Project Jasper:

A Canadian Experiment with Distributed Ledger Technology for Domestic Interbank Payments Settlement

PRJ.JASPER

White Paper prepared by:
Payments Canada, Bank of Canada[1] and R3

TABLE OF CONTENTS

[1] Discussion in this report is complementary to Chapman et al. (2017).

Abstract

Project Jasper is a collaborative research initiative by Payments Canada, the Bank of Canada, financial innovation consortium R3 and a number of Canadian financial institutions to understand how distributed ledger technology (DLT) could transform the future of payments in Canada. The project was launched in March 2016, and since then two phases of exploration into the use of DLT for wholesale interbank payments settlement have been successfully completed. This paper describes the project's findings to date. Since inception, the project team has had the opportunity to explore and compare two distinct DLT platforms for the chosen use case—an Ethereum platform and an R3 Corda platform—while also building some of the key functionality found in wholesale interbank settlement systems today. Specifically, in the second phase the project team successfully built a liquidity-saving mechanism (LSM) in the form of a central queue on top of the latter platform to help economize on liquidity and promote smooth intraday flow of payment transactions across the platform. Interestingly, the analysis thus far is suggestive that DLT platforms that employ a "proof-of-work" consensus protocol, as was built in Phase 1, do not deliver the necessary settlement finality and low operational risk expected of core settlement systems. Phase 2, however, built a distributed ledger system that employed an alternative consensus model on the basis of a "notary node" that could deliver improvements in regard to settlement finality, scalability and privacy. While there is still much analysis to be completed, the project team continues to work on improving the ability of a DLT platform to observe the international Principles for Financial Market Infrastructures (PFMIs) that must be met by a distributed ledger wholesale interbank payments settlement system.

Acknowledgement of contributors

The following individuals are acknowledged for their contribution to this White Paper.

Payments Canada:

Andrew McCormack
Joshua Burrill
Kathryn Martin
Neville Arjani

Bank of Canada:

Scott Hendry
James Chapman
Dinesh Shah
Adrian Guerin

R3:

Rod Garratt
Clemens Wan

section 1 Introduction

In recent years, material global resources have been devoted to developing and implementing solutions in the area of financial technology—or fintech—which leverages technology to support innovation and improvement in the provision of financial services. Product solutions developed by players in this space, including venture start-ups, financial institutions, industry consortia, academics and central authorities, can encompass new business models, applications and processes. Some solutions that are already having an impact on the provision of financial services fall in areas such as trade finance, asset management, capital markets, retail and business lending, supply chain management and, of course, payments. Fintech solutions are marketed with the prospect of reduced complexity and cost, greater transparency, enhanced product and service customization, and improved access for consumers of financial services around the world.

One fintech innovation that has garnered much attention recently is distributed ledger technology (DLT). DLT, which includes blockchain technology, is perhaps best known as the technology underpinning the cryptocurrency system Bitcoin.[2] New use cases seeking to leverage DLT to improve the client experience and/or improve the efficiency of asset transfer emerge almost daily and extend well beyond the creation and exchange of digital currencies. Indeed, DLT demonstrates much promise in a number of global industries, given the benefits mentioned above.

This report presents early findings from Project Jasper, a collaborative research initiative undertaken by Payments Canada, the Bank of Canada, the financial innovation consortium R3 and a number of Canadian financial institutions to understand how DLT could transform the future of payments in Canada.[3] Project Jasper was launched in March 2016, and since then two phases of exploration into the use of DLT for wholesale interbank payments settlement have been successfully completed. The report discusses the background and rationale for the project, highlights governance and organization, and, perhaps most importantly, elaborates on the development of a proof-of-concept DLT design for interbank settlement (hereafter referred to as the "Jasper platform") as a way to inform the hypotheses of the project team. A technical appendix (**Appendix 2**) also accompanies the main text, offering further information on the development of the Jasper platform, including in the context of the international Principles for Financial Market Infrastructures (PFMIs) (**Appendix 3**).[4] The PFMIs represent global guidance for the management and control of key risks including credit, liquidity, operational and general business risk. The report concludes by outlining possible next steps for the project.

[2] See Nakamoto (2008).

[3] The Bank of Montreal, Bank of Nova Scotia, Canadian Imperial Bank of Commerce, Royal Bank of Canada, and TD Bank participated in both Phases 1 and 2 of the project. National Bank of Canada and HSBC Bank Canada joined for Phase 2.

[4] The PFMIs, which were published by the Bank for International Settlements (BIS) in 2012, can be found on the BIS website at **http://www.bis.org/cpmi/publ/d101a.pdf.**

section 2

Anticipated Benefits of DLT and Its Application to Interbank Settlement

What is special about DLT, and what does it aim to address that current systems and processes do not? In August 2016, the World Economic Forum (WEF) published an extensive report on the prospect of implementing DLT across a range of financial services areas.[5] The WEF report describes DLT as a repository of information (or database) underpinning asset exchange between parties over one or more peer-to-peer network platforms. Importantly, the notion of a "distributed" ledger means that a common database is maintained and shared amongst all parties to the arrangement.[6] This allows each party to an agreement to maintain a consistent copy of the same transaction record on a proprietary ledger, and every recorded change to the database to be synchronized across all copies of the ledger.

As well, strict protocols are established to govern by whom, and how, changes to the database are carried out to prevent unauthorized (and possibly malicious) edits to ledger content. These protocols are also designed to eliminate incidents of "double-spending" where a party to the arrangement either purposely or inadvertently tries to transfer the same asset over the platform more than once. In virtually all implementations of DLT, multiple parties must come to a consensus on the legitimacy of a transaction before it can be posted to the repository (i.e., no two parties should have a conflicting view of the transaction for it to be recorded). The database is updated in real time as transactions are validated, which affords timely exchange of value and access to trade and/or position information from the same trusted source.

DLT enables collaboration among parties to maintain a single, updated record of transaction activity. This shared representation of data and common process can reduce or eliminate the need for error-prone internal record-keeping by each party. With DLT, each party records transaction activity as an agreed set of data signed by all counterparties, eliminating the need to reconcile internal transactions since the data have been agreed and attested to by all parties and cannot be changed by one party acting alone. The likelihood of disputes is therefore reduced, and regulatory compliance requirements can more easily be met through reliance on a mutually agreed transaction history.

The need for a central database and, more importantly, a trusted database operator to house transaction records between parties may also be eliminated with DLT. In some instances, it can prove quite difficult to find such a trusted entity. A trusted central database and operator would still represent a single point of failure warranting careful attention to business continuity and disaster recovery in implementation.[7]

[5] See WEF (2016).

[6] This section describes DLT in a general sense. Depending on use case, some variants of DLT design limit the information that is distributed to each party; for example, where privacy concerns are prevalent.

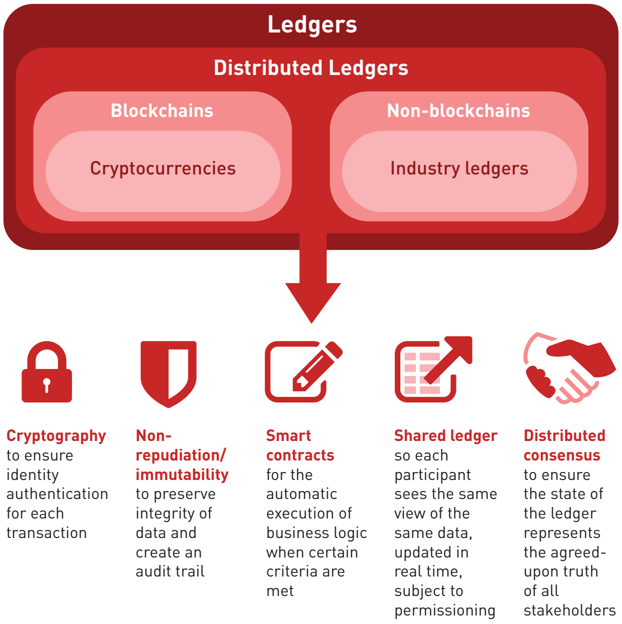
[7] Again, this section describes DLT in a general sense. As will be discussed, not all DLT solutions eliminate this single point of failure problem so cleanly.

Building in such resilience, however, may add to the cost of the solution. Moreover, in a central database or central party solution, transacting parties would still need to align their internal reconciliation systems and processes with the central database. DLT solutions could obviate or eliminate the need for additional reconciliation once transaction details have been attested by all (or a majority of) parties.

DLT arrangements are also becoming more flexible and cost-effective with time, which enhances DLT's appeal in more areas of commerce.[8] The WEF report (2016) makes it clear that DLT solutions are emerging in all shapes and sizes depending on the use case. For example, the ability to access the DLT platform or to add new transactions need not be open to everyone, unlike the Bitcoin system. Instead, the platform can be accessed only by parties that meet certain eligibility criteria. This is referred to as a "permissioned" DLT.

Moreover, where privacy and confidentiality concerns among parties are paramount—as is the case in financial services—it is also possible to limit the information from the database that each party is privy to. For example, access could range from every party seeing all information if transparency is desirable, to information being restricted solely to each party's own activity if privacy is preferred. Alternative protocols for validating transactions and recording them on the database are also emerging to substitute for more computational-intensive and time-consuming consensus mechanisms based on "proof-of-work" as in the Bitcoin system, for example. The introduction of a "notary node" on R3's Corda DLT platform that verifies transaction uniqueness as part of the consensus protocol is an example of such a development.[9] Thus, DLT continues to evolve to support transaction immutability, privacy of information and enhanced speed of transaction processing. Moreover, by facilitating the creation and execution of "smart contracts" by codifying the terms and conditions of economic agreements directly on the platform, DLT can help to reduce contract uncertainty and counterparty risk. See **Figure 1** for a summary of the key benefits of DLT.

Figure 1: Benefits and Key Features of DLT


Ledgers
Distributed Ledgers
Blockchains
Non-blockchains
Cryptocurrencies
Industry ledgers
Cryptography to ensure identity authentication for each transaction
Non-repudiation/ immutability to preserve integrity of data and create an audit trail
Smart contracts for the automatic execution of business logic when certain criteria are met
Shared ledger so each participant sees the same view of the same data, updated in real time, subject to permissioning
Distributed consensus to ensure the state of the ledger represents the agreed-upon truth of all stakeholders

[8] One might argue that such developments have led to a shift away from the original principles of the DLT grassroots movement. Examples include openness and inclusion, decentralization and operational resilience.

[9] While reaching consensus can be easier and faster using these alternative protocols, and privacy maintained, concerns around the operational resilience of the platform may be heightened.

Potential Application of DLT to the Payments Industry

These features suggest that the payments industry is a viable target for a DLT use case. For example, in regard to clearing and settlement, which represents the back-end of the payments value chain, the reliance by participating financial institutions on central databases is ubiquitous. In most cases, these central databases are owned and operated by a trusted party such as a central bank or industry association. Given the importance of these centralized arrangements in underpinning the payments ecosystem and broader economy, significant investment in business continuity and disaster recovery is warranted. These systems are highly regulated, where the overall scope of regulation can extend beyond the safe and efficient operation of the arrangement itself to other areas such as prudential regulation, anti-money laundering (AML) and anti-terrorist financing (ATF), as well as a multitude of consumer protection and privacy laws. Indeed, many regulators and central agencies maintain a keen interest in payment clearing and settlement arrangements.

Even a single payment message or file transfer sent by one party to another in a clearing and settlement arrangement can trigger a complex train of activity involving multiple entities. Participation in these systems is typically tiered, meaning that a financial institution that is a direct participant in an arrangement may represent not only its own interest, but may also act on behalf of other financial institutions.[10] These latter institutions are commonly referred to as "indirect" participants because they connect to the clearing and settlement arrangement through the services of a direct participant. As such, a single payment message or file transfer between direct participants in a central clearing and settlement arrangement has the potential to be captured in the internal record-keeping systems of multiple financial institutions. Given the sheer volume of activity that flows through these systems—e.g., amounting to tens of thousands of payment messages and batch-file entries per day in Canada—erroneous and duplicate entries can and do occur, sometimes escalating into costly disputes between participating financial institutions. Such errors and disputes typically require manual or semi-automated resolution by the affected institutions.

The merits of DLT appear particularly relevant in the area of cross-border payments where, despite global efforts to centralize the settlement function through the Continuous Linked Settlement (CLS) service, traditional correspondent banking networks are still prevalent in moving value across regions and currencies. Trillions of dollars in value are transferred across global borders each day. Supply chains in this space can be highly complex, involving a large number of financial institutions in different countries and time zones around the world, all relying on their own internal record-keeping to capture proprietary aspects of each currency transfer. Counterparty risk can run high in this space. Moreover, regulatory compliance measures related

[10] Moreover, these arrangements have the potential to introduce counterparty credit risk that needs to be managed, and which might benefit from smart contract technology.

to AML/ATF[11] in the cross-border space have the potential to present an excessive number of false positives which, in some cases, can take weeks or even months to resolve. Affected financial institutions and their clients have to pore through and try to reconcile their internal records to aid in legitimizing each transaction. For the client of one of these incorrectly flagged payments, weeks or months is an exceptionally long time to have working capital or investment funding tied up unnecessarily. For these reasons, DLT is increasingly being referenced for cross-border payments where innovators such as Ripple Labs are partnering with global financial institutions to exploit gaps in this service area.[12]

In most jurisdictions, including Canada, domestic payments services are viewed as relatively efficient compared with cross-border services. This reflects consumers' trust in highly regulated national infrastructure providers and financial institutions, and in the central monetary authority which typically serves as the settlement institution in these arrangements. Compared with the cross-border experience, consumers of domestic payments services are offered greater variety and lower-cost payment options; well-established rules, standards and procedures governing the use and acceptance of these various payment instruments; and reliable and/or timely funds availability. Nevertheless, as described earlier, there is still potential for costly error, duplication and disputes among parties to domestic payment clearing and settlement arrangements, largely driven by inconsistencies across the back-office record-keeping systems both within and across participating financial institutions. As well, central database solutions for domestic payments without high-availability designs could act as a single point of failure, and therefore require significant investment in business continuity and disaster recovery, which increases the overall cost of the arrangement.

As the shift to digital commerce and the seamless integration of the "payment experience" into the overall "customer experience" continues, it is critical that payment arrangements keep pace with other aspects of the user experience. Payments based on legacy technology could be an impediment to achieving a near frictionless end-to-end customer experience. For these reasons, exploration of emerging technologies such as DLT that have the potential to improve the payment experience is warranted. Herein lies the basis for launching Project Jasper. See Figure 2 for a summary of the potential benefits of DLT in the payments space.

Figure 2: Benefits and Key Features of DLT for payments industry

- Improved back-office payment processing and reconciliation within and across participating FIs
- Reduced likelihood of costly errors and disputes between participating FIs
- Potential improvement in addressing false positives tied to sanctions screening, and providing a clear and consistent audit trail for financial transactions
- Continued system transparency and monitoring for central banks and regulators
- Continued preservation of information privacy among FIs
- Improved automation through the use of smart contracts

[11] The acronyms AML and ATF refer to anti-money laundering and anti-terrorist financing, respectively

[12] More information on Ripple Labs can be found at **https://ripple.com/.**

section 3

Project Jasper: Background, Scope and Objectives

As the owner and operator of Canada's national clearing and settlement systems, and with a legislated mandate to facilitate the development of new payment methods and technologies, Payments Canada maintains a keen interest in understanding how emerging technologies such as DLT could transform the future of payments.[13] In early 2016, in collaboration with the Bank of Canada, the financial innovation consortium R3 and a number of major Canadian banks, Payments Canada launched Project Jasper with the goal of exploring the use of DLT to settle interbank payments.[14] The focus of the exploration includes operational, legal, policy and regulatory considerations. The launch of Project Jasper marked a significant milestone in the payments industry because, to our knowledge, it was the first time in the world that a central bank participated in a DLT experiment in partnership with private financial institutions. An overview of the two phases of Project Jasper can be found in **Figure 3.**

Staff from Payments Canada and the Bank of Canada have been instrumental in driving the Jasper Project and were key resources in defining the business requirements and producing research deliverables. Representatives from R3 led the design

Figure 3: High-Level Overview of Project Jasper Phases 1 and 2

	Phase 1	Phase 2
Purpose	Experimental review of DLT's applicability to wholesale interbank payments settlement	Further evaluation of the scalability and flexibility of DLT by building a Corda experiment. Accommodate multiple settlement options covering RTGS and LSM. Provide a data-driven simulation exercise with operational data sets to evaluate platform and LSM performance.
Platform	• Ethereum • Custom middleware • Solidity (3 contracts) • Angular JS front end	• Corda • Kotlin • Extended Phase 1's Angular JS front end
Innovations	• Atomic exchange (transfer without reconciliation) • Maintained knowledge of ecosystem distribution • Impacts on PFMI	• Netting options • Decentralized node authorities • Corda 1-N atomic flows • Operational impact
Notable output	• Ethereum prototype • Research in PFMI context	• Corda prototype • Platform comparison • 2 LSM comparisons • Research White Paper

[13] The legislated mandate and duties of Payments Canada are articulated in the *Canadian Payments Act*, which came into force in 2001 (succeeding the *Canadian Payments Association Act*). It is available for download on the Payments Canada website at **www.payments.ca.**

[14] The term "interbank" is synonymous with "wholesale" and "high-value."

and build of the Jasper platform, supported by an exceptional team of architects and developers from the participating Canadian banks. In addition, Payments Canada provided overall project management for the initiative, and worked with the Bank of Canada and R3 to define the project scope. Moreover, senior representatives from all of these organizations came together to form the project's Steering Committee, which met regularly to provide high-level thought leadership to the project team. The success of Project Jasper to date has, without question, resulted from the close collaboration between all of the organizations involved.

