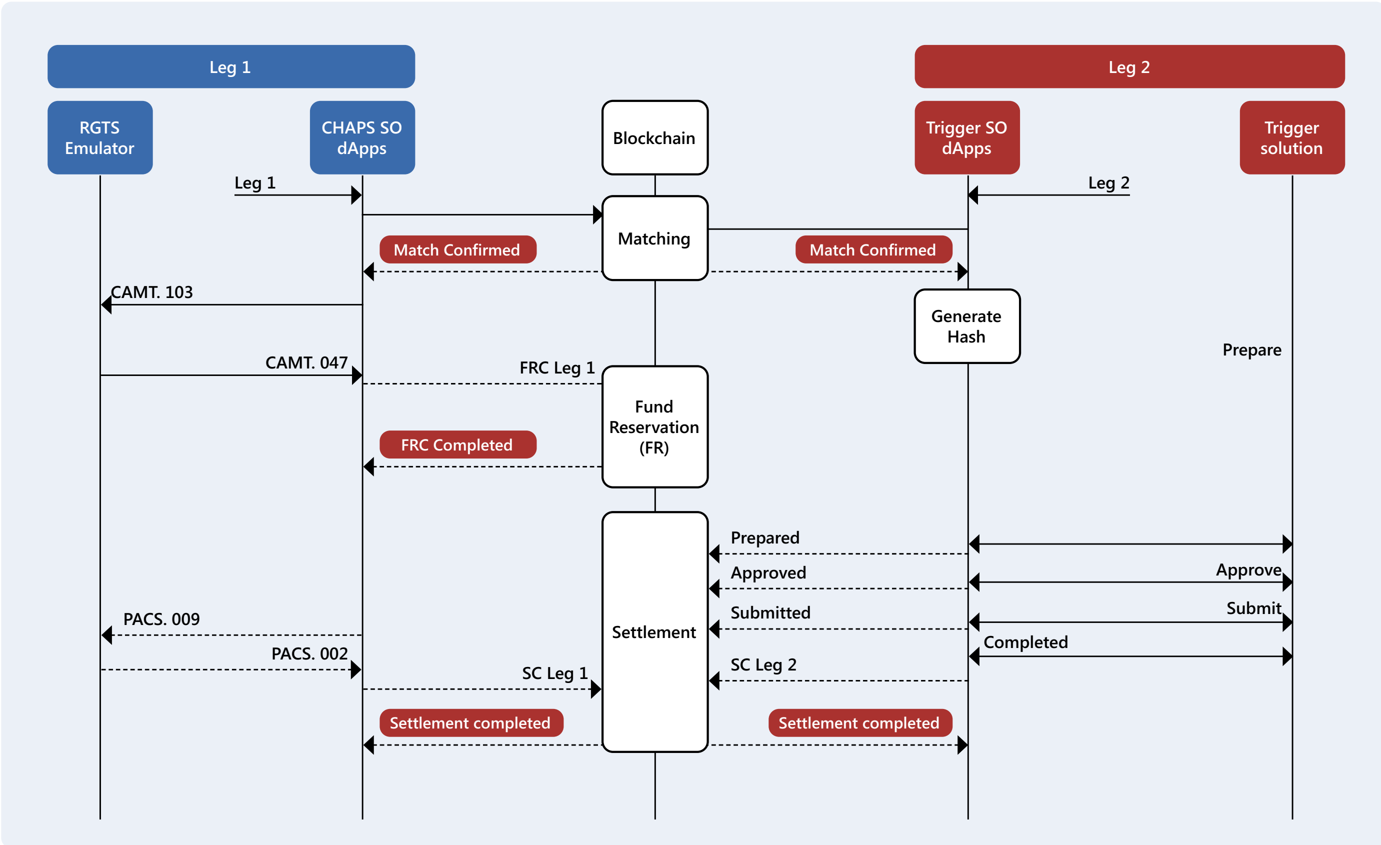
To settle a payment via the Trigger Solution, SO node makes three API calls to the Trigger Solution. The APIs use the create, approve and submit endpoints. Upon success, the Trigger Solution returns a complete status back to the SO node.

