QUANT

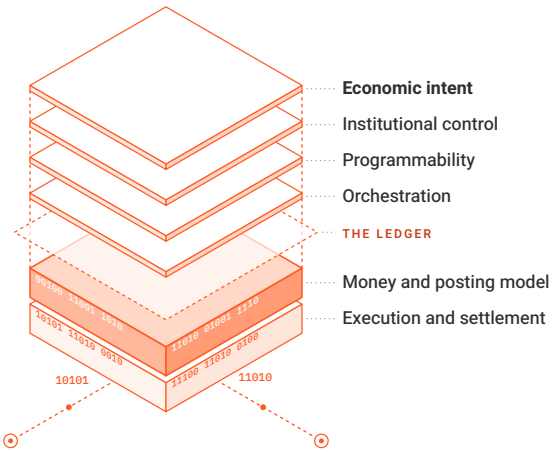
A QUANT WHITEPAPER ON TOKENISED DEPOSITS

Above the Ledger

Economic intent and what a commercial bank builds on tokenised deposits

Lenna Russ
Chief Commercial Officer, Quant · September 2026

Read online at quant.network/perspectives/above-the-ledger.html



Tokenised deposits can make commercial bank money programmable through several valid architectures. The enduring opportunity is to preserve the customer's economic instruction as money moves across accounts, ledgers and settlement systems.

This paper is written for commercial banks. It sets out what to build on the tokenised deposit networks now being put in place, distinguishes the record of value from the purpose of a transaction and gives a framework for turning tokenised deposits into banking products.

Executive summary

A tokenised deposit creates value when it helps a bank deliver a useful financial outcome. Recording the deposit on a new ledger makes that outcome possible. The service is defined by what the bank does with the record.

The question for a commercial bank is what to do now that the networks exist. Four are being built: Great British Tokenised Deposit (GBTD) in the UK, where seven banks have completed the first live customer transactions on a shared platform built by Quant; The Clearing House's On-Chain Money Initiative in the US, which has selected Quant to power it, opening to participating institutions in the first half of 2027; the joint initiative of Canada's six largest banks, announced on 22 September 2026; and in Europe the Commercial Bank Money Token (CBMT) initiative of the German banking industry, now in pilot. The product on each of them is the bank's to build.

A corporate customer wants an invoice paid under an agreed policy, liquidity available before a deadline or cash exchanged for an asset. Each requires a coordinated transaction with a clear meaning and an accountable institution behind it.

The starting point for programmable banking is economic intent. Intent captures the purpose, parties, authority, conditions, dependencies, outcome and failure behaviour of a transaction. It gives the bank a durable description of the product while allowing the underlying money and execution infrastructure to evolve. **It also makes the infrastructure cumulative: each new use case reuses the programmability, interoperability and orchestration the first one put in place.**

Whether the tokenised deposit is the book of record, mirrors one held in the existing banking environment or sits in a pooled arrangement, the argument holds. The choice of posting model follows the product, the legal arrangements and the bank's operating capability.

Research published by the Bank for International Settlements gives the argument its starting point. Its analysis of public blockchain data shows how little a technical record reveals about the economic activity behind it. **A shared ledger records value. The instruction that gives a movement its meaning has to live above it, with the bank.** The institutional implications drawn here are Quant's analysis.[1]

> A tokenised ledger can be the book of record. The bank still needs an explicit account of what the transaction is intended to achieve and the conditions under which it may proceed.

Commercial value comes in several forms: chargeable automation, stronger payment and FX relationships, better use of liquidity and lower operational effort. The banks that capture it will be the ones that treat the tokenised deposit as the start of a product, with the customer's instruction, the controls and the evidence carried alongside it.

The framework in this paper separates six concerns: economic intent, institutional control, programmability, orchestration, money and posting model, and execution and settlement. These are responsibilities for the bank to make explicit; how many systems carry them is a design choice.

Banks can begin inside their own group and join a network when they want reach and ready-made counterparties. Either way, the product rests on a handful of design decisions: where the deposit is recorded, which controls apply, who owns an exception, what happens when a condition fails and how the service is run day to day. Quant works through them with the bank, and that first product is what turns tokenised deposits into revenue.

What the BIS evidence establishes

A ledger records what moved. It rarely records why.

A tokenised deposit ledger exposes a transaction identifier, an amount and a timestamp. Which customer obligation those discharged is a separate fact. Public blockchain measurement shows how far apart the two can be.

BIS Working Paper 1377 measures activity on Bitcoin, Ethereum and Tron. Monthly estimates of Bitcoin transfer value differ by up to sixfold depending on how self-transfers and change are treated. Reused token symbols and the proliferation of contracts complicate classification on Ethereum, and the same stablecoin serves different purposes on different chains. The authors conclude that on-chain indicators approximate economic activity and have to be read with that in mind.[1]