Phase 1 of Project Jasper was launched in March 2016 and concluded in June 2016. The objective of Phase 1 was to build wholesale interbank settlement capability on an Ethereum DLT platform and demonstrate its ability to exchange value in the form of a central bank-issued digital settlement asset.[15] The settlement asset established in Phase 1 was a digital depository receipt (DDR) reflecting a claim on Canadian-dollar deposits held in accounts at the Bank of Canada. Only banks participating in the Jasper Project were able to transact using DDR. It follows that Phase 1 also required procedures to pledge Canadian dollars in exchange for DDR and, conversely, to redeem DDR in exchange for Canadian dollars.

Phase 1 was successful in delivering the intended functionality in a non-production setting. Importantly, the consensus mechanism utilized by the Phase 1 platform was based on "proof of work," which required all R3 members to perform validation of a proposed exchange of DDR between two participants as a requirement for that exchange to be recorded to the database. However, only participating Canadian banks in the project could transact in DDR over the ledger.[16] The use of a proof-of-work consensus mechanism raised concerns from the perspective of operational efficiency, since it was demonstrated that this consensus method could not provide the throughput required as volumes increased. As well, the Ethereum solution provided full visibility into the central ledger for all participants in the system. Although this transparency was helpful for monitoring the status of all participants in the system, the platform did not support participant requirements for data privacy.

In a proof-of-work solution, it is unclear when settlement finality has been achieved, if ever. From a policy perspective, this presents a substantial challenge. The Phase 1 solution was also not integrated with any external systems (including collateral management capabilities) and relied on a prefunded account, which is very inefficient and, if adopted, would substantially increase settlement costs for FI participants.

DLT is a distributed technology, and the existing settlement process relies on centralized infrastructure. As a result, there were several design challenges to overcome in adapting the existing centralized settlement process for Project Jasper. It was unclear if a central bank settlement model could be supported

[15] Launched in 2013, Ethereum is a general-purpose DLT platform that allows any type of digital asset to be defined, created and traded. It also enables smart contracts, which allow a DLT to execute the terms of a contract automatically. These capabilities combine to provide far more functionality than simply the transfer of one specific type of digital asset. See Buterin (2013).

[16] See Garratt (2017). This was done as a matter of convenience. R3 has a version of the Ethereum platform operating for the use of its members that utilizes proof-of-work mining built into Geth for consensus and transaction validation. This platform was utilized for the Phase 1 simulation, with some modifications to eliminate the costs of ether and, with non-participants in the Jasper Project playing a passive role.

with a distributed solution. These important aspects of wholesale interbank settlement will require further investigation. For more information on Jasper Phase 1, please see **Appendix 1**.

Jasper Phase 2 was launched in September 2016 to build on the learnings from Phase 1. A major goal of Phase 2 was to evaluate the scalability and flexibility of DLT by moving to an alternative technology platform and by continuing to build in more of the functionality observed in today's interbank settlement solutions.

As part of Phase 2, the Jasper platform was transitioned from the Ethereum DLT platform to R3's Corda DLT platform, which, as noted above, introduces the concept of a "notary node" at the core of its consensus protocol. The Phase 2 platform was built in a test environment only, and there was no integration with external systems such as those that support collateral management and optimization. Transition to the Corda platform continued to rely on the pledge/redeem functionality determined in Phase 1. As part of the Phase 2 implementation, the project team established additional participant nodes to support two additional Canadian banks that joined the project.

The Phase 2 platform was built to accommodate multiple settlement options similar to those currently available in incumbent infrastructures, representing a major achievement by the Jasper development team, and demonstrating the creativity and capability of its members. The two settlement options supported by the platform are an "atomic" option and a "liquidity-saving mechanism" option (LSM)[17]. A cornerstone of the LSM option is a central queue that draws on a payment-matching algorithm to routinely settle batches of queued payments on a net basis. Netting promotes funding efficiency and enables a smoother intraday flow of payments. Indeed, the Phase 2 platform appears to be one of the first instances of implementing a central queue with a payment-matching algorithm on a distributed ledger platform.[18]

Lastly, data-driven simulation exercises were completed in Phase 2 to evaluate the operation and performance of the Jasper platform. Specifically, the operation of the central queue and payment-matching algorithm was evaluated under a range of unique circumstances, or "edge-cases" using smaller carefully crafted data sets. As well, the payment-processing capacity of the broader platform was evaluated using simulation by drawing on much larger data sets reflective of daily transaction volumes observed in the Large Value Transfer System (LVTS) today.

It is important to emphasize that exploration of DLT for wholesale interbank settlement under Project Jasper is still in the early stages. Important considerations (e.g., technical and operational, legal and governance) require further reflection before concluding whether DLT introduces a net benefit for this use case. As such, the findings presented in this

[17] In this context, "atomic" settlement is akin to settlement on a real-time gross basis.

[18] The Bank of Japan has also done a DLT experiment incorporating an LSM. **http://www.boj.or.jp/en/announcements/release_2017/rel170208a.htm/.**

paper should not be viewed as a "referendum" on DLT in terms of its ability to support wholesale interbank settlement in Canada, nor is the intent to deliver a comparison of DLT against the incumbent infrastructure. Evaluation against the LVTS, while a possible long-term objective of Project Jasper, was not included in the scope for Phases 1 and 2. It is expected that these insights will come naturally as results emerge from further research.

A final word is warranted on how Project Jasper relates to ongoing efforts led by Payments Canada to modernize the Canadian payments ecosystem. While insights learned from Project Jasper are expected to inform modernization, and vice versa, it is important to confirm that the project is separate from the modernization agenda and there are no plans to include DLT as part of the improvements being contemplated when this paper was being written.[19]

[19] That said, the rapid maturity of this and other new technologies could be considered part of the modernization initiative. To the extent that they are feasible, modular payments system designs that have the capacity and flexibility to incorporate new technical solutions as and when they are ready should be considered.

section 4

Key Hypotheses and Considerations Addressed by the Project

With the objective of determining the viability and feasibility of settling wholesale interbank payments on a DLT platform, the Project Jasper team pulled together a number of hypotheses for evaluation over the course of the project. Certain of these hypotheses, of course, remain to be assessed given the particular scope of Phases 1 and 2.

Solving for the "Last Mile"

An underlying premise of the project is that payments settlement represents the final leg of most economic transactions, where some types of transactions may already be, or have the potential to be, supported by DLT solutions in other areas of the contract chain. For example, trade finance and supply chain management are market examples where DLT and the use of smart contract technology within the contract process are already highly applicable. From an efficiency standpoint, and regardless of the technology under consideration, unnecessary frictions in economic exchange introduced by the payments leg of a contractual agreement between parties should be identified and eliminated. Moving payments settlement to a DLT platform and aligning it with other DLT arrangements supporting the same economic contract is an appealing proposition in promoting reduced friction and cost. Moreover, by incorporating the use of smart contract technology, for example, to automate payment release based on specific terms and conditions being met, it is possible to imagine both reduced contract uncertainty and reduced reliance on trust between contract counterparties.

Principles for Financial Market Infrastructures (PFMIs)

The international PFMIs will serve as a helpful guide in the design and evaluation of the Jasper platform, since any DLT arrangement for wholesale interbank settlement will be expected to comply with these principles. On some principles, the Bank of Canada has issued risk-management standards for designated FMIs that fully incorporate the PFMIs and Key Considerations articulated in the PFMIs. This additional guidance should also be factored into the design of the Jasper platform. For example, the platform should reflect a real-time gross settlement (RTGS) model and avoid reliance on a central bank commitment to settle accounts, which is a characteristic of Canada's current LVTS.[20] Moreover, there is no intraday credit facility envisaged in the Jasper framework. All value exchange over the platform is prefunded. It is worth noting that the platform design in Phase 1 was also implemented as a "pure" RTGS model, with no intraday credit provision. There are, of course, some aspects of a non-production DLT platform that are difficult to evaluate against international principles (e.g., governance, legal

[20] See Leduc (2017). More information on the LVTS settlement and risk model can be found in the next section, and in **Appendix 4** that accompanies this report.

basis, some operational aspects). While this report is not a formal assessment of the platform in relation to the PFMIs, it does provide for initial discussion, with additional detail provided in **Appendix 3** (PFMI Considerations).

Of note, the LVTS is one of the most collateral-efficient wholesale settlement models in the world and meets key guidelines for settlement risk established in the PFMIs. The PFMIs also emphasize maintenance of a liquidity-efficient settlement arrangement for participating financial institutions. Turnover in the LVTS Tranche 2 payments stream (described below) stands at roughly 24x (i.e., meaning $24 in processed payments value is supported by $1 of pledged collateral). Despite having to adopt an RTGS model and, more importantly, having to abandon the Tranche 2 risk model and use of partial collateral pooling, any DLT solution for wholesale interbank settlement will also need to demonstrate a high turnover rate to reduce cost and risk exposure for participants.

Management of Settlement Risk

Settlement of a payment is defined as the unconditional and irrevocable transfer of value between an originator and beneficiary. When these conditions are met, settlement is considered "final" since the beneficiary can use the funds without risk of reversal in any circumstance. Timely settlement finality is crucial for financial and economic stability and represents a requirement of the international PFMIs. While final settlement occurs at a clear, well-defined point in time in current wholesale settlement arrangements, and is backed by a strong legal basis, this concept of final settlement may be less clear in a DLT environment.

Scalability and Operational Resilience

A common question around DLT is whether it can handle the transaction volumes observed in the incumbent infrastructures. A DLT-based wholesale settlement solution in Canada must be able to process, at a minimum, over 50,000 payment items per day, and at least 14 items per second, based on observation of the LVTS today. In DLT arrangements and, in particular, those that rely on proof-of-work consensus protocols, there could be a high computational requirement to validate proposed transactions and record them on the ledger. Moreover, it can take time to achieve consensus among platform participants, particularly when communication and information-sharing between nodes occurs over a public Internet connection. Meeting the current volume benchmark using this arrangement would require careful attention to scalability in the design of the DLT solution. As performance of DLT platforms continues to improve, it is likely that these transaction volumes could be met. For example, testing in Jasper Phase 2 illustrated that current-day average volumes could be processed within an acceptable window.

A second hypothesis is that operational resilience may be achieved in a more cost-effective manner on a DLT platform. This is predicated on the observation that centralized systems without high-availability designs would represent a single point of failure, and therefore operators must invest heavily in business continuity and disaster-recovery arrangements.

Operating Efficiency

DLT provides the opportunity for parties to a transaction to maintain and share identical and mutually agreed records of transaction activity between them. This should deliver reduced incidence of errors and related disputes between counterparties, which are commonly caused by duplication and inconsistencies within and across the back-office record-keeping systems of the parties. DLT shows potential to drive a reduction in operating costs, given that manually addressing errors and disputes can be costly to a bank, and also recognizing the resources required to continue to synchronize transaction information across separate information repositories. This hypothesis is, however, difficult to evaluate in a non-production environment.

Global Integration

Beyond payments settlement, to effectively support global commerce in the future, interoperability among domestic and global financial markets and infrastructures is needed. At the end of the day, DLT presents an opportunity to store, update and facilitate sharing of information related to virtually any economic contract agreed to between parties. The prospect of all involved parties having real-time access to the same contract terms and conditions has much appeal, and more so when combined with the possibility of automated contract enforcement built into the solution. DLT has the potential to deliver a more seamless and lower-cost transaction experience, particularly where exchange across multiple asset classes can be accommodated by the same shared database. Indeed, Brown et al. (2016) introduce a long-term vision of a “global logical ledger” with which all economic actors will interact and which will allow any parties to record and manage agreements amongst themselves in a secure, consistent, reliable, private and authoritative manner. This hypothesis is likely to take some time to evaluate, of course, and is beyond the scope of Phases 1 and 2. Nevertheless, it is interesting to contemplate possibilities for DLT well into the future.

section 5

Current Environment for Wholesale Interbank Payments Settlement

Before presenting the main tenets of the Jasper platform, it is worth taking time to review existing wholesale interbank settlement arrangements in Canada. This section introduces the LVTS, which is owned and operated by Payments Canada, and draws on observations of wholesale interbank settlement arrangements for liquidity management to provide guidance on the design of the Jasper platform. Terminology used here will also establish context for introducing the Jasper platform in the next section. Interested readers can find more information on the LVTS in **Appendix 4** to this report.

Overview of Canada's Large-Value Transfer System (LVTS)

Introduced in 1999, the LVTS lies at the centre of the Canadian financial system, supporting payment clearing and settlement among 17 participating financial institutions ("LVTS participants") on their own behalf and on behalf of their clients. The LVTS also supports settlement of end-of-day positions at other financial market infrastructures (FMIs) in Canada, including securities, foreign exchange and derivatives. As well, it serves as the platform for the daily implementation of Canadian monetary policy. As an indication of its prominence, the LVTS clears total payments value of over $170 billion daily, which is equivalent to clearing the value of Canadian nominal gross domestic product every nine business days. The Bank of Canada has designated the LVTS as a systemically important payment system in Canada, and is responsible for oversight of the LVTS in accordance with international PFMIs.

The LVTS employs a real-time net settlement model where individual payment messages that pass the real-time validation checks (e.g., liquidity availability) are processed immediately, and settlement of participants' multilateral net positions occurs on the books of the Bank of Canada at the end of each business day. Once a payment message is processed, funds can be distributed to the beneficiary on a final and irrevocable basis. In this way, the LVTS supports payment finality before the system achieves settlement finality, which makes it a unique arrangement globally. Through the use of credit exposure limits, collateral posted by participants and a central bank commitment to settle accounts, LVTS settlement is guaranteed to occur in all states of the world.[21] The LVTS risk model supports two tranches (or payments streams), and both are underpinned by the same settlement model. These tranches differ in both collateral requirement and loss-allocation procedure in the event that a participant is unable to meet its settlement obligation. Tranche 1 employs

[21] More information on the LVTS, and the central bank commitment to settle accounts, can be found in CPA By-law No. 7 Respecting the Large Value Transfer System, which is available at **http://laws.justice.gc.ca/PDF/SOR-2001-281.pdf.**

a "defaulter-pays" loss-allocation model, where draws against the Tranche 1 credit exposure limit are secured fully by collateral pledged by the sending participant. Tranche 2 utilizes a "survivors-pay" loss-allocation model, where participants' draws against Tranche 2 credit exposure limits are partially secured by a pool of collateral pledged collectively by participants, with the remainder secured by a central bank commitment to settle accounts.

In 2016, the LVTS processed a total payments value of $175.245 billion each day, on average, representing more than 34,000 transactions. The maximum daily value cleared by the LVTS in 2016 was $271.797 billion and more than 53,000 transactions. At some points in 2016, the LVTS was processing up to 14 transactions per second, in addition to addressing other queries that may have been run at the time. Payments Canada maintains a firm commitment to business continuity and disaster-recovery planning for the system, given the importance of the LVTS to the Canadian economy and its role as a centralized infrastructure.

Liquidity-Saving Mechanisms (LSMs) in Wholesale Interbank Settlement Systems

Liquidity in wholesale interbank settlement systems can be described as the ability of a participant to meet its payment obligations as they become due. Sources of liquidity for participants include the value of incoming payments, which can subsequently be recycled as outgoing payments, and the ability to draw against an intraday credit facility, which is usually limited by some form of cap.[22] Participants will draw on intraday credit whenever incoming payments are insufficient to support outgoing payments as they become due.

To manage risk, intraday credit provision is typically secured by high-quality (and therefore costly) collateral. This creates an incentive to avoid such cost and delay one's payments as long as possible while awaiting incoming payments.[23] In the case of time-sensitive payment obligations, there is little room for delay, so participants will typically bear the liquidity cost to complete the payment on time. However, the vast majority of payments are non-time-sensitive with flexibility for delay. This can give rise to adverse outcomes, including payments gridlock, where no participant is willing or able to "be the first" to send funds to another participant. System operators have long understood this issue and have responded by implementing LSMs, described as mechanisms to improve the coordination of incoming and outgoing payments to support the smooth intraday flow of payments.[24]

[22] The provider of intraday credit can differ according to settlement model. In RTGS systems, central banks usually serve as providers of intraday credit, whereas in net settlement systems intraday credit is provided by the system.

[23] This incentive is less defined in Tranche 2 of the LVTS, given its use of a survivors-pay collateral pool.

[24] See Bank for International Settlements (1988, 2005) for a discussion of the rationale behind, and new challenges introduced by, LSMs in wholesale settlement systems.

LSMs can take different forms, including the use of central queues that may employ payment-matching algorithms. In this case, non-time-sensitive payments can be placed in the queue by participants rather than kept in internal schedulers. A payment-matching algorithm could then perform routine or ad hoc offsetting of queued payment values during the day, where only funds equal to participants' net obligations from the offset are required to process the entire batch of queued payments. This can contribute to a smoother intraday flow of payments and reduced funding costs for participants, while also creating an incentive to submit non-time-sensitive payments earlier in the day.[25] Norman (2010) speaks to the prevalent use of LSMs—and specifically queuing with offsetting algorithms—in wholesale payments systems around the world and points to international evidence indicating liquidity savings of 15 to 20 per cent through the use of these algorithms.

