43rd International Conference on Very Large Data Bases

Tutorial: Blockchains and Databases

C. Mohan

IBM Fellow
Distinguished Visiting Professor
Tsinghua University, Beijing

IBM Almaden Research Center, San Jose, USA

@seemohan cmohan@us.ibm.com http://bit.ly/CMwIkP

29 August 2017

Links to Videos, Slides, Bibliography & Twitter Handles @ http://bit.ly/CMbcDB

IBM

Agenda

Goal: Educate DB people about private/permissioned blockchains (BCs) to convince them to get more involved to improve them

- Origin of blockchains
- Related distributed systems/databases topics
- Evolution: Private BCs, Smart Contracts, …
- Applications
- Market Scene
- Benchmarks
- Architectural Choices and Relationship to DB Replication
- Technical Details of Representative Systems:
  Enterprise Ethereum, Hyperledger Fabric, R3 Corda, BigchainDB, Sawtooth, Ripple
- Futuristic Topics

Blockchain (BC)

- Origin in digital currencies, in particular Bitcoin (Satoshi Nakamoto, 2008) – anonymity, **open/public/permissionless** environment
- Numerous organizations across the world working on various aspects of it: security, consensus, database, benchmarks, …
- Banks, regulators, universities, startups, big technology companies, services companies, governments, … individually or as part of consortia
  - February 2017: First commercial deployment of BC technology by IBM and Guernsey's Northern Trust for admin of private equity fund managed by Unigestion
  - July 2017: Hyperledger Fabric **1.0** Released
  - Hyperledger Fabric on IBM Cloud - IBM Blockchain Platform (formerly HSBN) on highly secure Linux on mainframes (System Z) with security hardware – announced August 2017
- Grand View Research: Global BC Tech Market **$7.74B** by 2024
- **My focus**: **Private/Permissioned** BC Systems!

Distributed Systems

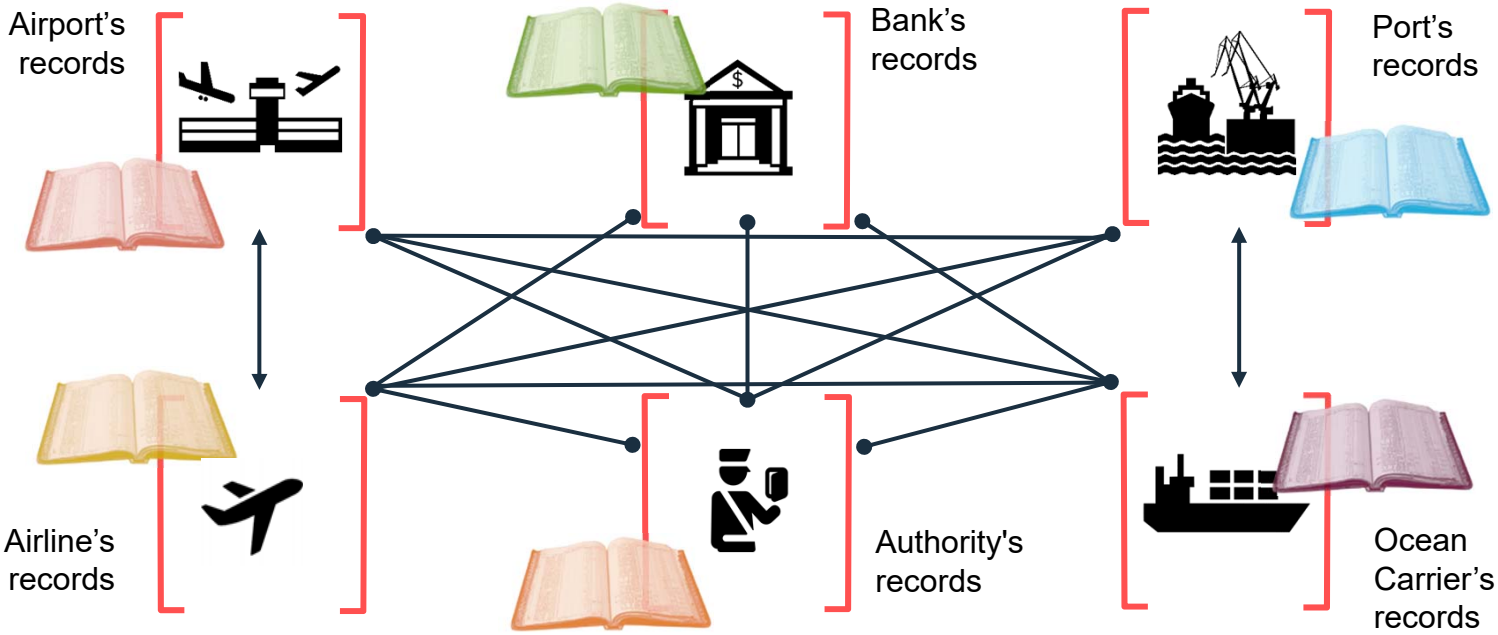
- Distributed operating systems
- Distributed virtual memory
- Message passing in distributed computations and distributed checkpoints
- Clock synchronization and event ordering (e.g., Lamport clocks)
- Byzantine agreement and distributed consensus
- Two phase commit optimizations (e.g., Presumed Abort)
- (Transactional) RPCs and distributed file/object systems
- Asynchronous computation via message queues and pub-sub
- Distributed event-based systems
- Client-server, mobile computing and caching, WWW
- Workflow or business process management systems
- Service Oriented Architecture (SOA)
- Public cloud and hybrid cloud
- 

Data Systems

- Relational DBMSs (e.g., System R) and SQL
- Data consistency, degrees of isolation and fault tolerance
- Distributed databases (e.g., R*) and distributed transactions/queries
- Synchronous and asynchronous replication with primary copy
- Update anywhere (multi-master) replication and eventual consistency
- Stored procedures, user-defined types/functions, data provenance, …
- Data warehousing and parallel DBMSs – OLTP vs OLAP
- Shared Nothing Vs Shared Disks
- Object-oriented databases, XML, schema chaos, data integration, …
- Web2.0-inspired NoSQL, sharding & massive scaling (e.g., Spanner), JSON, …
- Big Data: Map-Reduce, Hadoop, Spark, …
- Data privacy, multitenancy and trans-border data flow restrictions
- Multi data centers and disaster recovery
- …

Problem Being Solved

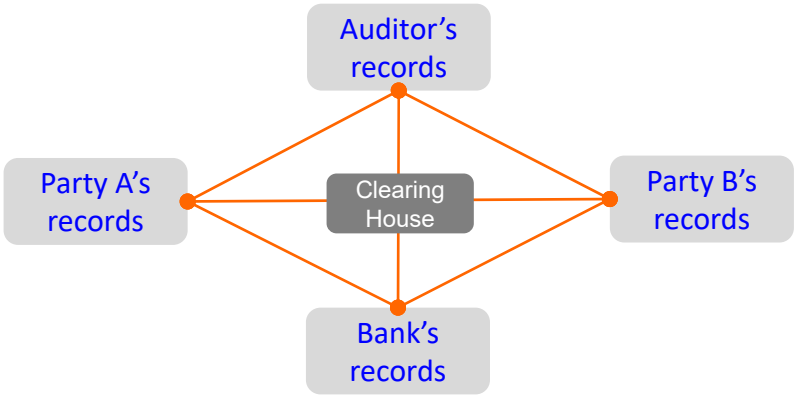
Recording of events is becoming much more complex…



… Inefficient, expensive, vulnerable, lack of transparency

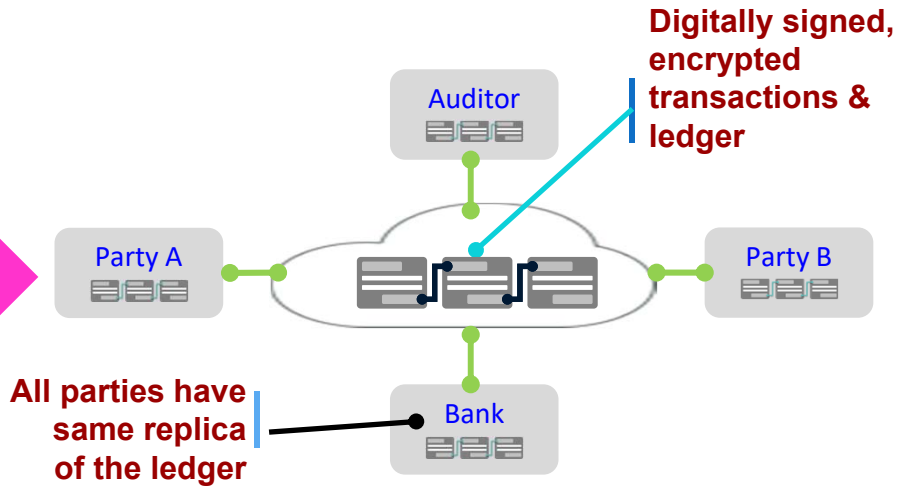
Basic Change to Business Processes

Traditional Way



… Inefficient, expensive, vulnerable

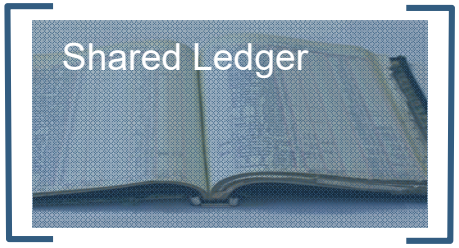
Blockchain Way



… Consensus, provenance, immutability, finality

Blockchain for Business

Append-only distributed **system of record** shared across business network



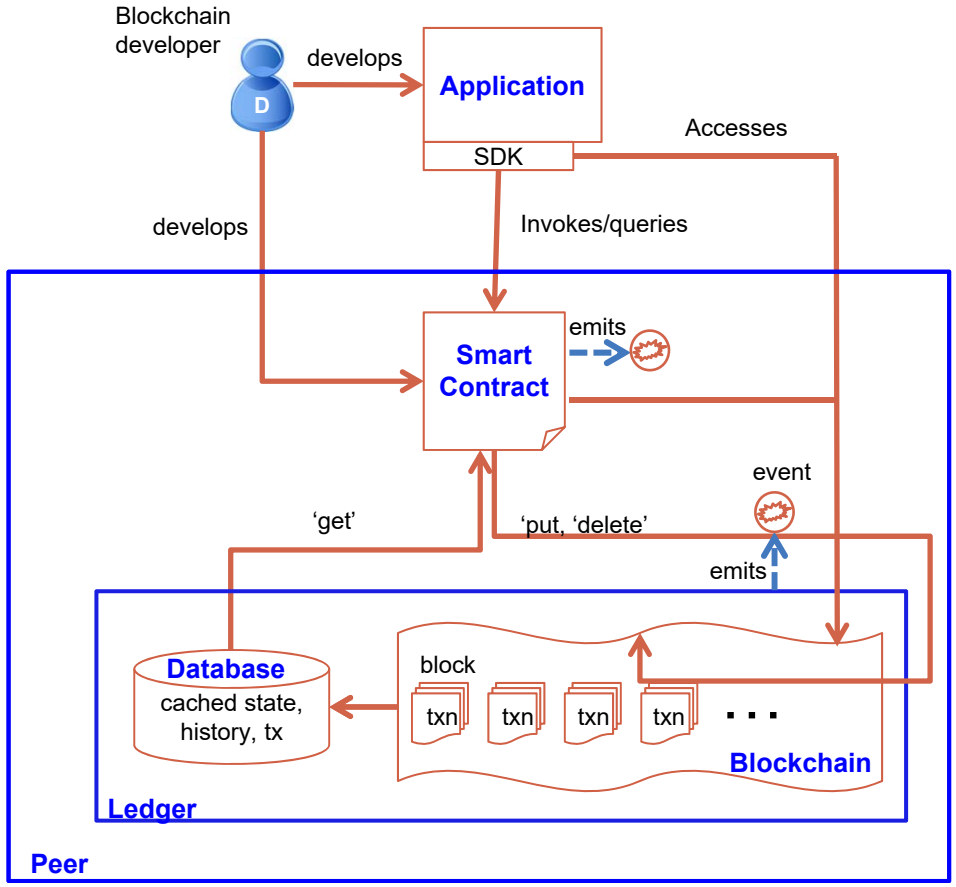
Business terms embedded in transaction database & executed with transactions

Ensuring appropriate visibility; transactions are secure, authenticated & verifiable