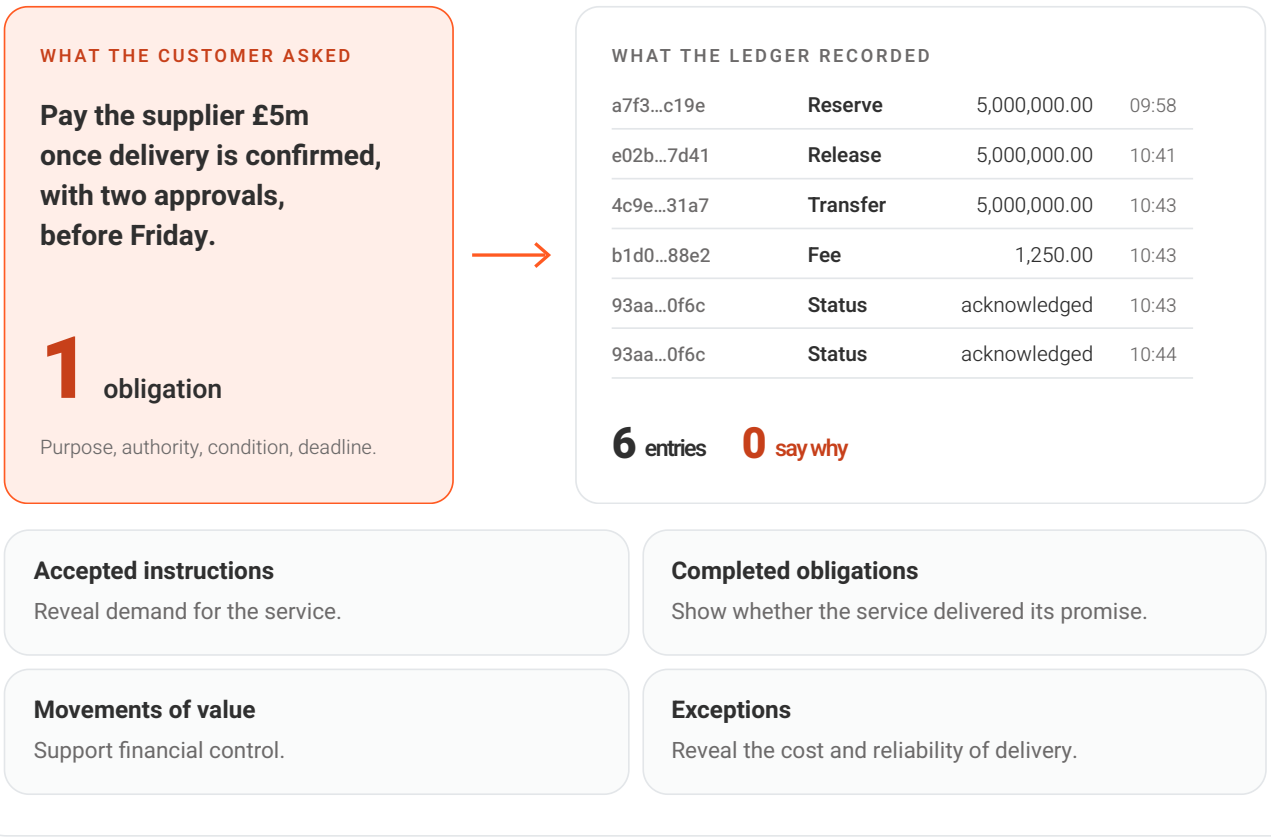
The bank must define the unit it is measuring

For a bank considering programmable money, the first implication is analytical. What counts as a transaction? An instruction may produce a reservation, a transfer, a release and several status events. Counting all of them as completed customer payments would confuse processing activity with business activity. Counting only the final transfer could conceal the operational work required to complete it.

A useful management view therefore distinguishes accepted instructions, completed obligations, movements of value and exceptions. These measures answer different questions. A single transaction-volume headline cannot replace all four.

FIGURE 1

The ledger counts six; the customer counts one



Counting ledger entries counts the plumbing. The bank's unit is the obligation, and the ledger cannot say which entries served it. The four measures beneath answer four different questions about the same instruction. This instruction is followed through the paper.

The inference for a bank

Banks hold information a public-chain analyst does not: identified customers, mandates, account relationships and the commercial instruction. The opportunity is to keep those connections attached as the money moves. A later enquiry then has an answer: which obligation the movement fulfilled and under whose authority.

The inference is Quant's and it is narrow. The meaning of a technical record depends on assumptions and context supplied from outside it. **A bank can design that context into its service, whichever ledger records the money and whoever operates it.**

Turning context into institutional control

Economic context changes the actions a bank may take.

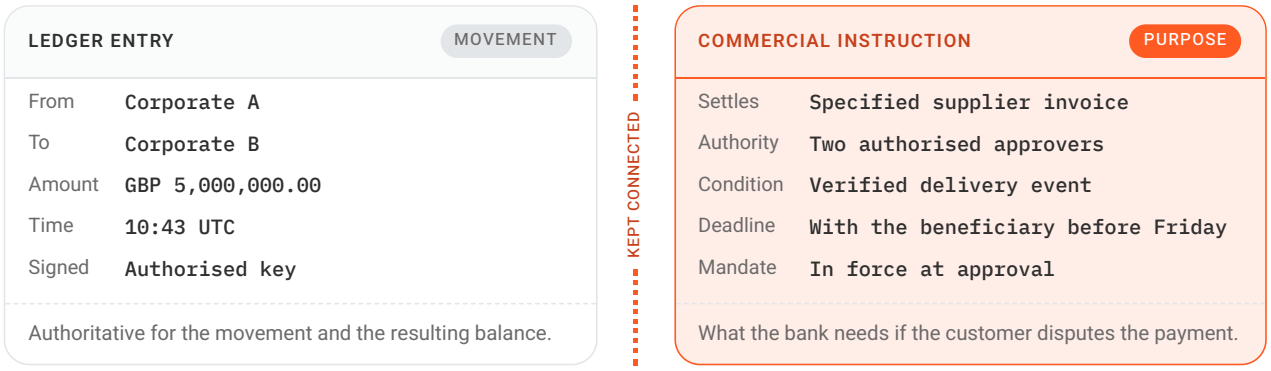
The same amount sent to the same beneficiary can require a different treatment when it represents payroll, collateral, an investment or a supplier payment. Purpose alone does not determine every control, but it helps the bank identify the relevant mandate, product terms, dependencies and evidence.

A record of movement and a record of purpose

Take the transfer row from Figure 1. The ledger entry shows that Corporate A transferred £5 million to Corporate B at 10:43 UTC. That entry can be authoritative for the movement and resulting balance. The associated commercial instruction might say that the payment settles a specified invoice, requires two authorised approvers, depends on a verified delivery event and must reach the beneficiary before an agreed deadline.

FIGURE 2

What a single entry cannot tell you



Illustrative. A technically valid signature and a valid business authority are related, but distinct, propositions.

Those facts should remain connected. If the customer disputes the payment, the bank needs more than proof that an authorised key signed a transfer. It needs to identify the mandate in force, the instruction that was approved and the evidence on which a condition was treated as satisfied.

Evidence has an owner

Programmability can act on a delivery confirmation or an asset-availability signal. It cannot make the underlying statement true. The bank must know who supplies the evidence, how current it is, what scope it covers and what happens if it is withdrawn or disputed. A workflow that consumes unreliable evidence can execute perfectly and still deliver the wrong commercial result.

There is also a privacy boundary. Preserving context does not require distributing every invoice, identity document or approval record to every participant. A system can retain an appropriate reference and evidence of a decision while keeping sensitive material within an authorised environment. The level of disclosure should follow the product's needs and the parties' responsibilities.

Make completion meaningful

The product should specify the outcome the customer can rely on. An instruction accepted for processing, a value transfer recorded and funds finally available to the beneficiary are different milestones. Calling all three “complete” creates ambiguity for the customer, operations team and balance-sheet owner.

