

WATSON
BRAZIL
DEVELOPER
SUMMIT
2017

BlockChain

Laboratório Parte 1

[Carlos L Rischio] - [carlosr@br.ibm.com]

[Marcos Tadeu Brisola Vieira] - [marcos.vieira@br.ibm.com]

[Percival Silva de Lucena] - [plucena@br.ibm.com]



WATSON
BRASIL
DEVELOPER
SUMMIT
2017

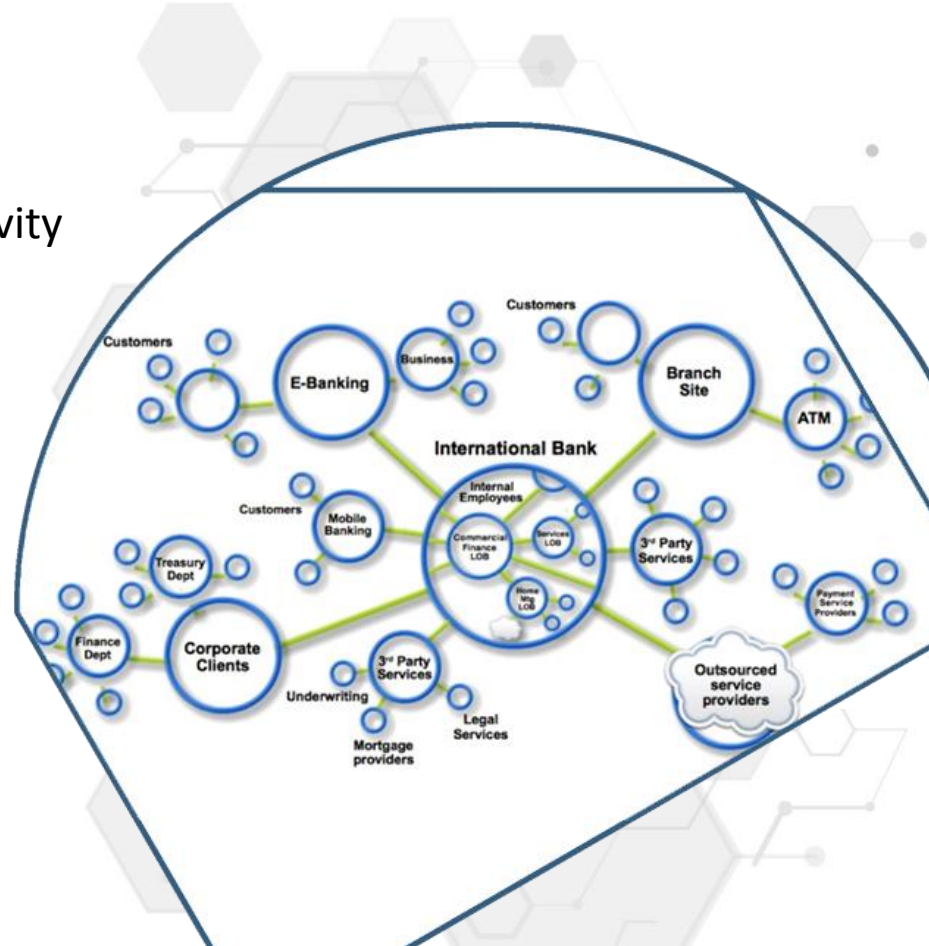


- Overview BlockChain



Business networks, wealth & markets

- Business Networks benefit from connectivity
 - Participants are customers, suppliers, banks, partners
 - Cross geography & regulatory boundary
- Wealth is generated by the flow of goods & services across business network in transactions and contracts
- Markets are central to this process:
 - Public (fruit market, car auction), or
 - Private (supply chain financing, bonds)



Transferring assets, building value

Anything that is capable of being owned or controlled to produce value, is an asset



Two fundamental types of asset

- Tangible, e.g. a house
- Intangible, e.g. a mortgage



Intangible assets subdivide

- Financial, e.g. bond
- Intellectual, e.g. patents
- Digital, e.g. music



Cash is also an asset

- Has property of anonymity

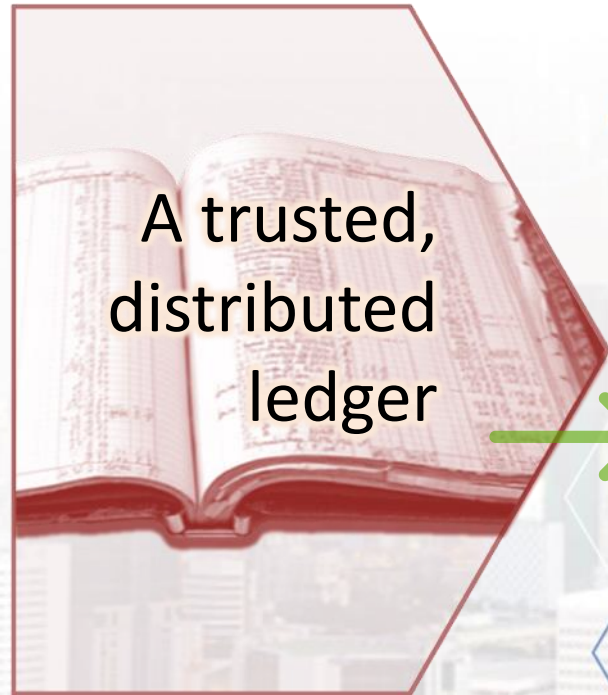
Ledgers are key ...

Ledger is THE system of record for a business. Business will have multiple ledgers for multiple business networks in which they participate.

- **Transaction** – an asset transfer onto or off the ledger
 - John gives a car to Anthony (simple)
- **Contract** – conditions for transaction to occur
 - If Anthony pays John money, then car passes from John to Anthony (simple)
 - If car won't start, funds do not pass to John (as decided by third party arbitrator) (more complex)



