Project Jasper

Workshop On Fintech, Payments, and Financial Inclusion

Dinesh Shah
Program Manager, E-money and Fintech Research
Bank of Canada

The Canadian Context

Payment Systems Operator

Payment System Oversight

Modernization Program: Faster, easier, data-rich innovation

↕ *independent* ↕

Jasper: experimental

Motivation

DLT Features

- Common data and processes
- No Single Point of Failure
- Feature Programmability using "Smart contracts"

Uses in Financial System

- Codify and automate contracts
- Consolidate information and align processes *across* institutions

Potential Benefits

- Cost savings:
  - Efficiencies
  - Reduced errors
- Simpler Resilience

Q: To realise these benefits, do we need cash on ledger?

Objectives of Project Jasper

What would a DLT based large value payment system mean for:

- Standing against Principles for Financial Market Infrastructure (PFMI)
- Costs (3 levels: core systems; participants, financial system)
- Transparency of information
- Barriers to direct participation by institutions
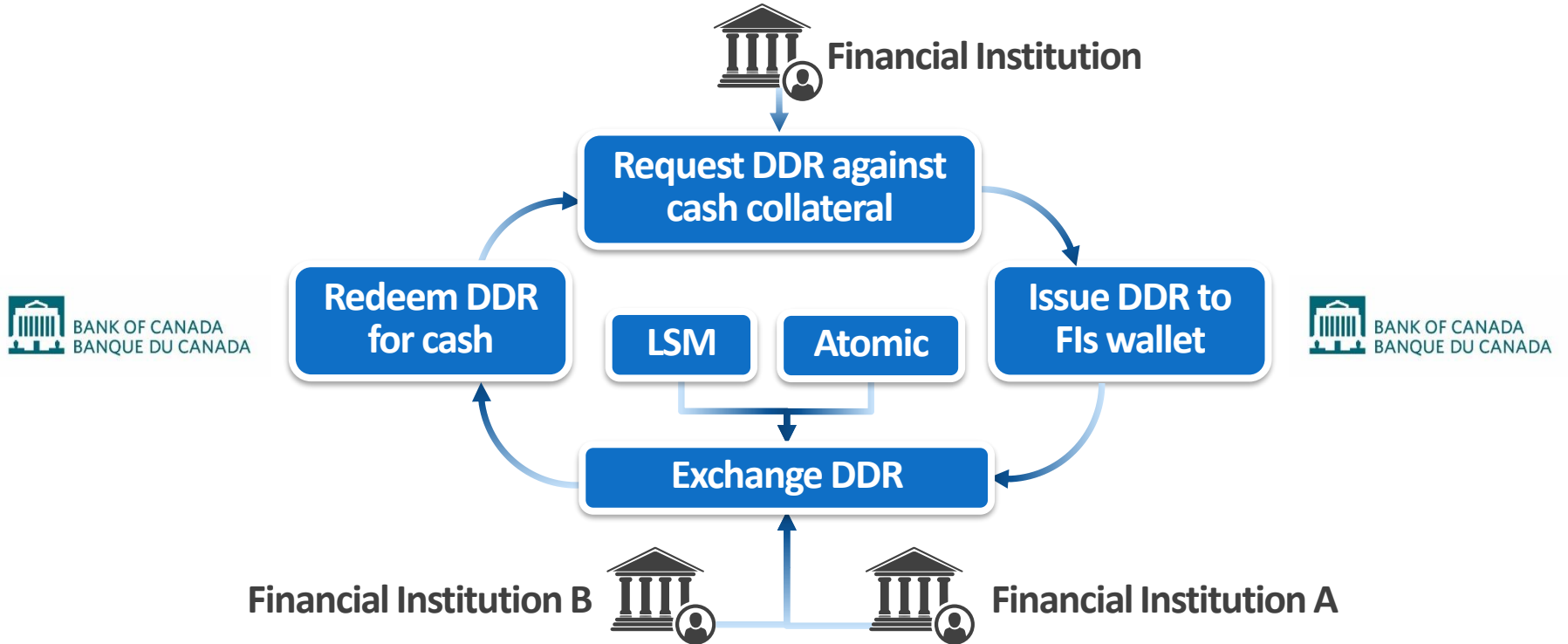
Approach