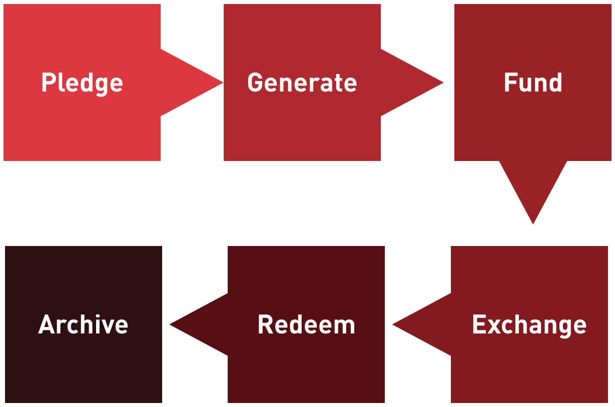
[25] See Davey and Gray (2014).

section 6

Overview of the Jasper Platform

This section describes the sequence of functions used to support exchange of value over the Jasper platform. Additional technical details can be found in **Appendix 2.** The Jasper platform encompasses a distributed database consisting of mutually agreed and notarized records of transaction activity between platform participants. The distributed database allows each party to a transaction to maintain a common copy of the records on their own proprietary ledgers, as well as broadcasting synchronized changes to each entitled party's ledger in real time. **Figure 4** illustrates the high-level sequence of events that occurs to exchange value over the platform.

Figure 4: High-Level Sequence of Events to Exchange Value over the Jasper Platform



Notion of a "State"

The notion of a "state" is integral to understanding the functioning of the Jasper platform (see **Figure 5** and Brown et al., 2016 for a fuller description of Corda). For example, the current "state" of the distributed database is observed at a point in time and reflects all transaction activity recorded on the database up to that point. As platform participants engage in further activity (e.g., exchanging DDR, redeeming DDR, etc.) the state of the database will change as each new transaction is validated and recorded. In other words, participants' activity on the platform essentially defines the state of the database. As indicated, strict conditions must be met before effecting any change to the state of the database (i.e., a consensus protocol must be in place). The evolving state of the database serves as evidence of participants' activity on the platform: for example, the amount of DDR each participant has transacted gives rise to its aggregate DDR balance. DDR represent a claim on Canadian-dollar deposits held at the Bank of Canada that result from a linked pledge and issuance process.

In addition to the state of the database, the state of individual transaction records between participants will also evolve over time. To understand this point, the concept of an "object" is introduced. Participants engage in activity on the Jasper platform, for example, to exchange DDR through either the atomic or

LSM settlement options. Each of these actions is recorded with the use of an object. For example, and as will be elaborated on below, to initiate an exchange of DDR with another participant, one must create a "DDR Object," which contains relevant information to the transaction (referred to as "data attributes") such as the sender, receiver and value of the transfer. For the DDR Object to be recorded to the database, consensus on the legitimacy of the transaction must be reached among multiple parties. The consensus protocol is described below. If consensus is achieved, the DDR Object will be recorded in the database, and will undergo a state transition from "unspent" to "spent," with the sender's unspent DDR balance now reflecting a lower amount.

Parties Involved

There are three types of nodes on the Jasper platform—a *participant node*, a *notary node* and a *supervisory node*. Each node maintains a proprietary ledger that captures up-to-date transaction record information.[26] Information is updated and synchronized across all proprietary ledgers as soon as any changes are recorded. To preserve privacy among banks, the ledgers for participant nodes contain only information that pertains to their own transactions. In contrast, the ledgers maintained by the notary and supervisory nodes are intended to capture all transaction records, which gives them full view of all activity taking place over the platform.

Of note, it is also possible for the supervisory and notary roles to be combined into a single node on the Jasper platform.[27] The Bank of Canada serves in both roles on the current version of the platform. Before a production implementation, the risk and policy implications associated with each potential candidate to perform one or both of these roles should be thoroughly explored.

The Jasper platform is a private-permissioned distributed ledger. Participant nodes on the platform are managed by the Canadian banks involved in the project, as well as the Bank of Canada, which also must exchange DDR over the platform. For example, the Bank plays a critical role as "settlement agent" in the LSM settlement option and must exchange DDR for this purpose. It is likely that those managing a participant node would be subject to eligibility criteria, including meeting the technical requirements of the platform and any operational and regulatory requirements established by the Bank of Canada as overseer. The notary node is a cornerstone of the Jasper platform's consensus protocol and represents a shift away from the proof-of-work consensus mechanism employed in Phase 1. A trusted third party is expected to manage the notary node. As noted, the Jasper platform also supports the inclusion of a supervisory node, which is expected to have full view of the ledger to aid in system oversight and compliance monitoring. The supervisory node can query its copy of the ledger at any time to determine the unspent DDR balance of one or more participant nodes, which supports real-time intraday liquidity monitoring at the participant level.

[26] For clarity, reference to a "proprietary ledger" throughout this section refers to a participant's maintenance of its own copy of the main ledger.

[27] To avoid confusion, reference is still made to the notary/supervisory node in the remainder of this section.

The Bank of Canada assumes the role of issuer of DDR. Participant nodes pledge Canadian dollars to the Bank in exchange for DDR, which can then be exchanged with other parties over the platform. A digital wallet application is introduced in the form of a user interface to assist participants with navigation through the available options on the platform, e.g., DDR exchange, DDR redemption, transaction queueing, etc.[28] Each node is also allocated a cryptographic signature that represents its identity on the platform and serves as part of the consensus protocol in transaction validation.

Consensus Mechanism

Consensus on the Corda platform is achieved through two functions: a validation function and a uniqueness function. The validation function is performed by the participants on the platform (i.e., the transacting banks). The uniqueness function is performed by a special trusted participant, known as the notary. In Jasper, the role of the notary is played by the Bank of Canada; however, in general applications the notary can be any trusted agent or group of agents.

The validation function ensures that the details ("data attributes") of the transaction are correct and have been agreed to by both sender and receiver, where agreement is demonstrated by these nodes reviewing and attaching their digital signatures to the proposed transaction. As part of the validation function, the receiver must verify the chain of custody related to the DDR the sender is proposing to send back to the issuer, in this case, the Bank of Canada. This requires that that the receiver sees the history of transactions related to the proposed DDR Object. This history may go back to the start of the payments day, or it may be shorter, if the DDR Object used in the proposed transaction emerged from the exhale phase of the LSM.[29] An implication of the validation function is that participants in the system see information about more than their own transactions, but no more than is absolutely necessary to allow independent verification. As such, the version of the Corda platform used in Phase 2 represents a significant advancement over Ethereum and, as far as we know, comes closest to meeting the user requirements for privacy among existing DLT platforms.[30]

The notary node is responsible for confirming the uniqueness of the transaction. That is, the notary ensures that the DDRs proposed for exchange have not been previously spent by the sender. This is done by marking any validated transaction object as historic and only validating transactions that do not involve objects with historic designations. The uniqueness function performed by the notary prevents the "double spend" problem without relying on a costly proof-of-work mechanism.

[28] The wallet application is "CordApp," a distributed application that sits above the Jasper platform.

[29] This is a rather fortuitous, unplanned bonus of the LSM design. The inhale/exhale procedure resets the time of issuance throughout the day and shortens the length of chains back to issuance. See page 23

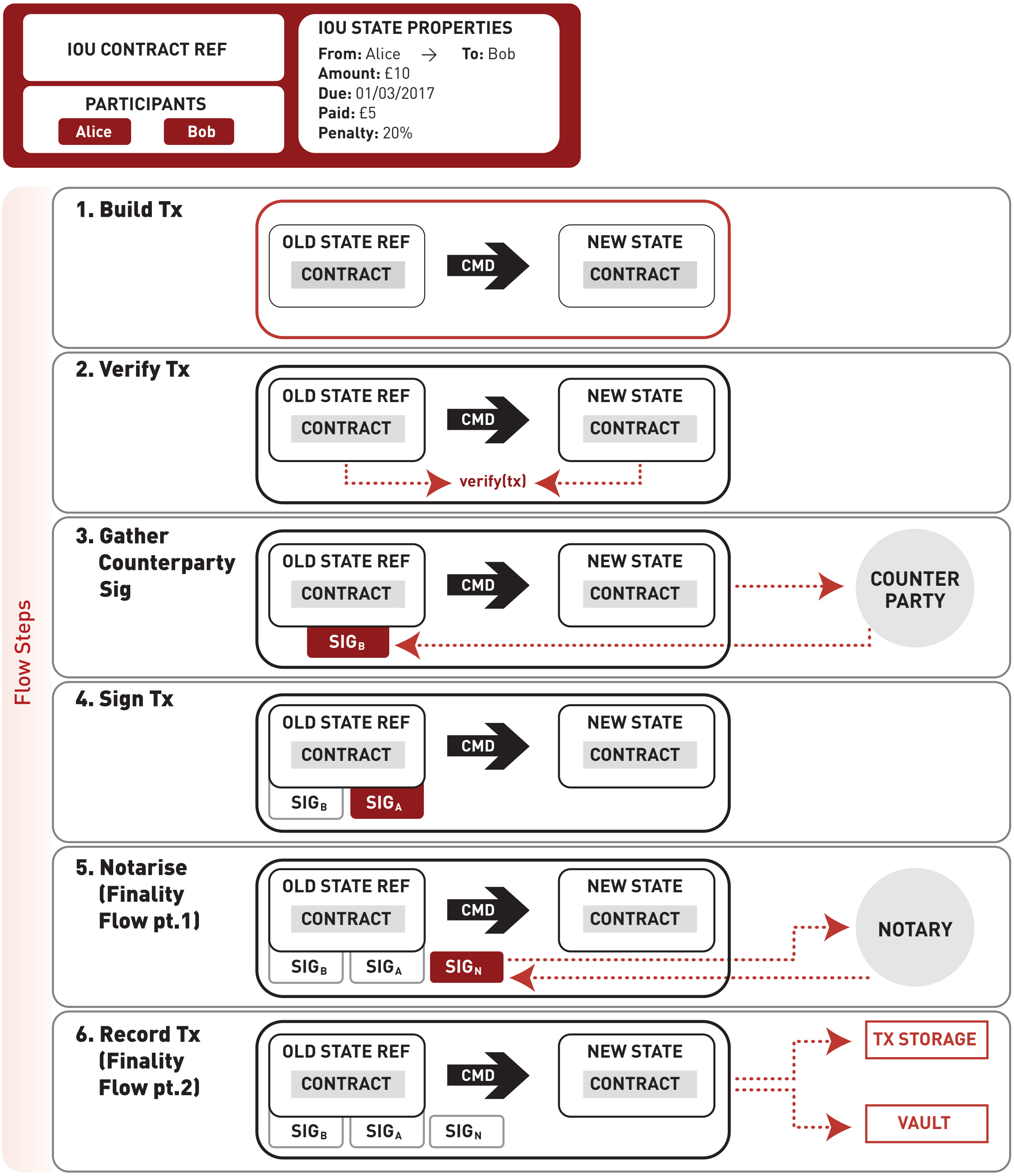
[30] Two options for meeting user privacy requirements are currently under development. One is for the sender to perform the validation themselves and then convince the receiver beyond doubt that they have done so, without providing additional transaction information. This is the idea behind zero knowledge proofs, and Corda is designed to adopt this technology as it matures. The other option is to send the necessary data to the recipient's *computer* for verification, but to do so in such a way that the recipient cannot see it, even though they have full control of their computer. This is what homomorphic encryption will allow as it matures and is what secure enclave technology, such as that provided by Intel's SGX, offers today. Corda is designed to take full advantage of Intel SGX.

Consensus on the Corda platform, as built for Project Jasper, requires the signatures of the transacting participant nodes, the notary node and the supervisory node. The notary node's signature is always the last to be added before recording an Object to its complete ledger and then broadcasting the results to the sender and recipient. Without these four signatures, a transaction is not considered to be valid and cannot be recorded on the database of each node that is entitled to see those transaction results. Note that requiring the supervisory node to sign each transaction could also be viewed as contributing to the operational resilience of the platform, since the supervisory node creates a proprietary ledger that reflects the full state of the ledger at a point in time (together with the ledger of the notary node).

DDR as a Concept

As mentioned, DDR is a central bank-issued digital asset that is exchanged over the Jasper platform. DDR is not "native" to the Jasper platform—it is a digital representation of a Canadian-dollar deposit balance held in an account at the Bank of Canada. This exchange rate is perpetually fixed; a DDR balance is always equivalent to the same value in Canadian dollars. Essentially, DDR performs all the same functions as money on the platform: a medium of exchange, store of value and unit of account. Ownership of DDR is critical to the functioning of the platform; it changes with each transaction and is reflected in the proprietary ledgers of the transacting parties to determine how much a party can "spend" at some point in time (i.e., exchange value over the platform or redeem DDR with the Bank of Canada for Canadian dollars).

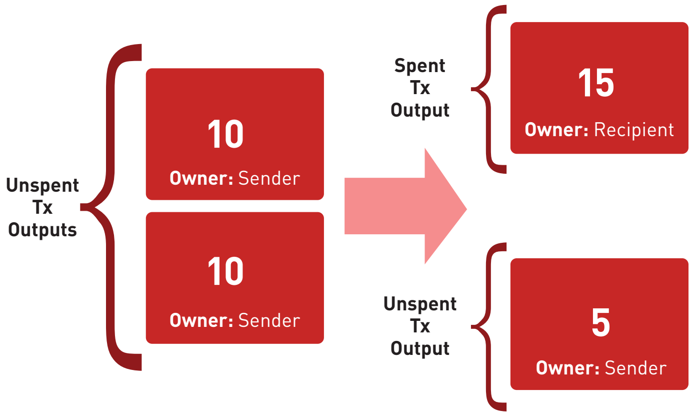
Figure 5: Corda – Key Concepts



The UTXO Model

The wallet application situated on top of the Jasper platform utilizes the Corda UTXO model, which stands for Unspent Transaction Outputs (**Figure 6**). As indicated above, to support the consensus protocol the platform parses "unspent" transaction outputs recorded on the sender's ledger to determine the validity of a transaction output—i.e., whether the sender has sufficient unspent DDR to affect the proposed exchange of DDR. Of note, the UTXO model also accommodates parallel (simultaneous) transactions, where the platform does not need to wait for a participant's account balance to be updated before another transaction can be sent.

Figure 6: UTXO Model and an Atomic Transaction



- This diagram represents the atomic transaction process with each block representing a DDR object
- The sender wants to send 15DDR to the recipient, so the platform confirms that the sender has enough DDR to send (based on the Unspent Tx Output blocks). The Unspent Tx Outputs are the inputs that fund the transaction.
- Note that two blocks are created: a block of 15DDR changes ownership to the recipient and a block of 5DDR (the sender's remaining balance) is created with the sender as the owner.

Pledging and Generating DDR

The process of funding a participant's wallet begins with the participant pledging Canadian dollars to a pooled deposit account at the Bank of Canada, with a request that the Bank generate and distribute DDR at par against the pledged dollars. This request generates an object on the platform called a "DDR Obligation," which must be signed by both the requesting participant and the Bank of Canada. Once both signatures have been added to the DDR Obligation and the supervisory node's signature has been obtained, the validation checks are performed by the platform, and the notary node performs its uniqueness checks on the DDR Obligation, including confirmation of correct signatures and review of data attributes (e.g., date, denomination, issuer, owner, etc.). Once these steps are complete, the notary will record the DDR Obligation on its ledger.

With the DDR Obligation recorded on the notary's ledger, the Bank of Canada formally accepts the pledge of Canadian dollars and, after signing the Object and recording it to its own ledger, generates a DDR balance consistent with the amount specified in the DDR Obligation. The DDR Obligation is then "consumed" and a state transition occurs as the DDR Obligation becomes a "DDR Object." The DDR Object contains the same data attributes as the DDR Obligation, and most importantly, assigns ownership of the DDR Object to the pledging participant. The DDR Object is signed by the Bank of Canada and is sent to the participant for review and counter-signature, and subsequently returned to the Bank. Next, the DDR Object is reviewed and signed by the notary and supervisory nodes. The DDR Object is then posted to the proprietary ledgers of the participant and the

Bank of Canada, and the participant is now in a position to initiate exchange of value over the platform using either the atomic or LSM settlement option.

To be clear, a holder of a DDR balance maintains the right to the equivalent value of Canadian-dollar deposits held at the Bank of Canada, and it may redeem the DDR for a cash deposit on its account on demand.

Atomic Exchange

The platform must receive a request from a participant to send an atomic transaction to process a transaction using the atomic settlement option. Once this request has been made, a DDR Object is created with the data attributes identifying owner (sender), receiver, value, issuer[31] (Bank of Canada), and date. The DDR Object is signed by the sender and sent to the recipient for inspection and agreement on the data attributes. If the recipient agrees with the contents of the DDR Object, it counter-signs and returns the Object to the sender. Note that this counter-signature procedure is distinct from current payments practice. The signature of the supervisory node is then sought, and upon obtaining this signature, the sender initiates the last leg of the transaction by submitting the DDR Object to the notary for its signature.

The notary performs its uniqueness checks on the DDR Object, including confirmation of the signatures of the sender and recipient, and inspection of the data attributes. If satisfied, the notary then adds its signature and records the DDR Object on its ledger. The proprietary ledger of the sender is updated with commitment of the DDR Object, and this commitment is broadcast to the recipient, which records the DDR Object on its proprietary ledger to complete the transaction.