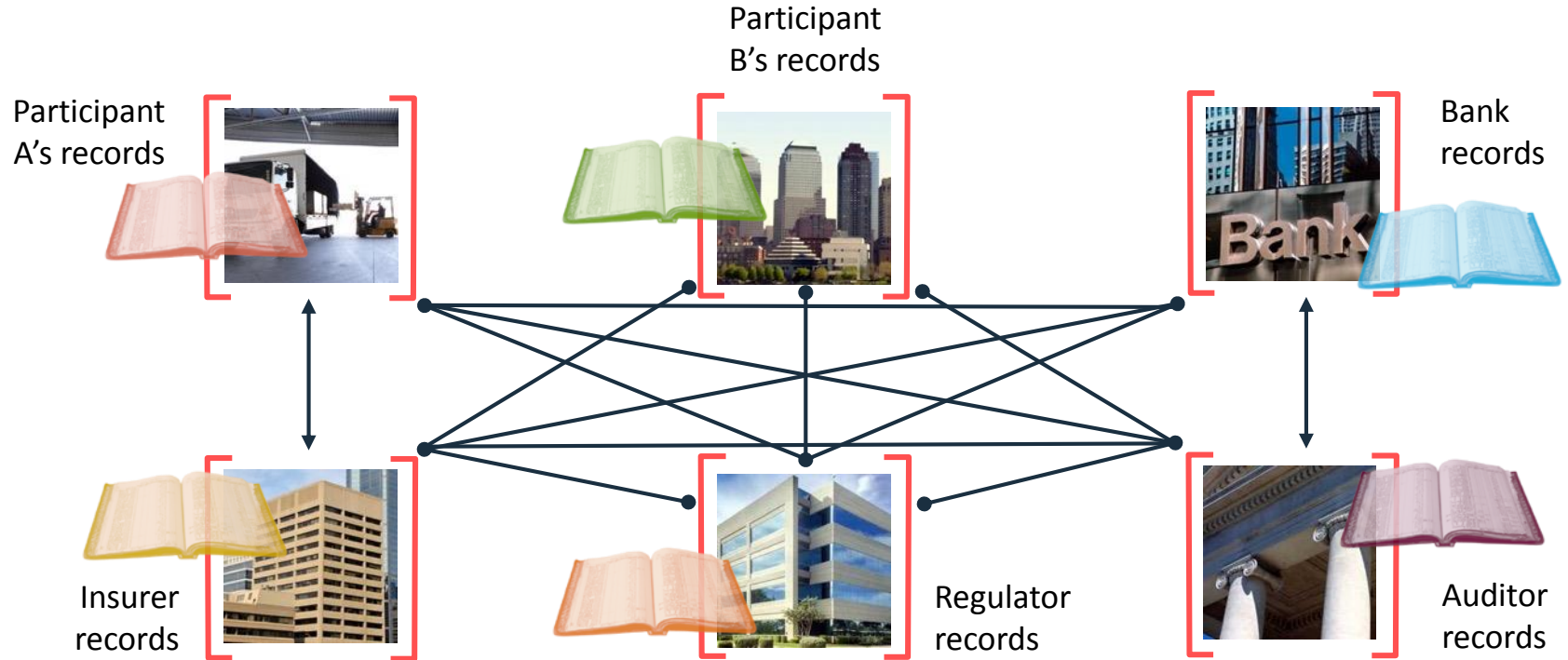
Introducing Blockchain



with shared
business
processes

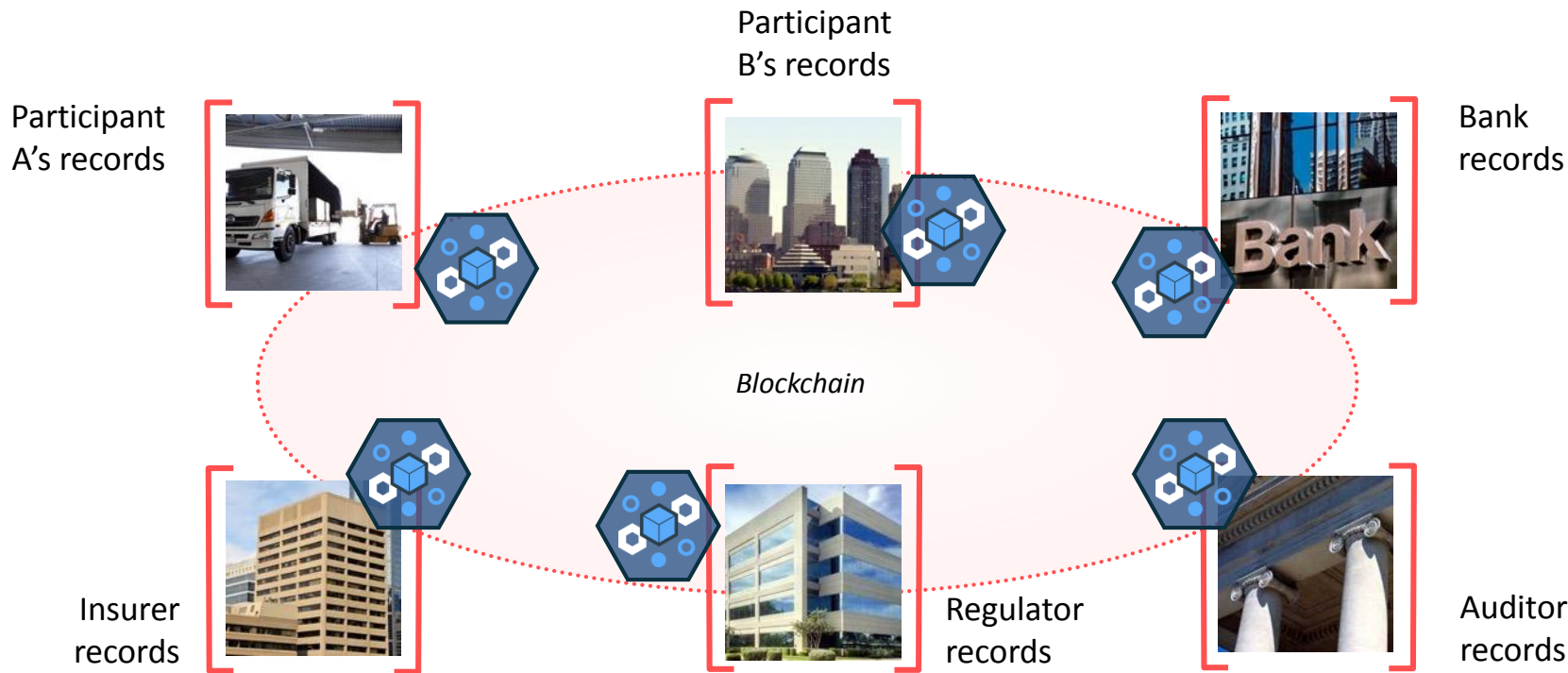


Problem ...



... inefficient, expensive, vulnerable

A shared replicated, permissioned ledger...