For management, this distinction improves the quality of the business case. A faster technical confirmation has limited value if the commercial obligation still waits for another system or operating window. Better coordination reduces uncertainty even when settlement time stays the same. Banks should measure the customer's outcome alongside the performance of the component systems.

Economic intent is therefore a practical control concept. It ties authority and evidence to a defined outcome, providing the context within which the bank can safely automate.

02 PART II

Economic intent and the ledger beneath it

The anatomy of economic intent

Economic intent is a structured description of the financial outcome a customer or bank seeks.

It connects the instruction to the controls and dependencies needed to fulfil it. The following seven elements provide a practical framework for product discussion. They are a conceptual model; the software schema and the account terms come later.

FIGURE 3

Seven elements, applied to a conditional payment

Pay a supplier £5 million after delivery is confirmed — Illustrative instruction

ELEMENT	QUESTION THE BANK MUST ANSWER	IN THIS INSTRUCTION
1 **Purpose**	Which obligation or business objective does this instruction serve?	`Discharge the identified supplier invoice`
2 **Parties**	Who owns the value, acts for the customer and receives the benefit?	`Corporate payer, its approvers, the named beneficiary`
3 **Authority**	Which mandate, entitlement and approval permit the action?	`Customer's approval policy plus the bank's own controls`
4 **Conditions**	What evidence must be valid before value may move?	`Delivery confirmed before the instruction expires`
5 **Dependencies**	Which cash, asset or information events must occur with it?	`External delivery event from a known source`
6 **Outcome**	What result fulfils the instruction and can be reported as complete?	`Funds received by the beneficiary`
7 **Failure behaviour**	What is permitted when the intended outcome cannot be achieved?	`On expiry, close the instruction and handle any reservation`

> The customer is asking for an authorised obligation to be discharged: a payment with a purpose, an authority and a deadline.

An illustrative conditional payment

A corporate asks its bank to pay a supplier £5 million after delivery is confirmed. The instruction identifies the invoice and beneficiary, requires the customer's specified approvals and expires if the delivery condition is not met in time. The bank applies its own controls as well as the customer's mandate.

The delivery event creates a dependency on external information. The expiry introduces a time boundary. The approval policy establishes authority, while the required receipt of funds defines completion. If a condition fails before execution, the bank needs an agreed way to close the instruction and deal with any reservation. Once execution has occurred, the available responses change.

Intent needs a scope

An instruction should not imply unlimited discretion. The service must define which accounts, amounts, currencies, time windows and counterparties are covered. A standing liquidity policy may authorise repeated actions within limits. A one-off payment authorisation has a different scope. Treating them as interchangeable would weaken control even if both use the same execution capability.

Changes also matter. If the beneficiary, amount or underlying condition changes after approval, the bank must determine whether the existing authority still applies. Programmability should make this decision explicit. Approval granted to one instruction does not carry silently into another.

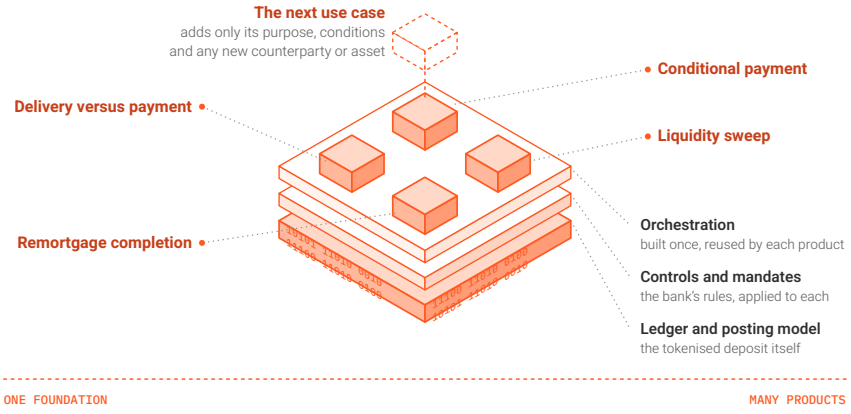
The value of the framework is shared understanding. Product, technology, operations and risk teams can discuss the same transaction before debating a ledger or integration pattern. That reduces the chance that a technically successful implementation solves a different problem from the one the customer brought to the bank.

One infrastructure, many products

The seven elements are the same whatever the product. A conditional supplier payment, a liquidity sweep, a remortgage completion and a delivery-versus-payment exchange differ in their purpose, conditions and outcome, and build on the same orchestration and controls, whichever ledger sits beneath them. **The first use case builds that foundation. Each one after it reuses it and adds only what is new**: its purpose, its conditions and any new counterparty or asset. Great British Tokenised Deposit shows the pattern: remortgages and a marketplace purchase first, with digital-asset settlement to follow on the same platform.[2]

FIGURE 4

One foundation, many products



The first use case lays the foundation. Every product after it is a new economic intent on the same rails.