All parties agree to network verified transaction

… Broader participation, lower cost, increased efficiency

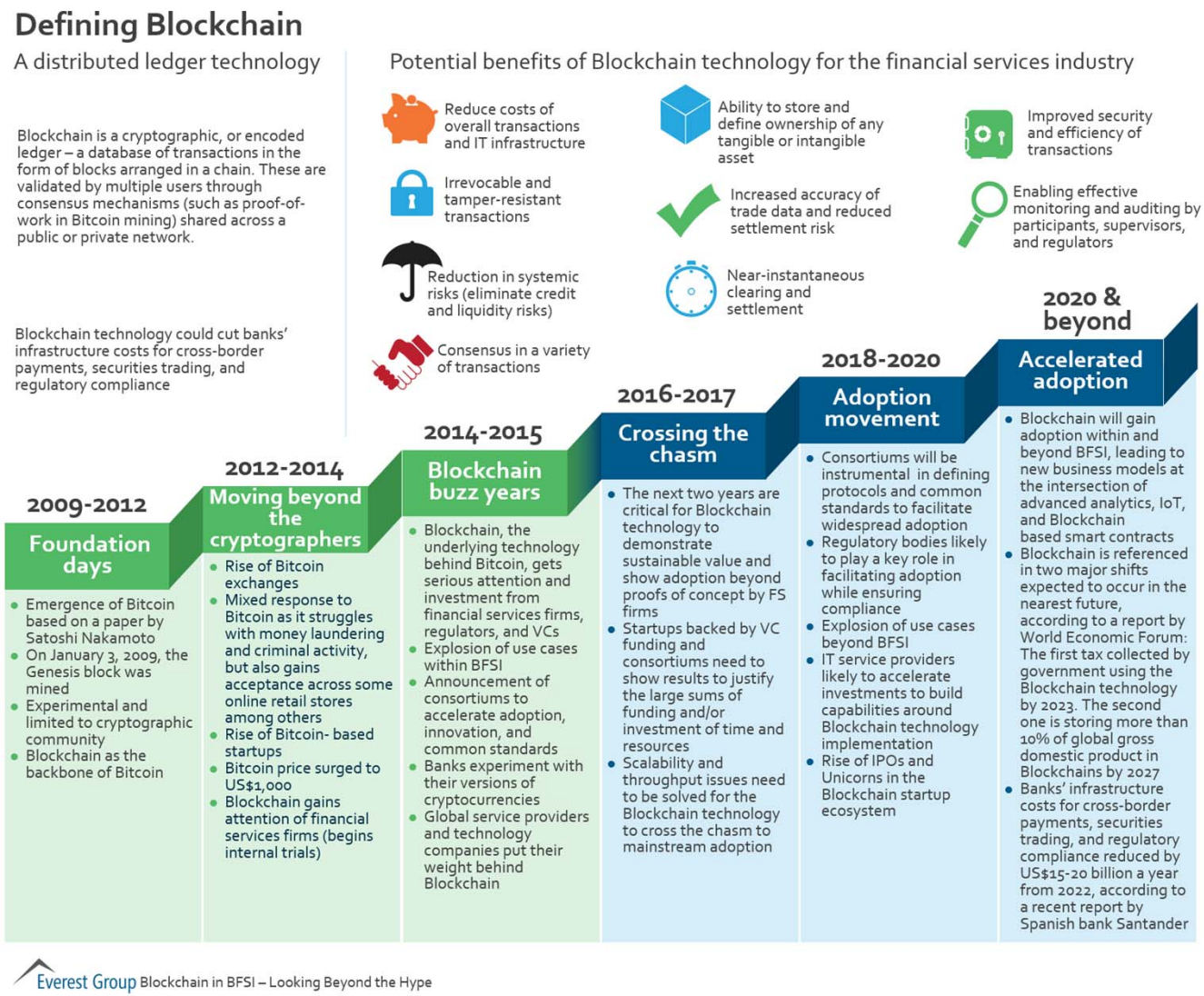
Overview of Application Flow



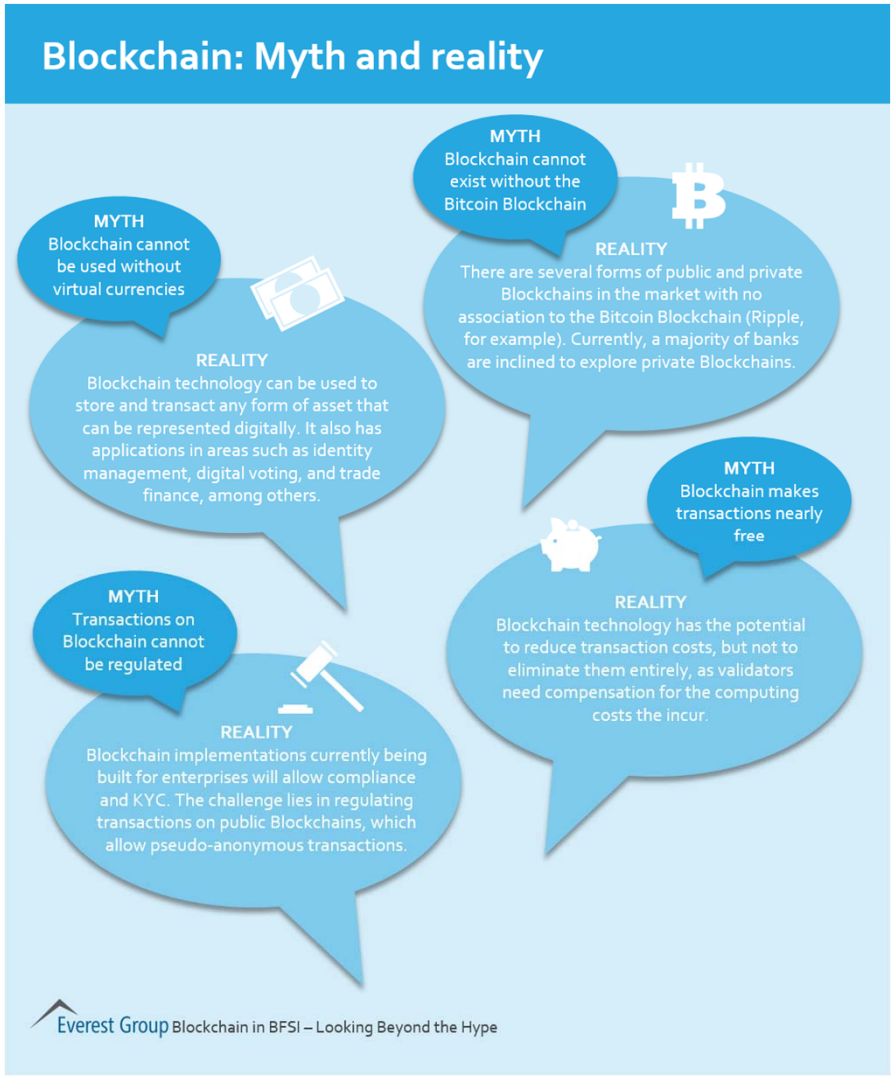
- Developers create application and smart contracts (chaincodes)
  - Chaincodes are deployed on the network and control the state of the ledger
  - Application handles user interface and submits transactions to the network which call chaincodes
- Network emits events on block of transactions allowing applications to integrate with other systems

Smart Contracts Everest Group

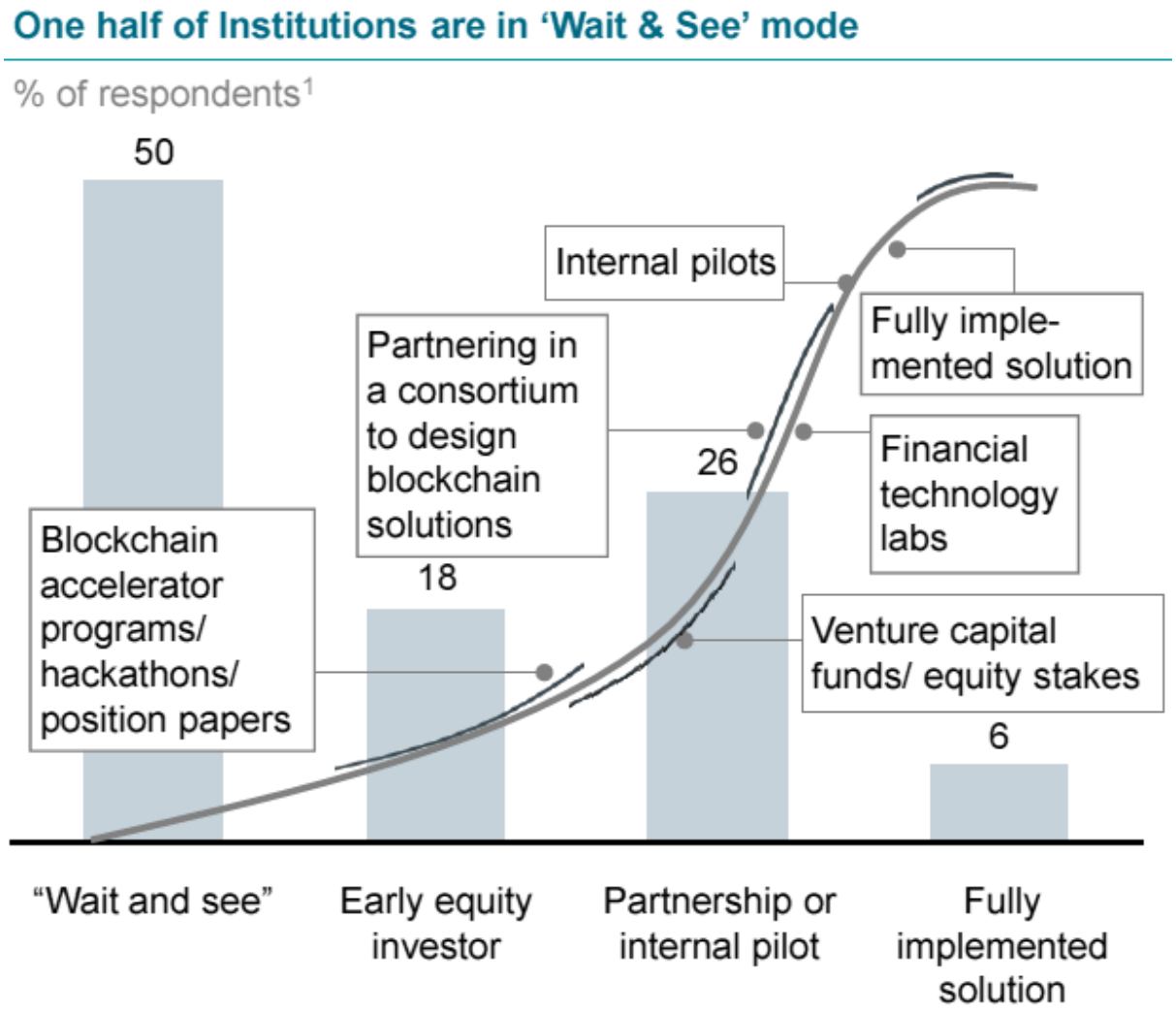
Blockchain: Distributed Ledger Technology Everest Group



Blockchain: Myth & Reality Everest Group



McKinsey Survey of Finance Execs Early 2016



Source:https://www.treasury.gov/initiatives/fio/Documents/McKinsey_FACI_Blockchain_in_Insurance.pdf

Blockchain Applications

- Track provenance, ownership, relationships & lineage of assets
- Supply Chain – Food Safety (Walmart), Logistics (Maersk)
- Insurance Claims
- Know Your Customer
- Derivatives Processing
- Trade/Channel Finance (IGF)
- Trade Information Warehouse (DTCC)
- Post-Trade Reconciliation/Settlement
- Private Equity Fund Management (Unigestion)
- Syndicated Loans
- Diamond/Valuables Tracking and Protection – Provenance Management (Everledger)
- Cross-Border Payment, Payments for/by Unbanked Populations
- Low volume stock trading (JPX)

"Blockchain Revolution: How the Technology Behind Bitcoin Is Changing Money, Business, and the World", Don Tapscott, Alex Tapscott, ISBN 978-1101980132.

Client/Application Examples

CLS

FX Netting

everledger

Diamond Provenance

Crédit Mutuel Arkéa

Identity Management

DTCC

Credit Default Swaps

BAML HSBC

Trade Finance

IBM Global Financing

IBM GLOBAL FINANCING

Channel Financing

Japanese Stock Exchange

JPX

Low Liquidity Securities Trading & Settlement

Walmart

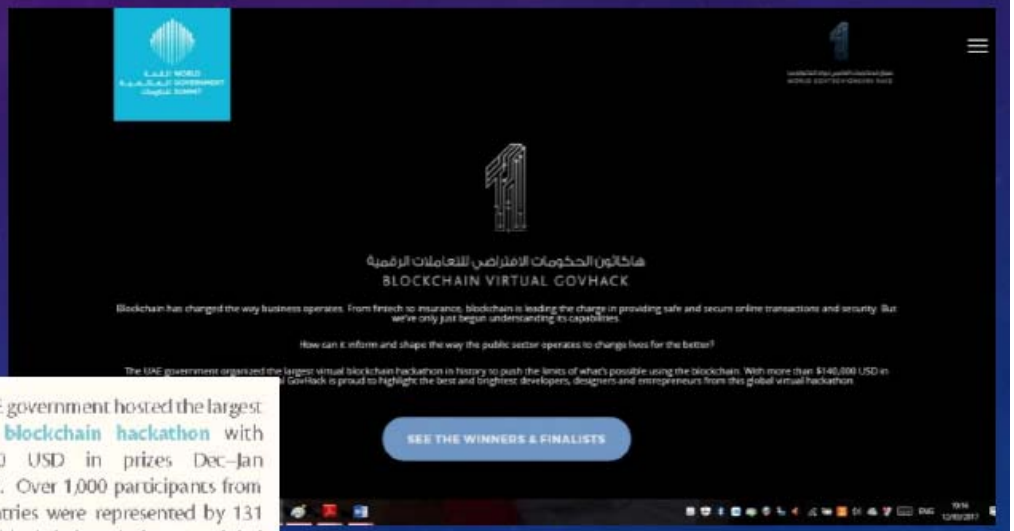
Food Safety

FDA

Health Data Exchange

UAE Initiatives Supporting Blockchain

- **The Blockchain Virtual GovHack**
  - is challenging participants to innovate their public sectors by using blockchain technology to tackle one of the challenges:
  - Re-inventing global identity
  - Reducing paper footprint
  - Fighting fraud and crime
  - Future of health
  - Building smart cities



The UAE government hosted the largest virtual blockchain hackathon with $140,000 USD in prizes Dec–Jan 2016/17. Over 1,000 participants from 41 countries were represented by 131 unique blockchain solutions to global government challenges in these areas:

1	Global Identity	20 %
2	Reducing Footprint	13 %
3	Crime & Fraud	11 %
4	Health	14 %
5	Smart Cities	10 %
6	Other	32 %

www.GovTechioneersRace.com

Source: Saeed Al Dhaheri, Etisalat Academy 3/2017

Dubai Blockchain Strategy

- Aims for Dubai to become the first blockchain-powered city by 2020
- For Dubai government to become paperless by shifting all transactions to Blockchain, and empower Dubai Smart city experience for all
- Based on Three pillars:
  - ✓ Government Efficiency: implementing blockchain technology in government services
  - ✓ Industry Creation: supporting the creation of a blockchain industry through empowering start ups and businesses
  - ✓ International Leadership: leading global thinking on blockchain techonolgy
- the Smart Dubai Office SDO launched Blockchain Challenge in partnership with global accelerator 1776
  - aims to identify the most innovative blockchain ideas from startups around the world and bring them to Dubai
- SDO launched a city-wide effort to implement blockchain in city services
- Partnerships with IBM as a Blockchain Lead Strategic Partner, and Consensys as Blockchain City Advisor.