... with consensus, provenance, immutability and finality

Requirements of blockchain for business

Append-only distributed system of record shared across business network

Shared ledger



Smart contract



Business terms embedded in transaction database & executed with transactions

Ensuring appropriate visibility; transactions are secure, authenticated & verifiable

Privacy



Trust



Transactions are endorsed by relevant participants

How IBM can help



Technology



HYPERLEDGER

BLOCKCHAIN TECHNOLOGIES FOR BUSINESS

Hyperledger
Fabric

Hyperledger
Composer



Hosting and Support



High Security Business
Network



IBM Bluemix



docker



**Making blockchain real
for clients**



Garages



Engagement

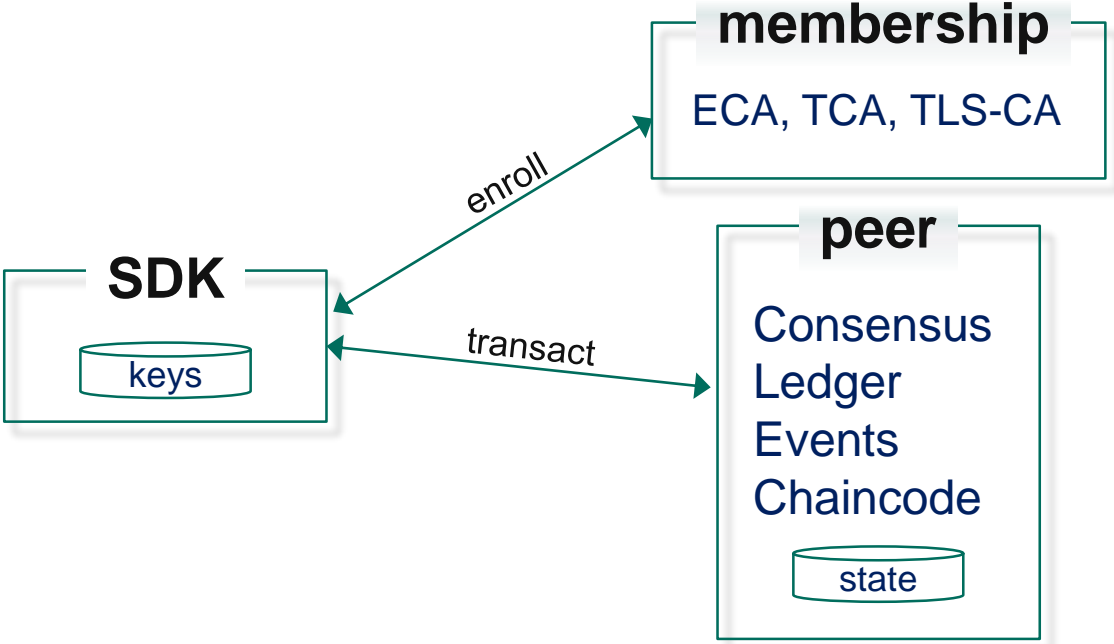
Hyperledger Members



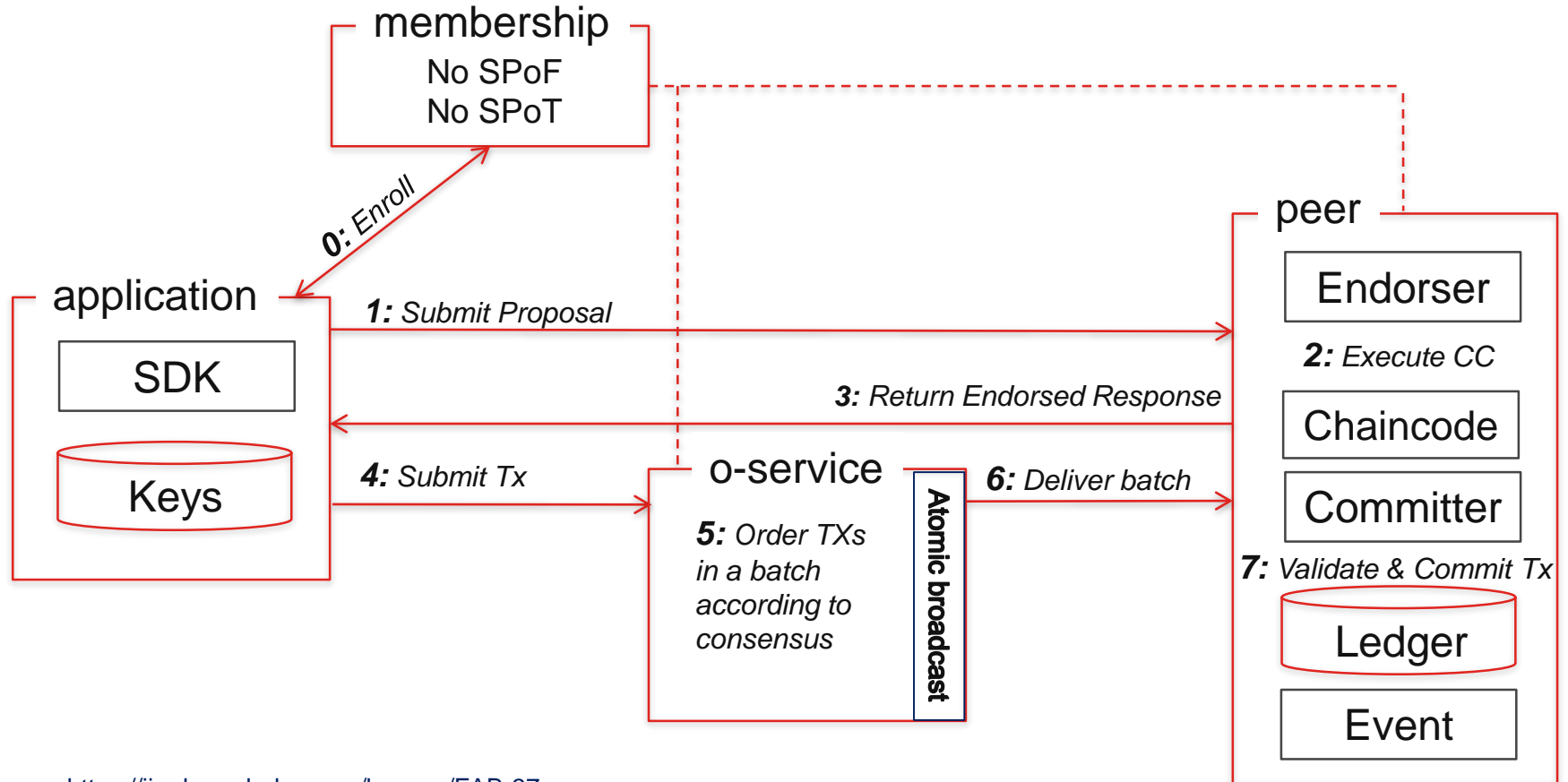
Premier

General




Architecture of Hyperledger Fabric v0.6



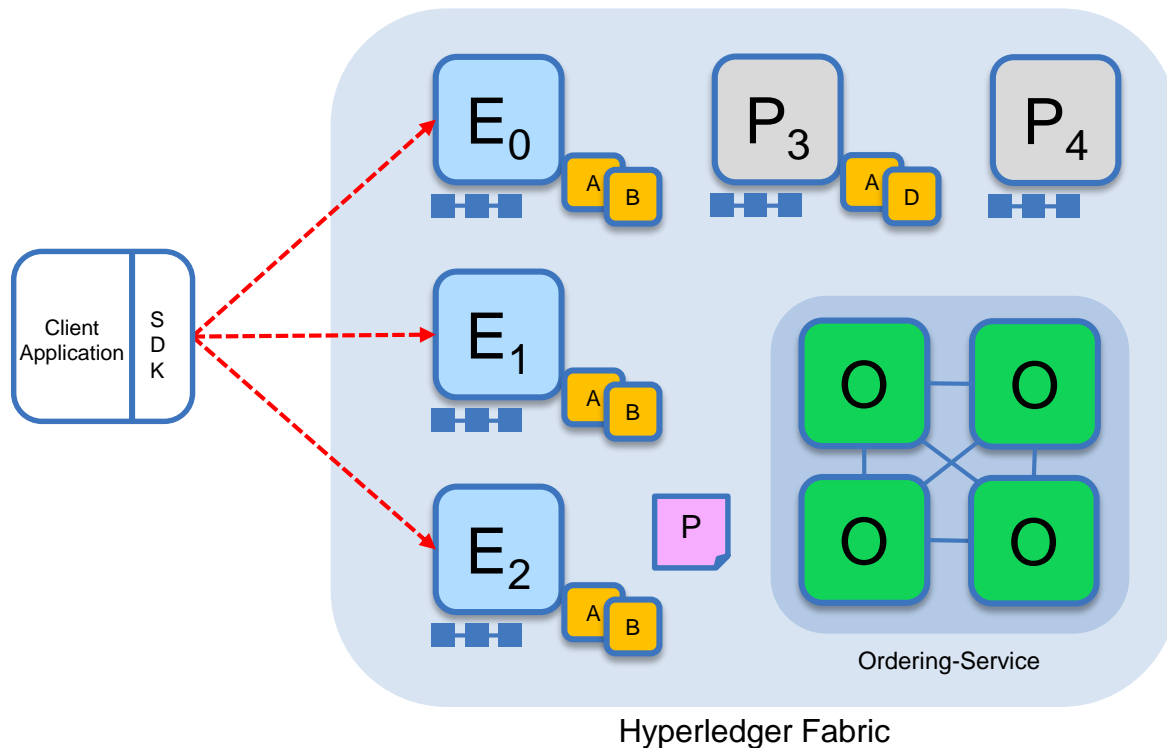
Architecture of Hyperledger Fabric v1



Nodes and roles

	<p>Committing Peer: Maintains ledger and state. Commits transactions. May hold smart contract (chaincode).</p>