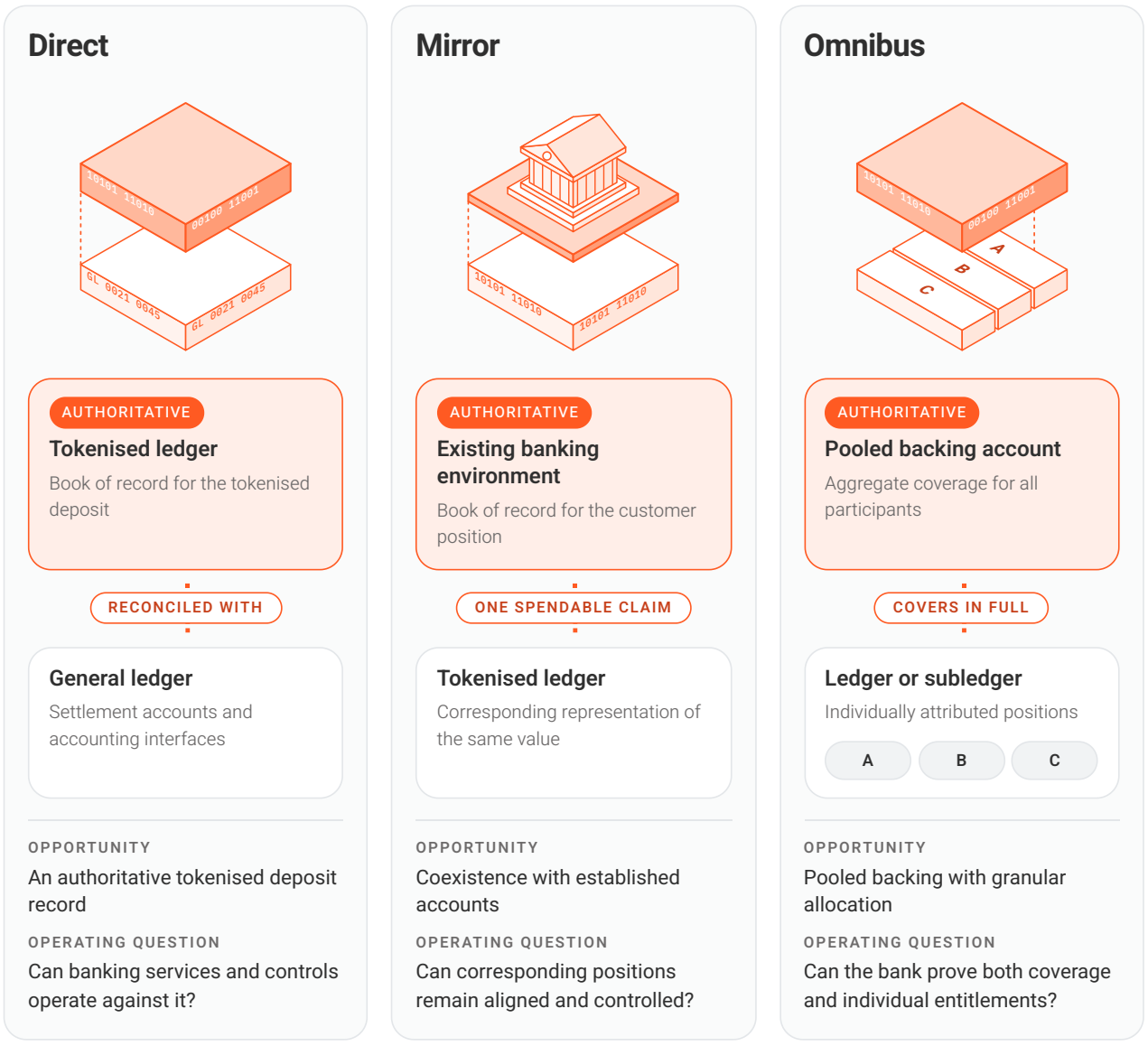
Three valid posting models

Posting architecture determines where positions are authoritative and how the bank maintains financial integrity.

The three patterns below differ in where the authoritative position sits. Economic intent sits above all of them.

FIGURE 5

Where the authoritative position sits



All three are valid designs. A bank may run more than one at once, and none is a stage on the way to another.

Each model carries the operating question shown in the figure, and none removes the need for alignment with settlement accounts, external systems and the general ledger.

Choose for the proposition

The objective is to keep the customer proposition intelligible across these choices. An authorised conditional payment retains its economic purpose whichever route reserves, records or transfers the money.

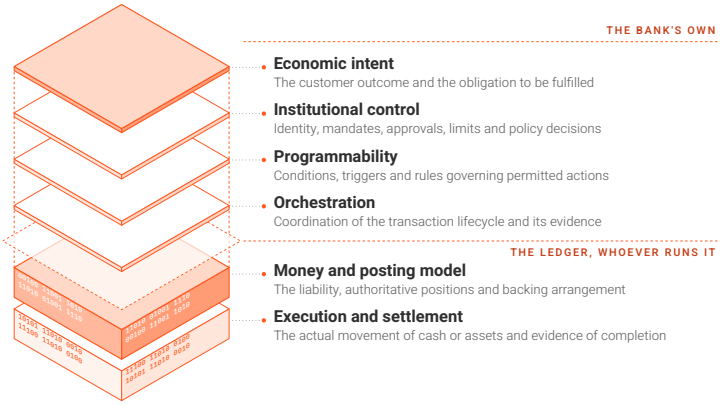
Six architectural responsibilities

Above the Ledger describes a separation of concerns.

It allows banks to distinguish the economic service from the infrastructure that records and moves value, while retaining the constraints of that infrastructure. The six responsibilities below can be implemented together or distributed across several systems.

FIGURE 6

The service and the ledger beneath it



Four responsibilities belong to the bank. Two sit with the ledger, whoever runs it.

Separation preserves accountability

A condition may be expressed in a financial workflow, but the bank remains responsible for deciding which conditions are sufficient. An orchestration service may coordinate several systems, but it does not acquire the authority to rewrite their financial records. A ledger may enforce restrictions, but those restrictions still need to correspond to the product's mandate and operating rules.

“Above” describes responsibility, and functions can run on-chain, off-chain or inside the bank's core. **The transaction record and money record can reside on the same technology. Their responsibilities remain distinct.** A direct tokenised ledger can be authoritative for the deposit while associated services retain

the commercial context and coordinate dependent actions.

Interoperability includes meaning

Connecting two systems is only the beginning. One may report acceptance while another reports settlement. One may support a reliable reservation while another offers only a request to cancel. An integration must preserve these distinctions or the product will inherit false assumptions about what its infrastructure can guarantee.

For the bank, a useful architecture makes those differences visible at the point where a product is approved. The commercial promise should be no stronger than the weakest relevant dependency. Where a route cannot support the required outcome, the bank chooses a different route or scopes the product to fit.

How Quant approaches the problem

Quant's PayScript® expresses programmable payment rules. Quant Flow runs programmable financial workflows inside the bank, with Tokenised Deposit Services for the deposit itself. Overledger connects banking infrastructure and distributed ledgers, and QuantNet coordinates settlement between participating institutions. Together they keep financial logic intact across execution environments.

A bank does not need to be tier one to take part. The Clearing House network, powered by Quant, gives financial institutions of all sizes a path to participate. A bank that does not yet have tokenised deposit capability can issue and move tokenised deposits through Quant's shared platform, without building new infrastructure itself. The pattern holds on any network: the rails are shared, the bank-side capability and the orchestration come from Quant, and the bank's investment goes into the product and its controls.[3]

The posting model, custody arrangements and settlement infrastructure remain the bank's decisions, and capability and scope are established for the chosen use case.