Dubai launches Blockchain strategy to become paperless by 2020

Hamdan unveils ambitious plan to save 25 million work hours annually through paperless transactions

GULF NEWS

Source: Saeed Al Dhaheri, Etisalat Academy 3/2017

Dubai Blockchain Strategy

- Aims for Dubai to become the first blockchain-powered city by 2020
- For Dubai government to become paperless by shifting all transactions to Blockchain, and empower Dubai Smart city experience for all
- Based on Three pillars:
  - ✓ Government Efficiency: implementing blockchain technology in government services
  - ✓ Industry Creation: supporting the creation of a blockchain industry through empowering start ups and businesses
  - ✓ International Leadership: leading global thinking on blockchain techonolgy
- the Smart Dubai Office SDO launched Blockchain Challenge in partnership with global accelerator 1776
  - aims to identify the most innovative blockchain ideas from startups around the world and bring them to Dubai
- SDO launched a city-wide effort to implement blockchain in city services
- Partnerships with IBM as a Blockchain Lead Strategic Partner, and Consensys as Blockchain City Advisor.


Dubai launches Blockchain strategy to become paperless by 2020

Hamdan unveils ambitious plan to save 25 million work hours annually through paperless transactions

GULF NEWS

Source: Saeed Al Dhaheri, Etisalat Academy 3/2017

Dubai Blockchain POC Project

- Pilot Project: Trade finance and logistic solution
- Dubai customs, Dubai trade, DU, EmiratesNBD, Banco Santander, Aramix, and IBM as a technology provider
- Blockchain solution for goods importing and re-exporting in and out of Dubai
- Using hyperledger fabric as an open source cross-industry trade finance solution and IBM infrastructure
- Aim to replace paper-based contracts with smart contracts
- This integrates all the key trade process stakeholders from ordering stage, in which the importer obtains a letter of credit from their bank, through the intermediary stages of freight and shipping, and ending with customs and payment clearance

Source: Saeed Al Dhaheri, Etisalat Academy 3/2017

Blockchain Roadmap

Source: American Banker 3-2017

Blockchain Companies/Consortia & Banks

New kid on the block

Enterprise Ethereum Alliance is the latest addition to a list of companies and consortiums focused on blockchain where banks serve as investors or partners

	No. of partners
Utility Settlement Coin	5
Global Payments Steering Group	6
Chain	9
Digital Asset Holdings	15
Enterprise Ethereum Alliance	30
R3	81
Ripple	90+
Hyperledger	122

Source: Staff research

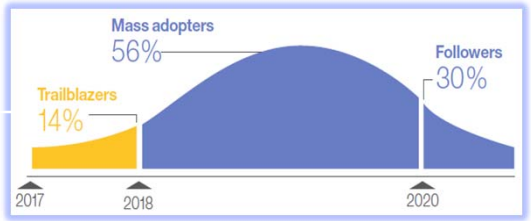
Source: American Banker 2-2017

Adoption Indicators

Survey Readout

14%

in FSS survey of 200 expect to go production at scale with Blockchain in 2017

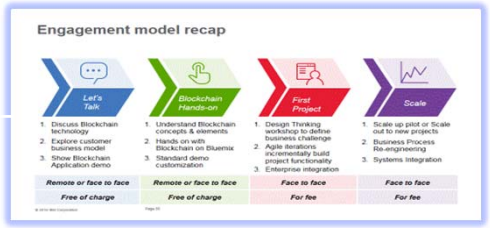


Report: "*Blockchain rewires financial markets*"
IBM Institute for Business Value

Practitioner Readout

400+

Client engagements on IBM Blockchain



IBM Blockchain Garage
https://www.ibm.com/blockchain/getting-started.html

Operator Readout

4

Networks launched on IBM Blockchain

IBM
NORTHERN TRUST
SECURE KEY
everledger

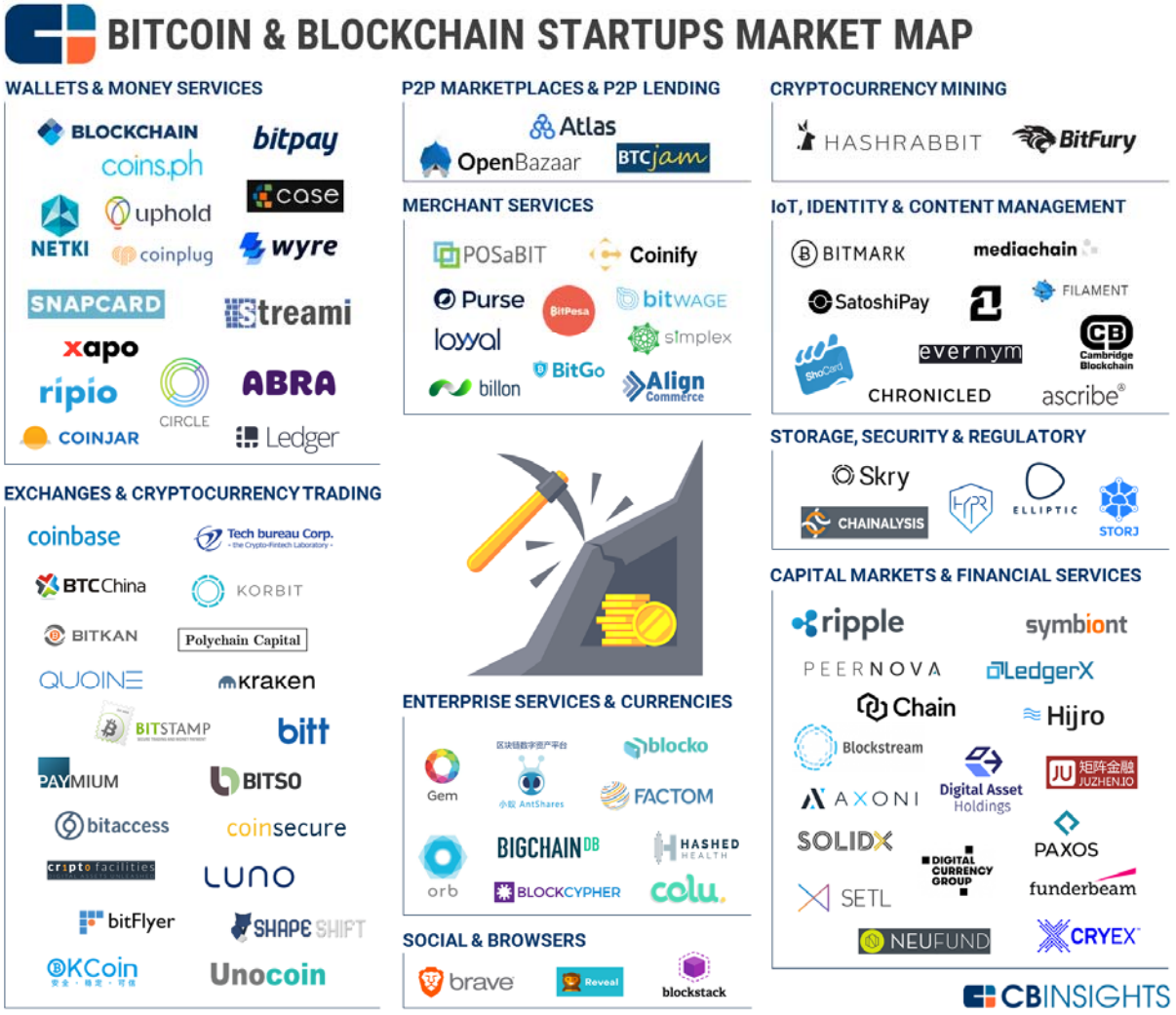
www.ibm.com/blockchain

1.2M diamonds today on the IBM Blockchain Platform

Ongoing Industry Projects/Efforts

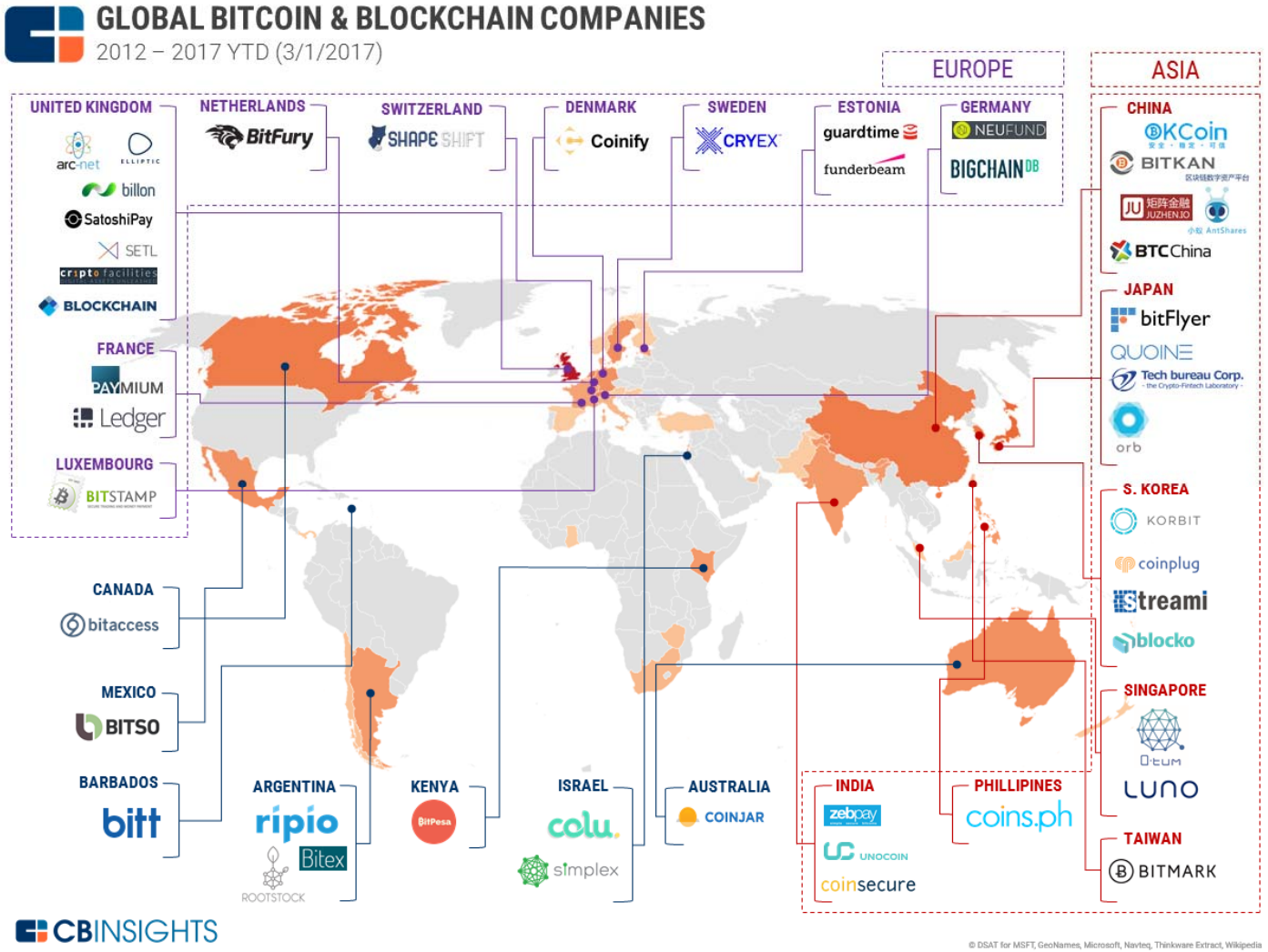
- 11/2016: R3 open sources Corda
- 2/2017: DTCC (Depository Trust & Clearing Corp) Selects IBM, AXONI and R3 to develop DTCC's distributed ledger solution for derivatives processing – expected to go live in early 2018
- 2/2017: Enterprise Ethereum Alliance launched
- 3/2017: Fabric graduates, Incubation to Active
- 7/2017: V1 released
- BigchainDB (Berlin): Starts from DBMS end to add BC features
  - Uses a single RethinkDB Cluster
  - MongoDB support being added