	<p>Endorsing Peer: Specialized committing peer that receives a transaction proposal for endorsement, responds granting or denying endorsement. Must hold smart contract</p>
	<p>Ordering Nodes (service): Approves the inclusion of transaction blocks into the ledger and communicates with committing and endorsing peer nodes. Does not hold smart contract. Does not hold ledger.</p>

Sample transaction: Step 1/7 – Propose transaction



Application proposes transaction

Endorsement policy:

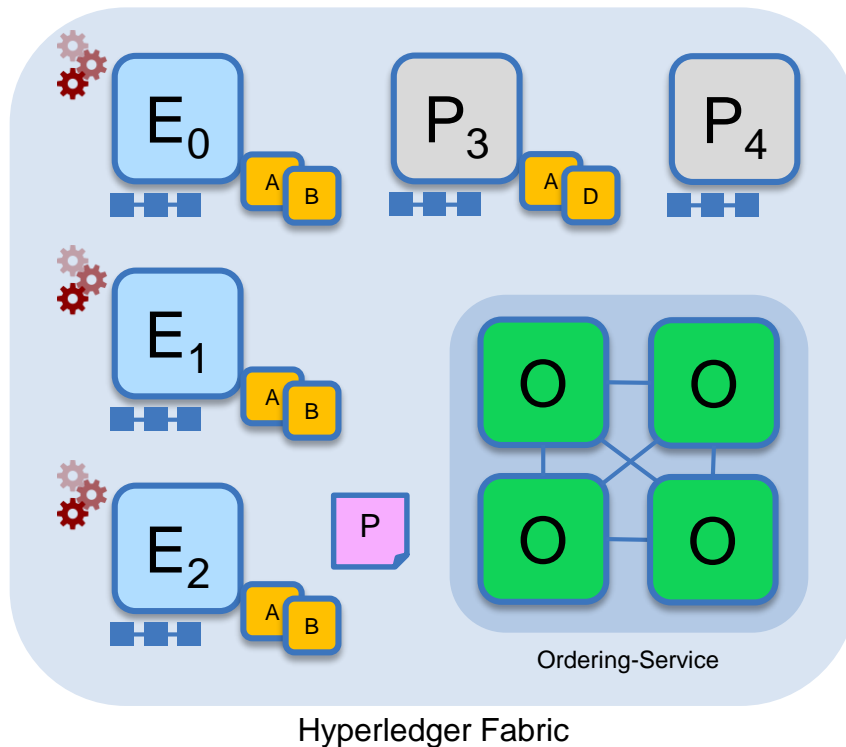
- “E₀, E₁ and E₂ must sign”
- (P₃, P₄ are not part of the policy)

Client application submits a transaction proposal for **Smart Contract A**. It must target the required peers {E₀, E₁, E₂}

Key:

Endorser			Ledger
Committing Peer			Application
Ordering Node			
Smart Contract (Chaincode)			Endorsement Policy

Sample transaction: Step 2/7 – Execute proposal



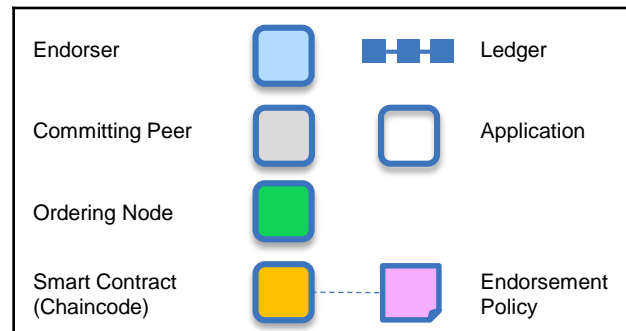
Endorsers Execute Proposals

E₀, E₁ & E₂ will each execute the *proposed* transaction. None of these executions will update the ledger