WHERE THIS IS LIVE

Great British Tokenised Deposit (GBTD) runs on shared UK infrastructure built by Quant. Its first live retail transactions were two remortgage completions, with funds locked and released automatically at completion, and a consumer marketplace purchase in which the buyer's money was released only when the goods changed hands. For a remortgage, the customer can keep earning interest on the funds until completion.[2] Capital markets come next: the participating banks plan to issue three digital bonds in the first quarter of 2027, traded and settled with tokenised deposits, and to set up the company, rulebook and governing framework that take GBTD into full production.[4]

With Quant integrated into MX.3, Murex clients gain access to tokenised deposit and digital bond issuance and settlement inside the trading, risk, position-keeping and regulatory reporting workflows they already run, with no parallel system to reconcile. Overledger connects MX.3 to public and private ledgers through a single integration. Conditional payments, automated corporate actions and multi-step settlement sequences run under the institution's own controls, and custody stays the institution's choice.[5]

03 PART III

Lifecycle and infrastructure choices

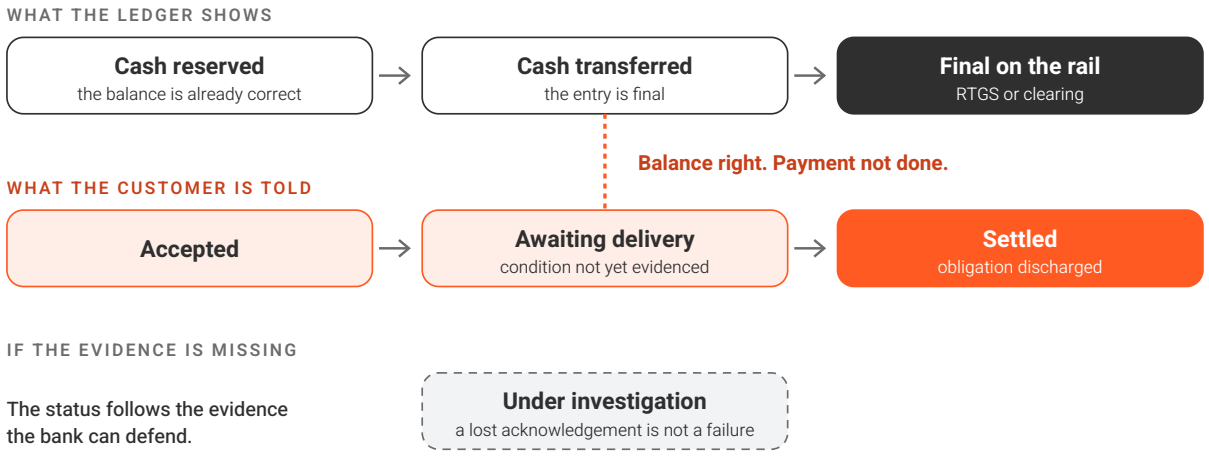
From a transfer to a transaction lifecycle

Institutional transactions extend over time.

An instruction can be received, checked, authorised, reserved, submitted and settled at different moments. These are useful distinctions even when successful execution is fast. They determine what the bank may report to the customer and which actions remain safe.

FIGURE 7

Two different facts: where the money is, and whether the payment is done



A reserved balance is correct and the payment is still incomplete. The customer's status follows the evidence the bank holds, and a lost acknowledgement reads "under investigation" until the facts are established.

Money state and transaction state

Money state describes the relevant position: who holds value, what is available and what is restricted. Transaction state describes progress towards an economic outcome. A balance can be correct while a transaction remains incomplete. For example, cash may be reserved correctly while the bank waits for evidence that an asset is available.

The opposite risk also exists. A customer-facing service may display completion while the financial records have not reached the required condition. This can lead the customer to release goods, use a balance or close an obligation prematurely. The service must connect its status language to evidence it can defend.

Completion has several dimensions

An execution confirmation establishes that a system accepted or processed a request. Settlement evidence establishes a different fact. Legal finality depends on the relevant arrangement and cannot be assumed solely from a technical confirmation. Operational closure often needs reporting or reconciliation after funds have settled.

Customers need only a few of these distinctions, stated accurately. “Accepted”, “awaiting a condition”, “settled” and “under investigation” can communicate useful differences without exposing the system's internal design. A clear status can itself be part of the product's value when customers currently rely on enquiries to understand a payment.

The lifecycle crosses organisational boundaries

An invoice platform, bank, payment network and beneficiary institution each know only part of the transaction. No participant should assume that another party's silence proves failure or that a transport acknowledgement proves funds arrived. Shared references and agreed status meanings help participants coordinate, but each fact still needs a responsible source.

For product leaders, this is a service-design issue. Who answers the customer's question while an instruction is pending? When should the bank notify a delay? Which commitments continue overnight or outside market hours? A ledger that is open all hours sets the question of which of screening, liquidity, customer support and connected settlement will be open with it.

The lifecycle therefore belongs in the proposition from the beginning. A bank that defines the successful outcome but leaves pending and exceptional states unspecified has described only part of the service. The missing part is often where customers experience the greatest uncertainty and where operations incur the greatest effort.

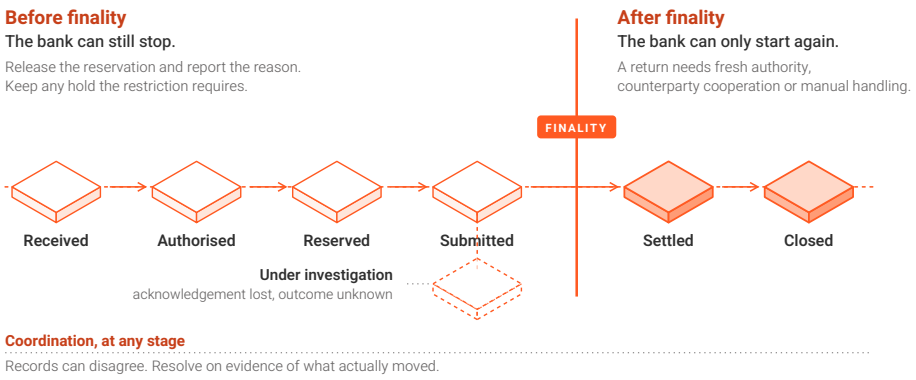
Failure behaviour is part of the product

Automation needs a defined response when the intended outcome cannot be delivered.

Three broad classes are enough to allocate business responsibility. The recovery design itself is a separate discussion.

FIGURE 8

The finality line changes what the bank can do



> Before finality the bank can stop, release the reservation and say why. After it, undoing the movement is a new transaction with its own authority. A lost acknowledgement sits between the two as an honest state until the evidence is in.