BANK OF CANADA
BANQUE DU CANADA

PAYMENTS CANADA   PAIEMENTS CANADA

RBC   TD   BMO

Scotiabank   CIBC

NATIONAL BANK

HSBC

R3

- Set requirements
- Pose Hypotheses or Questions

Design and build proof of concept

Conduct analysis

Functional Lifecycle

Architecture – Ethereum Platform

Designed for the internet and pseudonymous access

- Data and function replicated everywhere

- Little confidentiality

- High resilience

- Any changes are validated and recorded by a consensus of whole network

- The consensus method (PoW) means changes become more permanent over time

All ledgers are identical
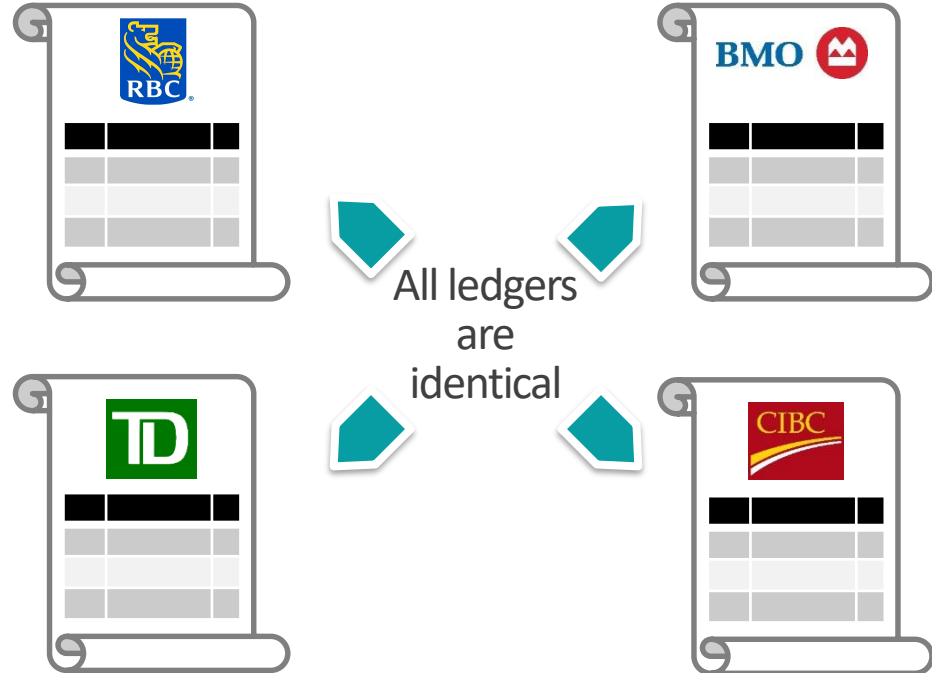
Architecture – Corda Platform