TIPS Hash-Link integration

The SO integrates with TIPS Hash-Link through ReST like APIs exposed by the API gateway component. Interaction between the SO node and the TIPS Hash-Link is based on an asynchronous model. The SO sends requests to the API gateway, the API gateway handles the processing of the request and returns the responses to the SO's designated endpoint. The body of each HTTP request sent in these interactions is composed of an ISO20022-compliant XML message required by the TIPS Hash-Link protocol.

To synchronise a PvP payment with TIPS Hash-Link solution, the SO node makes two API calls (init and instruct-payment) to the API gateway. The responses are delivered asynchronously to the SO node's API layer (init-answer and instruct-payment-answer).

Figure 9: TIPS Hash-link integration



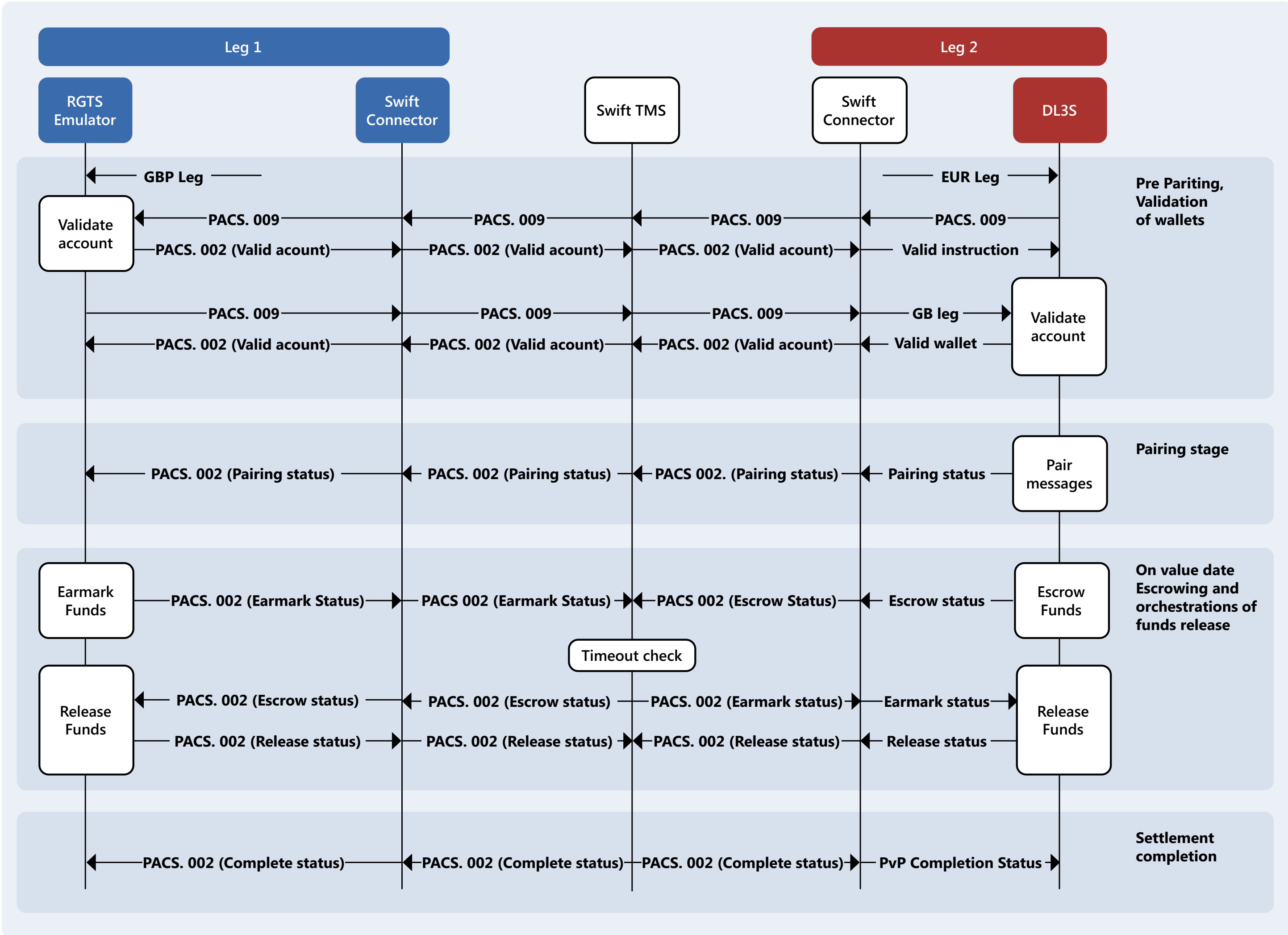
DL3S integration

In the context of DL3S, the SWIFT interoperability solution plays the role of the SO.

The solution is composed of two components: a SWIFT CBDC connector and a Transaction Management Simulator (TMS). DL3S was already integrated with this solution. DL3S implemented its own translation of ISO20022 messages to a DLT format and vice versa through the SWIFT CBDC connector. This component ensures communication between DL3S and TMS, which orchestrates the exchanges with the RTGS emulator.