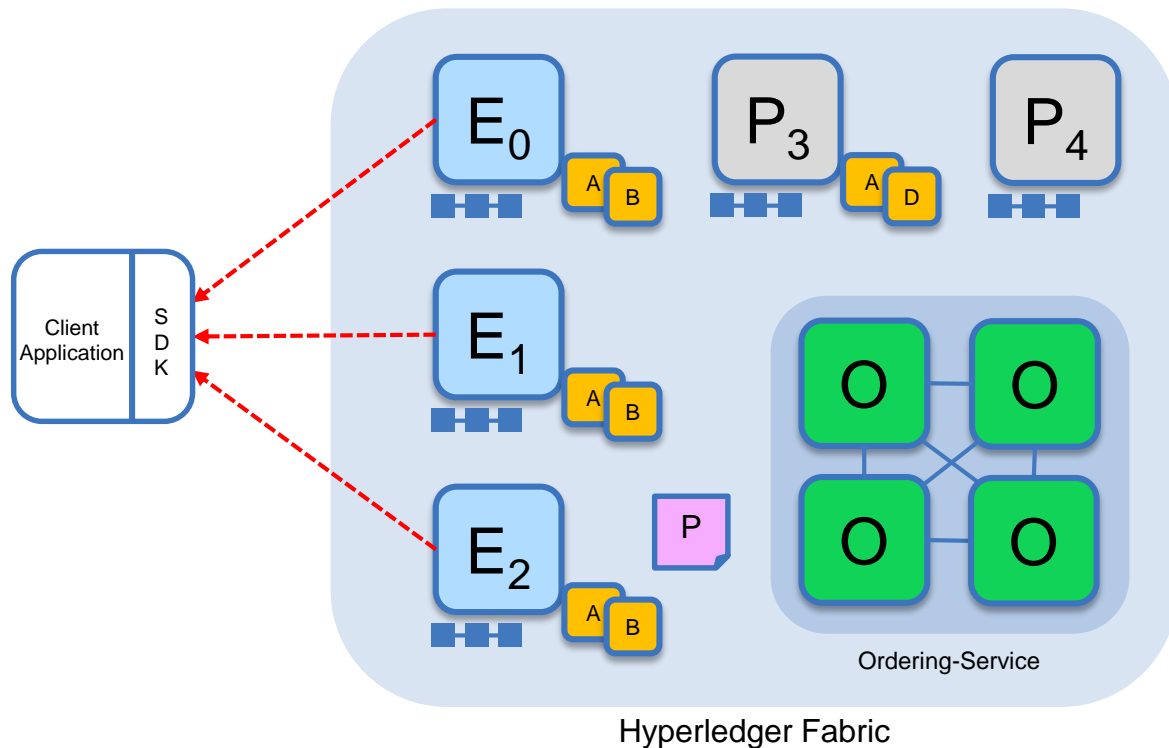
Each execution will capture the set of **Read** and **Written** data, called **RW sets**, which will now flow in the fabric.

Transactions can be signed & encrypted

Key:



Sample transaction: Step 3/7 – Proposal Response



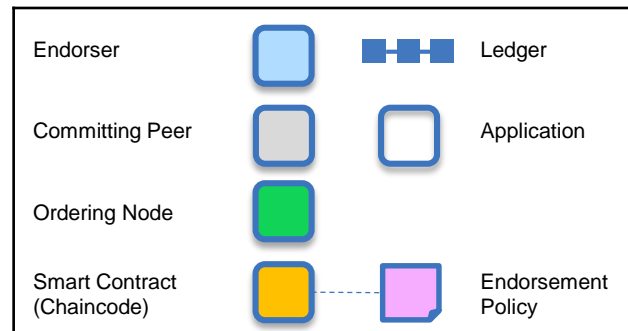
Application receives responses

RW sets are asynchronously returned to application

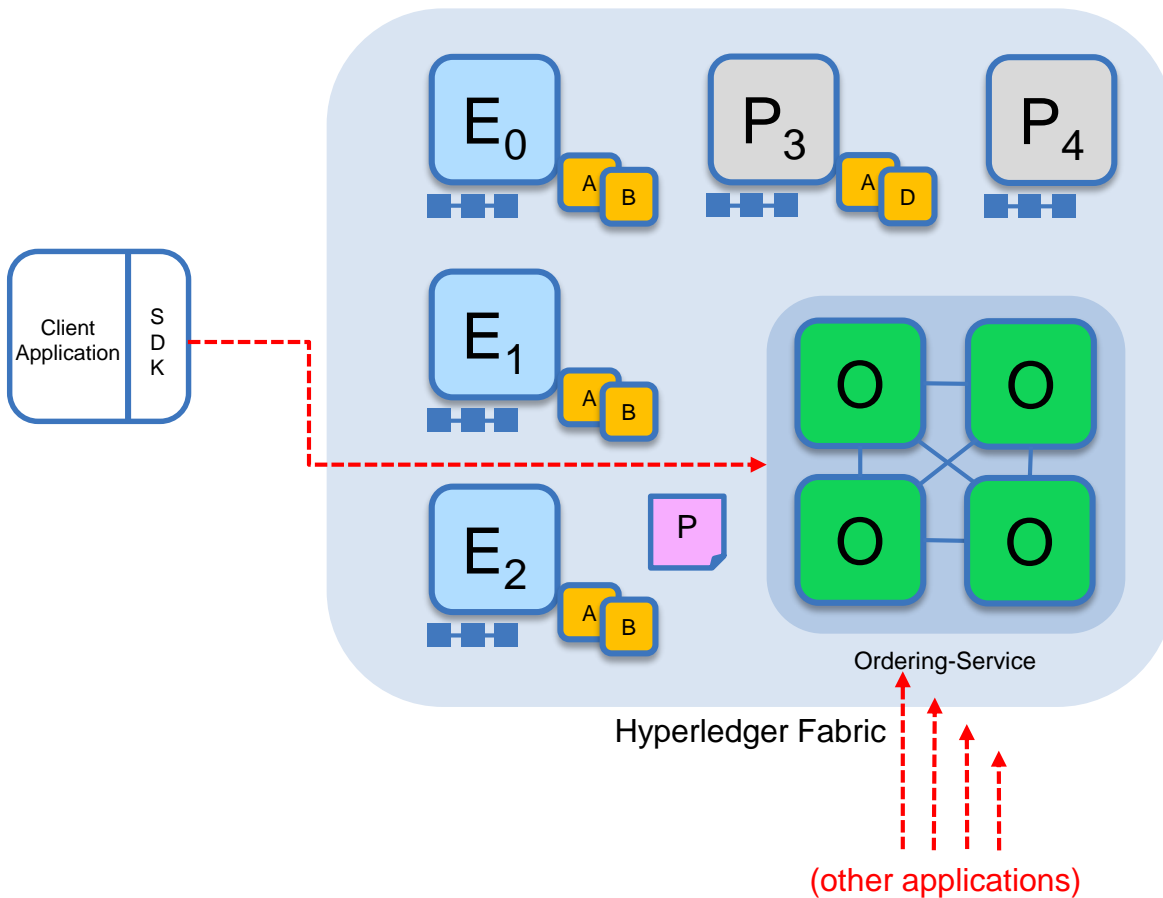
The RW sets are signed by each endorser, and also includes each record version number