Business condition failure

The transaction may no longer be eligible to proceed. A required approval can expire, available funds may be insufficient or screening may prevent execution. The appropriate outcome is often to stop and report the reason within permitted disclosure rules. Repeating the request does not make the underlying condition valid.

Any reservation must be handled according to the applicable restriction and policy. A failed commercial condition does not always permit immediate release: a compliance restriction may require the bank to retain a hold or take another controlled action. Failure behaviour therefore follows the reason for the restriction.

Execution failure

The instruction can remain valid while a required system becomes unavailable. An interruption before submission differs from a lost acknowledgement after submission. In the latter case, the bank may not yet know whether value moved. Retrying indiscriminately could create a second economic effect; cancelling prematurely could contradict a successful settlement.

The customer promise should accommodate uncertainty. A pending investigation is an honest state where evidence is incomplete. **A timeout is not evidence that the payment failed.**

Coordination failure

Different components may hold incompatible views of the transaction. One leg may have progressed while another has not, or one record may lag behind a confirmed movement. Resolving the difference requires evidence about what actually occurred and which positions are authoritative. Making the displays agree can hide an economic discrepancy.

Respect the boundary of finality

Releasing a reservation before execution is different from recovering value after settlement. Once a movement is final under the relevant arrangements, software cannot assume a rollback will undo it. A return or compensating transaction may require fresh authority, counterparty cooperation or manual handling. The precise response depends on the product and its legal framework.

Banks should therefore assess automation by both its successful path and its controlled behaviour under uncertainty. Recovery time, unexplained differences and customer communication are commercial service measures. A production proposition is credible when its owners can explain what happens when the intended outcome is unavailable.

Preserve the product through infrastructure change

Banking infrastructure will continue to change.

A product will reach different customer groups through different rails, and most banks will run conventional accounts alongside several forms of tokenised money. The economic proposition should remain understandable across that estate.

Consider a standing instruction to maintain a minimum operating balance and sweep surplus cash within a defined group of accounts. Its purpose, authority and liquidity limits need not change merely because the bank changes where a position is recorded. The implementation may change substantially: available-balance calculation, reservation, settlement timing and reporting can all differ.

The product travels with the intent

Separating intent from execution is what lets product logic move with the bank. The customer's objective, permitted parties, limits and completion criteria stay constant. How each route fulfils and evidences them adapts to that route's hours, reservation capability and settlement rules. Before a route takes over, the bank confirms it delivers the promised outcome, or updates the promise to match.

Shared logic, governed once

Reuse is where the economics improve. An approval rule or workflow component written once serves every product that needs it, and a change made once reaches all of them. Ownership, versions and evidence that each change suits the customers using it keep that consistency deliberate, which is exactly what the bank's risk and audit teams want to see.

For customers, continuity means account terms, availability of funds, cancellation rights and reporting that stay clear as the technology underneath changes. Where a migration alters any of them, the bank handles it as a product change and tells customers so.

Each product makes the next one faster

Separation pays back with every product added. The first use case carries the cost of the orchestration, controls and integration. The second and third reuse them, so each new product reaches customers sooner and for less. A practical programme starts with the execution environments its first product needs and defines intent and controls so they move to the next environment without being rebuilt.

Commercial value in corporate banking

Programmability becomes commercially relevant when customers can buy an outcome that is difficult to obtain through isolated payments.

Banks should connect each use case to a defined customer problem, an operational response and a measurable source of value.

TREASURY AND LIQUIDITY

Fund and concentrate cash within agreed thresholds

Where the bank earns: cash-management fees and the balances that stay with the bank.

CONDITIONAL PAYMENTS

Pay suppliers on a verified business event

Where the bank earns: fee income, payment retention and less work investigating ambiguous instructions.

CROSS-BORDER AND FX

Run conversion, screening and payment as one instruction

Where the bank earns: retaining the payment and FX relationship with better status visibility.

Treasury and liquidity

Corporate cash is distributed across accounts, entities and currencies. A treasury team often sees those balances yet still relies on manual instructions to fund an operating account or concentrate surplus cash. A programmable service can apply agreed thresholds and permissions to initiate eligible movements and report their outcome.

The bank's value is a deeper cash-management relationship, service fees and the deposit balances a corporate keeps with the bank that runs its liquidity policy. Model where funds settle under the policy and what the customer will pay for the service.

Conditional corporate payments

A supplier payment can depend on a verified business event and the customer's approval policy. This can reduce repeated manual hand-offs between finance teams and the bank. It also introduces responsibility for the evidence used to satisfy the condition and for disputes when the commercial event is contested.

The bank can differentiate through a controlled payment service that fits the customer's operating process. The proposition needs a clear boundary: automating the payment condition does not turn the bank into the adjudicator of every commercial dispute between buyer and supplier.

Cross-border payments and FX

A customer's instruction may span currency conversion, liquidity, screening, payment submission and receipt. Coordinating these activities as one economic instruction can improve status visibility and reduce disconnected manual decisions. A cross-border route usually spans institutions with different hours, rules and settlement arrangements.

The bank can seek to retain the payment and FX relationship by making the whole process more useful to the customer. **Coordinating the conversion, the screening and the payment together is what makes a quoted rate deliverable within its validity**, and that is a service a corporate will pay for.

Build a business case around behaviour

For each proposition, compare the existing process with the proposed service. Identify the manual decisions removed, the decisions retained, the liquidity committed and the additional controls required. Early measures should include customer adoption, chargeable usage, support demand and financial exposure. That is a defensible investment case, and it strengthens with each product, because each one builds on the orchestration, controls and integration already in place, whatever ledger it settles on.

When money meets assets

Tokenised assets have moved faster than the cash that settles them.

A bond can move on a ledger in seconds while its payment still follows on a conventional rail, leaving one side of the trade exposed until the other arrives. The product a capital markets customer is buying is the exchange, and every obligation that follows from it.

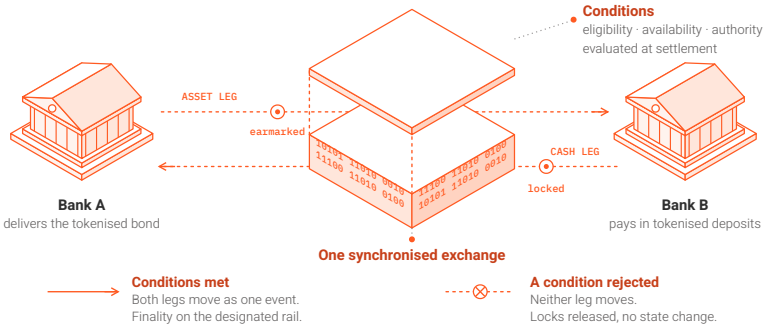
Delivery versus payment

Delivery versus payment, or DvP, ties the asset to its payment. Programmable DvP checks eligibility, availability and authority first, locks both legs, and then executes the exchange as one event or rejects it in full. Finality for each leg stays with its designated rail.