The RTGS emulator integrates with the TMS. ISO messages are forwarded from the TMS to the SWIFT CBDC connector installed at the RTGS emulator end. As the RTGS emulator is a non-DLT network, the SWIFT CBDC connector is configured to bypass ISO20022 to DLT translation. Thereby, incoming ISO20022 messages are processed by the RTGS emulator, and response ISO20022 messages are forwarded back to the TMS Swift via the SWIFT CBDC connector.

Figure 10: DL3S integration



To settle a PvP payment, the escrow of funds on DL3S and earmarking of funds on the RTGS emulator happen on the value date. Once the escrow and earmark are confirmed back to the TMS by DL3S and the RTGS emulator, respectively, the TMS instructs a simultaneous release of funds on both networks. The solution, using the escrow and earmarking functionalities and orchestrated by the TMS, is therefore able to achieve synchronised, simultaneous PvP transactions.

Extension Objectives

Liquidity saving approach

To optimise settlement execution success and reduce the risk of failures, an LSA capability is integrated into the SO. This integration aims to minimise settlement failures caused by insufficient liquidity from either counterparty, ensuring more reliable transaction processing.

The SO accepts payment instructions with a priority flag:

- **High priority:** These settlements are not included in the LSA, instead they are executed immediately. If they fail, they will not be re-tried.
- **Normal priority:** These settlements are included in the LSA and retry process.

The SO supports two forms of LSA: (i) gross settlement with queuing or (ii) gross settlement with offsetting LSA modes. The LSA mode and interval can be updated through configuration.