Designed for Financial Institutions and identified (permitted) access

- Data only shared and replicated on nodes of parties to trade
- Good confidentiality
- More costly resilience
- Changes validated only by parties and trusted notary
- Changes are deterministic
- Some centralised functions
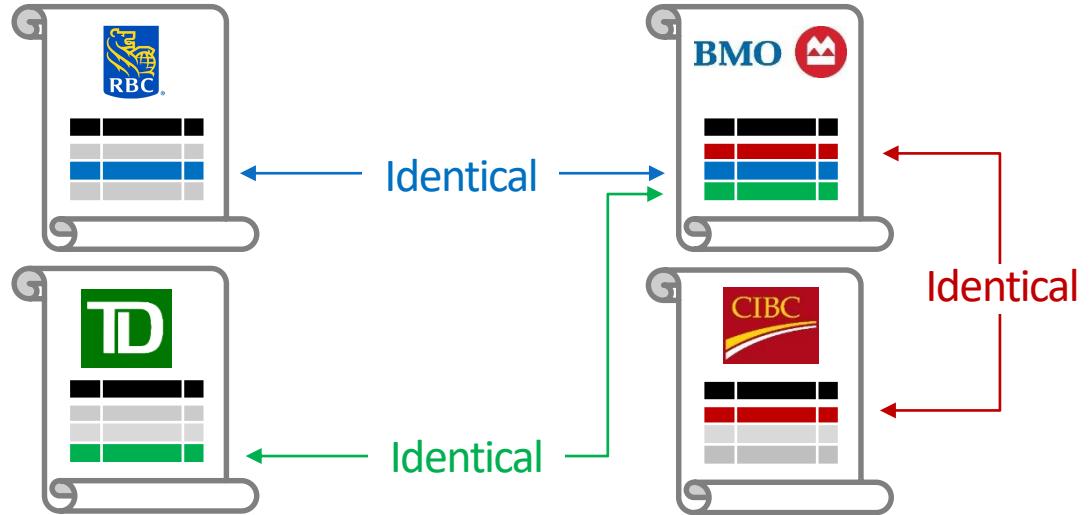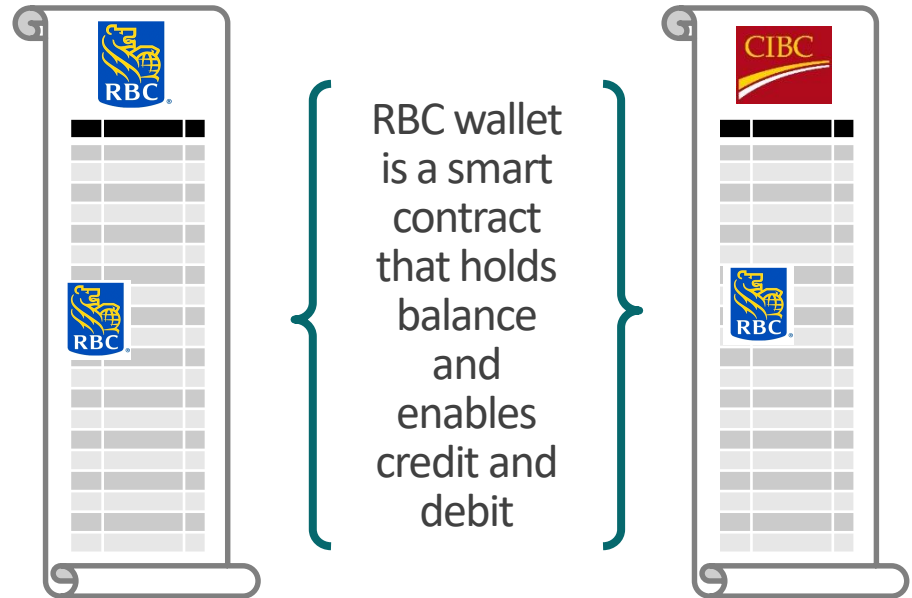