(This information will be checked much later in the consensus process)

Key:



Sample transaction: Step 4/7 – Order Transaction



Application submits responses for ordering

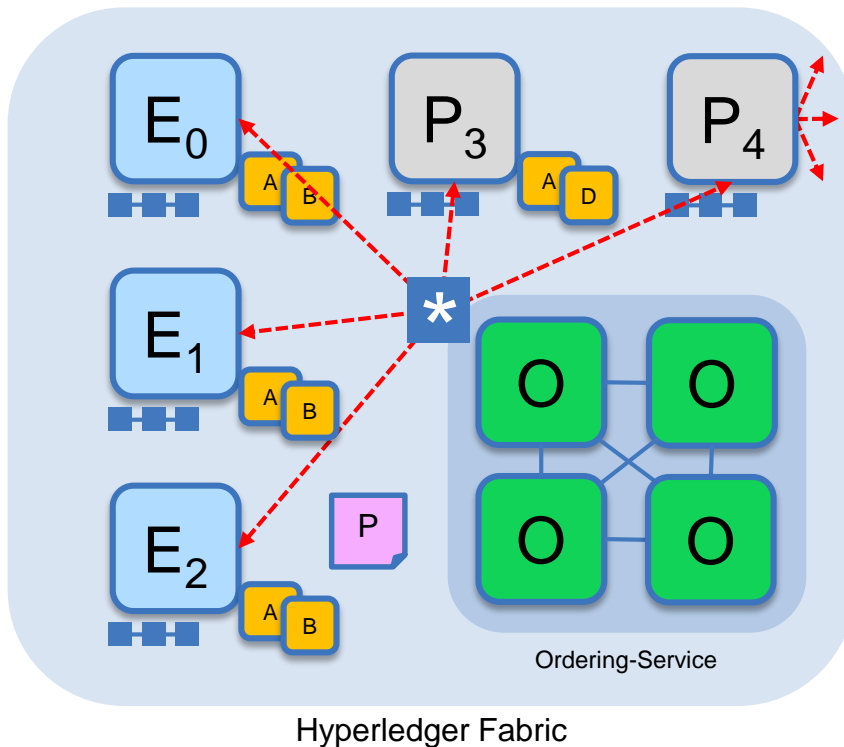
Application submits responses as a **transaction** to be ordered.

Ordering happens across the fabric in parallel with transactions submitted by other applications

Key:

Endorser			Ledger
Committing Peer			Application
Ordering Node			
Smart Contract (Chaincode)			Endorsement Policy

Sample transaction: Step 5/7 – Deliver Transaction



Orderer delivers to all committing peers

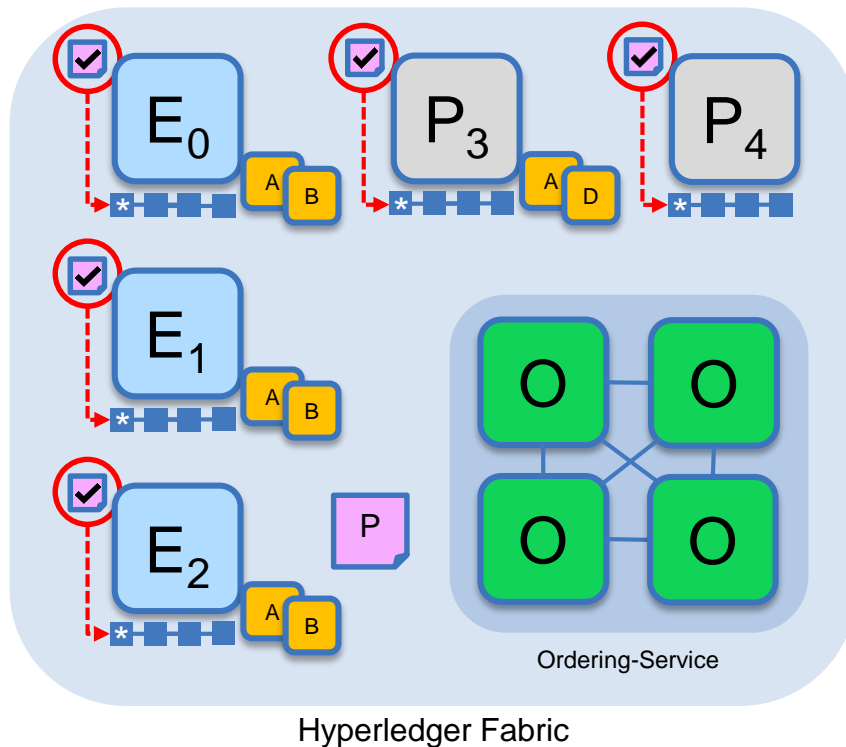
Ordering service collects transactions into proposed blocks for distribution to committing peers. Peers can deliver to other peers in a hierarchy (not shown)

- Different ordering algorithms available:
- SOLO (Single node, development)
 - Kafka (Crash fault tolerance)

Key:

Endorser			Ledger
Committing Peer			Application
Ordering Node			
Smart Contract (Chaincode)			Endorsement Policy

Sample transaction: Step 6/7 – Validate Transaction



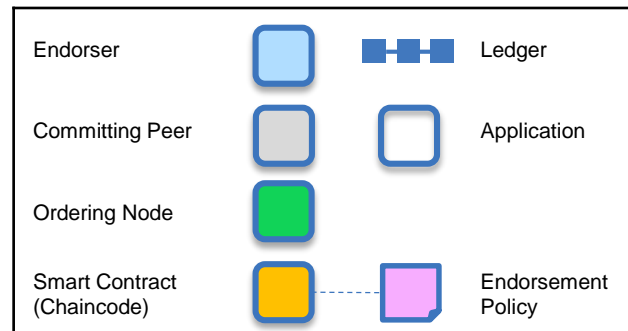
Committing peers validate transactions

Every committing peer validates against the endorsement policy. Also check RW sets are still valid for current world state