Technical description of LSA 1: gross settlement with queuing

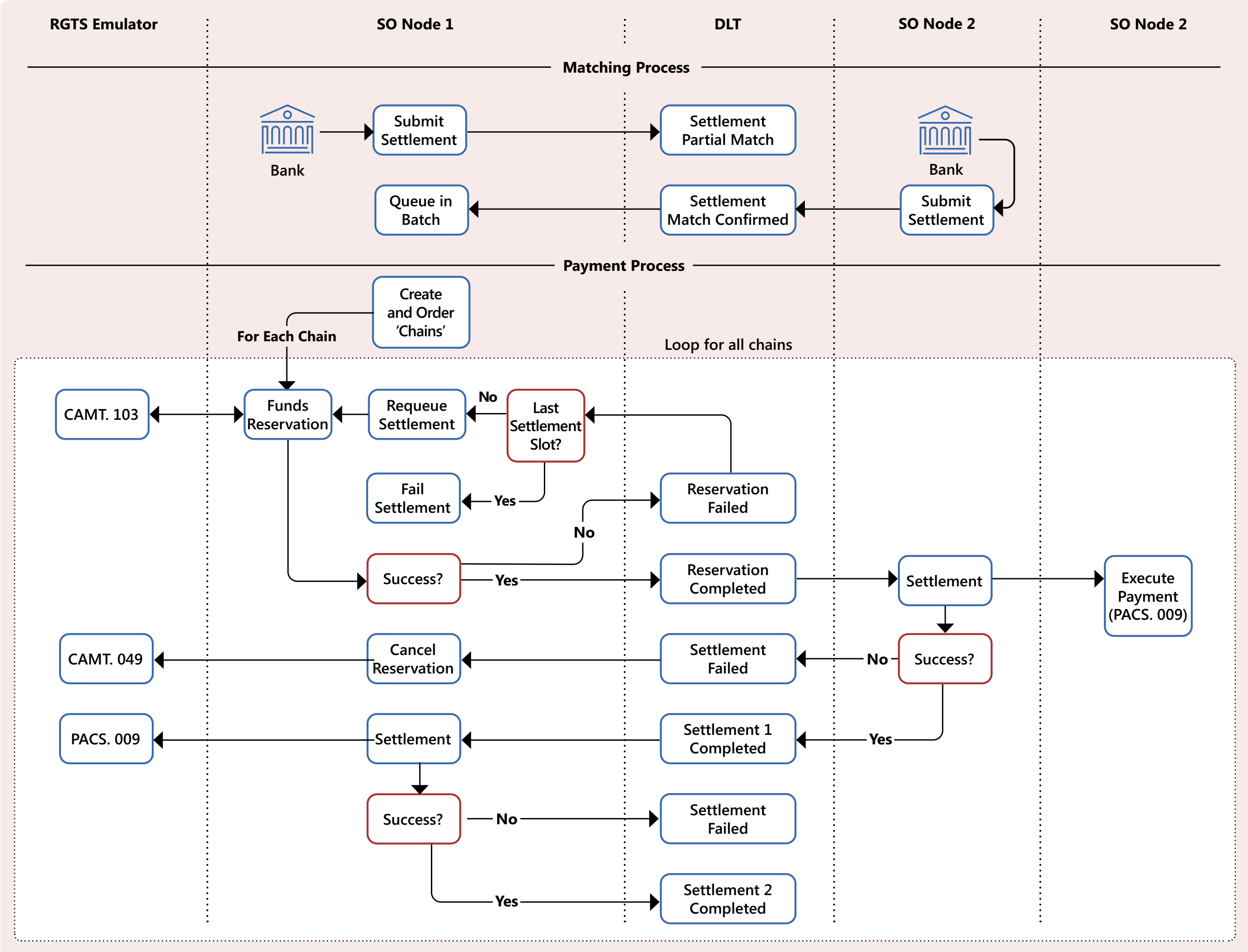
Normal priority matched settlements are queued for a configurable period before being re-sequenced using an LSA algorithm. When the LSA service triggers, the algorithm calculates the most optimistic settlements and organises them into groups called chains. These chains contain a set of settlements arranged such that each bank has a transaction paying the following bank.

The organisation criteria to determine the order is as follows:

1. outlier nodes in the graph, such as two banks only having settlements between one another.
2. nodes that do not have incoming transactions and calculating the longest possible chain of settlements.
3. chains and ordering them by length. Here, the priority is always moving the most volume (sum of settlements of the chain) in cases of length being the same or having circular chains. Circular chains are broken at the repeating node and ordered by volume as well.

Once the chains are created, an orchestrator service is called to start processing the five longest chains concurrently. The settlements inside a chain are processed sequentially using the high priority workflow. If a settlement fails, it is moved to QUEUED status and re-tried in the next LSA window. If this is not possible, the transaction fails with INSUFFICIENT_LIQUIDITY status.

Figure 11: Liquidity savings approach: gross settlement with queuing



Technical description of LSA 2: gross settlement with offsetting

To enhance performance and maximise liquidity savings of the "queuing" approach, the project explored a second option. This approach integrates LSA with netted fund reservations for each bank and facilitates multilateral settlement executions using pacs.029 messages. Minimising the number of fund reservations and settlements per batch and allowing clearing systems to execute these processes in parallel can significantly improve overall system performance while reducing liquidity needs. Additionally, calculating the net amount to reserve for each bank per batch can further optimise liquidity savings.