Centralized functions }

Notary: Validates and records *all* transactions it receives

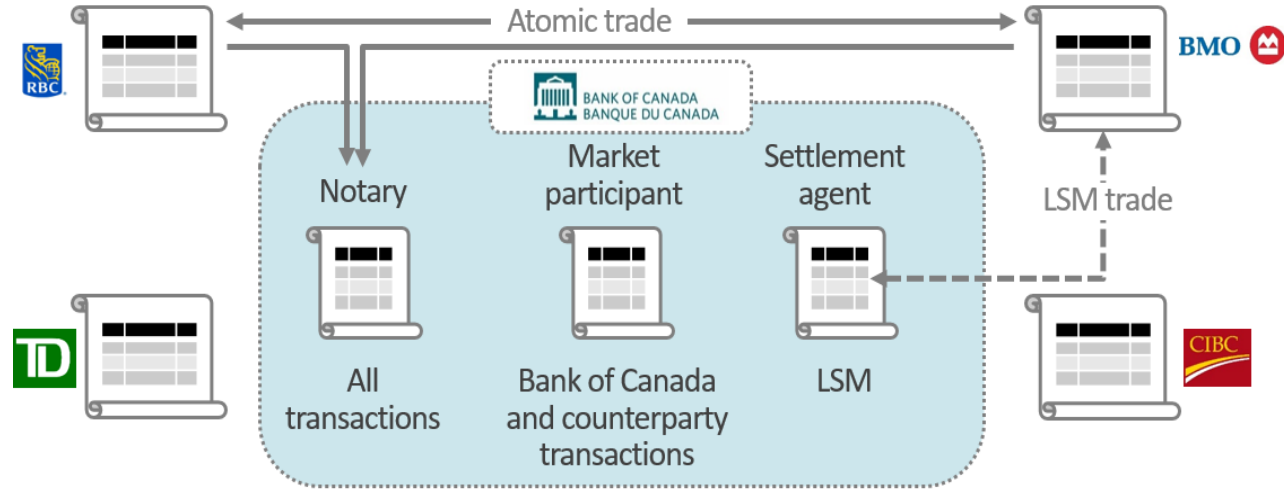**Doorman:** Identity and authorization service

Architecture – Phase 1, Proof-of-Concept on Ethereum

- Although the RBC wallet exists on all nodes, only RBC can invoke the function to spend from it (only it has the private key)

- Similarly, although all ledgers have the function to issue new DDR, only Bank of Canada can invoke that function

- If an FI's node fails, they should be able to continue transacting on the network