LSM Exchange[32]

The LSM settlement option encompasses a centralized queue mechanism that employs an algorithm to perform matching of queued payments at set intervals throughout the day. This payment matching (or offsetting) is intended to help economize on the use of costly DDR in the exchange of value (specifically, pledged Canadian-dollar deposits). Participants faced with non-time-sensitive payment obligations can place them in the queue during the day, rather than holding them in internal schedulers. Payments can be removed from the queue by the sender at any time, except during the running of the matching algorithm, when queue content is temporarily locked down until the algorithm runs its course. The design of the matching algorithm and of the LSM settlement option more generally was inspired by that used in the United Kingdom's CHAPS RTGS scheme. [33]

The steps listed above for initiating a transaction for atomic settlement generally apply to initiation of a transaction for LSM settlement. Importantly, the Bank of Canada plays a central role as "settlement agent" in this option. The Bank is involved in the receipt of DDR from, and transfer of DDR to, participants at the beginning and conclusion of each matching cycle, respectively. This has been dubbed the "inhale/exhale" procedure and represents a major innovation in thinking about how to enable an inherently centralized process like queue operation on an inherently decentralized DLT platform.

[31] The issuer is important for the validation process performed by the platform and helps the platform differentiate between assets when more than one type of asset is being transacted.

[32] **Appendix 2** contains a more in-depth written description, including diagrams, of the LSM settlement option.

[33] See Davey and Gray (2014).

The process of settling a transaction using the LSM option on the Jasper platform consists of adding a transaction to the queue; registering LSM liquidity limits and prefunding ahead of each matching cycle; performing and calculating the results from netting/offsetting transactions in the queue; and paying out the resulting settlement balances.

Adding Transactions to the Queue

This step involves a participant submitting a transaction item to the queue. For each proposed submission, the platform generates a "DDR LSM Object," which, similar to the DDR Object and DDR Obligation explained above, contains data attributes provided by the sender that identify the sender and receiver, the date and, importantly, the LSM liquidity-allocation limit specified by the sender (described below). Once the sending participant completes and adds its signature to the DDR LSM Object, it sends the Object to the Bank of Canada. The Bank will perform the review of the DDR LSM Object, add its signature and return the DDR LSM Object back to the sending participant. The Bank repeats this process for every proposed submission to the queue to keep track of queued items and to assure participants that the payment item has been recorded, as evidenced by the Bank's signature on the DDR LSM Object.

The platform then performs the necessary validation checks before sending the DDR LSM Object to the notary node, where the validation and uniqueness checks are completed, including verification of signatures and review of the data attributes.[34] If the notary node is satisfied, it will add its signature to the Object and record it on its ledger. The DDR LSM Object is then returned to the sending participant to be posted to its ledger and is subsequently broadcast to the Bank of Canada for posting to its ledger. It is important to note that although the DDR LSM Objects are added to the shared database and committed to parties' ledgers, these objects do not undergo a state transition, meaning they do not represent an *expenditure* of DDR. Rather the objects are saved until hopefully being redistributed by the Bank of Canada at the conclusion of a payment-matching cycle (see below).

The "Inhale" Process

A key step in the LSM process is the execution of a payment-matching algorithm that performs multilateral offsetting of queued payments. Before this step takes place, however, participants must first establish their liquidity-allocation limit, referred to as a "Limit Object." The Limit Object, which is established at the discretion of each participant and can be adjusted before any matching cycle, if desired, dictates the maximum amount of a participant's DDR balance that can be drawn on to support settlement of queued transactions when the matching algorithm runs. Establishing the Limit Object and submitting it to the Bank of Canada as the settlement agent for the LSM before each matching cycle is an *automated* process, similar to a preauthorized debit transaction in the traditional payments sense. Introducing this Object is also likely to improve the liquidity efficiency of the platform as a whole, since a participant need not tie up its entire DDR balance every time the matching algorithm runs but can still send other

[34] Note that there is no liquidity check performed at this stage. The liquidity check is performed as part of the "Inhale" process described in the next section, just prior to the matching algorithm running. Moreover, the volume of queued payments has no impact on a bank's ability to make atomic payments until the matching cycle begins.

atomic payments at the same time. In the event that a participant's wallet does not have sufficient DDR balance to meet the value of the Limit Object, the full wallet balance will be deducted as part of the process.

Creation of the Limit Object is a critical first step in what is referred to as the inhale process for the Jasper LSM settlement option. Specifically, the inhale represents the moment when the Bank of Canada redeems the Limit Object for the specified amount of DDR, which is debited from the participant's DDR balance and credited to the Bank of Canada's DDR balance. In the process, the LSM Object undergoes a state transition to become a DDR Object, denoting the transfer of DDR ownership from the participant to the Bank of Canada, which is recorded on the shared database and committed to the applicable ledgers. By redeeming the Limit Object of all participants, the Bank accumulates a pooled DDR balance that can be redistributed as part of the exhale process described below.

Netting of Transactions in the Queue

With the pooled DDR balance in place, the matching algorithm begins by performing multilateral offsetting of all queued transactions. This offsetting will produce a multilateral net settlement obligation for each participant, which could be positive or negative depending on the total value of the queued transactions sent by the participant relative to the total value of the queued transactions to be received. If the former is greater than the latter, the participant has a negative net settlement obligation based on the offset; i.e., it owes funds to the system based on the set of queued payments. In this case, the absolute value of the participant's negative net settlement obligation is compared with the value of its liquidity-allocation limit (i.e., the value of the "Limit Object"). This comparison is performed for all participants with a negative net settlement obligation resulting from the offset. Where the absolute value of all participants' negative net settlement obligations is less than their limit (hereafter referred to as the "liquidity condition being met"), each participant will be owed the difference between these amounts. Participants that had a positive net settlement obligation from the offset will be owed the sum of that position plus the value of their limit. The matching algorithm then concludes.

If instead the absolute value of one or more participants' negative net settlement obligations exceeds their limit (the liquidity condition is not met), a payment elimination process is triggered to see if the liquidity condition can be met for some partial set of queued transactions. The elimination process works by dropping one payment at a time and repeating the offset each time to see if the revised set of queued transactions can produce a scenario where the liquidity condition is met for all participants. This elimination process will continue until either all payments are eliminated by the algorithm or some partial set of queued payment items is identified where the liquidity condition is met. The payment eliminated in each step is the smallest payment item sent by the participant that exhibits the largest violation of the liquidity condition, unless elimination of this payment will cause another participant's liquidity condition to be violated for that offset round. In this case, the algorithm will leave the first payment alone and will move to the next smallest payment of the participant with the largest violation of the liquidity condition. Payments not matched by the algorithm remain in the queue unless otherwise removed by the participant during the next cycle. It also follows that summing across all participants' net settlement obligations following the offset will produce a balance of zero.

For further clarity, a numerical example of the procedures invoked during each matching cycle is provided in **Appendix 2.**

The "Exhale" Process

The exhale process in the Jasper LSM option involves the Bank of Canada paying out the pooled DDR balance that it accumulated during the inhale step to each participant in accordance with the outcome of the offsetting process. Again, if a participant ended the offset with a negative net settlement obligation, then it will be paid out the difference between the value of its limit and the absolute value of its net settlement obligation. On the other hand, if a participant ended the offset with a positive net settlement obligation, then it will be paid out the value of this position plus the full value of its limit. In this way, the pooled DDR balance accumulated during the inhale process is fully paid out via redistribution among participants based on the outcome of the matching cycle. To effect a payout to a participant, a DDR Object is created for each participant and signed by the Bank of Canada following the same process as that for any atomic transaction settled over the platform.

A number of factors will play a role in determining the performance or netting efficiency of the LSM settlement option and the Jasper platform as a whole. These factors include the value and number of transactions that enter the queue, the frequency of the matching algorithm, the value of the liquidity limits established by participants, and whether other transaction-release algorithms may be employed in addition to the matching algorithm (e.g., bilateral offsetting).

To be clear on the process under the LSM settlement option, while a queued transaction is added as a DDR LSM Object to the ledgers of the participants, the Bank of Canada and the notary, it will not affect participants' DDR balance until it is processed by the LSM matching cycle. DDR LSM Objects are archived by the Bank of Canada and by each participant once their status is designated as "completed" (the Bank and the participant can query the DDR LSM Objects). The exhale process reissues new DDR Objects between the Bank of Canada and each participant. All queued LSM payments not executed in that particular matching cycle remain as a DDR LSM Object until processed during a later matching cycle.

It should be noted that the inhale-netting-exhale mechanism is a highly automated procedure, despite the detailed description of each step provided above.

Redeem and Archive

Participants maintain the right to convert their DDR balance into the equivalent value in Canadian dollars pledged to the Bank of Canada at any time during the day, or at end of day. By initiating the redemption process, a DDR Object is generated that includes the amount and other data attributes, including the name of the redeeming participant and the date. The Object is signed by the redeeming participant and then sent to the Bank of Canada for approval. If approved, the Bank's signature is added to the DDR Object, and the Object is sent back to the participant. The Object is then sent to the supervisory node for signature after the platform has performed the validation checks, and it is finally sent to the notary for the usual validation and uniqueness checks. With the notary's signature, the Object is recorded in the notary's ledger and sent back to the participant. The notary and the participant both commit the Object to their respective ledgers, and a broadcast is made to the Bank of Canada. The Bank debits the Object from the participant's "unspent" DDR balance and credits the participant's Canadian-dollar settlement account for the same amount. Finally, the DDR Object with the redeemed balance is archived by the Bank to prevent the participant from spending DDR that is no longer available.

Note that in both Phase 1 and Phase 2, the underlying assumption is that DDR balances would all be redeemed at end of day, and that DDR would not exist for longer than one day. This constraint was imposed to simplify the model, particularly to avoid the need to apportion interest against DDRs rather than the underlying cash deposits. There is no technical reason to prevent the DDRs from existing for longer than one day, nor is there a technical reason preventing the DDR from being funded by an intraday and/or overnight line of credit rather than a cash deposit. DDRs could also be created to represent future claims. All of these are possible extensions to the simple use case of Jasper Phase 2 and are topics worthy of further exploration from a policy and technical perspective for future phases. Of all the processes, redeeming is the only process that must be manually approved by the Bank of Canada.

section 7

Evaluating the Merits of the Phase 2 Platform

Phase 2 provides insight on the effectiveness of the Jasper Project in addressing the following considerations.

Solving for the "Last Mile"

At the time of writing, multiple DLT experiments are occurring within and beyond the financial services ecosystem that involve or conclude with a payment. By constructing a domestic settlement capability that leverages DLT, the possibility of "end-to-end" support of a DLT use case is created. Assuming that interoperability between DLT platforms can be supported, payment settlement that leverages DLT could prove to be a valued capability in multiple markets. This is likely to be the primary benefit of introducing a DLT interbank cash payment platform.

Managing Settlement Risk

The interested reader should refer to **Appendix 3** for a more detailed assessment of the Jasper platform against the PFMIs.

The PFMIs stress appropriate management of settlement risk, including the need for timely payment settlement that represents a final and irrevocable transfer of value between sender and receiver. From an operational standpoint, there is really no defined moment when settlement finality is achieved in arrangements that leverage a proof-of-work consensus protocol such as was used in Phase 1. Rather, in these arrangements, settlement finality is "probabilistic"; a payment recorded on the ledger is never settled with certainty because there is always a non-zero probability that it could fail to remain in a blockchain and recorded. Put differently, the immutability of the recorded transaction may increase with time as additional contracts are added to the ledger, but immutability is technically never certain until the transaction has been included in a block and recorded across all nodes. In contrast, since the Phase 2 Jasper platform currently relies on a trusted third party as a notary node rather than on proof of work in its consensus protocol, settlement is not probabilistic, it is deterministic and should be achieved in a timely manner.

To ensure legal settlement finality, Project Jasper was structured such that exchange of DDR between platform participants would be equivalent to a full and irrevocable transfer of the underlying claim on central bank deposits. If an appropriate legal structure were in place to support this, one could argue that settlement of value exchange on the Jasper platform is ultimately achieved in central bank money (a key requirement of the PFMIs). For all intents and purposes, DDR appears to function as central bank money in the system.[35] Nevertheless, the strength of the legal basis for settlement finality

[35] Moreover, this is a "platform-agnostic" design feature of the Jasper platform. It exists whether on a Corda or an Ethereum platform.

on the Jasper platform warrants further discussion with legal experts in the payments field.

Operational Resilience

Using a private-permissioned ledger involving a small group of regulated financial institutions, consensus can be reached more quickly and easily than with public ledgers. With the Ethereum platform built in Phase 1, consensus was required only among R3 members, which supported timely transaction processing. However, while the platform demonstrated the ability to process 14 transactions per second (similar to throughput levels observed in incumbent systems), this represented maximum capacity. While sufficient for current daily transaction volumes, the Phase 1 platform could introduce constraints on future peak volumes. In contrast, increasing volumes are not expected to be a concern with the Phase 2 Corda platform, in part because it requires only the transacting parties, a supervisory node and the notary node to validate and record a transaction on the shared database. Importantly, the consensus protocol not only achieves more expedited transaction processing, it also supports privacy of information among participating financial institutions (as well as greater settlement certainty, as described above).

This presents an interesting potential trade-off in DLT design which deserves further exploration, and which is likely to evolve as the technology underpinning DLT advances. With this trade-off, the benefits described above can be achieved, but at the expense of the reduced operational resilience of the platform as a whole. As will be explained below, it is not possible to determine whether the Phase 1 Ethereum platform was more "operationally resilient" than the Phase 2 Corda platform. This comparison may boil down to an empirical question warranting further evidence.

The Ethereum platform built in Phase 1 is argued to have demonstrated high-availability at relatively low cost because, given the proof-of-work consensus protocol and the sharing of the full database content among all nodes afforded by the platform, the nodes essentially "backed each other up." If one or more nodes were to become corrupted, the platform could rely on the non-corrupt ledgers of other participants to make the platform whole again. This supports a risk-proofed platform environment without costly investment by each participant to manage a high-availability node. Moreover, the consensus protocol employed in Phase 1 requires agreement among only a majority of R3 members, not all, which means that no participant node could serve as a single point of failure for achieving consensus on proposed transactions.[36] It should be noted, however, that transaction replication across nodes would not obviate the need for each participant to safeguard its personal data in a production implementation. Relying on counterparties for data restoration in the event of an outage should not be assumed to be desirable, or possibly even feasible.

In contrast, the Jasper Phase 2 Corda platform partitions data such that each participant's proprietary ledger reflects only its transaction activity, with

[36] As discussed in **Appendix 3**, this made the arrangement relatively more susceptible to a "51 per cent/selfish miner attack" where a group of nodes could conspire and work together to effectively erase past transactions from the distributed ledger. There were 42 nodes operated by R3 members on the Phase 1 Ethereum platform.

only the notary and supervisory nodes maintaining the full content of the shared database on their respective ledgers. While this approach resolves data-privacy issues that were flagged with the Ethereum platform, the Corda configuration could introduce significant challenges for data replication across the network. Thus, with the Phase 2 Corda platform, each participant would presumably have to invest in a high-availability node to reduce the likelihood and impact of an outage. Moreover, given that the signatures of both the notary and supervisory nodes are needed for consensus in the Phase 2 platform, both must maintain high-availability systems to mitigate single-point-of-failure risk on the platform.

In comparing the full data replication of Ethereum against the state-based channel replication of Corda, it becomes apparent that there are important considerations for both technology platforms from an operational-resilience perspective. These considerations are outlined below.

Use of a private Ethereum solution reduces, but does not eliminate, the need to back up on-chain and off-chain data and to have that data available. In the event of node failure, participants are unlikely to share the ledger data due to confidentiality of information. Though the Ethereum platform in Phase 1 provided backups natively for all on-chain data, the shared ledger format did not meet the confidentiality requirements of participants. The Corda platform in Phase 2 was able to meet confidentiality requirements by only sharing transaction data between participants and the notary. This reduced the data resilience of on-chain data since the full ledger is not backed up on each node. Another option would be to create a backup node as a general service, but this would capture only common transactions data, not private data, and would not support full operational resilience for any individual participant.

To protect all the data stored on the ledger and associated with ledger activities, each participant should invest in data replication and archiving to ensure business continuity and the re-creation of the ledger. Although the full ledger is stored on every node in the Ethereum solution, both the Ethereum solution and the Corda solution would still require individual participants to back up all private data supporting activity on the platform (for example, private keys, customer account information, bank-private data and other data that the application running on the node needs access to but must not be shared with others).

Corda stores each node's data in a standard relational database and moves data using a message queue (MQ) system, both of which can be coordinated using a transaction manager running on the Java Virtual Machine. It is important to ensure that all blockchain platforms offer the same capabilities to protect private data in the event of a node failure.

High-availability is critical for nodes on a Corda platform. Ethereum provides high-availability for the system based on the design of the solution; additional high-availability requirements are limited to add-ons, including keys and applications.

Unlike the Ethereum Phase 1 platform, participants' individual nodes on the Corda platform must be operational to send or receive payments. Corda nodes can queue pending requests to other nodes in the event that a node is forced offline, so that when the node is back online transactions may be processed without loss of integrity of the record, but the impact of an offline node on ecosystem participants could be substantial. It is important that all the components required to support a transaction are highly available and that solutions are in place to ensure transactions can be processed in the event of any issue.

Centralized activities can create a single point of failure on both platforms. While LSMs are an important (some would argue necessary) feature of wholesale settlement arrangements, adding a centralized queuing mechanism to a DLT platform could introduce a single point of failure. If operated by a single node, the payment-matching algorithm would represent a single point of failure for operation of the LSM. It may be possible to mitigate single-point-of-failure risk using a highly available system design. Any DLT solution with a central queue/LSM component will have to consider the need for high-availability of this functionality.