Similar to the “queuing” approach, the system is triggered in configurable intervals and goes through the following workflow:

- **First negotiation phase**: Each node must agree with each clearing system (counterparty) on the list of settlements to include in the next batch. At the end of this phase, if there are “X” clearing systems, each one may potentially have “X-1” lists. The agreement between two clearing systems is done by presenting both lists and selecting only settlements that are present in both lists. The resulting list is called the “original list”.
- **Funds reservation phase**: This phase works in a loop until “consensus” between the two counterparties is reached.
  - **Net calculation:** Each clearing system SO will proceed to calculate the “net amount” for every bank (“outflows – inflows” given the settlements in the batch).
  - **Fund reservations**: One bank at a time (for those whose net amount is bigger than 0), clearing systems will try to reserve the necessary funds.
  - **If** fund reservation **partially succeeds:**
    - the camt.047 response will contain the maximum reserved amount (MRA). This value will be stored, keeping in mind that the reservation will not be able to extend beyond it.
    - one or many settlements where the bank is the payer must be removed from the list to reduce the amount to reserve until it complies with the MRA returned by camt.047. The removal will be done according to the REMOVAL ALGORITHM.
    - calculate net amounts for all banks again.
    - if the net amounts of banks that have been already reserved have increased, then:
    - if the bank’s previous reservation is fully completed, it is then possible to proceed to carry out a second reservation for the difference (“fund reservations” step).
    - if the bank’s previous reservation was partially completed, it will not be possible to increase it. More settlements will need to be removed to comply with the MRA.
  - **Otherwise,** it is possible to move to the next phase. The list of settlements that remain in the batch (original list – settlements removed during “funds reservation phase”) is called the “reserved list”.
- **Second negotiation phase**: Reserved lists are exchanged with the counterparty clearing system. Both counterparties agree to process only settlements that are present in both lists. The resulting list is called the “common list”.
  - If the Common list corresponds to both reserved lists, it is possible to move on to the next phase.
  - Otherwise, it is necessary to return to the funds reservation phase, where net amounts will be recalculated and fund reservations will be adjusted:
    - If the net amounts of banks that have been already reserved have increased, it is possible to proceed to carry out a second reservation for the difference.
    - If the net amounts of banks that have been already reserved have decreased, no action is required.

- **Settlement phase**: A pacs.029 message is sent with all the settlements included in the common list.

Removal algorithm

- Settlements are ordered by amount, from largest to smaller.
- The settlement with the smallest amounts is removed.
- If the new amount does not reach the reserved amount (that is, it fails), the settlement with the second smaller amount is removed.
- If fund reservation still fails, a calculation is performed to determine if the 3rd settlement amount is greater than the 2nd settlement amount plus the 1st settlement amount.
  - **TRUE**: removing the 2nd and 1st settlements is attempted. If this fails, removing the 3rd settlement is attempted.
  - **FALSE:** removing the 3rd settlement is attempted. If this fails, removing the 2nd and 1st settlements is attempted.
- If fund reservation still fails, the process moves on to the 4th settlement, following the same principle of removing the least possible funds, which can be either all settlements combined until settlement "i", or settlement "i" itself.

Due to the nature of the possible combinations of all settlements present, the removal algorithm will first detect the number of possible combinations and find the subset of settlements that respect the available liquidity in each bank. However, two inconveniences occur:

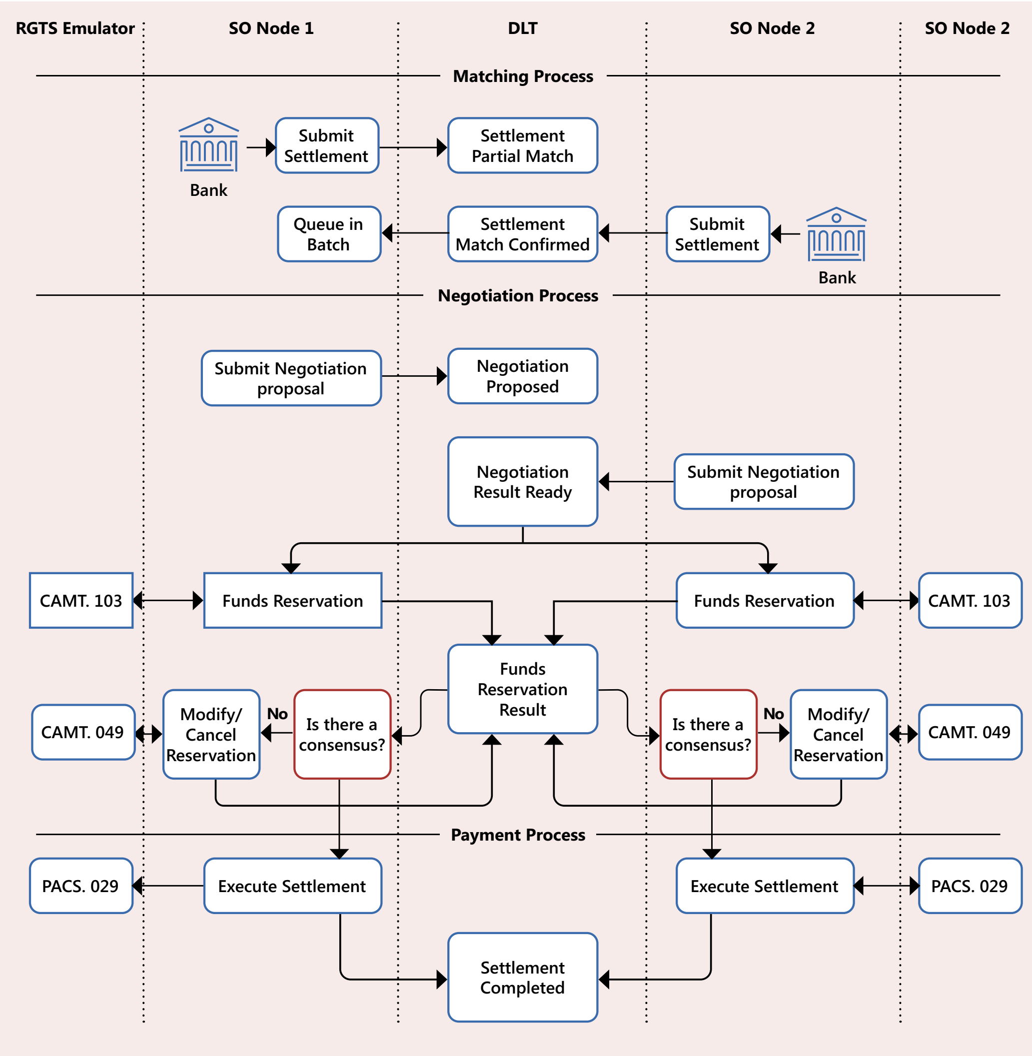
1. Removing a settlement due to a lack of liquidity affects two banks: the creditor side and debtor side.
2. The available liquidity of each bank is not known by default.

Testing with different algorithms resolves that the best approach is calculation by approximation. Above a certain list size, the aim is to obtain the best result in exchange for obtaining a valid result in a reasonable amount of time. With successive approximations, the process attempts to obtain the best possible results, in exchange for increasing the computational cost. The strict algorithm is O(n – 2^n).

The approximation algorithm is $O(k - n^2)$, where *k* is the approximation factor to be used. Determined by the performance information from the attached tests, the most suitable parameters are as follows:

- For lists below 25 elements the strict algorithm will be used.
- For lists with 25 or more elements, the approximation algorithm will be used with an approximation factor inversely proportional to the size of the list with values between 5 and 20, always ensuring a minimum factor of 5.