The opportunity for a bank is to provide the cash leg: to be the bank whose tokenised deposits its trading customers settle in, and to take the operational friction out of settlement for them. Cash and securities are reserved only while the exchange is pending, so they keep working until they are needed.

FIGURE 9

Delivery versus payment: two legs, one exchange



The bond is earmarked and the cash locked while the conditions are evaluated. This is the pattern Quant and Murex show live on the Discover Stage.

Payment versus payment

Payment versus payment, or PvP, applies the same principle to two currencies. Both legs exchange under agreed conditions or neither does, which removes the exposure of the party that pays first. Rate validity, currency availability and settlement windows are part of those conditions.

Repo and collateral

Repo and collateral run well past the opening settlement. Eligibility, valuation, substitution, margin calls and the unwind each change what has to happen next. Holding the instruction against the whole contract lets every one of those events run through the same rules, through to maturity or recall.

UK Finance and Oliver Wyman's *Building Digital Markets of the Future* (August 2026) puts the dependency plainly: every tokenised transaction has a cash leg, and the benefits arrive when the cash leg is as digital as the asset leg. It reports that the Digital Markets Champion's first report, in July, identified repo settled end to end on tokenised rails as the first cross-sector use case, with a live trial targeted for spring 2027.[6]

LIVE ON THE DISCOVER STAGE · MONDAY 28 SEPTEMBER, 10:30 TO 11:00

Quant and Murex show that transaction at Sibos: a USD repo against a tokenised US Treasury, with the cash leg in tokenised deposits, booked and operated entirely from MX.3. The first run takes the repo from booking through to the next-day recall. The second is rejected mid-execution and rolled back in full, with no state change on any ledger, account or inventory. In both, finality stays with the designated settlement rail.

Add the session to your calendar →

The return comes from collateral that is available when it is needed, less processing effort and more predictable funding, across the life of the trade.

Preserve the institution's operating context

Trading, risk and post-trade systems already hold the commercial context. Programmable settlement runs from them and returns every state change to the people responsible for positions and obligations, so the desk works in the system it already knows.

Quant and Murex present programmable settlement for tokenised assets at Sibos 2026, building on their joint whitepaper on synchronised delivery versus payment. That work applies the argument of this paper to capital markets. The principle is the same across payments and assets: the bank needs to coordinate an economic outcome, retain evidence of the conditions applied and account for the result across its operating environment.[7]

JOINT WHITEPAPER · QUANT × MUREX

Programmable Settlement

Synchronised delivery versus payment for tokenised assets, delivered through the trading and settlement platform banks already run.

Get the whitepaper →

Shared infrastructure and institutional reach

Networks now exist that no single bank could have built.

Great British Tokenised Deposit (GBTD), convened by UK Finance, gives UK banks a shared tokenised sterling deposit, and seven banks have completed its first live customer transactions.[2] The Clearing House is bringing the same to US banks through its On-Chain Money Initiative. Canada's six largest banks

announced on 22 September 2026 that they are jointly developing a Canadian dollar tokenised deposit solution, starting with moving tokenised deposits between institutions. In Europe, the Commercial Bank Money Token (CBMT) initiative of the German banking industry is in pilot with German banks and corporates, and banks elsewhere in Europe are joining its sandbox.[8] Each gives every member the same rails, the same participation rules and the same settlement certainty, and each leaves the product to the bank.[3][9]

Swift's messaging network and its new shared ledger, domestic payment systems, real-time gross settlement systems, tokenised deposit networks and asset platforms play different roles. A connection to one does not make the others redundant. Nor does connectivity alone establish access rights, settlement eligibility or a common legal treatment. An economic instruction often crosses several of them before the customer outcome is achieved.

Common rules and distinct propositions

A shared scheme can standardise what participants need to exchange and which outcomes they can rely on. Individual banks can still differentiate through customer mandates, servicing, liquidity, pricing and programmable products. This separation is commercially useful: common reach can support distinct customer services without requiring each bank to recreate every network function.

The boundary between shared and local responsibility must remain visible. The scheme may coordinate a common step while a bank retains responsibility for its deposit liability, customer controls and accounting. If a transaction is delayed, the customer should not have to navigate an organisational map to find the owner of the problem.

Reach is what membership buys

What a bank does with the reach is the differentiation: which customer instructions it carries, under which mandates, with which conditions and evidence, and how it reports the outcome. More connected systems bring more operating calendars and exception owners, and the intent layer is where those are made explicit and owned. That layer is the bank's own, and the sooner it is designed the sooner membership pays.

A bank that operates in several currencies will be a member of several networks, each with its own rules, its own hours and its own point of finality. Its customers' instructions will cross them: a euro payment against sterling, a dollar repo against a bond held elsewhere. Coordinating one instruction across two networks, each settling under its own rules, is the work above the ledger, and it is the same work whichever networks are involved.

THE CLEARING HOUSE · 24 SEPTEMBER 2026

The Clearing House has selected Quant to power its On-Chain Money Initiative, an interoperable payments network that will enable financial institutions of all sizes to clear and settle tokenised deposit transactions. Quant provides the network's interoperability, orchestration and transaction-management layer, with connectivity to the RTP® and CHIPS® networks. The network opens to participating institutions in the first half of 2027.[3]

> **In The Clearing House's words, Quant has "a proven track record of delivering on-chain capabilities in production and at scale".**

It is the pattern this section describes: The Clearing House owns and operates the network, its governance and its settlement framework, and each bank keeps its own deposit liability, customer controls and products.

Read the announcement →

05 PART V

Getting started

How Quant works with a commercial bank

The fastest route to realising economic intent runs through the bank's own group.

Tokenised deposits can move between the bank's own entities and offices, across markets, and between its own corporate customers, with every counterparty already inside the bank. A network, where one exists, adds reach and a ready-made set of counterparties. Whichever route, the product rests on four design decisions and a model for running it, and Quant works through each of them with the bank's product, operations, risk and technology teams.