Startup Market Map 2-2017



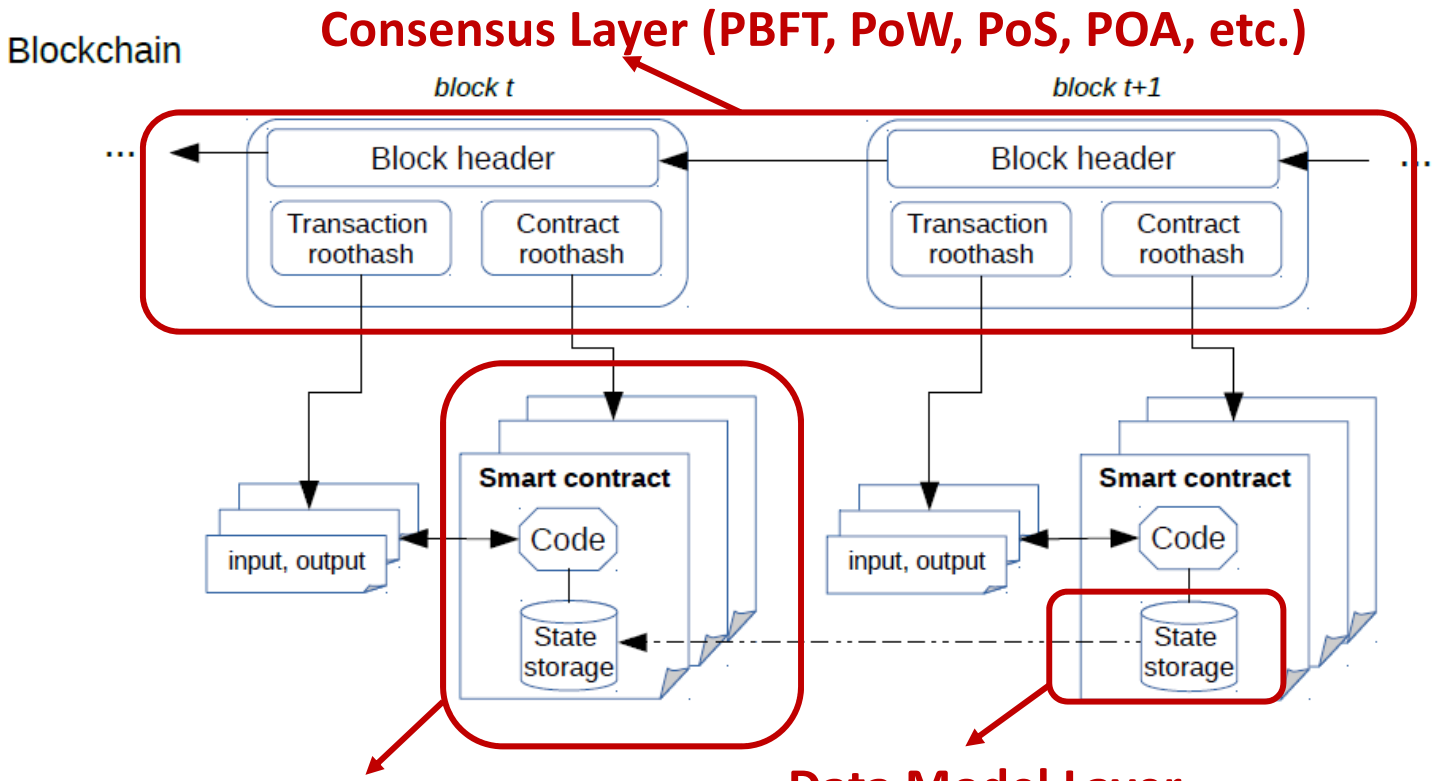
Source: https://www.cbinsights.com/blog/bitcoin-blockchain-startup-market-map/

Global Bitcoin/Blockchain Companies 3/2017



Source: https://www.cbinsights.com/blog/bitcoin-blockchain-startup-global-map/

BC Software Stack



Source: Anh Dinh, et al., SIGMOD 2017

Blockchain Architecture/Feature Choices

- Cryptocurrencies Vs Generalized Assets
- Permissionless/Public Vs Permissioned/Private
- Byzantine Vs Non-Byzantine fault model
- Consensus approach: PoW, PoA, PoET, PBFT, …
- SQL Vs NoSQL data stores
- Transactional stores Vs Non-transactional stores
- Versioned/Unversioned state database
- On-Chain Vs Off-Chain data
- Parallelism exploitation during different phases of transaction execution

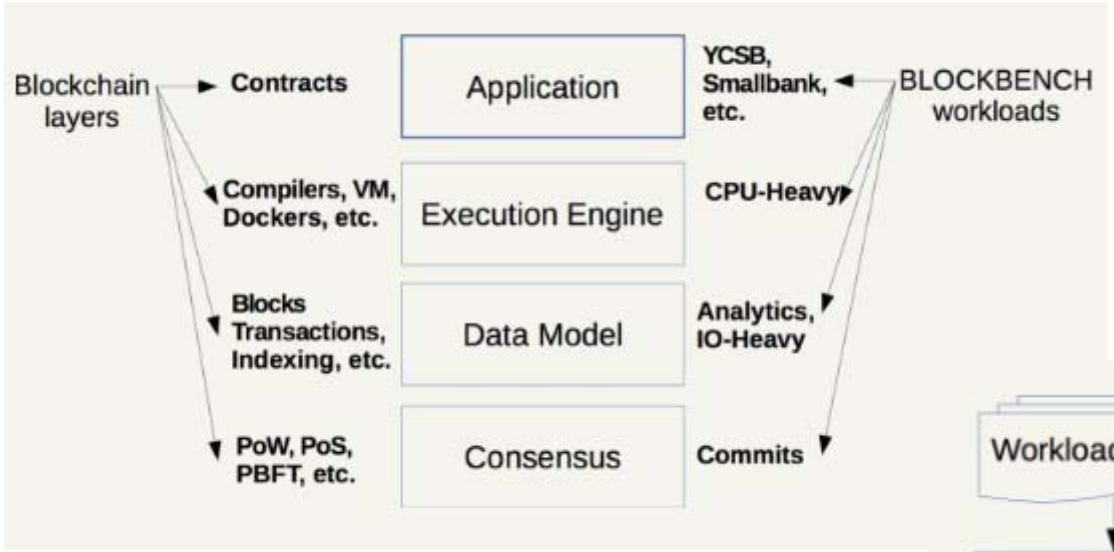
Good Survey Paper: Untangling Blockchain: A Data Processing View of Blockchain Systems, A. Dinh et al.

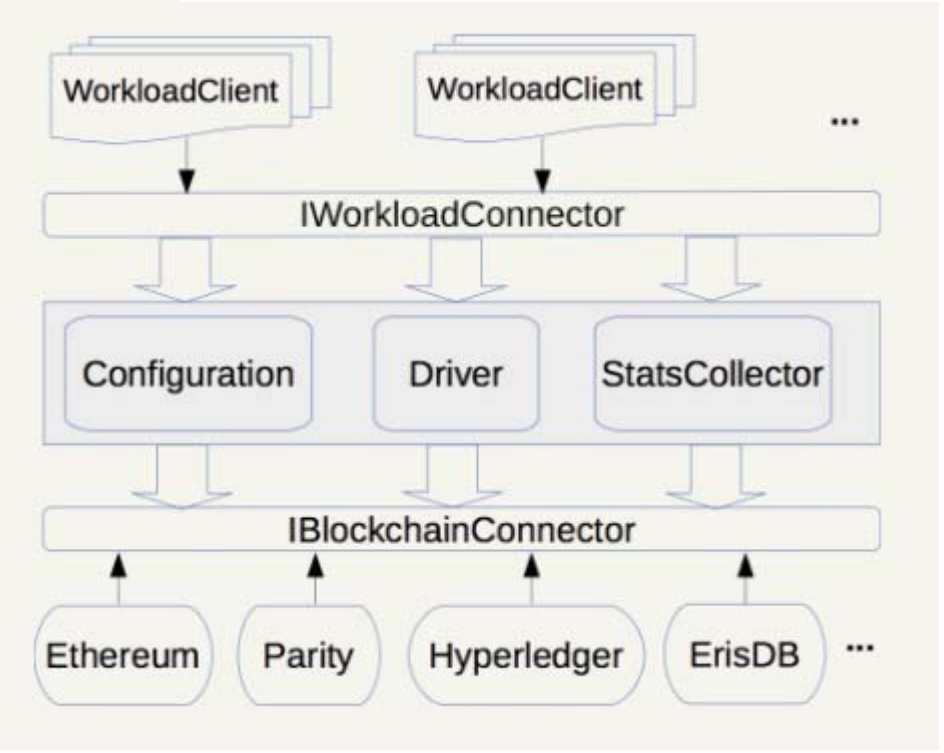
Database Replication

- Primary log replay at replica – homogeneous systems with full DB replicas, typically done for disaster recovery (DR) backup
- Log capture generates DML statements from what is logged and apply executes those statements (e.g., IBM Q Replication)
  - Can handle non-determinism and partial replicas
  - Requires dependency analysis to leverage parallelism at apply time
  - https://www.ibm.com/developerworks/data/roadmaps/qrepl-roadmap.html
- Capture DML statements as issued by application and re-execute them at replica (e.g., H-Store/VoltDB)
  - Cannot handle non-determinism
  - Typically, serial execution of transactions

Upfront (fairly random, unoptimized) ordering of transactions in blockchain systems – leads to all sorts of issues!

Benchmark Framework: BLOCKBENCH (NUS)





- OLTP & OLAP load measured
- Metrics: throughput, latency, scalability, fault tolerance, security
- Consensus methods: Ethereum (PoW), Fabric (PBFT), Parity (PoA)
- Old version of Fabric (pre-V1)
- Fabric performs better
- Fabric scales well up to 16 nodes

Source: Anh Dinh, et al., SIGMOD 2017

Ethereum

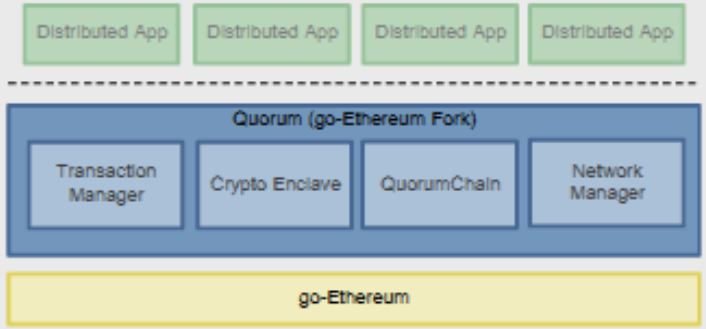
- Public BC like Bitcoin
  - Extends it with Smart Contracts
  - Uses PoW for consensus
  - Own machine lang & VM
  - *gas* charging!
- Most apps relate to its currency Ether
- *Enterprise Ethereum Alliance* (**EEA**): JPMorgan Chase, Microsoft, Intel, Accenture, Banco Santander, BNY Mellon, ConsenSys, Credit Suisse, ING, Thomson Reuters, UBS, Wipro
  - EEA will add confidentiality (Quorum), scalability (pluggable consensus) and permissioning to Ethereum
  - Focus on specification, **EntEth** 1.0 with Python reference client, benchmarking, compliance testing and tools
  - Develop standards for Ethereum: best practices, security, privacy, scalability, interoperability
- Quorum from JPMorgan

Quorum Overview J.P. Morgan

Highlights

- **Built on Ethereum**
  - First mover advantage. In production since July, 2015.
  - 50,000+ unit tests, Security Audits, Bounty Program
  - Largest Ecosystem of Developers, Tools, DApp's
  - Public Ethereum blockchain protects over $1B+ Ether[1]
- **Simple Privacy Design**
  - Supports both private and public transactions and smart contracts
- **Single Blockchain Architecture**
  - All public and private smart contracts and state derived from a single, common, complete blockchain of transactions validated by every node in the network
  - Private smart contract state validated by parties to contract only
  - Best of both worlds… every node validating the list of transactions while only exposing details of private transactions and contracts to relevant parties
- **High Performance**
  - Able to process **dozens to hundreds of transactions per second**, depending on system configuration; enough to support institutional volumes