In the Corda solution, the notary plays a centralized role and manages a node that reflects all transactions on its ledger. An outage of the notary node would prevent the processing of all payments. This highlights the need to ensure that the notary node in a production solution has redundancy and failover options available to ensure the availability of any critical centralized service. To address this risk, the notary node could run a BFT[37] or non-BFT notary cluster across a number of servers or even data centres; authorize other participants to run notary nodes under their instruction in the event of an outage; or implement a range of other options to ensure that the service continues to be available if the primary notary is out of service.

The architecture of Corda allows multiple servers to perform centralized processing in parallel. Each server can calculate proposed updates to the ledger (e.g., the results of an LSM algorithm) and propose it to the participants. The first proposal would get processed while the others are ignored as duplicates. This parallel processing removes the risk of a single point of failure for any such function. Other single points of failure can be overcome by straightforward high-availability designs in production.

Operating Efficiency

As described above, DLT solutions are anticipated to dramatically reduce the number of errors and exceptions associated with manual processes that currently support settlement in Canada. Unfortunately, without implementation in a production environment, it is very difficult to evaluate the capabilities of a DLT solution to reduce operating expenses associated with exception processing.

[37] BFT refers to Byzantine Fault Tolerance. The objective is to defend against Byzantine failures, in which components of a system fail with symptoms that prevent some components of the system from reaching agreement among themselves, where such agreement is needed for the correct operation of the system. Correctly functioning components of a BFT system will be able to provide the system's service, assuming there are not too many faulty components.

Global Integration

DLT is continually evolving and there are multiple platforms available in the market, although the number of production implementations is still small. For now, focus is perhaps best placed on the need for standardization, the development of protocols that permit interoperability between other ledgers and networks, and the reduction of computational intensity and costs. Of note, standardization efforts are already under way to address interoperability between DLT implementations. The International Organization for Standardization (ISO) is perhaps the highest body with this objective in mind, while R3 is also focused on trying to deliver interoperability between projects, particularly those involving central banks in different jurisdictions.

section 8

Performance of the Jasper Platform

As with all technology builds, much testing and trouble-shooting went into the design and construction of the Phase 2 platform design. Discussion in this section focuses on a particular component of this testing, involving the use of simulation analysis based on real and artificial payments data for participating banks involved in the exercise.

The simulation analysis focused on two areas.

1. Given the fundamental contribution in Phase 2 of an LSM settlement option employing a complex payment-matching algorithm, one objective of the simulation analysis was to ensure that the algorithm worked as intended in a range of scenarios, including some unlikely but still possible "edge-cases." Importantly, the scope of each of these scenarios (and the data utilized) had to be manageable so that, for each scenario, the project team could manually determine ahead of time what precisely should happen within the queue when the matching algorithm was applied, and evaluate performance using metrics including number of payments processed, number of payments remaining in the queue, etc.[38] Examples of scenarios evaluated include the following:

- Basic checks:
  - o Payments intended for atomic settlement cannot enter the queue under any circumstance
  - o Ability to adjust the frequency of the matching algorithm
  - o Ensure queue contents are locked down while the matching algorithm runs
  - o Ensure participants may use unallocated DDR to transact via the atomic settlement option while the payment-matching algorithm is running
- The full stock of queued payments is released in the first round of offsetting when the liquidity condition is met for all participants.
- Two payments with virtually identical attributes (e.g., sender, receiver and value) in the queue at any given time are treated as separate payment items by the elimination process.
- The elimination process works as intended for both single and multiple breaches of the liquidity condition. The algorithm does not enter into an endless loop if a zero-solution or empty-set of queued payments is identified.
- When testing the liquidity condition, a negative net settlement obligation is only tested against a participant's liquidity allocation amount, not against its total wallet size, when the latter is larger than the former.

[38] For tractability, at most, two runs of the payment-matching algorithm were considered in each of these scenarios, so additional performance metrics around payments delay were not really relevant.

2. A key question that emerges around DLT is whether it can handle the volume of transaction activity observed in incumbent systems today. A second focus of the simulation analysis, therefore, was to build hypothetical data scenarios reflecting both a low-and high-volume day in the current environment to see if the Jasper platform could manage these volumes. Recall that the project team's hypothesis was that the Corda platform may scale better to current and future payment volumes given its use of a notary node rather than a proof-of-work consensus mechanism.

   Since the number of participating banks in Project Jasper is lower than the number of LVTS participants, the data sets had to be augmented with artificial data to bring transaction counts closer to current volumes. This was achieved by taking the existing historical data and generating additional transactions using a recombinant approach, which means randomly pulling a subset of the original transactions and, while retaining the same sender and receiver for each transaction, shuffling the times and values attached to these original data to create new transaction content. The hypothetical operating day in the simulations is from 08:00 hours to 18:00 hours.[39] For each daily transaction file, the following payment attributes were provided: time of payment, sender, receiver, value and a binary indicator variable denoting "1" for time-sensitive and "0" for non-time-sensitive. For the low-volume day, the total number of transactions cleared is nearly 26,000 with a value of $104.5 billion. The high-volume day consisted of clearing nearly 37,000 transactions representing $227.9 billion. The shorter business day used in the high-volume simulation resulted in higher throughput rates than typically experienced by the LVTS. In particular, roughly 61 payments per minute occurred in the simulation, while only 49 payments per minute were experienced by the LVTS on its observed high-volume day of 53,000 transactions in 2016.

   In addition to evaluating the ability of the platform to handle this transaction volume, there is also an opportunity to (loosely) assess the benefit of the LSM in terms of economizing on DDR usage. For this purpose, two scenarios are created for each of the two days, producing four simulation runs in total. The scenarios are described as follows.

   1. Assume that all transactions are time-sensitive and are settled as atomic transactions at their original time stamp. The binary indicator is equal to "1" for all payments. This is equivalent to the settlement model employed by the Phase 1 Jasper platform.

   2. Assume that all transactions are non-time-sensitive and are settled via LSM, and that all are simultaneously offset by the matching algorithm at the end of the day. The binary indicator

[39] In comparison, the current LVTS daily cycle runs from 00:30 hours to 18:30 hours, or eight hours longer than the day contemplated here.

attached to each payment is "0," and *all* payments are entered into the queue at their original time stamp. There is only one run of the matching algorithm in this scenario, which occurs at the end of the day once the transactions for the full day have accumulated in the queue.

Two performance metrics are generated for each simulation. These are "liquidity need" and "average delay." The first is measured in dollars and represents the sum of each participant's maximum daily Canadian-dollar deposit balance at the Bank of Canada (or DDR need) necessary for every payment to be processed by end of day. Participants would need to ensure that their deposit accounts (and DDR wallet balance) contained this amount on a prefunded basis, generally before the start of each day's transaction activity. The second is measured in seconds and reflects, across all payments during the day, the average time in seconds between when a transaction is submitted by a participant (to either the atomic or LSM stream) and when it is successfully processed on the platform (i.e., the DDR Object is recorded on the ledger). For time-sensitive payments, which are assumed to be processed immediately through the atomic settlement option, the delay will be zero and these zero values are counted for the metric. For non-time-sensitive payments, which are assumed to be entered into the queue and processed with some delay using the matching algorithm, this delay will generally be greater than zero.

The liquidity need calculated in Scenario 1 (atomic settlement) is referred to as the "upper bound of liquidity" for all payments to be processed during the day. The liquidity need in Scenario 2 (LSM settlement), which maximizes the power of multilateral netting, is referred to as the "lower bound of liquidity" needed for all payments to be processed.[40] In the extreme Scenario 2, where all payments are multilaterally netted at the same time (and where limit allocations are assumed to be zero), the lower bound will reflect the minimum liquidity level needed for all payments to settle, but delay will be maximized. In Scenario 1, however, it is liquidity need that will be maximized (i.e., the upper bound) and delay will be equal to zero. The difference in dollars between the upper and lower bounds reflects the opportunity for liquidity savings to be introduced by the LSM, as well as other circumstances in place at the time (e.g., actual timing of payments, urgency of payments, limit allocations to the queue, etc.). The liquidity need in a mixed case could fall anywhere in this range and will depend on the parameterization of the LSM and the proportion of payments destined for the LSM, among other factors, as described earlier. This mixed-case scenario was not evaluated as part of the Jasper Project.

Results of the simulations are shown in **Table 1.** The simulations are intended only to demonstrate the effectiveness of the LSM and to ensure that the relative values for each of the metrics considered are as expected (i.e., liquidity need is higher in Scenario 1 than in Scenario 2; average delay is higher in Scenario 2 than in Scenario 1). The notion of "turnover" is also introduced in Table 1, reflecting the value cleared in each scenario per dollar of collateral pledged.

[40] See Leinonen and Soramäki (2003).

Table 1: Results of the Simulation Exercise

Average Day			
	Total liquidity need ($B)	Average delay (sec)	Turnover
Scenario 1 – All time-sensitive ("atomic" only)	19.129	0	5.46
Scenario 2 – All non-time-sensitive ("LSM" only)	11.236	6 hours and 27 mins	9.31

High Day			
	Total liquidity need ($B)	Average delay (sec)	Turnover
Scenario 1 – All time-sensitive ("atomic" only)	34.135	0	6.68
Scenario 2 – All non-time-sensitive ("LSM" only)	26.677	6 hours and 11 mins	8.54

section 9

Other Key Lessons Learned from Our Experience

As with Phase 1, participating organizations learned much about DLT in Phase 2. While not an exhaustive list, some key lessons are presented below.

Collaboration

The financial services environment in Canada is highly concentrated. This environment can be conducive to industry-level projects with clearly articulated objectives. The Jasper Project has benefited from several cohesive forces that have supported the engagement of Canada's largest financial institutions. In addition to industry-level recognition of the potential value of DLT to drive efficiencies and support innovation, many Canadian financial institutions are members of the R3 consortium and have benefited from R3's global perspective and focus on DLT initiatives. Additionally, financial institutions participating in the Jasper Project are all members of Payments Canada and are LVTS participants, as mentioned. Project Jasper represents an opportunity for Canadian industry members to work together to investigate opportunities that will benefit all players in the payments settlement space, and Canadians more generally. The engagement and involvement of the Bank of Canada has also been a strong cohesive force in the Jasper Project.

DLT Platforms

Participants gained a clearer and deeper understanding of Ethereum and Corda, DLT capabilities, and the strengths and weaknesses of multiple DLT platforms.

LSMs

Jasper Phase 2 introduced LSMs to a DLT platform. While this addition allowed all involved parties to better understand the mechanisms through which LSMs operate, it also significantly enhanced the complexity of the platform and its development.

section 10

Concluding Remarks and Next Steps

DLT is an emerging technology that demonstrates great potential in a number of commercial areas, including payments. DLT offers the prospect of an improved user experience in the form of reduced complexity and cost, greater transparency, enhanced product and service customization, and improved access for consumers of financial services around the world. In the area of payment clearing and settlement, DLT offers the prospect of reduced operating costs and potential improvements in achieving operational resilience.

Introduced in March 2016, Project Jasper is an exploration into the use of DLT for settling wholesale interbank payments in Canada. Two phases of the project have been successfully completed and have delivered substantial understanding and accomplishments regarding this emerging technology. Specifically, the project team has had the opportunity to explore and compare the capabilities of two distinct DLT platforms—the Ethereum and Corda platforms—to build out this settlement functionality, and to build and implement an LSM in the form of a central queue on the Corda platform. The analysis to date suggests that DLT platforms that employ a proof-of-work consensus protocol, as was built in Phase 1, do not deliver the necessary settlement finality and low operational risk required of core settlement systems. Phase 2 built a distributed ledger platform that employed an alternative consensus model using a "notary node" and could deliver improvements in settlement finality, scalability and privacy, but does not adequately address operational risk requirements. Further evaluation of DLT is still required as solution providers enhance their offerings and introduce enhancements to satisfy the PFMIs that must be met by any wholesale interbank payments settlement system.

The LVTS is a very low-cost system to operate and supports efficient use of collateral pledged by participants. It will be challenging for any DLT-based system to process payments more efficiently than the LVTS. However, this may be too narrow a perspective to consider when evaluating the overall efficiency of the settlement platforms. Currently, all participants in the LVTS expend significant resources in back-office reconciliation efforts to verify and validate the information they receive from the LVTS. If a DLT-based settlement system is able to reduce back-office reconciliation efforts, significantly more cost savings could be realized across all of the participants, which would reduce the overall cost of operation.

It is also possible that a cash-based settlement-solution system such as Jasper could prove to be the core upon which other distributed ledger platforms can be built to perform tasks such as settlement of financial asset transactions, manage syndicated loans and support trade finance. If such an ecosystem for payment and settlement can be fully realized, there could be significant benefits for the whole financial sector and the economy overall.

Table 2: Future Opportunities for Exploration

Opportunity	Description
Connect domestic settlement capability with another domestic DLT/blockchain use case	• What advantages would a DLT solution based on central bank-issued digital cash equivalents provide for related processes, such as securities settlement? What experiments could be conducted to join the Jasper settlement platform with DLT initiatives in capital markets?
Central bank to central bank digital cross-border currency exchange	• How might a single-currency DLT zone best be connected with other currencies? What are the optimal models for addressing friction in the current cross-border payment ecosystem?
Define the operational (non-technical) considerations that need to be addressed to support a DLT settlement solution	• What policy considerations and changes need to be addressed? What new or reformed legal constructs are required? • How would a production system be governed, maintained and supported? • Explore non-cash collateral, including high-quality liquid assets (HQLA) and other asset holdings in addition to cash?

DLT solutions have developed significantly since Bitcoin was launched in 2009, but it will take further evolution before they may become ubiquitous in the financial sector. Our analysis has shown that proof-of-work systems, like those built in Phase 1, do not deliver the necessary settlement finality and low operational risk required of core systems by the PFMIs. Phase 2, however, built a distributed ledger system that could deliver settlement finality and greatly improve the platform in terms of scalability and privacy, but contains features that introduce risks to operational reliability. While further analysis is still required, the industry is improving the ability of DLT platforms to observe the PFMIs, which must be met by any wholesale interbank payments system.

Without question, the success of the Jasper Project to date has been due to the close collaboration among all of the partners involved in achieving this common goal: staff from Payments Canada, the Bank of Canada, R3 and many major Canadian banks have been involved at all stages and levels of the project.

section 11

References

Buterin, V. 2013. Ethereum Whitepaper. Available at **https://github.com/Ethereum/wiki/wiki/White-Paper.**

Bank for International Settlements. 1997. Real-Time Gross Settlement Systems. CPSS Report. Available at **http://www.bis.org/cpmi/publ/d22.pdf.**

———. 2005. New Developments in Large-Value Payments Systems. CPSS Report. Available at **http://www.bis.org/cpmi/publ/d67.pdf.**

Brown, R.G., J. Carlyle, I. Grigg, and M. Hearn. 2016. Corda: An Introduction. Available at **https://static1.squarespace.com/static/55f73743e4b051cfcc0b02cf/t/57bda2fdebbd1acc9c0309b2/1472045822585/corda-introductory-whitepaper-final.pdf.**

Canadian Payments Association. 2017. By-law No. 7 Respecting the Large Value Transfer System. Available at **http://laws.justice.gc.ca/PDF/SOR-2001-281.pdf.** Version current to June 5, 2017.

Chapman, J., R. Garratt, S. Hendry, A. McCormack and W. McMahon. 2017. "Project Jasper: Are Distributed Wholesale Payment Systems Feasible Yet?" Bank of Canada Financial System Review. June. Available at **http://www.bankofcanada.ca/wp-content/uploads/2017/05/fsr-june-2017-chapman.pdf.**

Davey, N. and D. Gray. 2014. "How Has the Liquidity Saving Mechanism Reduced Banks' Intraday Liquidity Costs in CHAPS?" Bank of England Quarterly Bulletin. Second quarter. Available at **http://www.bankofengland.co.uk/publications/Documents/quarterlybulletin/2014/qb14q207.pdf.**

Garratt, R. 2017. CADCOIN versus Fedcoin. R3 Report. Available at **https://www.r3.com/wp-content/uploads/2017/06/cadcoin-versus-fedcoin_R3.pdf**

Leduc, S. 2017. "Upgrading the Payments Grid: The Payoffs Are Greater Than You Think." Remarks to 2017 Payments Canada Summit. May. Available at **http://www.bankofcanada.ca/wp-content/uploads/2017/05/remarks-250517.pdf.**

Leinonen, H. and K. Soramäki. 2003. Simulating Interbank Payment and Securities Settlement Mechanisms with the BoF-PSS2 Simulator. Bank of Finland Discussion Paper, No. 23-2003. Available at **https://www.suomenpankki.fi/globalassets/en/rahoitusjarjestelman_vakaus/bof-pss2/documentation/bof_dp_2303.pdf.**

Nakamoto, S. 2008. Bitcoin: A Peer-to-Peer Electronic Cash System. Available at **https://bitcoin.org/bitcoin.pdf.**

Norman, B. 2010. Liquidity Saving in Real Time Gross Settlement Systems – An Overview. Bank of England Financial Stability Paper No. 7. May. Available at **http://www.bankofengland.co.uk/financialstability/Documents/fpc/fspapers/fs_paper07.pdf**

World Economic Forum. 2016. The Future of Financial Infrastructure: An Ambitious Look at How Blockchain Can Reshape Financial Services. Future of Financial Services Series. August. Available at **http://www3.weforum.org/docs/WEF_The_future_of_financial_infrastructure.pdf.**

Appendix 1 Jasper Phase 1 Overview

Project Jasper was launched in March 2016 as a research project with the R3 Lab and Research Center (R3 LRC) in partnership with Payments Canada and the Bank of Canada to assess the potential application of DLT within the payment infrastructure of Canada's settlement process. Project Jasper is an ongoing joint collaborative research effort that evaluates the question "What if we were to issue, transfer, and settle central bank-issued assets (denominated in Canadian dollars, or CAD) on a distributed ledger network?"