- Settlement does not have a definite point-in-time finality

- Everyone on the network an see *all* trades

RBC wallet is a smart contract that holds balance and enables credit and debit
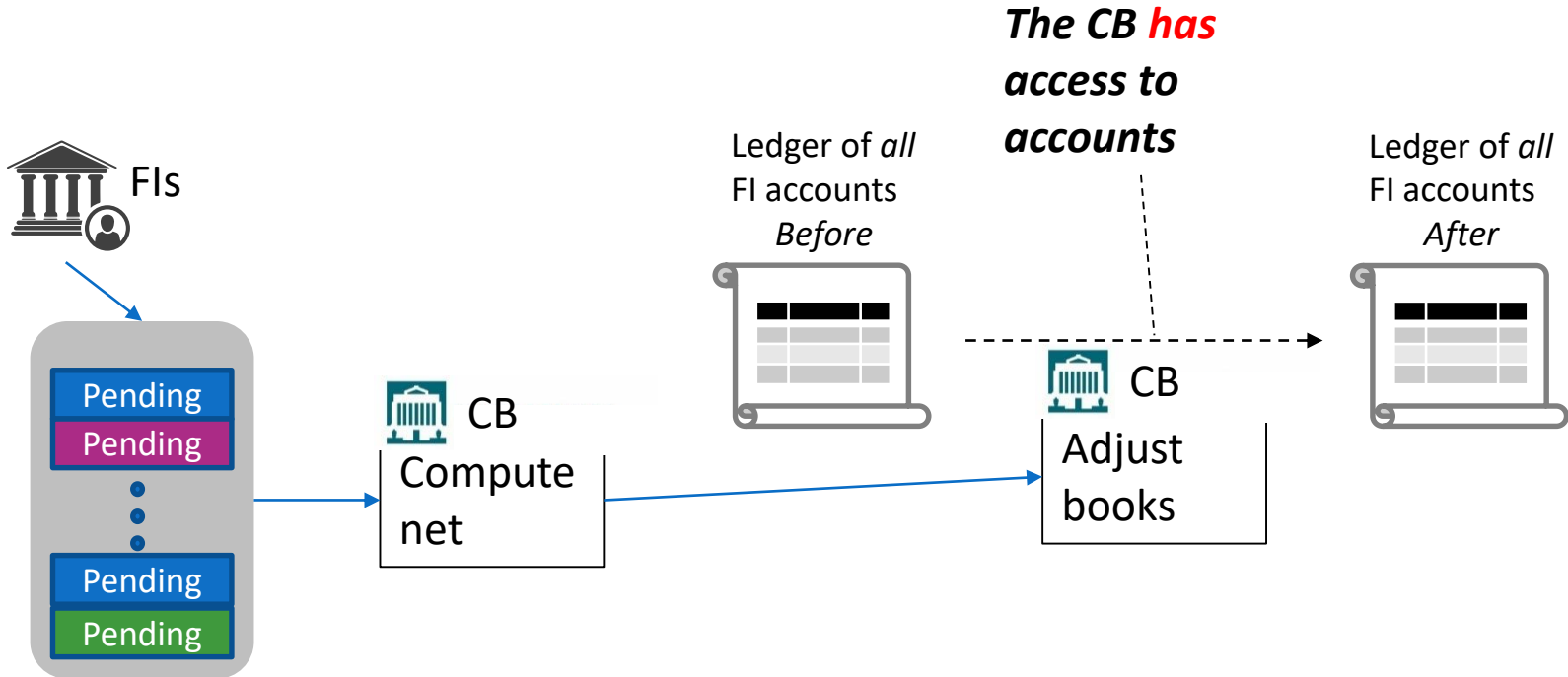
Architecture – Phase 2, Proof-of-Concept on Corda
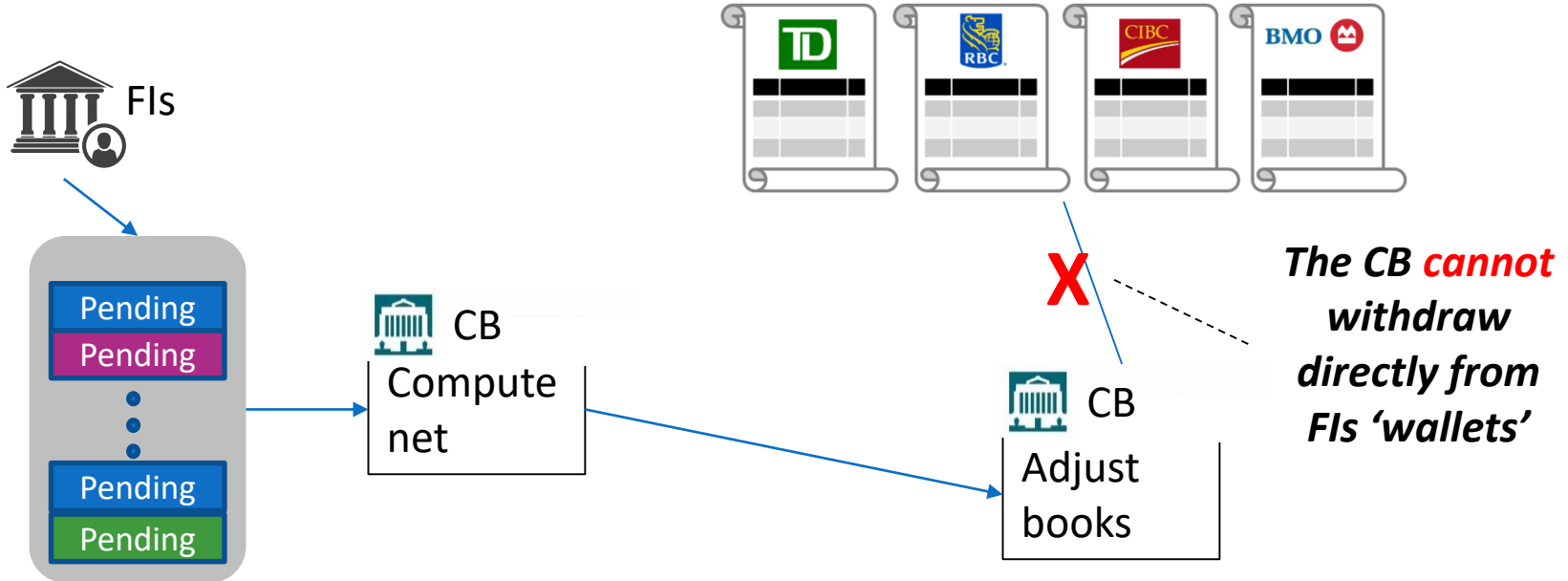
- Hybrid: decentralised / centralised