Architecture

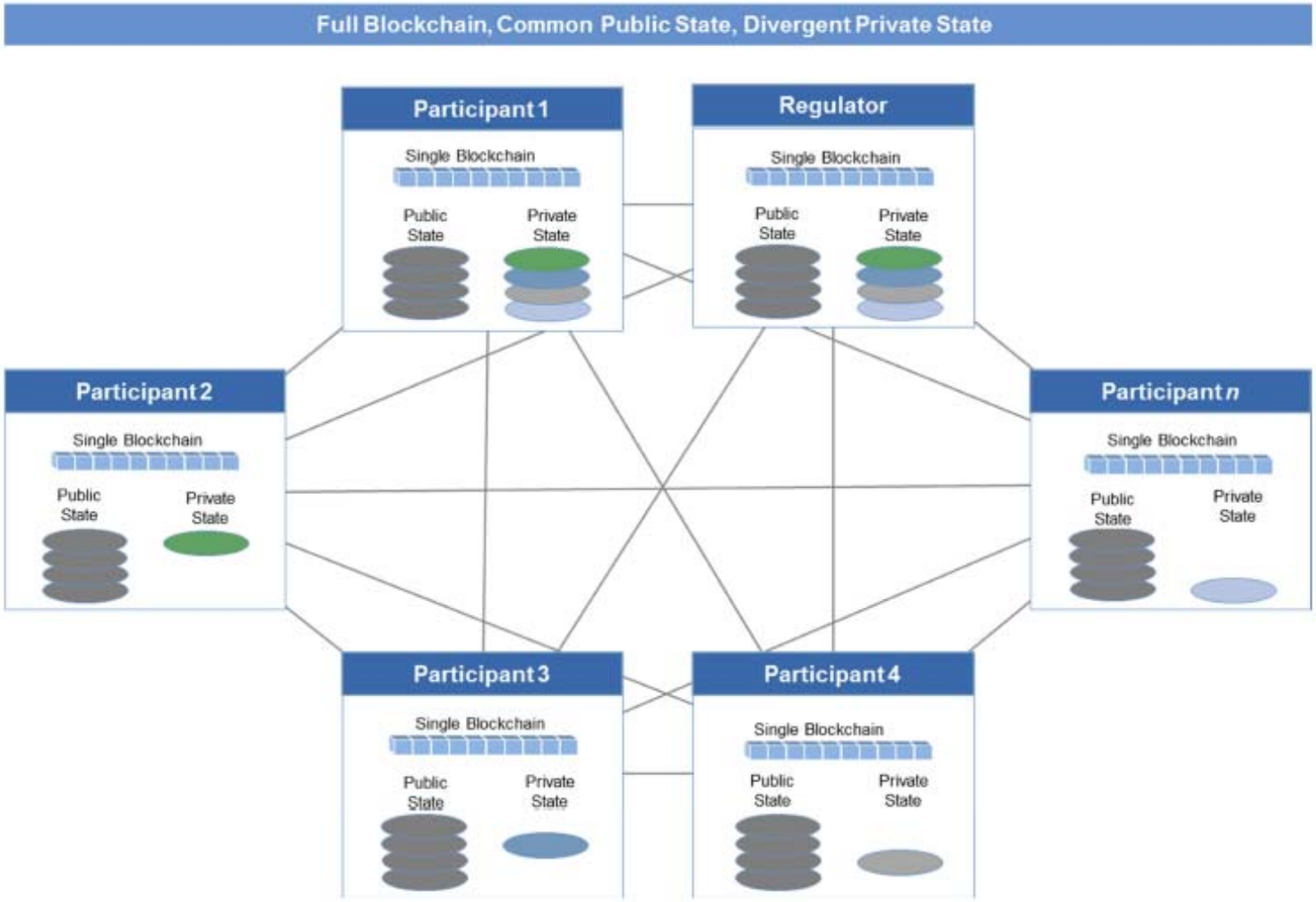


Components

- **Transaction Manager** – allows access to encrypted transaction data for private transactions, manages local data store and communication with other Transaction Managers
- **Crypto Enclave** – responsible for private key management and encryption and decryption of private transaction data
- **QuorumChain** – voting-based, BFT-hardened consensus mechanism that utilises core Ethereum features to verify and propagate votes through the network
- **Network Manager** – controls access to the network, enabling a permissioned network to be created

[1]As of 22-Sep-2016

Quorum Network J.P. Morgan



Linux Foundation's Hyperledger Project

- *Open Ledger Project* announced December 17, 2015 with **17** founders, now over **100** members
- *Hyperledger Project* rebrand in February 2016
- Collaborative effort to advance Blockchain technology by identifying and addressing important features for a cross-industry open standard for distributed ledgers that can transform the way business transactions are conducted globally
- Open source, open standards, open governance

Enable adoption of shared ledger technology at
a pace and depth not achievable by any one company or industry

QUICK FACTS	
Chairman	Blythe Masters/DAH
Executive Director	Brian Behlendorf
Technical Chair	Chris Ferris/IBM
Contribution	44,000 lines of code in February 2016
Sprint to one codebase with unified thinking	Staged releases

www.hyperledger.org

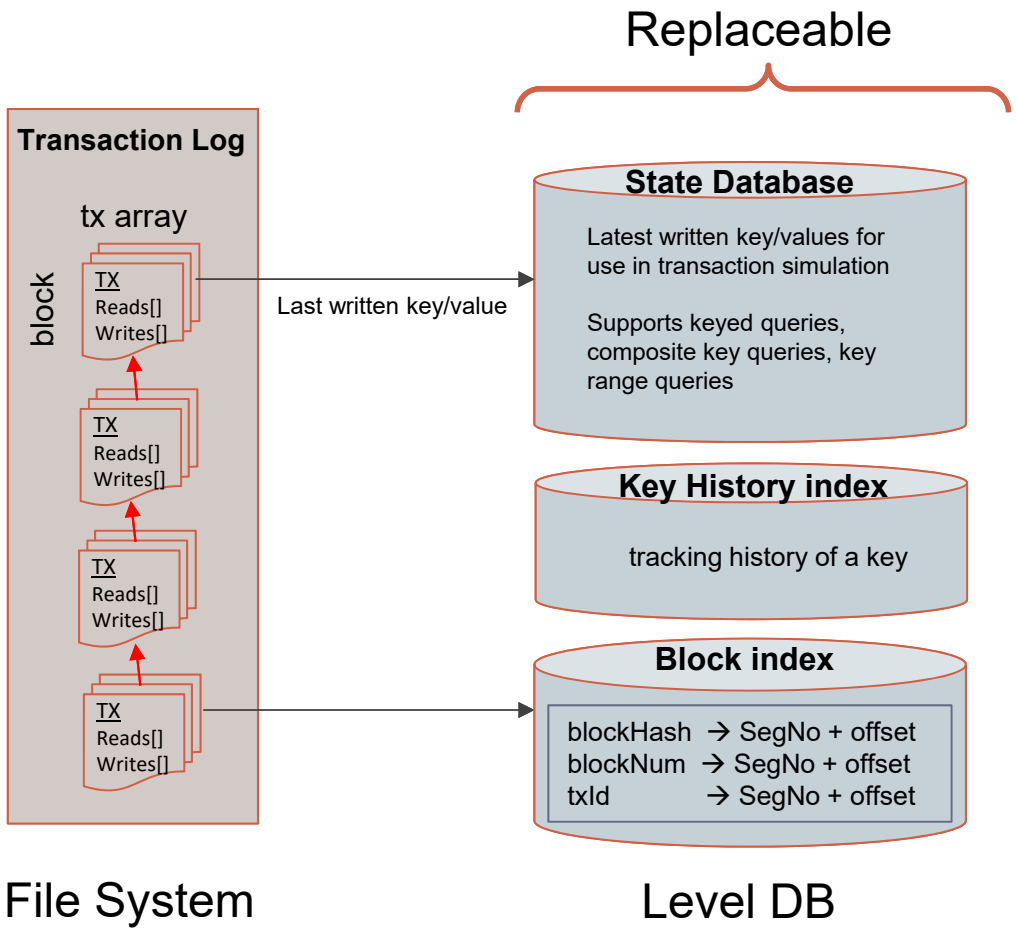
Hyperledger Fabric Project

- Initiated by IBM with IBM open source ledger contribution
  http://hyperledger-fabric.readthedocs.io/en/latest/
- Significant change in architecture from V0.6 to V1
  - Chaincode trust flexibility
  - Scalability
  - Confidentiality
  - Consensus modularity
- Used PBFT for consensus before V1 Miguel Castro, Barbara Liskov: Practical Byzantine Fault Tolerance and Proactive Recovery. ACM Trans. Comput. Syst. (TOCS), 20(4), 2002.
- Other Hyperledger Projects: Iroha, Sawtooth, Composer, …
  https://www.hyperledger.org/
- Customers doing trials: Bank of Tokyo, London Stock Exchange, Maersk, Northern Trust, Walmart

Fabric V1 Architecture

- Elements of the Architecture
  - Chaincode: System and regular ones. Deploy and Invoke latter.
  - State: Versioned KV model - stored in Level DB or CouchDB
  - Ledger data: Blockchain has full state, including history
  - Transactions
- Kafka for Ordering:
  - No Byzantine fault handling
  - Done to improve performance
  - Pluggable consensus permits other methods

Fabric V1 Ledger



CouchDB (external option) supports keyed queries, composite key queries, key range queries, plus full data rich queries (beta in 1.0)

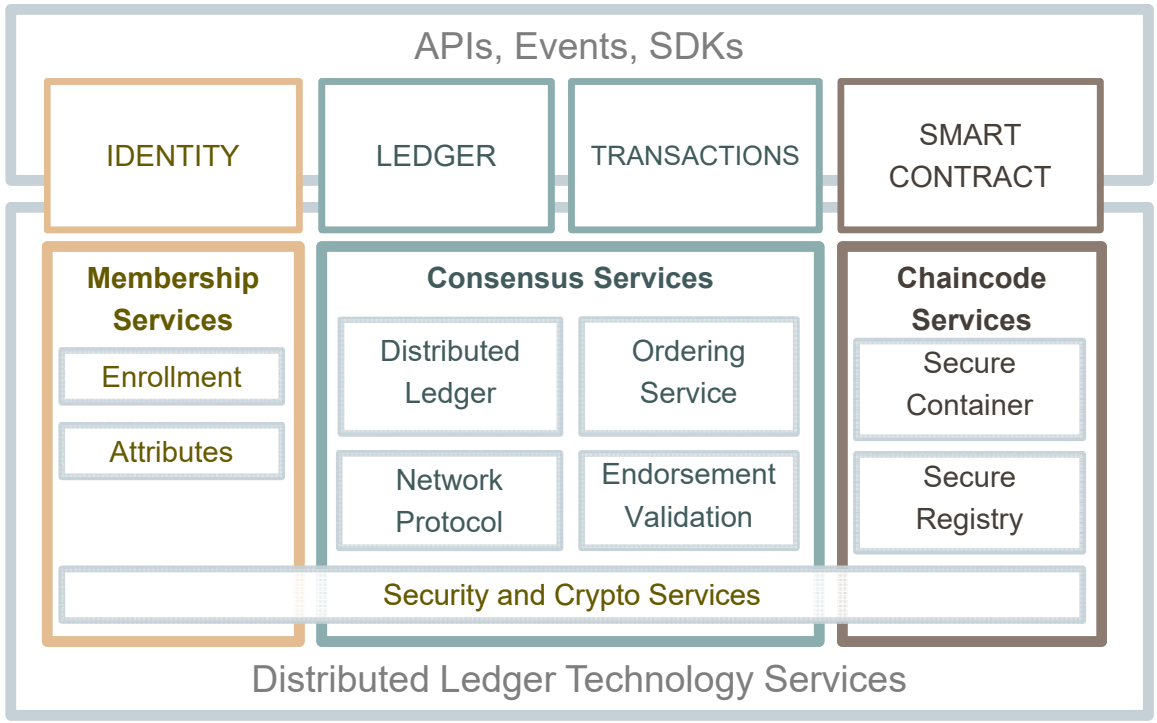
Hyperledger Fabric V1 Components

Hyperledger Fabric enables the creation of blockchain networks that protect information with the accountability needed by regulated businesses.

Hyperledger Fabric is implemented as a modular architecture, consisting of the following components:

CLIENT SDK	MEMBERSHIP	LEDGER	CHAINCODE	CONSENSUS
Enables the creation of applications that deploys and invokes transactions atop a shared ledger. Support is provided for both Node.js or Java SDK	Provides a specialized digital certificate authority for issuing certificates to members of the blockchain network Leverages cryptographic functions provided by Hyperledger Fabric	Provides a distributed encrypted ledger, including an append-only data store. Provides ability to query, and write data across distributed ledgers	Transactions logic is written as chaincode (in the Go or Java languages). Chaincode executes in Docker containers	Service enabling digitally signed transactions to be proposed and validated by network members

Fabric V1 Reference Architecture



IDENTITY
Pluggable, Membership, Privacy and Auditability of transactions.

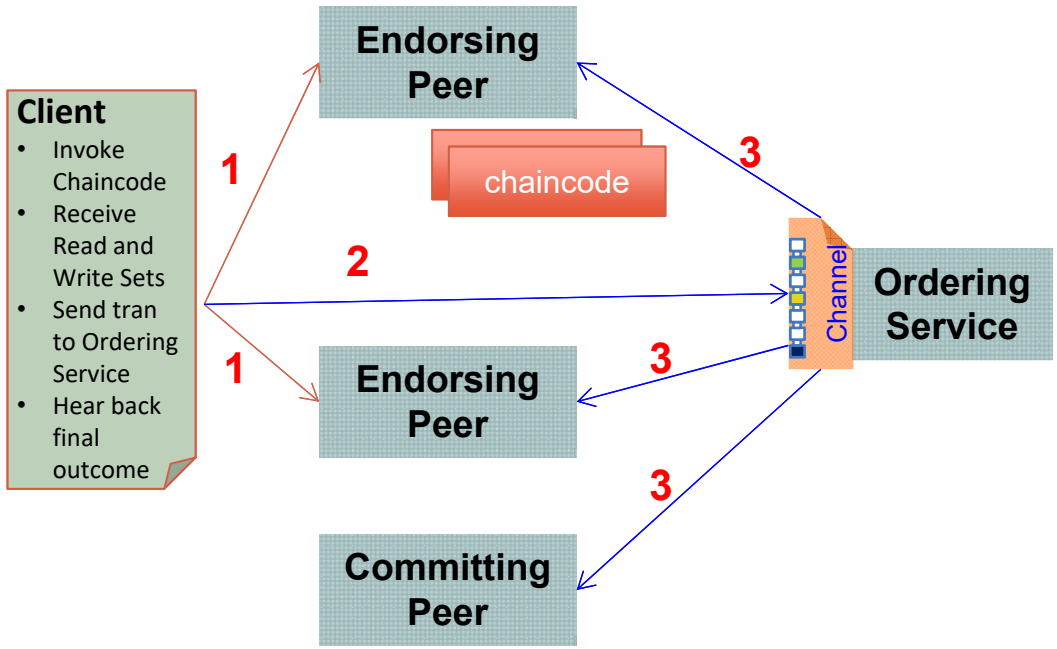
LEDGER | TRANSACTIONS
Distributed transactional ledger whose state is updated by consensus of stakeholders

SMART CONTRACT
"Programmable Ledger", provide ability to run business logic against the blockchain (aka smart contract)

APIs, Events, SDKs
Multi-language native SDKs allow developers to write DLT apps

Transaction Execution Overview Fabric V1

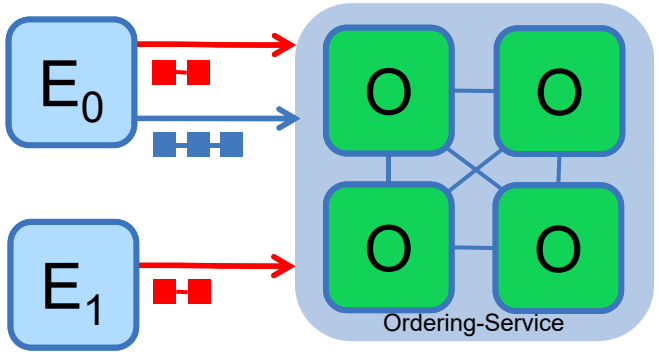
3 Stage Execution: Endorsement, Ordering, Validation/Commit



- Transaction is sent to the counter-parties represented by Endorsing Peers on their Channel
- Each Peer simulates transaction execution by calling specified Chaincode function(s) and signs result (Read-Write Sets)
- Each Peer may participate in multiple channels allowing concurrent execution
- Ordering Service accepts endorsed transactions and orders them according to the plug-in consensus algorithm then delivers them on the channel
- All (Committing) peers on channel receive transactions: on successful validation, commit to ledger. No chaincode execution.