Phase I efforts involved building a framework to evaluate the suitability of a central bank-issued asset transferred between participants on a distributed ledger network for CAD domestic large-value wholesale payments. Project Jasper efforts have focused on evaluating the suitability of DLT for the issuance, transfer and settlement of CAD payments from business, technical, operational, monetary policy and regulatory perspectives. Results and insights will provide valuable input on domestic payments regulation, financial system stability and monetary policy research.

Phase I Goals

- Build a proposal for a central bank-issued digital currency, including issuance, transfer, settlement and destruction
- Leverage rapid prototyping to test and validate business, operational and technical hypotheses; modelling will focus on the on-ramp/off-ramp access points to the central bank ledger

Guiding Hypotheses

We identified the following hypotheses to guide our research:

Area of Focus	Description
Cost	The overall cost of the system per participant will be less with a DLT solution (operating costs, collateral costs etc.) than with a centralized system
Resilience	A DLT system will be more resilient than a centralized system due to the distribution of technology across participants
Accessibility	Barriers to entry will be reduced in a DLT system relative to a centralized system, allowing for an increased number of direct participants
Control	Information will be protected/released in a more granular and policy-determined manner

Proposal to Prototype

DDR Asset Model

- The underlying asset is not a token but a ***depository receipt*** that reflects the balance of the claim between the two counterparties; this ensures that there is no impact on the monetary supply.
- The design decouples the business rules from the movement of funds in order to maximize flexibility and allow for stages of implementation with an incremental realization of benefits (ROI).
- DDR is NOT a one-to-one token for $1 CAD on deposit. It is a receipt conveying title to a net balance of CAD payable on redemption by the Bank of Canada.
- Fractional DDR balances may be divisible to two decimal places.

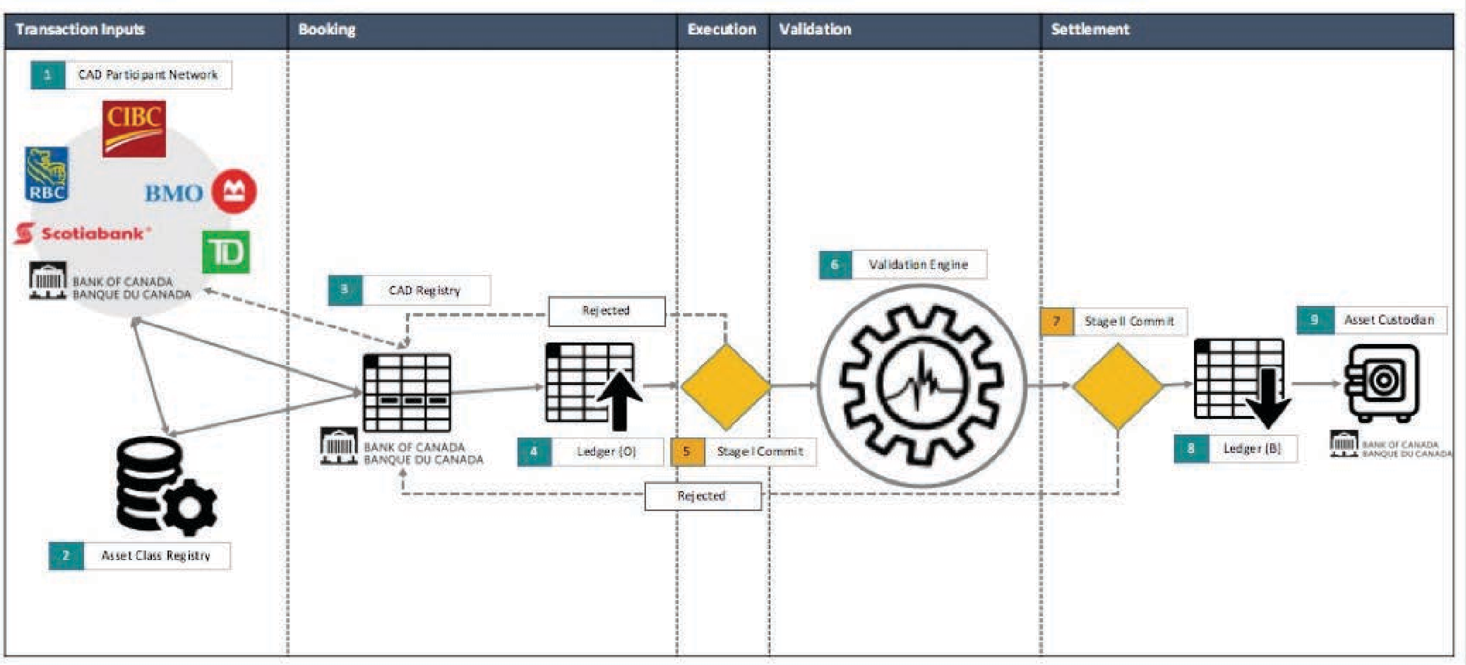
Shared Transaction Life Cycle

A transaction is defined as an **action** taken according to **validated instructions.** The **DDR life cycle** consists of **multiple sets of transactions.** See figure below.

Design Overview – Phase 1

- The Jasper prototype was built using Ethereum Open Source Code (written in Solidity).
- The Jasper prototype used the proof-of-work mining built into Geth for consensus and transaction validation, but added customizations to eliminate the costs of ether for the pilot.
- Each bank was given access to its own node and the ability to create its own accounts and send transactions/deploy contracts.
- Only the R3 node could accept transactions.

Shared Transaction Life Cycle



Prototype Functionality

- Front to back application with shareable code—code fully published and works as defined in specs
- UTXO design of contracts for portability to Corda—transactions-based structure for all end-points
- Atomic transactions grouped for asset movement—single call used for group transactions and setting
- Demonstration of multi-signature —temporary escrow of funds before sending
- Reusable middle layers for all further iterations—foundation work for extending our infrastructure
- Simple build of a user management layer for private key storage—key storage is a significant element of blockchain identity, and a rough version of this has been created to support this use case and others
- Exploration of different clients and layers—we've explored other middle layer and blockchain clients for Ethereum

1 Transaction agent deployment	Bank of Canada (BoC) deploys and takes ownership of the transaction agent smart contract. The transaction agent will be the autonomous 3rd party in all transactions (pledging, redeeming and transferring).
2 Wallet creation	BoC instructs the transaction agent smart contract to create and own wallets on behalf of each individual bank. All future interactions involving wallets need to be initiated by the respective banks.
3 Pledge	A bank can create a request via the transaction agent to pledge CAD in exchange for DDR in its wallet. The BoC can accept the request once it handles collateral off-chain, and DDR will be credited to bank's wallet.
4 Transfer	A bank can initiate a transfer of DDR through the transaction agent, which debits and credits the wallet.
5 Redemption	A bank can create a request via the transaction agent to redeem DDR from its wallet in exchange for CAD. The BoC can accept the request once it handles collateral off-chain; DDR is in escrow before being destroyed.
6 Cancel	All pledge and redeem transactions in ledger have initial state of PENDING and can only be changed once to a new state of ACCEPTED, REJECTED or CANCELLED. Banks that have initiated a transaction can cancel it if pending, and the transaction agent credits the wallet if needed.

Findings

Considerable centralization and organization were required. Given the early stage of the technology, capabilities were limited, and we were required to implement additional measures to enable DDR transactions.

- Account/Key management—Keys are often managed through a central service that is separate from the Ethereum nodes.
- Node management—A separate node is required for every bank to separate the storage of accounts. If a participant has access to an account, it can potentially use "Ethereum.accounts" and view all active users.
- Slow mining and expensive network—R3N1 still uses Proof of Work and it isn't that much faster than the public network.
- Frequent errors—Geth often has issues with maintaining connectivity.
- Money held in escrow—For multi-signatures to occur, money for redemptions is held in escrow until redeemed. Pledging is easier because these balances are updated separately.
- Unreliable events in Ethereum—Certain events don't execute consistently.
- No updates of contracts—The current solution does not yet have the capability to update/upgrade contracts that are already published.
- **Local storage of addresses**—The current solution has smart contracts published by a trusted third party and executed by that party.
- **No array calls**—Solidity does not allow for easy dashboard array calls of data: versions of contracts from different submit times can be viewed as they're stored with the state, but it is challenging to pull large chunks of data.

Data and Transparency

Currently the BoC and Payments Canada observe the full network of payment flows, but private banks do not. Private banks track their net positions with all counterparties, but they do not know their counterparties' vectors of net positions, which could suggest that their counterparties will face liquidity shortages. Under current practices, it would be quite difficult for the BoC to share full information on payments flows to banks in real time. The DL, on the other hand, delivers this capability. However, there are significant privacy trade-offs that conflict with collateral management advantages. Ultimately, the DL offers advantages for sharing information, but decisions will have to be made as to how much data should be made available to which participants and when.

Appendix 2

Technical Overview of Jasper Phase 2

In Phase 2, the prototype was built using the Corda platform. Corda was selected after a detailed review of the options. Unlike some other DLT platforms, Corda has been designed from the ground up by and for financial institutions and delivers a DLT solution engineered to meet the unique privacy requirements of regulated financial institutions.

The capabilities of the Jasper platform were expanded in Phase 2 to include the capability to support multiple settlement options. In addition to atomic settlement, the platform included a liquidity-saving mechanism (LSM) that supported netting of payments.

Phase 2 High-Level Corda Design

The Jasper prototype that was created over Phase 1 and 2 supports eight processes. In Phase 1 we created the ability for participants to pledge, fund a DDR wallet, exchange payment, redeem DDR and archive DDR. In Phase 2 we expanded the exchange capabilities to include both atomic and deferred net settlement. Details of each process, the triggers, precondition checks and post events for these stages are detailed in the table below.

A unique shared ledger approach

Blockchain-inspired: takes best attributes from Bitcoin, Ethereum, and others.

Enterprise grade: built specifically for financial markets.

Data privacy: transactions info propogated only to relevant nodes.

Consensus: achieved at individual deal level, rather than system level. Supports a variety of consensus mechanisms.

Regular-focused: design directly enables regulatory/supervisor observer nodes.

Smart contract: strong link between legal prose and smart contract code.

Easy integration: reuse existing developer skills and make integration with bank systems easy and safe. Query and join the ledger to existing DBs with SQL, and code contracts in modern, standard languages like Java.

Process Step	Description	Initiator	Trigger Event	Pre-conditions	Post-conditions
Pledge	Pledge of CAD balance to the BoC	FI Participant	Submit DDR Obligation to BoC	DDR Obligation is correctly issued by FI	DDR Obligation received by BoC node
Generate/ Fund	Generate CAD digital depository receipts	BoC	Acceptance of DDR Obligation from FI	DDR Obligation reviewed by BoC	DDR issued to requesting FI
Exchange - Atomic	Exchange DDR	Sending FI	Send DDR to Receiving FI	DDR available to support exchange	DDR consumed for Sending FI; DDR transferred to Receiving FI
Exchange – LSM	Add Payment – submit payment to queue	Sending FI	Send DDR LSM to Bank of Canada.	DDR is available to fund LSM exchanges	DDR LSM consumed on Sending FI -DDR LSM transferred to PC
	Netting – run LSM algorithm	LSM algorithm	LSM cycle time	DDR LSM Objects in LSM Queue available for aggregation	DDR LSM Objects reviewed with proper balance logic to conduct netting; triggers Exchange (LSM) Execute
	Execute – exchange net payments	PC	Exchange (LSM) Netting	Netting algorithm provides instructions for atomic exchange	Instructions sent to relevant FIs to support multiple party atomic exchange
Redeem/ Archive	Request redemption of DDR and archive	Sending FI	Submit DDR Obligation (Redeem) to BoC	DDR available for redemption of DDR Objects	DDR Object consumed on Sending FI -DDR Obligation (Redeem) sent to BoC -Copy of DDR Obligation (Redeem) remains with Sending Bank -DDR Object of requested amount sent to BoC where it is consumed (but saved)
Return	Return new net balance of DDR at BoC	BoC	Redeem accepted by BoC manually	Redeem DDR available	DDR Obligation (Redeem) returned to Sending FI for confirmation of archived DDR.

Corda State Objects and Their Data Attributes

The table below describes the attributes for each of the state objects required to support settlement on the Jasper Platform:

State Object Name	Field Name	Value
DDR	Issuer	Text
	Issuer Date	Date (dd-mm-yyyy)
	Amount	Float
	Currency [CAD by default]	ENUM: CAD
	Owner	Text
DDR **Obligation**	Requester	Text
	Requester Date	Date (dd-mm-yyyy)
	Amount	Float
	Currency [CAD by default]	ENUM: CAD
	Owner	Text
	Type	ENUM: Pledge, Redeem
	Status	ENUM: Request, Approved
DDR LSM	Requester	Text
	Requester Date	Date (dd-mm-yyyy)
	Amount	Float
	Currency [CAD by default]	ENUM: CAD
	Owner	Text
	Status	ENUM: Request, Executed
	Max LSM Daily Amount	Float
	Max LSM Cycle Amount	Float

Jasper Phase 2 LSM Requirements

Jasper Phase 2 requirements were created by settlement experts from the Bank of Canada, Payments Canada and R3. These requirements are summarized in the table below:

Function	Description
Queue Attributes	• The LSM queue can be configured by Payments Canada to settle on a multilateral basis every X minutes • The Regulator (i.e., notary) must be able to configure the duration of the multilateral exchange cycle (i.e., global system-wide parameter) • Upon the completion of each matching cycle, the system will update and display the projected date and time of the next matching cycle to all users • The user can submit payment items to the queue throughout a matching cycle • The queue depth should be unlimited in size • The user should be able to see all payment items that have been submitted to the LSM queue (i.e., expected debits owed by user); all payment items in which she is the intended receiver via the LSM queue (i.e., expected credits owed to user) • Throughout the matching cycle, the user should be able to request a projection of her multilateral net position based on the current state of the payment items in the queue • The user can set constraints on submitted payment items • The user can specify the maximum amount of funds to contribute to a matching cycle (i.e., Max Allowed LSM Matching Cycle Amount) • A maximum on the value of payments sent to all other users in excess of payments received from all others during the course of the day (i.e., Max Allowed LSM Daily Amount). This type of global limit was not implemented in Phase 2 • Details of payment items in the queue to be displayed are: Sender, Receiver, Amount (to two decimals), Type (debit or credit), Status, Date/Time item was sent to the queue, amount of time the item has been in the queue
Netting Algorithm	• The netting algorithm will attempt to net all payment items in the queue on a multilateral basis and according to any user-specified constraints in order to determine the long (credit) or short (debit) position for each participant with queued payment items.

Function	Description
Netting Algorithm	• The algorithm begins matching payment item(s) with other queued item(s) immediately after the expiration of the matching cycle time frame • Once the netting algorithm starts to execute, any payment item submitted to the queue after execution will be ignored and placed in the next matching cycle • The algorithm will then determine, based on the short positions, which participants must provide liquidity and validate that: ○ Sufficient DDR funds exist: the user has sufficient DDR funds in their wallet ○ Max Allowed LSM Matching Amount: The DDR amount from the user is less than or equal to their pre-set limit for each LSM matching cycle; and ○ Excess Multilateral Daily Limit: The DDR amount from the user is less than or equal to their total pre-set limit for all LSM cycles that day • Assuming sufficient funds exist and no constraints are violated, the system will generate a new payment using the output of the algorithm to redistribute liquidity from the users' wallets who are short to the users' wallets who are long • The wallets of users in a short position after the algorithm has matched items should be frozen when the algorithm is accessing users' wallets to exchange DDR based on the netted positions. This is to ensure that wallet positions do not change after the algorithm determines netted positions, but before these positions are settled
Exchange	• For each user in a short position: The system will create new payment transaction(s) that draws funds from user's wallet and sends funds to the central bank wallet • For each user in a long position: The system will then create new payment transaction(s) that withdraws from the central bank wallet and sends fund to the users in a long position • The above transactions are zero-sum impact on central bank wallet in that the total funds from users equals the total funds to users for each cycle • The algorithm will also mark as complete all of the queued payment items that were successfully netted during the matching cycle • All of the payment items successfully completed as a matching cycle should be easily identified and included in the respective users' transaction logs

Function	Description
Exchange	• The transaction logs should show the status (completed, rejected, removed) of: ○ user-generated atomic transaction ○ payment items that were algorithmically netted and matched ○ system-generated atomic transaction (to/from central bank wallet) resulting from algorithmic netting obligations ○ payment items submitted to the queue

Illustration of the LSM Flow

Simple Illustration of the Netting Algorithm

The following illustration considers only three banks and assumes that each bank has submitted one payment for each other bank into the netting queue. In reality, and in the phase 2 simulation, banks submit multiple payments to some or all other banks into the queue during a queueing period. However, the basic principle on which the netting algorithm works is captured by a simple case.

Start with a matrix A, which represents all the payments in the queue at the end of the queueing period. Element *ij* is the payment from *i* to *j* in the queue.[41] So, for example, in the matrix below, bank 2 inserted an $8 payment to bank 1 into the queue at some point during the queueing period.

$$A = \begin{pmatrix} 0 & 10 & 1 \\ 8 & 0 & 2 \\ 3 & 1 & 0 \end{pmatrix}$$

We also have a vector of limits. These are either the user limits or DDR available, whichever is lower. Here, all three banks have limits equal to 2.

$$l = \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix}$$

A and *l* are the inputs to the process.