01

Define the outcome before the route

What the customer will be able to rely on: when the money moves, what it waits for and what counts as complete. Agreed first, so the technology choices can be compared against it.

WHAT WE COVER TOGETHER

- The first use case and who it serves, often a corridor between the bank's own entities with a pilot corporate customer, and the patterns that follow it
- The successful path in plain steps: reserve, mint, transfer, book, settle
- When funds are final for the customer, and which legs settle behind it, such as a nostro that squares when a core reopens

- The operating calendar: cut-offs, weekends across time zones, and what waits while a system is closed
- What the first release includes, and what is deliberately parked

02

Establish the financial authority

Where the deposit liability and individual entitlements are recorded, whether direct, mirror or omnibus, and the role of the general ledger, mapped against the bank's existing core and accounting.

WHAT WE COVER TOGETHER

- Which system is the golden source: core banking with the tokenised ledger reflecting it, or the tokenised ledger as the book of record
- The backing model for each segment: mirror accounts for hundreds or thousands of corporate accounts, omnibus at larger volumes, direct where the ledger becomes the record, and how they run side by side
- How foreign-currency and nostro positions are represented
- The single integration seam with the existing estate: which system the tokenised ledger talks to, with which messages, and what stays behind it
- Reconciliation between the core, the backing accounts and the chain, and the history kept for audit

03

Make controls and ownership explicit

Which mandates permit the activity, who supplies the evidence for a condition, who can change a live rule and who owns an exception, designed into the service before it goes live.

WHAT WE COVER TOGETHER

- Approval rules: who can approve what, by payment type, channel, destination and value, with group policy and each jurisdiction's rules held together
- Onboarding: how a corporate's wallet and backing account are created and linked, and where the relationship manager's process fits
- Keys and custody: the bank's own HSM estate or a provider, and who signs what
- Change control: how the bank's teams adjust rules within governed releases
- The split of responsibilities between the bank, Quant and any integrator, from release pipelines to operations

04

Demonstrate dependable behaviour

How the service behaves when a condition fails, a system is unavailable or evidence is incomplete. The failure path is proved alongside the successful one, which is what a production case rests on.

WHAT WE COVER TOGETHER

- The exception scenarios the bank prioritises: insufficient funds or limits, a screening hit, an expired approval, a lost connection, partial success, duplicate messages, cancellation after remittance and a reconciliation break
- For every way a transaction can fail: how it is detected, who resolves it and what the customer is told
- Recovery that starts from a recorded fact, never from silence
- A defined recovery for each case, with an approval step wherever a person should decide
- Where people step in: high-value settlement failures and infrastructure incidents, and the rota that covers them

05

Run it as a live service

What operating a tokenised deposit service looks like once real money moves: who watches it, who can step in and how every action is evidenced. The bank's operations team rehearses it before anything touches its systems.

WHAT WE COVER TOGETHER

- A single view of the service for the operations team: what has moved, what is pending and what needs attention
- Early warning on what matters to the bank: value, liquidity and cut-offs

- Intervention under the same controls as the service itself, with every action attributable to the person who took it
- Evidence that auditors and security teams can use in the systems they already run
- Support and resilience that grow with the service, from the first live transactions to round-the-clock operation

A durable basis for programmable banking

The record of money can change, and the ledger that holds it can be the bank's own or one it shares with others. In each case, economic intent gives the bank a consistent way to define the service and assess whether the surrounding systems can deliver it.

The commercial opportunity is to make existing banking relationships more useful through controlled, coordinated financial outcomes. The banks that build on tokenised deposits now, within their own group or across a network, will take their returns from the layer above the ledger.

GOING DEEPER

A second edition of this paper, for one-to-one discussion, takes the framework into implementation: posting authority, instruction design, transaction state, coordination, retry safety, reconciliation, operational responsibility, migration and production evidence.

Book a conversation at Sibos →

ABOUT THE AUTHOR

Lenna Russ

Chief Commercial Officer, Quant

Speak with Lenna at Sibos →

ASK ME ABOUT

Tokenised deposit use cases across banks, capital markets and treasury; what production has taught us; ROI and where to start.

Sources and scope

1. Timothy Aerts, Ronald Heijmans, Jan Paulick and Violeta Vuletić. *Hidden by complexity? Measuring stablecoin, crypto and decentralised finance ecosystems.* BIS Working Papers No 1377, September 2026. See the introduction and sections 4.1 to 4.3 for aggregation, contract classification and stablecoin comparability. The views are those of the authors and do not necessarily reflect those of the BIS or its member central banks. **Read the BIS working paper**
2. UK Finance. *UK banks complete first live customer transactions using tokenised sterling deposits.* Press release, 24 September 2026. **Read the announcement**
3. The Clearing House and Quant. *The Clearing House Partners with Quant to Advance the On-Chain Money Initiative.* Press release, 24 September 2026. **Read the announcement**
4. Phoebe Seers and Elizabeth Howcroft. *UK banks make first interbank transactions using tokenised deposits.* Reuters, 24 September 2026. **Read the article**
5. Murex and Quant. *Murex, Quant Network embed tokenized deposits, digital bond settlement into capital markets infrastructure.* Press release, 25 March 2026. **Read the announcement**

6. UK Finance and Oliver Wyman. *Building Digital Markets of the Future: how the UK can unlock deep, liquid tokenised markets across the lifecycle.* August 2026, published 8 September 2026. **Read the report**

7. Quant and Murex. *Programmable Settlement: Synchronise Cash and Asset Movement, Without Changing Your Operating Model.* Joint whitepaper, 23 September 2026. **Get the whitepaper**

8. Ledger Insights. *BNP Paribas, ABN AMRO join CBMT deposit token sandbox as German initiative goes European.* September 2026. **Read the article**

9. BMO, CIBC, National Bank of Canada, RBC, Scotiabank and TD Bank Group. *Six Canadian banks explore development of a secure CAD tokenized deposit solution.* Joint release, 22 September 2026. **Read the release**

Interpretation and examples

The economic-intent framework, posting-model comparison and commercial implications are Quant's analysis. They should not be attributed to the BIS. Examples are illustrative, do not identify a bank or customer, and do not describe a committed implementation. Potential benefits depend on the relevant product, infrastructure, legal arrangements and operating model.

Direct, mirror and omnibus describe broad design patterns for discussion. Their precise account treatment and legal effects must be specified for the chosen arrangement. This paper does not provide a detailed implementation methodology or assert universal atomicity across payment and asset systems.