V1 Channels Provide Data Partitioning

SDK or the CLI can create separate channels which will isolate and segregate transactions and ledger.



- Chaincode is installed on peers that need to access the asset states to perform reads and writes
- Chaincode is instantiated on specific channels for specific peers
- Ledgers exist in the scope of a channel
  - Ledgers may be shared across entire network of peers
  - Ledgers may be only on a specific set of participants
- Peers can participate in multiple channels

Fabric V1 Transaction Flow

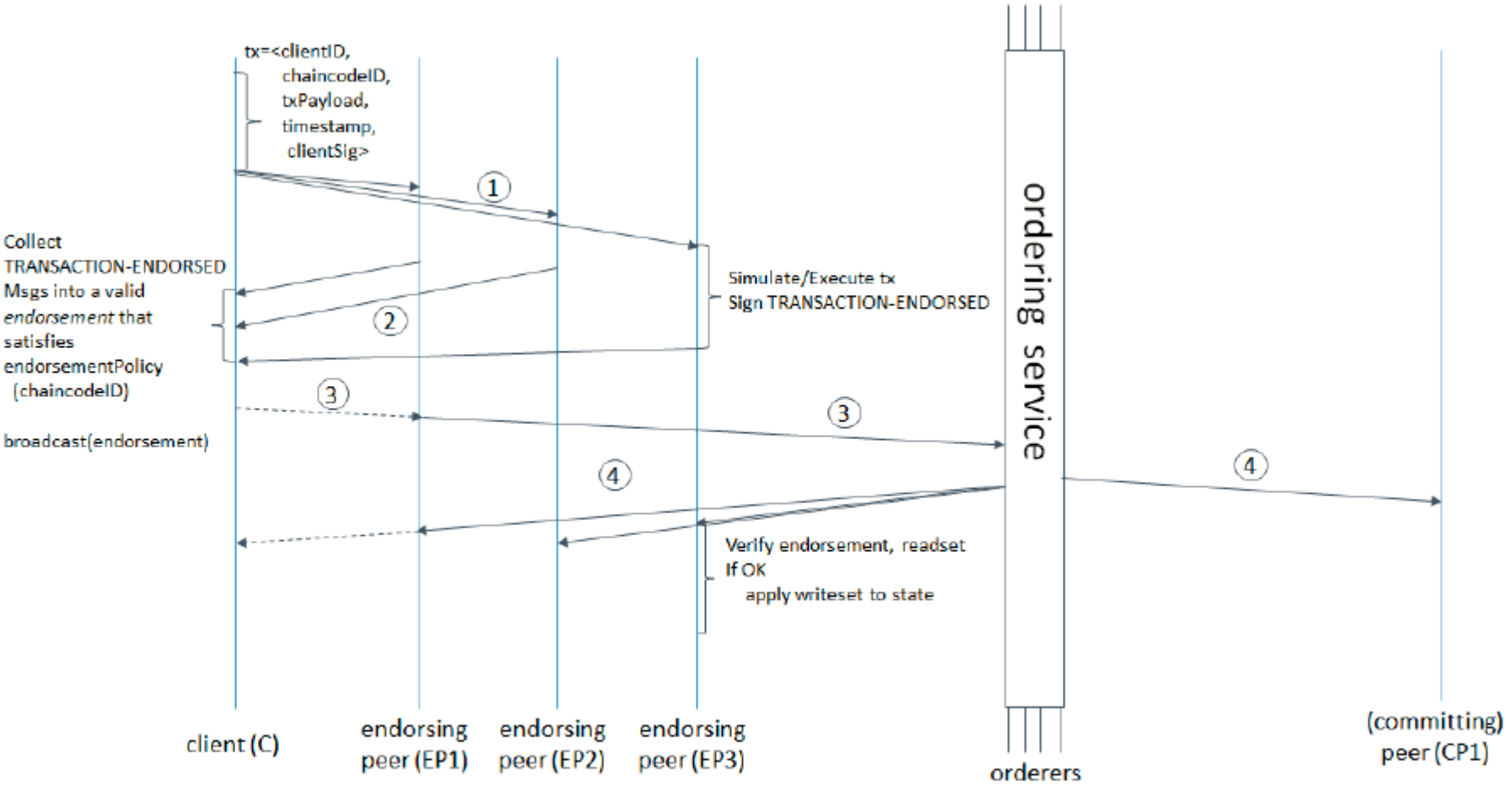


Illustration of the transaction flow (common-case path).

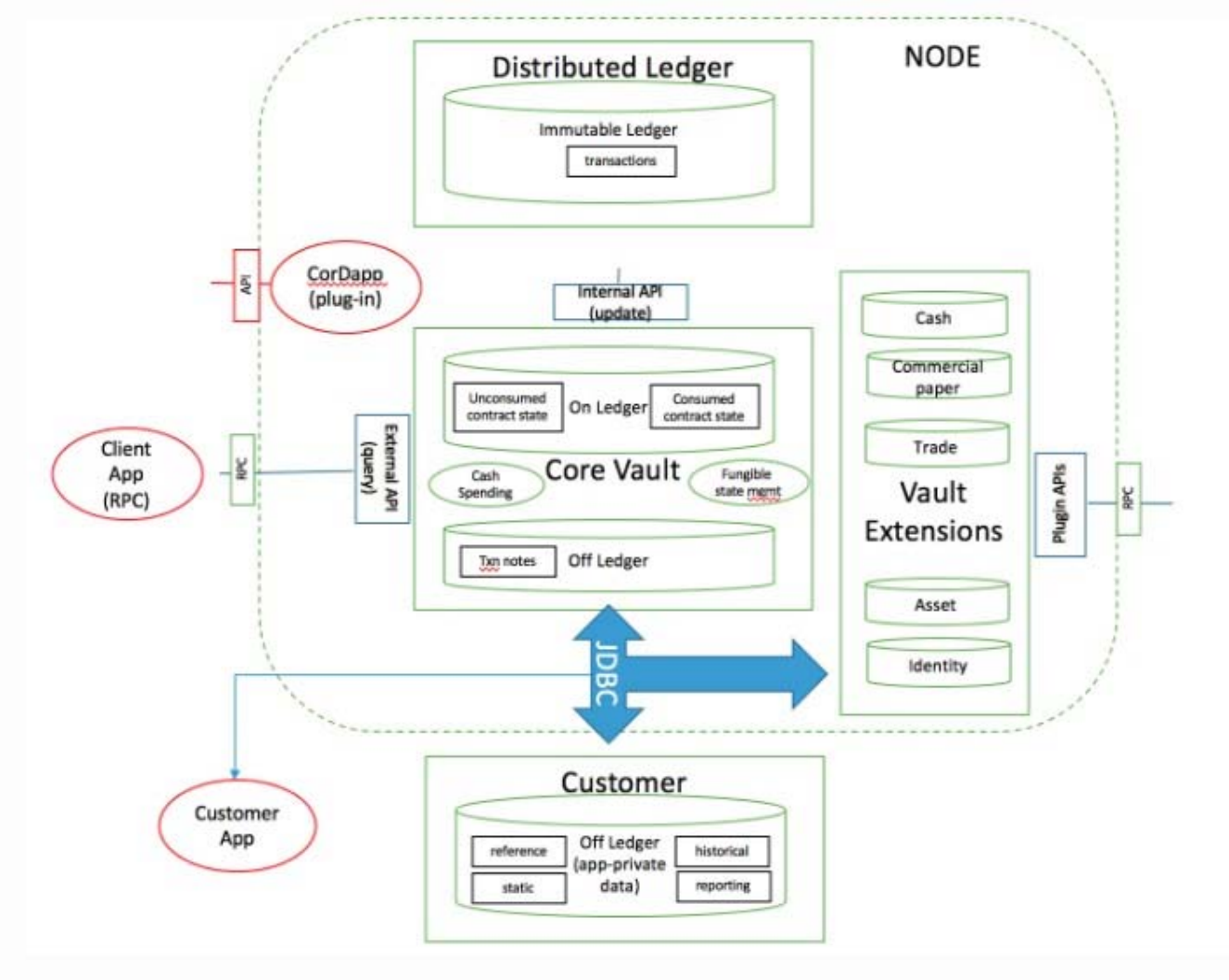
DBMS Implications

- Simulation concept requires layer between chaincode and State DB having to take on analysis of DBMS calls
  - Update statements split into two: read part and write part
  - Read alone sent to DBMS with modifications to retrieve version #s for items read
  - Writes not sent to DBMS but processed and cached locally – doesn't allow for read your own write by chaincode transaction
- During Commit phase, read sets validated by retrieving each item's version # individually and then, if validation succeeds, writes also done one at a time
- Dealing with phantoms requires reexecution of query during commit phase to be sure simulation read set same as read set at Commit time
- Chaincode portability across different State DBMSs hard to do
- Lots of open questions and research issues in this area

R3 Alliance & Corda

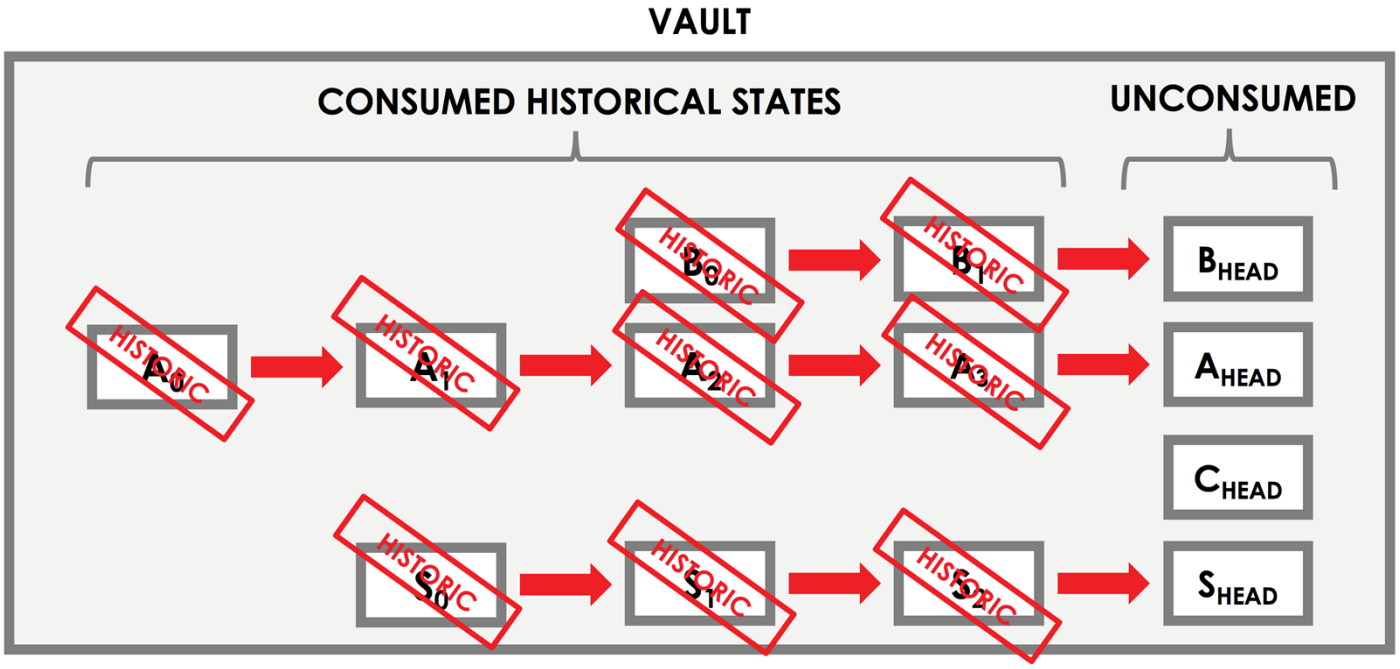
- Barclays, BBVA, Commonwealth Bank of Australia (CBA), Credit Suisse, J.P. Morgan, State Street, Royal Bank of Scotland, UBS
- Special features for JVM to guarantee deterministic behavior
- Hearn, M. Corda: A distributed ledger, Version 0.5, November 2016. https://docs.corda.net/_static/corda-technical-whitepaper.pdf
- Nodes backed by RDBMS, ledger data SQL queryable and joinable with private tables
- Corda written in Kotlin (simpler Scala with much better Java interoperability) from JetBrains – contracts in Kotlin/Java
- Contract execution is deterministic and its acceptance of a transaction is based on the transaction's contents alone. A transaction is only valid if the contract of every input state and every output state considers it to be valid