[41] In the actual algorithm used, each element *ij* in matrix A is the vector of payments from *i* to *j* in the queue.

Now the algorithm runs. The first thing it does is compute the aggregate amount each person owes and is owed. The amount each bank owes is the row sum: r_i. The amount each bank is owed is the column sum: c_i. The net obligation of each bank is $n_i = r_i - c_i$.

$$\text{Let } n = \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix}$$

In the above example:

Owes	Owed	Net Amount Owed
Bank 1 owes 11	Bank 1 is owed 11	0
Bank 2 owes 10	Bank 2 is owed 11	-1
Bank 3 owes 4	Bank 3 is owed 3	1

Thus, we compute the net obligations (owes minus owed) as $\text{n=}\begin{pmatrix} 11 \\ 10 \\ 4 \end{pmatrix} - \begin{pmatrix} 11 \\ 11 \\ 3 \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \\ 1 \end{pmatrix}$

Then we check if $n \leq l$. Here the largest obligation is 1, which is less than 2, so everything is fine and all the payments are processed by drawing $1 from bank 3 and giving it to bank 2.

Sometimes (for different payment files), however, we will obtain a violation of the liquidity constraint, i.e., $n \leq l$. In that case, we need to remove payments in order to find a valid solution.

In order to maximize the total value cleared by the queue, we could consider all of the $2N$ -1 possible combinations of payments (where *N* is the total number of payments in the queue) and start by removing the smallest value combination and seeing if this allows a valid solution. If not, we could put that combination back in the queue and move to the next smallest combination and so on. This would eventually lead to a solution, but for large payment files the number of cases is too large for this approach to be computationally feasible.

Instead we use a simpler algorithm that seeks to increase the total value of liquidity cleared by targeting small payments for removal from the queue first.

Process Steps

1. Start with a matrix that represents all payments to be used in the matching cycle.
2. Element ij represents a payment from i to j.
3. l represents a vector of limits, or users' balances, whichever is lower.
4. The amount each bank owes is the row sum r_i and the amount each bank is owed is the column sum c_i.
5. Hence, the net obligation of each bank can be represented as $n_i = r_i - c_i$.
6. Next, we check if $n \leq l$, if all $n \leq l$ the netting was successful; otherwise we must proceed to the next step.
7. If $n > l$ we identify the single lowest payment from the party with the largest liquidity shortfall, and ask if removing this payment would put any other bank into a liquidity shortfall. If not, we remove the payment and start over. If it does, we leave that payment in the queue and remove the next smallest payment from the party with the largest liquidity shortfall.
8. We proceed through this process until we find a solution or have exhausted all possibilities.

We finish up with another example, to illustrate the process just described. Let

$$A = \begin{pmatrix} 0 & 10 & 1 \\ 8 & 0 & 3 \\ 5 & 2 & 0 \end{pmatrix}$$

We also have a vector of limits. These are either the user limits or DDR available, whichever is lower. Here, all three banks have limits equal to 2.

$$l = \begin{pmatrix} 2 \\ 2 \\ 2 \end{pmatrix}$$

Once again, A and l are the inputs to the process.

Now the algorithm runs. The first thing it does is compute the aggregate amount each person owes and is owed. In the above example:

Owes	Owed	Net Amount Owed
Bank 1 owes 11	Bank 1 is owed 13	-2
Bank 2 owes 11	Bank 2 is owed 12	-1
Bank 3 owes 7	Bank 3 is owed 4	3

Thus, we compute the net obligations (owes minus owed) as $\mathbf{n}= \begin{pmatrix}11\\11\\7\end{pmatrix} - \begin{pmatrix}13\\12\\4\end{pmatrix} = \begin{pmatrix}-2\\-1\\3\end{pmatrix}$

Then we check if $n \leq l$. Here the largest obligation is 3, which is greater than 2 so we remove the smallest payment from bank 3, which is the $2 payment from bank 3 to bank 2. After this payment is removed

$$A = \begin{pmatrix}0 & 10 & 1\\8 & 0 & 3\\5 & 0 & 0\end{pmatrix}$$

Now $n = \begin{pmatrix}11\\11\\5\end{pmatrix} - \begin{pmatrix}13\\10\\4\end{pmatrix} = \begin{pmatrix}-2\\1\\1\end{pmatrix} \leq \begin{pmatrix}2\\2\\2\end{pmatrix} = l,$ so we are done. We clear all but $2 worth of payments in the queue, and we cannot do any better than that.

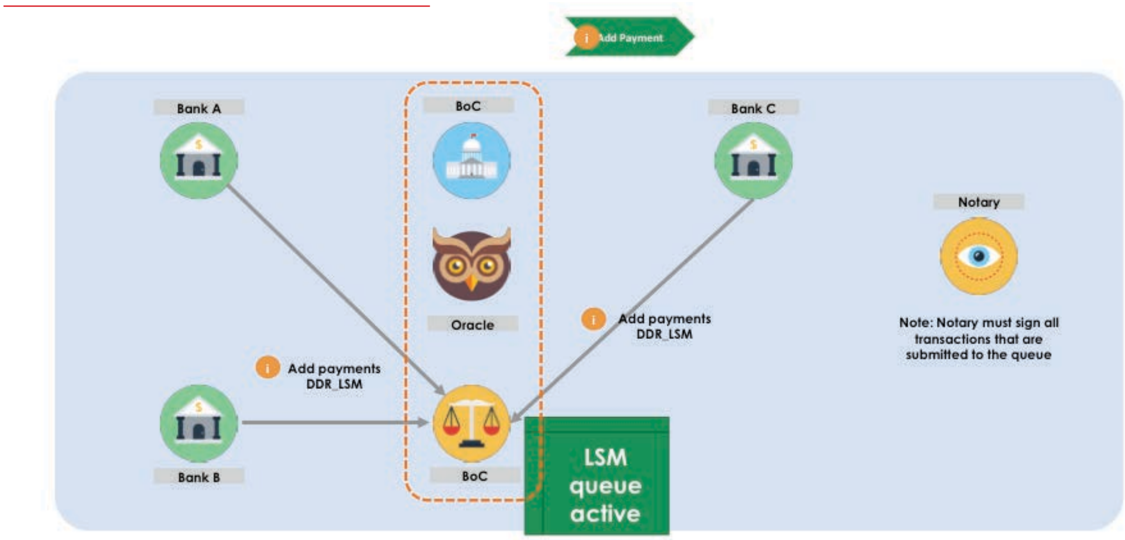
Strengths

- Converges in at most N steps
- Does not stop if attempting to remove a payment would make the recipient "negative" (i.e., l-n<0). It just leaves that payment in the queue and moves on.

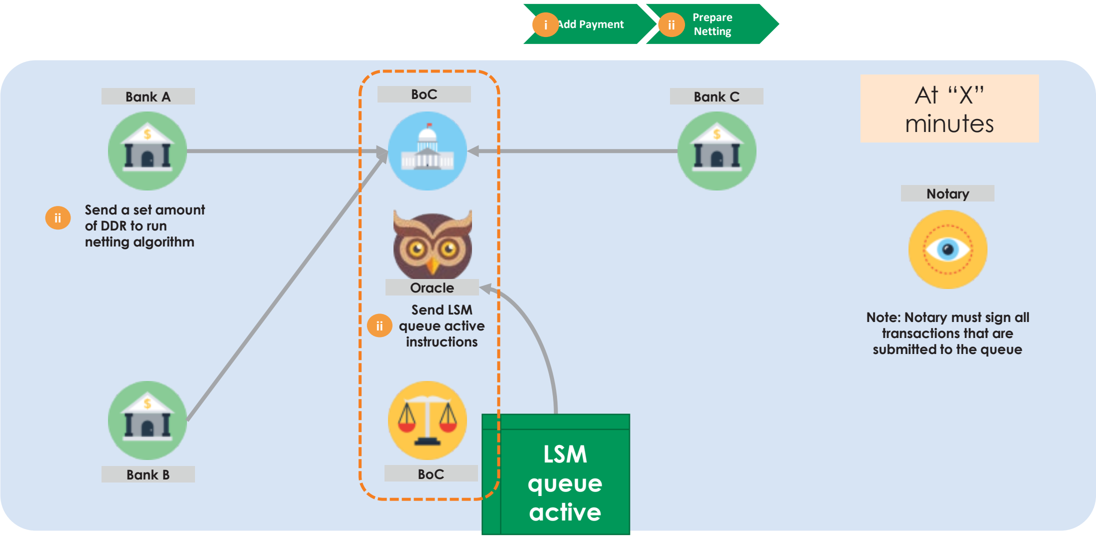
Weaknesses

- Some scenarios produce a result worse than the one derived from the exhaustive method described above.

Step Through LSM_Queue (i)



Step Through LSM_Queue (ii)

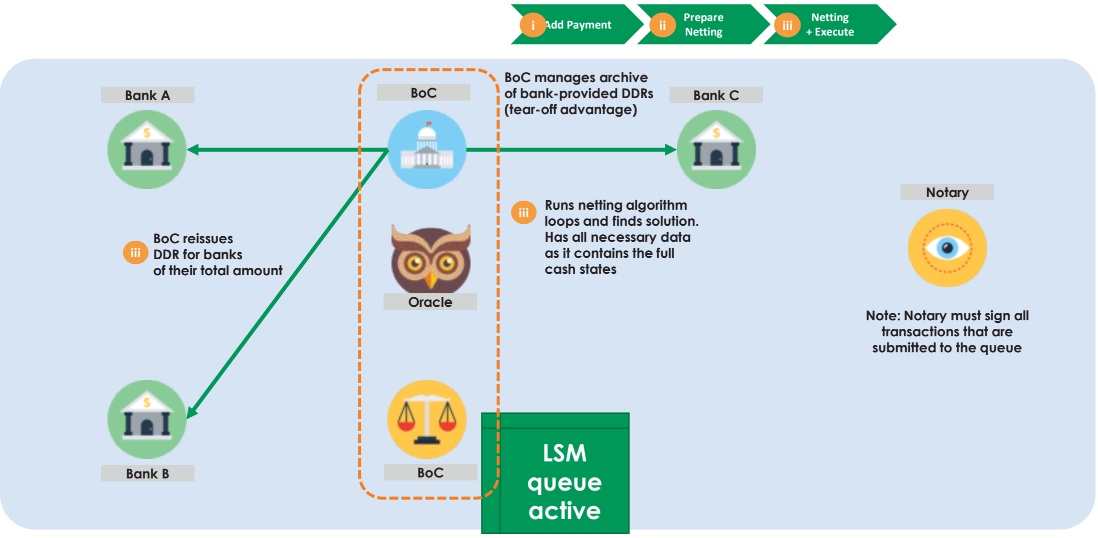


ii Send a set amount of DDR to run netting algorithm

ii Send LSM queue active instructions

Note: Notary must sign all transactions that are submitted to the queue

Step Through LSM_Queue (iii)



BoC manages archive of bank-provided DDRs (tear-off advantage)

iii Runs netting algorithm loops and finds solution. Has all necessary data as it contains the full cash states

iii BoC reissues DDR for banks of their total amount

Note: Notary must sign all transactions that are submitted to the queue

Jasper Phase 2 Platform Use Cases

The requirements called for the following use cases to be tested and validated using the Jasper Phase 2 platform.

Execution	Testing / Validation
Banks: Set individual LSM Limit	• Banks and BOC can confirm limits have been set
Banks: **Pledge** DDRs from BOC	• BOC will accept and generate pledges from banks
Banks: **Transfer** DDRs between each other atomically	• Banks confirm DDR received by parties • Check validity through transactions
BOC: Activates **LSM Netting**	• Banks confirm DDRs have been sent • Banks confirm LSM DDR balance zeroed and show transactions
Banks: Transfer DDRs to **LSM** Queue	• Banks confirm LSM DDR balance increases appropriately • BOC confirms there is not a queue of LSM transactions
Banks: **Redeem** DDRs from BOC	• BOC will accept and redeem cash out of the ecosystem

All use cases were successfully validated.

Recognition

We would like to highlight the core development contributions of the following individuals:

Resource	Sprint Activities	Role
Francis Guttridge	UI, APIs, Full Cash framework, Limits, Heartbeat design, Core Corda Design, Netting Algorithm in Kotlin	Principle Developer
Alex Shpurov	LSM Queue Inhale/Exhale and Atomic Flow Deployment on Staging Environment	Developer
Kevin Kofler	Pledge & LSM Redeem Contract	Developer
Michael DeLuca Rod Garratt Richard Green	Netting Algorithm in Python	Developer
Elena Litani Andrew McCormack Ross Nicoll Dinesh Shah Clark Thompson Clemens Wan	LSM Queue Design and checks with business viability	Platform Stream Design / Architects

Appendix 3

PFMI Considerations

Overview

The Principles for Financial Market Infrastructures (PFMIs) were published by the Committee on Payment and Settlement Systems of the Bank for International Settlements in April 2012. Seven principles, 4, 5, 7, 8, 9, 17 and 18, were defined as in scope for review. These relate to collateral, credit risk, liquidity risk, settlement finality, money settlements, operational risk, and access and participation requirements. The 11 remaining principles relate primarily to governance and legal aspects that would only be relevant for a production-level FMI.

Project Jasper meets the key aspects of the PFMIs concerning collateral, credit risk, money settlements and liquidity risk. This result was expected given that the design of Project Jasper relies heavily on linkages to the current wholesale payments system in Canada (LVTS), which is PFMI-compliant. The Project Jasper Phase 1 and 2 platforms do not fully meet the PFMIs concerning settlement finality, operational risk, and access and participation requirements.

Partial Assessment of Project Jasper Relative to the PFMIs

Principle 4: Credit Risk

An FMI should effectively measure, monitor, and manage its credit exposures to participants and those arising from its payment, clearing, and settlement processes. An FMI should maintain sufficient financial resources to cover its credit exposure to each participant fully with a high degree of confidence.

Transfers of DDR are transfers of a claim on central bank money, for which there is no credit risk because the central bank is not subject to default. The LSM introduced in Phase 2 executes payments on a net basis subject to amounts of liquidity provided in advance by each participant and hence does not introduce any credit risk. Overall, nothing in the proof-of-concept design was identified as fundamentally incompatible with the credit risk principle.[42]

One caveat, however, is that this presumes a legal structure is in place that ensures that a transfer of DDR is equivalent to a full and irreversible transfer of the underlying claim on central bank money so that, in the event that a participant defaults, that participant and its creditors only have claim to an amount corresponding with their DDR position at the time of default.

[42] The aspects of the PFMIs relating to establishing a robust framework to manage credit exposures were deemed not applicable to the proof of concept.

Principle 5: Collateral

An FMI that requires collateral to manage its or its participants" credit exposure should accept collateral with low credit, liquidity, and market risks. An FMI should also set and enforce appropriately conservative haircuts and concentration limits.

In Project Jasper, DDRs are a digital representation of Canadian-dollar deposits held in accounts at the Bank of Canada. DDRs are issued to a participant's wallet by the Bank of Canada following a transfer of that deposit amount from the participant via an LVTS payment. An LVTS payment has no credit, liquidity, or market risk; it provides immediate finality of payment to the Bank of Canada upon receipt and cannot be unwound or revoked under any circumstances. Project Jasper therefore meets Principle 5, since the DDRs are backed by Canadian-dollar deposits held at the Bank of Canada, transferred through the LVTS.[43]

It is worth mentioning that the procedure used to fund DDRs in Project Jasper was adopted as a matter of convenience, because it works without making any modifications to existing infrastructure; participants simply need to make an LVTS payment to the Bank of Canada to acquire DDRs. In a production system, banks would likely acquire DDR by allocating collateral directly in the same way as collateral is allocated to the LVTS, rather than using LVTS payments. If DDR was funded in this way, the distributed ledger (DL) would have the same collateral eligibility rules as the LVTS and would thus continue to meet Principle 5.

Principle 7: Liquidity Risk

An FMI should effectively measure, monitor, and manage its liquidity risk. An FMI should maintain sufficient liquid resources in all relevant currencies to effect same-day and, where appropriate, intraday and multiday settlement of payment obligations with a high degree of confidence under a wide range of potential stress scenarios...

In Phase 1 of Project Jasper we simulated a small stream of relatively low-value payments between banks. The payments were settled on a gross basis and there was sufficient liquidity in the system by design to complete all payments. In Phase 2, the plan was to test the platform performance using payment files that were representative of an actual day's payment activity. Gross settlement of payments on a fully collateralized basis produces large liquidity demands on participants, and to mitigate liquidity risk, the risk that a participant would have insufficient DDRs to make a payment, Phase 2 of Project Jasper incorporated an LSM.

In Phase 2 of Project Jasper we tested the settlement of randomly generated payment files based on sampled historical data and allocated payments for either atomic or LSM settlement based on assumptions about participant preferences. This analysis shows that the LSM algorithm developed for Jasper Phase 2 was more efficient in terms of collateral requirements when compared with atomic settlement; however, the LSM settlement scenario required more time to complete the settlement of payments. Testing also confirmed that the Jasper Phase 2 platform can successfully process payment volumes representing a high-volume day for the LVTS.

[43] LVTS payments ultimately reflect a transfer of a claim to Canadian-dollar deposits on the books of the Bank of Canada. Payments are immediately final and irrevocable upon processing, and while final settlement at the Bank of Canada does not occur until the end of the day, it is guaranteed, backed by collateral and a collateral management system that satisfy Principle 5.