- FIs do not receive information on trades they are not party to

- Since data and functions are not replicated, an FIs node must be available for it to transact

  - Requires 'traditional' high availability and disaster recovery architecture

  - As with current systems, central function failure => system wide failure

- Settlement has definite finality

LSM Netting Functions – Privileged Access in Centralised Systems

FIs

Pending
Pending
•
•
•
Pending
Pending

CB Compute net

Ledger of *all* FI accounts *Before*

The CB has access to accounts

CB Adjust books

Ledger of *all* FI accounts *After*

LSM Netting Functions – A Challenge in DLT

FIs

Pending
Pending
Pending
Pending

CB
Compute
net

CB
Adjust
books

*The CB **cannot** withdraw directly from FIs 'wallets'*

LSM Netting Functions – A Solution in DLT

Financial Institution A **Financial Institution B**   **Settlement agent**

Time

Add payment to queue ➔   Add to queue

Add payment to queue ➔   Add to queue

                                 Freeze queue

⬅ Send me DDR to cover your payments

⬅ Send me DDR to cover your payments

Time

Send DDR
Send DDR

'inhale phase'

Run netting algorithm to compare net positions bound by DDR received

Send net DDR
Send net DDR

'exhale phase'

Financial Institution A
Financial Institution B

BANK OF CANADA
BANQUE DU CANADA

Settlement agent

Findings – Against PFMIs

Principle	LVTS	Phase 1 (Ethereum)	Phase 2 (Corda)
Credit & Liquidity risk	Satisfied	DDR backed 1:1 with cash collateral on LVTS	
Settlement Finality	Satisfied	Probabilistic finality	Satisfied (requires further investigation)
Operational Risk – resilience	Central system engineered for HA and DR. Minimal participant work.	Tolerant to partial failure – rely on network, not individual nodes. High redundancy.	Requires engineering for HA and DR at central node *and* participant's nodes
Operational Risk – scalability	high throughput	Ethereum limited to about 12 tx/sec	No inherent limitation. Central functions can cause bottlenecks (e.g. notary)

Other Lessons

- Unlikely to be net benefits using DLT over a centralised system for the *sole* purpose of interbank payments

- Benefits more likely to come from enabling efficiencies in the broader FIs and FMIs via simplified processes (e.g. back office reconciliation)

- LSM implementation is complex – inherently centralised functions in decentralised systems is tricky

Next Steps

- Jasper phase 3: partnership with Payments Canada and TMX to examine a DvP use case using DLT

- Jasper-Ubin: Partnership with Monetary Authority of Singapore to examine cross-border / cross-currency payments using DLT

Can Project Jasper be replicated in SSA?

- Certainly! But should it?
  - What exactly are the questions being asked?
  - Will it be possible to actually measure what needs to be measured to answer the question?
  - Are you prepared to throw away what ever was built?
  - Can all the parties agree on a common objective (at least a common denominator)?
  - If on a best-effort basis, will the parties provide equitable effort?

Thank you.