R3 Corda Vault

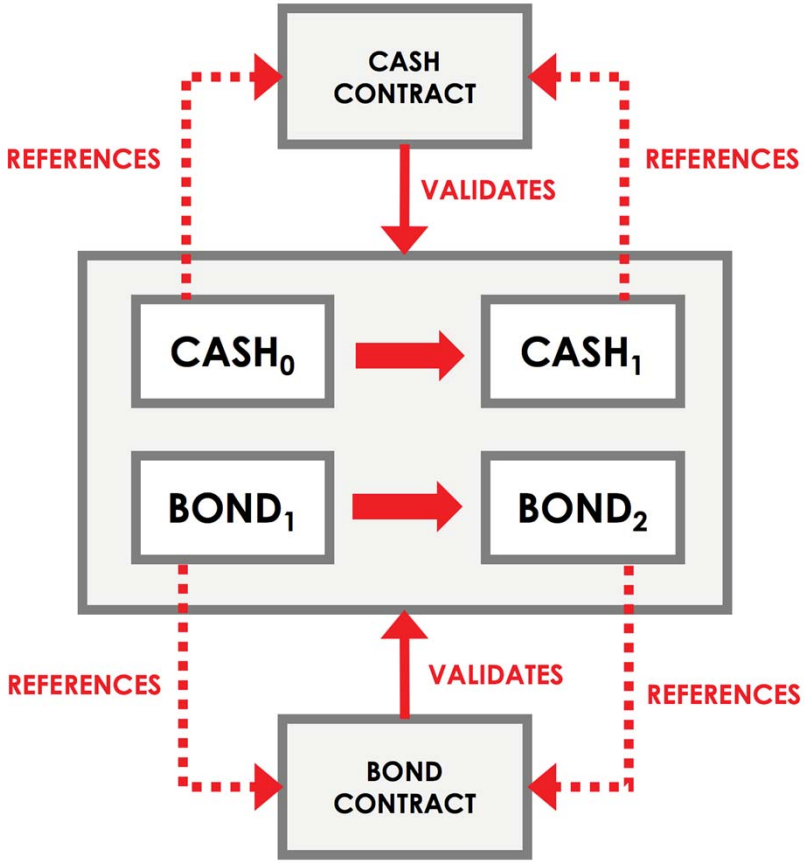


Corda Vault and State

Each node on the network maintains a *vault* - a DB where it tracks all the current and historic states that it is aware of, and which it considers to be relevant to itself

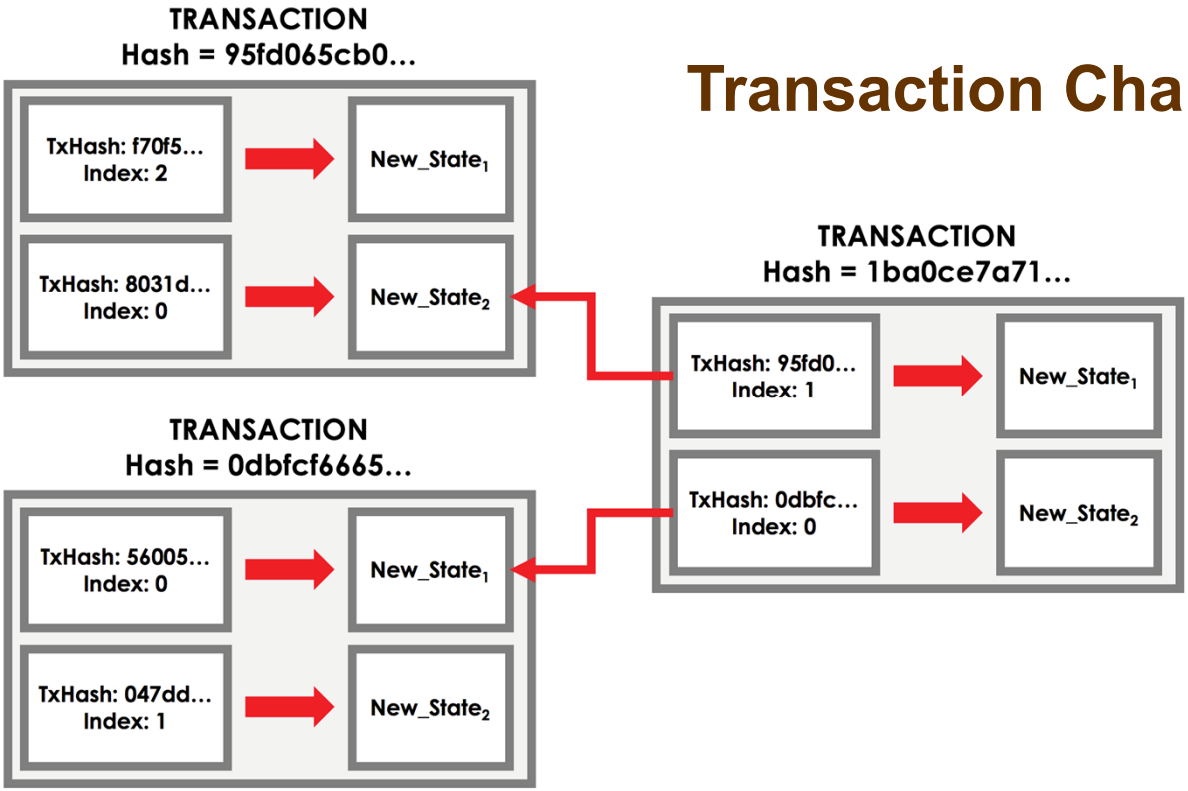


Corda Contract Validity



A transaction is only valid if it is digitally signed by all required signers. However, even if a transaction gathers all the required signatures, it is only valid if it is also **contractually valid**.

Corda Transactions

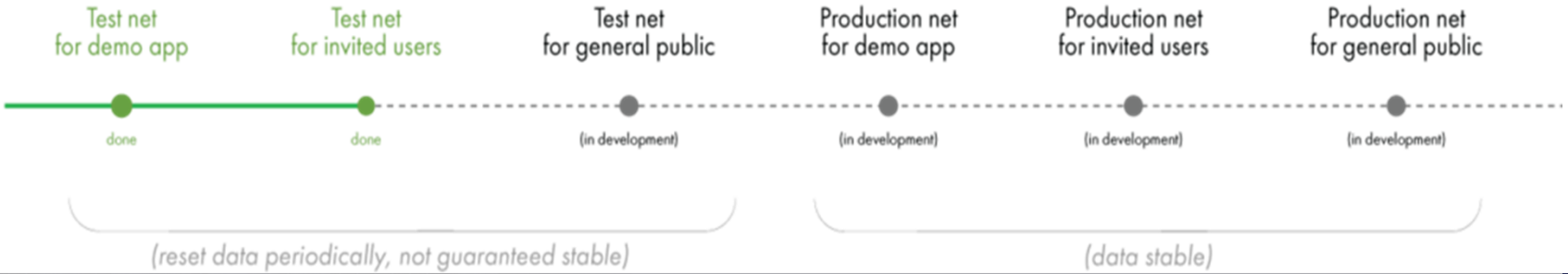


Notary and Regular Transactions

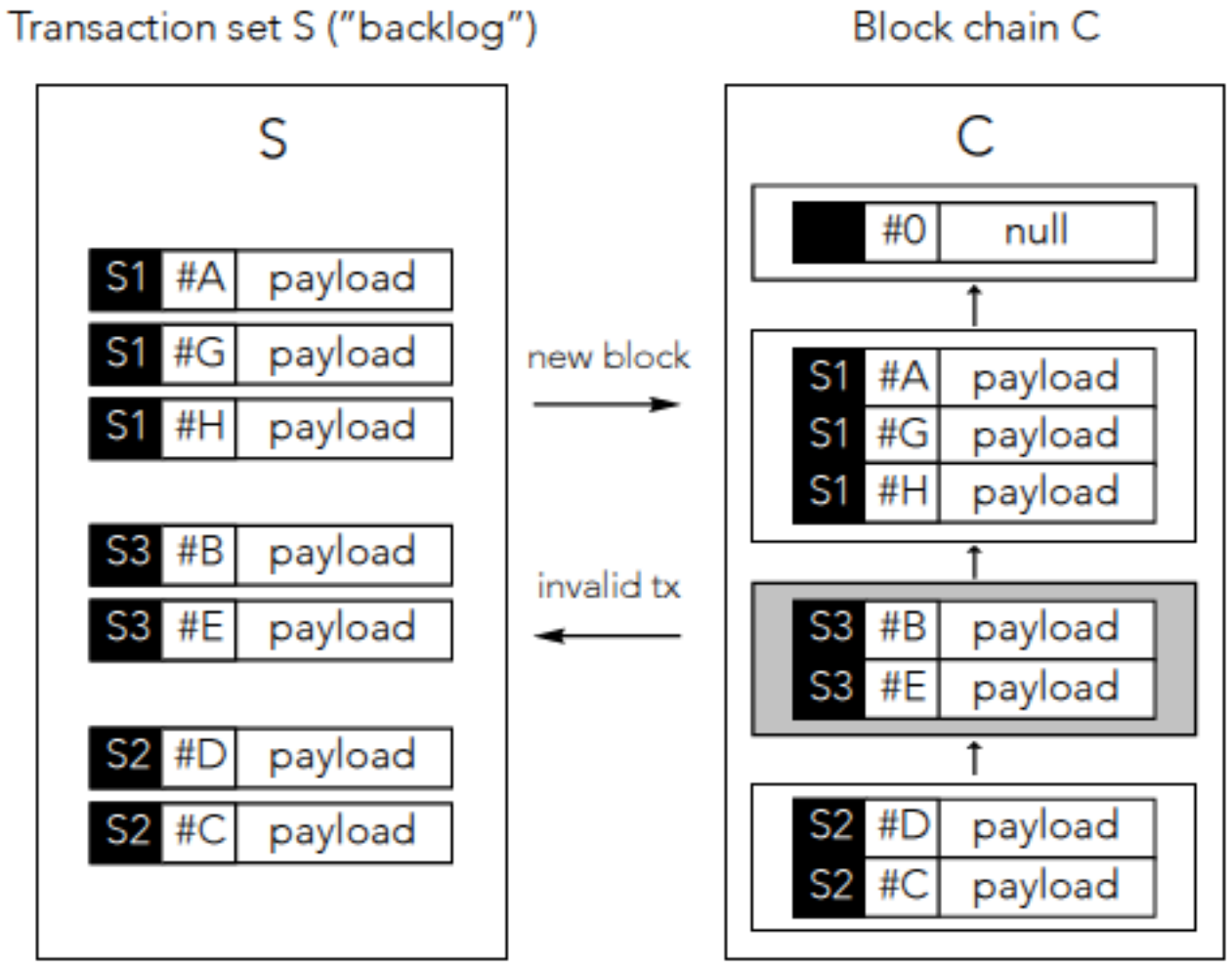
Every state has an appointed notary, and a notary will only notarize a transaction if it is the appointed notary of all the transaction's input states.