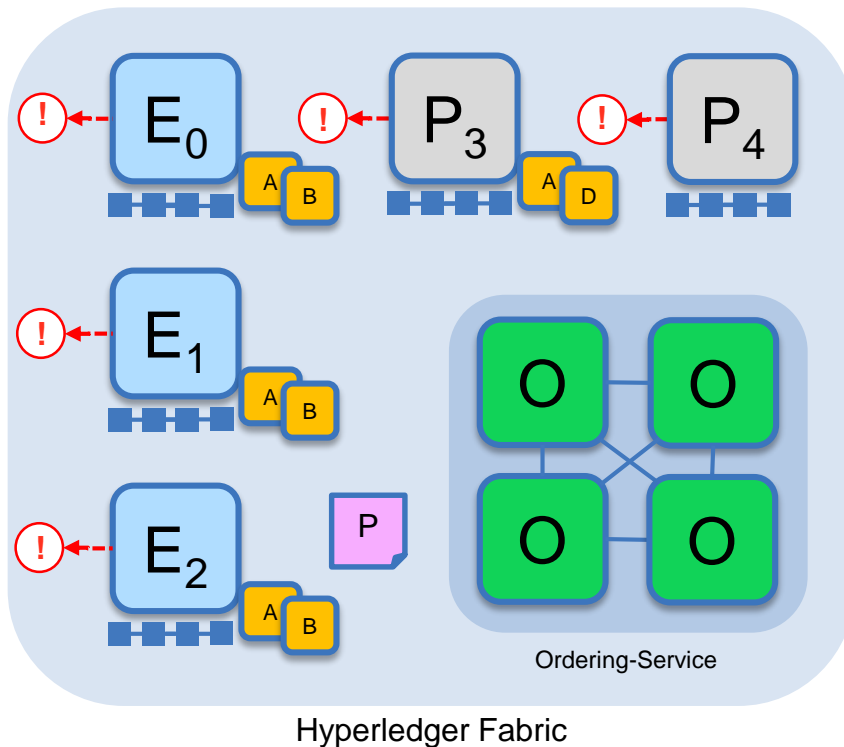
Validated transactions are applied to the world state and retained on the ledger

Invalid transactions are also retained on the ledger but do not update world state

Key:



Sample transaction: Step 7/7 – Notify Transaction



Committing peers notify applications

Applications can register to be notified when transactions succeed or fail, and when blocks are added to the ledger

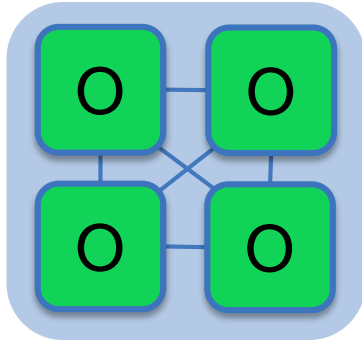
Applications will be notified by each peer to which they are connected

Key:

Endorser			Ledger
Committing Peer			Application
Ordering Node			
Smart Contract (Chain code)			Endorsement Policy

Ordering Service

The ordering service packages transactions into blocks to be delivered to peers. Communication with the service is via channels.



Ordering-Service

Different configuration options for the ordering service include:

– **SOLO**

- Single node for development

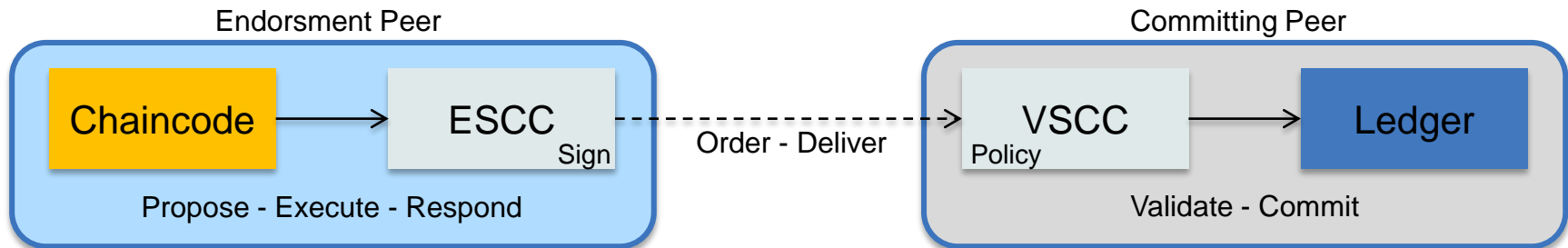
– **Kafka** : Crash fault tolerant consensus

- 3 nodes minimum
- Odd number of nodes recommended

Endorsement Policies

An endorsement policy describes the conditions by which a transaction can be endorsed. A transaction can only be considered valid if it has been endorsed according to its policy.

- Each chaincode is associated with an Endorsement Policy
- Default implementation: Simple declarative language for the policy
- ESCC (Endorsement System ChainCode) signs the proposal response on the endorsing peer
- VSCC (Validation System ChainCode) validates the endorsements



Endorsement Policy Examples

Examples of policies:

- Request 1 signature from all three principals
 - AND('Org1.member', 'Org2.member', 'Org3.member')
- Request 1 signature from either one of the two principals
 - OR('Org1.member', 'Org2.member')
- Request either one signature from a member of the Org1 MSP or (1 signature from a member of the Org2 MSP and 1 signature from a member of the Org3 MSP)
 - OR('Org1.member', AND('Org2.member', 'Org3.member'))