There is nothing inherently different about making payments on a distributed ledger that would create any additional liquidity risk implications, and we do not believe that the Phase 2 design introduces any new obstacles to meeting the Liquidity Risk Principle.

A DLT-based payment platform could coexist as a permanent facility alongside a conventional payment platform such as the LVTS, requiring banks to allocate liquidity to each system and introducing the risk that one facility may not be adequately funded. An assessment of liquidity risk in this scenario would be dependent upon details that are currently unknown. Further developments in this direction would require additional consideration and assessment with respect to Principle 7.

Principle 8: Settlement Finality

An FMI should provide clear and certain final settlement, at a minimum by the end of the value date. Where necessary or preferable, an FMI should provide final settlement intraday or in real time.

Settlement of a payment (whether submitted for atomic settlement or the LSM settlement available in Phase 2) in Project Jasper occurs at the point when a bank's wallet is updated on the DL. Queued payments are not considered settled until they are consumed as inputs during a matching cycle and the sending and receiving bank's wallets have been updated on the DL with the corresponding net payment and a record of the completed payment appears.

Settlement finality requires (i) a legal structure that ensures a transfer of DDR is equivalent to a full and irreversible transfer of the underlying claim on central bank money, and (ii) a DLT that ensures transfers of DDR are immutable.

- For (i), the system would need to be covered by existing laws and protections regarding settlement finality (e.g., the *Payment Clearing and Settlement Act*).

- For (ii), an individual or group of individuals must not be able to take actions to modify or undo payment transactions that have been validated. The vulnerability of a DL to modification depends greatly on the DLT platform.

The Phase I platform was built on Ethereum, which (at the time) relied on the proof-of-work consensus mechanism. Proof-of-work consensus is vulnerable to a 51 per cent/selfish miner attack, which can effectively erase past transactions from the DL.[44] Larger networks help prevent this type of attack by reducing the likelihood that a sufficient amount of computer power could be mobilized to compromise the DL. With only 42 nodes run by members of the R3 consortium for the Phase 1 proof of concept, it is possible that a bank or group of banks could alter the ledger, undermining settlement finality.

The Phase 2 platform was built on Corda, which makes use of the notary function to verify transactions and commit them to the ledger. The Phase 2 platform is therefore not susceptible to a 51 per cent

[44] The gist of the 51 per cent attack problem is that a bank or group of banks with 51 per cent of the mining power can conceivably create a fork in the blockchain at a point in the past that does not include the current set of validated transactions. Since miners build off of the longest existing blockchain, this requires validating a number of consecutive blocks from the new fork before others extend the original non-forked blockchain. A participant can only do this with non-negligible probability if he has a sufficient amount of mining power, namely, more than 51 per cent.

attack, but we still have to be convinced that no individual or group of individuals can take actions to modify or undo payment transactions that have been validated. The protection of the notary against compromise is a key requirement.

For both the Phase 1 and Phase 2 platforms, a regulator node would be able to see all ledger activity and could make use of backups and checks to ensure that no malicious activity has occurred. Since participants on the DL would be affiliated with legal entities and identifiable to the regulator node, it is conceivable that overarching legal structures could be in place to ensure that participants follow system rules and do not work to undermine the integrity of the DL or settlement finality.

Principle 9: Money Settlements

An FMI should conduct its money settlements in central bank money where practical and available. If central bank money is not used, an FMI should minimize and strictly control the credit and liquidity risk arising from the use of commercial bank money.

Transfer of DDR is a transfer of a claim on central bank money. Settlement would therefore occur in central bank money provided that a legal structure is in place linking ownership of the DDR to ownership of central bank money.

Principle 17: Operational Risk

An FMI should identify the plausible sources of operational risk, both internal and external, and mitigate their impact through the use of appropriate systems, policies, procedures, and controls. Systems should be designed to ensure a high degree of security and operational reliability and should have adequate, scalable capacity. Business continuity management should aim for timely recovery of operations and fulfilment of the FMI's obligations, including in the event of a wide-scale or major disruption.

This assessment should not be considered exhaustive but, rather, an initial identification of potential sources of operational risk and key issues identified.

Operational Reliability

In terms of resilience, a key question was whether a DLT-based wholesale payment platform could provide more cost-effective resilience by having no single point of failure. For the proof of concept, Principle 17 would require Payments Canada and the Bank of Canada to maintain high-availability systems, as the functions they perform are essential to the operation of the system. Jasper Phase 1 demonstrated a lower cost for high-availability nodes, since the nodes operated by all participants essentially served to back each other up, insofar as their shared data were concerned. This guaranteed high-availability without extra risk-proofing of each node. However, once privacy and additional functionality, such as with an LSM, are added to the system, the susceptibility to a single point of failure can return if resilience is not carefully considered in the implementation design.

There are two main reasons why this happens. First, additional technology components—such as identity and system access management—do not have the distributed characteristics of DLT. Thus, these important components suffered from the same single-point-of-failure challenges that existing centralized systems face. It must be noted, however, that the full broadcast blockchain systems, such as the Ethereum platform used in Phase 1, are not a full data backup solution. In particular, any of the operators of a blockchain node will invariably have private data in addition to data that are shared with other users on the network. This includes private keys (the loss of which, if not adequately backed up, would prevent access to data (which may extend to assets owned by the node), as well data that the application running on the node must access but cannot be shared.

Second, the comparison to existing systems can be taken a step further in the case of a notary system, such as Corda, which introduces single points of failure and reduces system resilience. Unlike proof-of-work systems, participants' individual nodes must be operational to send or receive payments,[45] and the notary node must be operational to process payments. Notaries do not have to be a single point of failure, as the system could rely on a notary cluster across a number of servers or data centres, authorize other participants to run notary nodes under their instruction in the event of an outage, or implement a range of other options to ensure that the service continues to be available should the primary notary be out of service.[46]

The Corda DLT platform examined in Jasper partitions data such that each participant's node has access to and maintains only a subset of that data. While this approach resolves data-privacy issues, it introduces significant challenges to data replication across the network. Unlike public blockchain schemes, where all nodes share a copy of the exact same database (e.g. the Ethereum blockchain), these permissioned systems have a point of failure at every node. That is, each node requires data replication and archiving to ensure business continuity rather than each node providing resilience to the system, as in the case of the Ethereum blockchain.

In Phase 1 and Phase 2, the Jasper architecture also has a user interface (UI), as well as "middleware" that require each participant to run multiple system components in order to conduct transactions. This is a pragmatic aspect of the architecture implemented to minimize and simplify participants' interaction with the DL. A disruption to the UI/middleware of a participant would prevent interaction by that participant with the underlying DL. A disruption to the UI/middleware used by the Bank of Canada node would disrupt interaction with the DL, including any pledge/redeem activity at that time.

At this time, an overall evaluation suggests that permissioned distributed ledger schemes, if not carefully designed, may decrease operational

[45] Corda queues pending requests to other nodes so that as and when the node is back online transactions may be processed without loss of integrity of the record.

[46] While not implemented in Phase 2, the architecture of Corda allows multiple servers to perform centralized processing in parallel. Each server can calculate proposed updates to the ledger (e.g., the results of an LSM algorithm) and propose it to the participants. The first proposal is processed, and the others are ignored as duplicates. This parallelization removes a single point of failure for any such function.

resilience when compared with a centralized platform or an open DLT platform. Meeting the requirement for strong operational resilience is more challenging for the Jasper Phase 2 based on Corda than the Ethereum platform in Phase 1. In a Phase 2 style production system, it is likely that each participant would have to invest in a high-availability node to reduce the chance of an outage. Other single points of failure can be overcome by straightforward high-availability designs in production.

In summary, there are issues that need to be resolved before the current platform would meet the resilience requirement of Principle 17. However, the specific version of Corda used for the Phase 2 proof of concept does not reflect the full capabilities of the platform, and work is under way to ensure compliance in the future.

Incident Management

System issues such as errors associated with the execution of processes on the DL would need to be managed through an incident management plan, complicated by the fact that there is no convenient mechanism to stop transactions. There would have to be significant coordination among participants to execute the plan (e.g., conducting opposite transactions to reverse fraudulent or mistaken transactions). Additionally, the system operator(s) could be given powers to implement a hard fork resolution to re-adjust the ledger.

Scalable Capacity

The PFMIs require an FMI to ensure that it has the scalable capacity to handle stress volumes. The LVTS processes 32,000 transactions per day, with a peak throughput of roughly 10 transactions per second (TPS). In DLT arrangements, there is a computational cost to distributing functionality. In proof-of-work platforms such as the current Ethereum platform, there is capacity to scale to approximately 14 TPS, as the platform has been designed for the public Internet, where speed limitations could challenge information flow between nodes. While 14 TPS would be sufficient to process current daily LVTS volumes, it could create constraints in times of market stress or volatility and as transaction volumes increase over time. In contrast, scalability would not be a constraint in the Corda platform. This is because it does not have a fixed-time-based consensus method, and only requires the nodes of the involved parties and notary to verify transactions.

Physical and IT Security

The PFMIs require comprehensive physical and information security policies to address all potential vulnerabilities and threats. In contrast to existing wholesale payment systems that rely on a central operator, Project Jasper relied on computers (nodes) hosted by participants active on the R3-CEV network to process transactions on the DL. While Phase 1 is resilient to the failure or compromise of multiple nodes, it would be crucial that the network collectively has sufficient IT and physical security to prevent a 51 per cent attack.

To comply with the PFMIs, a more comprehensive IT security analysis would need to be completed to determine the full scope of the measures required to counteract cyber threats. This includes a more detailed assessment of the security controls for the R3 network and the components hosted at each of the nodes (i.e., UI, middleware). Importantly, the proof of concept highlighted the security risks related to the middleware run by each participant that invokes smart contract functions and stores digital keys (with no ability to retrieve lost or stolen private keys).

In general, all of the above security issues are partially mitigated by the platform being hosted on a smaller, trusted network (R3). This might reduce the incentive for propagating an attack and increases the ability to detect an attack."

Interdependencies

Operations that are outsourced to a third-party service provider (e.g., data processing and information systems management) should meet the same requirements as if the services were provided internally by the FMI. It is unclear whether the operating nodes that are part of the consensus and verification mechanism would be considered critical service providers (operations being "outsourced") either individually or collectively. Further, there may be limited ability to implement minimum requirements or standards of operational reliability for those nodes.

The risk-management standards for observing the Principle for Operational Risk are extensive for systemically important FMIs, and it is not surprising that the Jasper proof-of-concept platform does not meet these standards. It is important to recognize the need for rigorous and extensive testing of all system components, to acknowledge policies for change-management/project management, contingency procedures, and other key requirements. Further, it is expected there would be significant system architecture challenges associated with the integration of a DLT system that would need to be considered for a production-ready system.

Principle 18: Access and Participation Requirements

An FMI should have objective, risk-based, and publicly disclosed criteria for participation, which permit fair and open access.

The proof of concept did not formalize a set of access and participation requirements within a set of rules. Participation was limited to LVTS participants for simplicity, since DDR must be funded directly through a LVTS payment.[47] As a result, to take part in the proof of concept, a participant must have met the LVTS participation requirements, which are PFMI compliant. The exception is operational risk, where each participant in the proof of concept could use different middleware, user interfaces, automation and key management processes. To meet the PFMIs, the system should have operational requirements for participants, which would need to meet a minimum standard of operational resilience. As a point of reference, participants must maintain at least 98 per cent availability in the LVTS.

One potential advantage of DLT technology is the ability to increase access. Future phases should consider structures more independent from the LVTS in order to test this hypothesis.

[47] Technically, a non-LVTS participant could have participated in Project Jasper if an arrangement were in place for an LVTS participant to make and receive payments to and from the Bank of Canada to fund and redeem DDR on their wallet. However, participation in Project Jasper was limited to LVTS participants.

Appendix 4

An Overview of Canada's Large Value Transfer System

For interested readers, this appendix explains the LVTS settlement and risk models in greater detail and also outlines at a high level the LVTS participant interface and LSM arrangement.

LVTS Settlement Model and Risk Model

The LVTS employs a real-time net settlement model. Payment messages are exchanged between participants each business day, where messages are processed immediately by the LVTS subject to the sending participant maintaining sufficient capacity to send the payment. This capacity is underpinned by the use of net debit caps, which limit the volume of settlement exposure that a participant can generate against the system during the day.[48] Every payment submitted to the LVTS is subject to a real-time risk-control test to ensure that, if the payment were processed, the sending participant's cumulative exposure generated against the system would still be within its net debit cap. This cumulative exposure is more commonly referred to as a participant's multilateral net position, calculated as the total value of payments received less the total value of payments sent at the time of determination. When a participant sends more value than it receives it incurs a "negative" multilateral position, which is constrained by a net debit cap.

Legally enforceable novation netting is performed on each payment message at the time of processing. When a payment message passes the real-time risk-control test, the original bilateral obligation between the sending and receiving participant is extinguished and replaced with a multilateral settlement obligation between the sender and the system. An obligation on the part of the system to ensure that the receiving participant is "paid out" at the time of settlement is simultaneously created. Once a payment message is processed, funds can be distributed to the beneficiary on a final and irrevocable basis. It follows that "payment finality" in the LVTS is achieved before any exchange of settlement assets between participants.

This is a key difference between the LVTS and wholesale settlement arrangements in other jurisdictions. Other countries have typically adopted a real-time gross settlement (RTGS) model operated by the central bank. In an RTGS environment, final and irrevocable transfer of the settlement asset across participants' central bank accounts occurs immediately as each payment is processed by the technology platform. Conversely, final settlement of LVTS multilateral net positions is effected at the end of each day. Of note, LVTS participants are required to maintain a settlement account at the Bank of Canada, where final settlement occurs in central bank money through the transfer of value across these accounts.

[48] Reference to "the system" in this context pertains to all other participants.

Two tranches (or payments streams) underpin the LVTS risk model. Participants can use either stream to send payment messages, and each stream employs a net debit cap constraint. The sum of a participant's multilateral positions in these two tranches reflects its overall multilateral net position in the LVTS. The tranches differ in both collateral requirement and loss-allocation procedure in the event that a participant is unable to meet its settlement obligation. Tranche 1 employs a "defaulter-pays" loss-allocation model, where draws against the Tranche 1 net debit cap are secured fully by collateral pledged by the sending participant. Tranche 2 utilizes a "survivors-pay" loss-allocation model, where participants' draws against Tranche 2 net debit caps are partly secured by a pool of collateral pledged collectively by participants. [49]

While the above risk controls ensure that there is sufficient collateral pledged to effect LVTS settlement, given the inability to settle on the part of a single participant, collateral may be insufficient to cover the extreme but plausible scenario that more than one participant defaults on the same day. In this case, exposures not covered by pledged collateral are back-stopped by a central bank commitment to settle accounts. Through the use of net debit caps, collateral posted by participants and the central bank commitment to settle accounts, LVTS settlement is guaranteed to occur in all states of the world. [50]

LVTS Participant Interface

The LVTS is a web-enabled mainframe application, where authorized representatives from each participant institution interact with the LVTS through proprietary workstations. A variety of other internal processes and record-keeping systems are also employed by participants to support LVTS activity, including in the areas of treasury and collateral management, client banking, and account processing and reconciliation. The LVTS is a "Y-copy" scheme that leverages the global SWIFT messaging service. Participants send and receive SWIFT messages containing payment details over the SWIFT network, where select details are intercepted by the LVTS, validated and checked against the real-time risk-controls, and notification of either successful or unsuccessful processing is communicated by the LVTS over the SWIFT network to the relevant parties. The LVTS application supports queries to access real-time and historical reports related to balances, payment flows, and caps and limits. Net debit caps and bilateral credit limits (BCLs) are also established and adjusted via the LVTS application during the day. Participants in the LVTS are able to view pending incoming and outgoing payment messages in the central queue (described below) through their proprietary workstation.

[49] Contributions to the pool are based on a pre-established formula and are underpinned by the notion of bilateral credit limits (BCLs) which are granted from a receiving participant to a sending participant as a means of limiting the settlement exposure that the latter can pose to the former. In the event of a participant's default on its settlement obligation, these BCLs would factor into the loss-allocation formula for surviving participants, with their Tranche 2 collateral contribution serving as a cap on their loss exposure to the default. Of note, no participant is required to extend a BCL to any other participant.

[50] More information on the LVTS, and the central bank commitment to settle accounts, can be found in CPA *By-law No. 7* Respecting the Large Value Transfer System, which is available at **http://laws.justice.gc.ca/PDF/SOR-2001-281.pdf.**

LVTS Jumbo Queue

In addition to the collateral-efficient Tranche 2 stream, the LVTS employs a central queue referred to as the LVTS "jumbo queue." Only payment messages with a value greater than $100 million that fail the real-time risk-control test are eligible to enter the queue. Payments may be released from the Jumbo queue on a First In First Out (FIFO) basis when a participant's liquidity is augmented (e.g., it receives a payment or its net debit cap is increased), or if the payment is processed by the matching algorithm. A Queue Expiry algorithm is also applied at routine intervals throughout the day that removes any payments that have been sitting in the queue for an extended period. This is absolutely critical in the case where FIFO release (with no by-pass capability) is used.[51]

[51] For an in-depth description of the LVTS central queue, including a numerical example, see A Primer on the LVTS, available at **http://www.bankofcanada.ca/wp-content/uploads/2010/05/lvts_neville.pdf.**

PRJ.JASPER