Bigchain DB

- Designed to merge best of DB & blockchain worlds
  - Scale, permissioning and NoSQL querying from DB side
  - Decentralization, immutability, and assets from blockchain side
  - All data as JSON documents in RethinkDB/MongoDB
  - Intended to play well alongside decentralized file systems (e.g., IPFS) and processing (e.g., Ethereum)
  - Modelled around *assets*, and *inputs* and *outputs* are the mechanism by which control of an asset is transferred
- Release 1.0 to ship 6/2017
- IPDB (Interplanetary Database) Network and a Foundation for its governance - management of personal data, reputation, and privacy, along with secure attribution, metadata, licensing, and links to media files https://ipdb.foundation/



BigchainDB Architecture



BigchainDB

- Central notion: assets
- Mechanisms for asset transfer: inputs and outputs
- Amount of asset encoded in outputs of a transaction
- Each output might be spent separately
- To spend an output, its conditions must be met by an input that provides corresponding fulfillments
- Simple signature condition: asset given to entity controlling a corresponding private key

BigchainDB

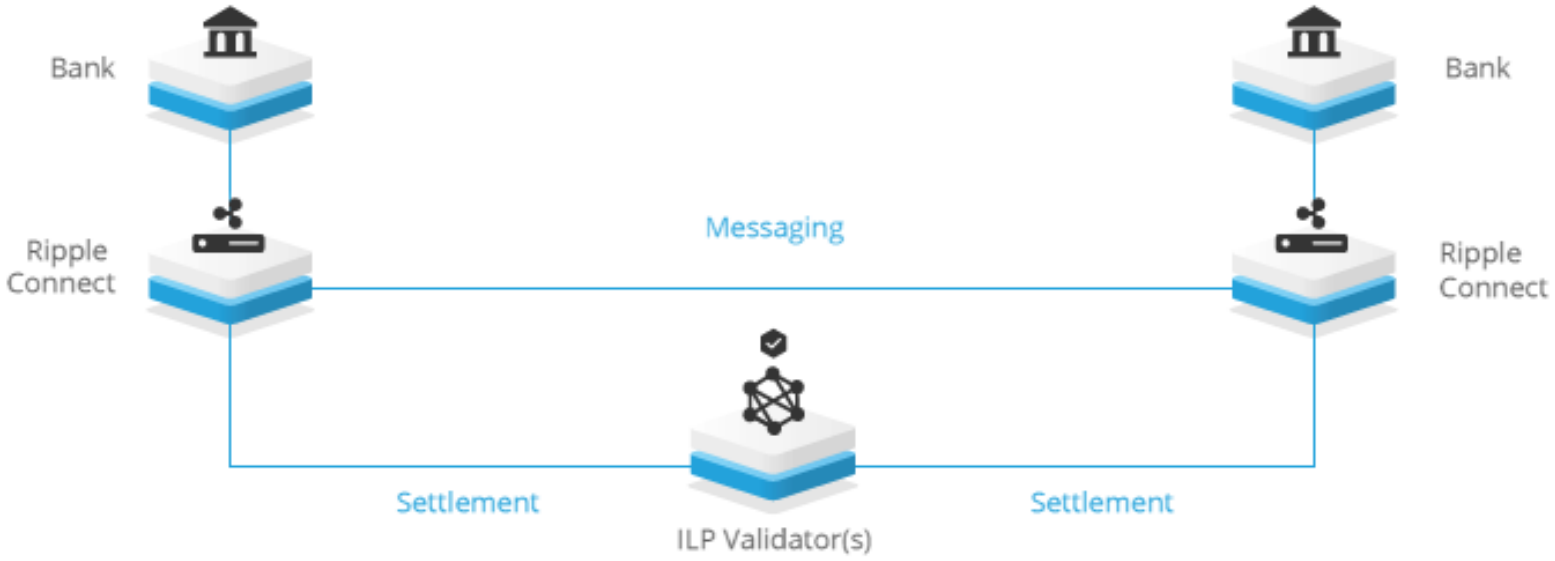
	Traditional Blockchain	Traditional Distributed DB	BigchainDB
High Throughput; increases with nodes↑	-	✓	✓
Low Latency	-	✓	✓
High Capacity; increases with nodes↑	-	✓	✓
Rich querying	-	✓	✓
Rich permissioning	-	✓	✓
Decentralized control	✓	-	✓
Immutability	✓	-	✓
Creation & movement of digital assets	✓	-	✓
Event chain structure	Merkle Tree	-	Hash Chain

Sawtooth (Intel)

- Incubation project of Hyperledger
- Proof of Elapsed Time (PoET) – Consensus Protocol
  - Every validator requests a wait time from a trusted function
  - Validator with shortest wait time for a particular transaction block is elected leader
  - Guaranteed wait time
  - Randomness in leader election (~ to lottery algorithm)
- Intended to run in a Trusted Execution Environment (TEE), e.g., Intel's Software Guard Extensions (SGX)
  - Currently experimental and not secure
- Concept of Transaction Family and Transaction Dependencies
- https://intelledger.github.io/introduction.html

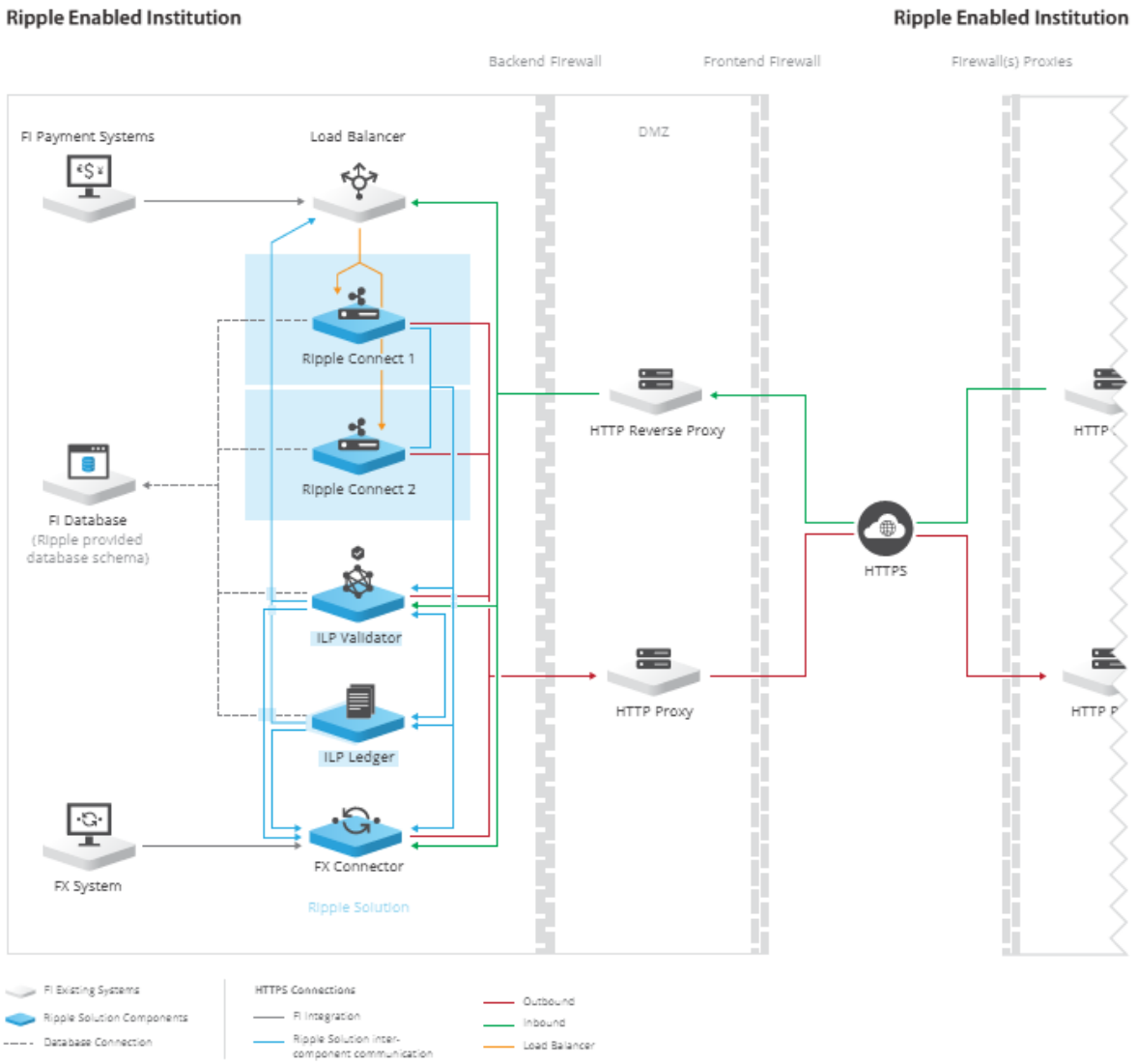
Ripple

Ripple's solution is built around an open, neutral protocol (ILP) to interoperate different ledgers and networks. It offers a cryptographically secure end-to-end payment flow with transaction immutability and information redundancy. It is designed to comply with your bank's risk, privacy and compliance requirements. It is architected to fit within your bank's existing infrastructure, resulting in minimal integration overhead and business disruption.



Works with PostgreSQL 9.4 or Microsoft SQL Server 2012

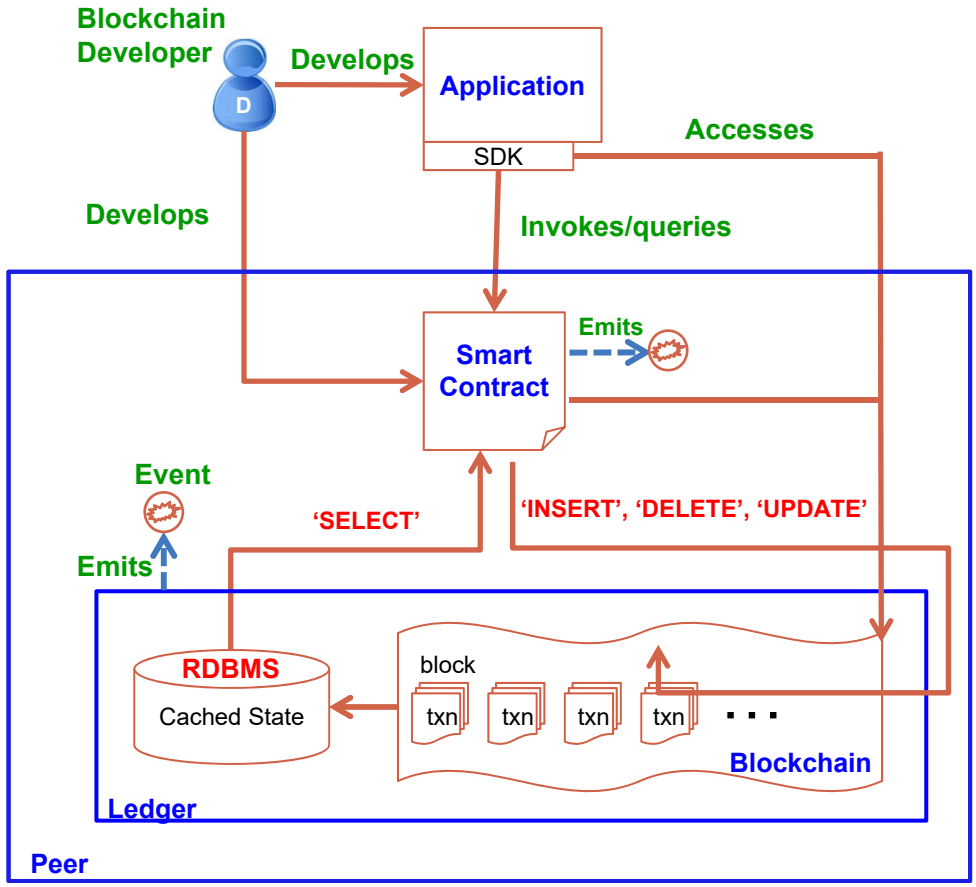
Ripple


Ripple Enabled Institution
Ripple Enabled Institution
Backend Firewall
Frontend Firewall
Firewall(s) Proxies
FI Payment Systems
Load Balancer
DMZ
Ripple Connect 1
Ripple Connect 2
HTTP Reverse Proxy
HTTP
FI Database
(Ripple provided database schema)
HTTPS
ILP Validator
HTTP Proxy
HTTP P
ILP Ledger
FX System
FX Connector
Ripple Solution
FI Existing Systems
Ripple Solution Components
Database Connection
HTTPS Connections
FI Integration
Ripple Solution inter-component communication
Outbound
Inbound
Load Balancer

Blockchain Architecture/Feature Choices

- Cryptocurrencies Vs Generalized Assets
- Permissionless/Public Vs Permissioned/Private
- Byzantine Vs Non-Byzantine fault model
- Consensus approach: PoW, PoA, PoET, PBFT, …
- SQL Vs NoSQL data stores
- Transactional stores Vs Non-transactional stores
- Versioned/Unversioned state database
- On-Chain Vs Off-Chain data
- Parallelism exploitation during different phases of transaction execution

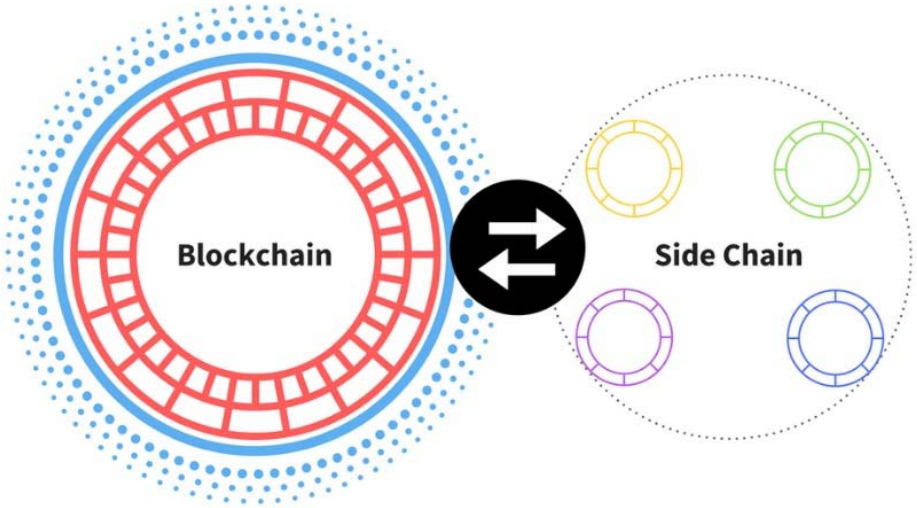
Application Flow with RDBMS (In Progress)



- Developers create application and smart contracts (chaincodes)
  - Chaincodes are deployed on the network and control the state of the ledger
  - Application handles user interface and submits transactions to the network which call chaincodes
- Network emits events on block of transactions allowing applications to integrate with other systems

Futuristic Topics

- Chaincode portability and power of data APIs
- DBMS enhancements to add BC features
- Standards across BC systems
- Cross channel transactions
- Non-deterministic actions
- Analytics on chaincode data
- Many app design issues
- Design tools for endorsement decisions



Numerous research possibilities for database and distributed systems people in this new era of distributed computing!

More Information

Links to Videos, Slides, Bibliography, Twitter Handles

http://bit.ly/CMbcDB

Follow me on

Twitter: @seemohan

Facebook: http://www.facebook.com/cmohan

LinkedIn: http://www.linkedin.com/in/seemohan/