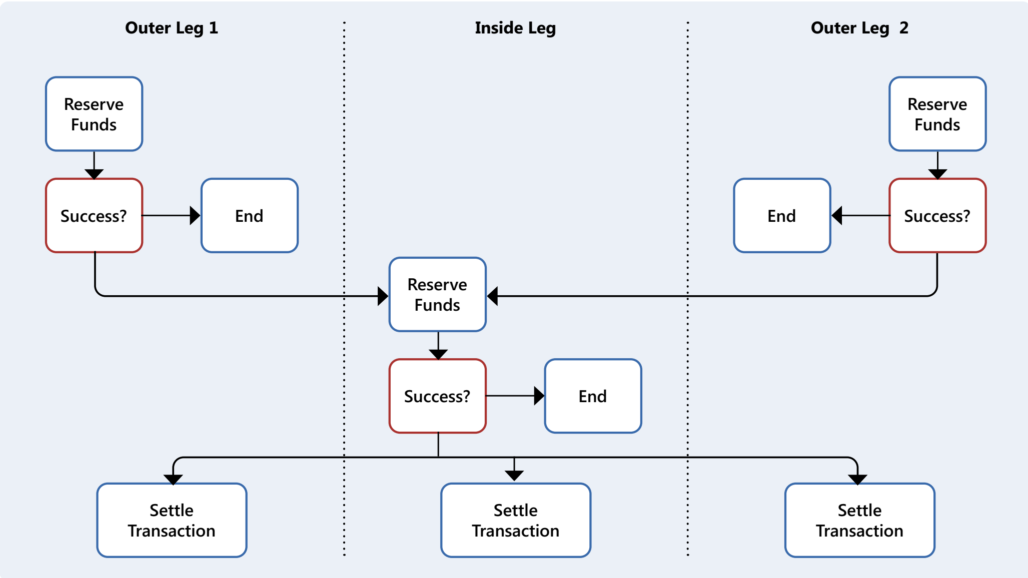
Figure 12: Liquid savings approach: gross settlement with offsetting



(PvP) + (PvP) use case

(PvP) + (PvP) extends the PvP principle to transactions involving three parties. It ensures that all payments in a multi-party transaction are executed only if every party fulfils their payment obligation. Project MFX implements the (PvP) + (PvP) as described below.

Figure 13: (PvP) + (PvP)



Initialisation

1. Settlement submission by agents:

- Agent from EXP1 submits a new (PvP) + (PvP) settlement.
- Agent from UK RTGS submits a new settlement, which matches the (PvP) + (PvP) settlement created by EXP1. This settlement does not require the agent to know it will be a part of the (PvP) + (PvP).
- Agent from UK RTGS submits a new (PvP) + (PvP) settlement, with the (PvP) + (PvP) ID of other settlements.
- Agent from EXP2 submits a new settlement, which matches the previous (PvP) + (PvP) settlement. This settlement does not require the agent to know that it will be a part of the (PvP) + (PvP).
- The DLT will emit the (PvP) + (PvP) partial matched and (PvP) + (PvP) settlement confirmed events, which confirm that all settlements are now linked and matched.

Reservation process

2. Outside nodes:

- EXP1 and EXP2 nodes will make reservations for the settlement in their system, using the CAMT.103 message.
- Depending on whether it is successful:
  - If **YES**, the fund reservation outside event is emitted.
  - If **NO**, the fund reservation outside failed event is emitted.
- If any of them fail, the process is aborted, and the settlement is marked as failed.

3. Inside node:

- UK RTGS will make the reservation once both outside nodes have completed the reservation, using the CAMT.103 message.
- Depending on whether it is successful:
  - If **YES**, the full fund reservation event is emitted.
  - If **NO**, the fund reservation inside failed event is emitted.

Payment process

4. Execute payment:

- Once all nodes have completed their reservations, each of them proceeds to execute the payment:
  - Outside nodes will use the PACS.009 message format.
  - The Inside node will use the PACS.029 message format.
- When all payments are complete, the settlements are marked as completed on all nodes.

Disclaimer: Figure 13 shows that the order of the matching and initialisation of settlements does not matter to the outcome, as they will be matched all the same.

BIS Innovation Hub

Project Meridian FX
Exploring synchronised settlement in FX

bis.org