WATSON
BRASIL
DEVELOPER
SUMMIT
2017



- Car Lease Sample

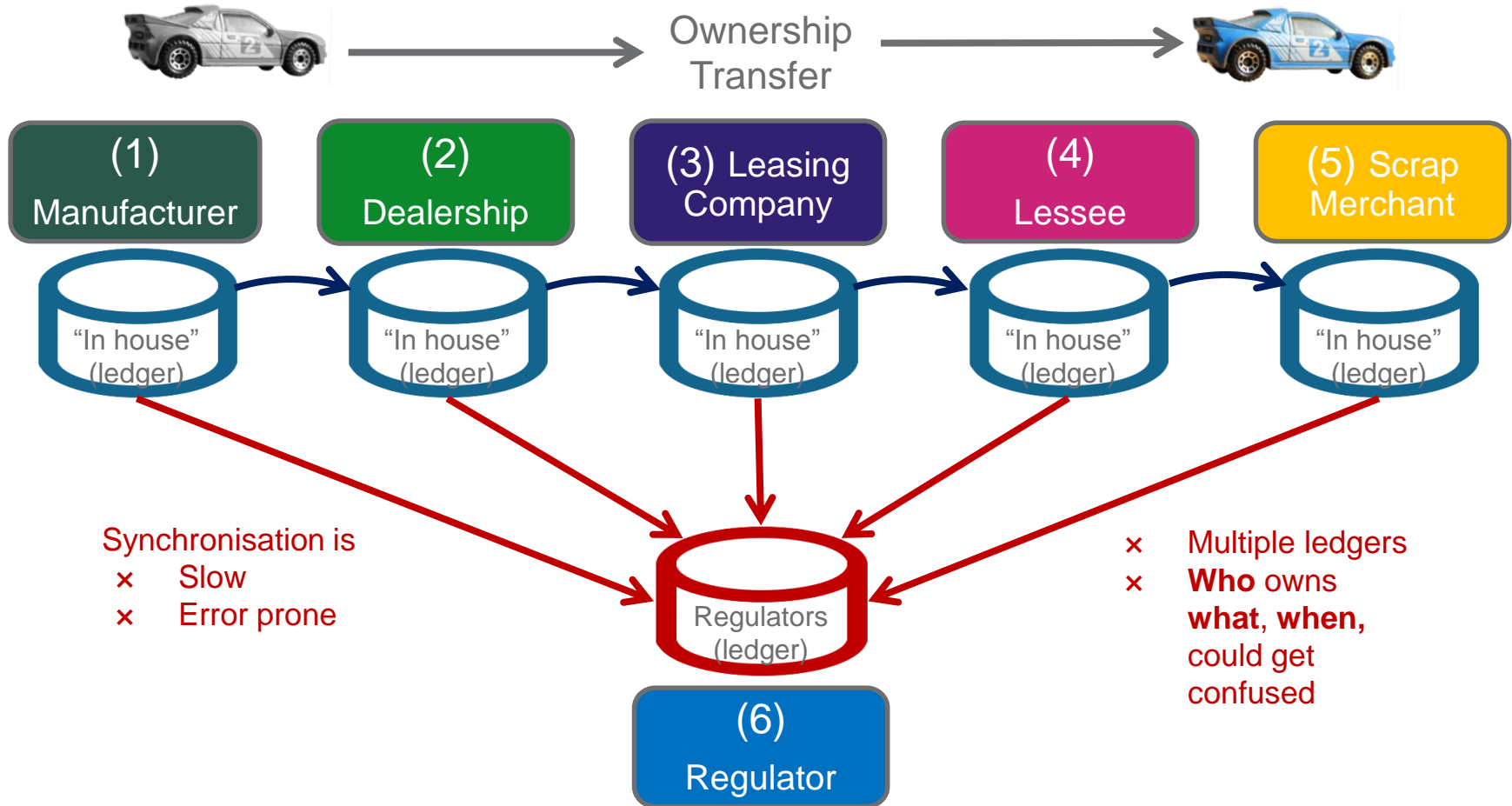




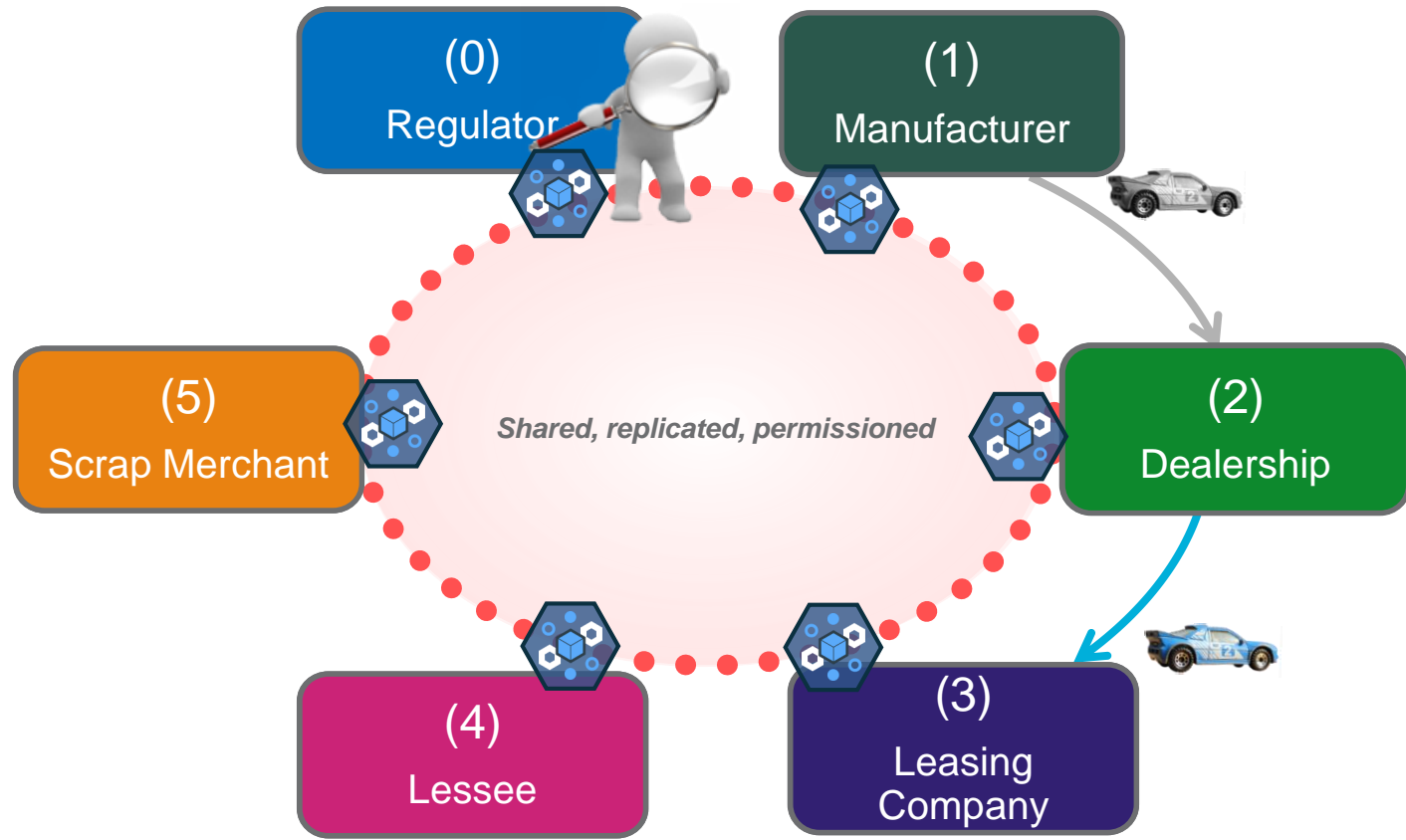
In this Blockchain Asset Transfer Demo we will be transferring cars
(...but it could be anything)



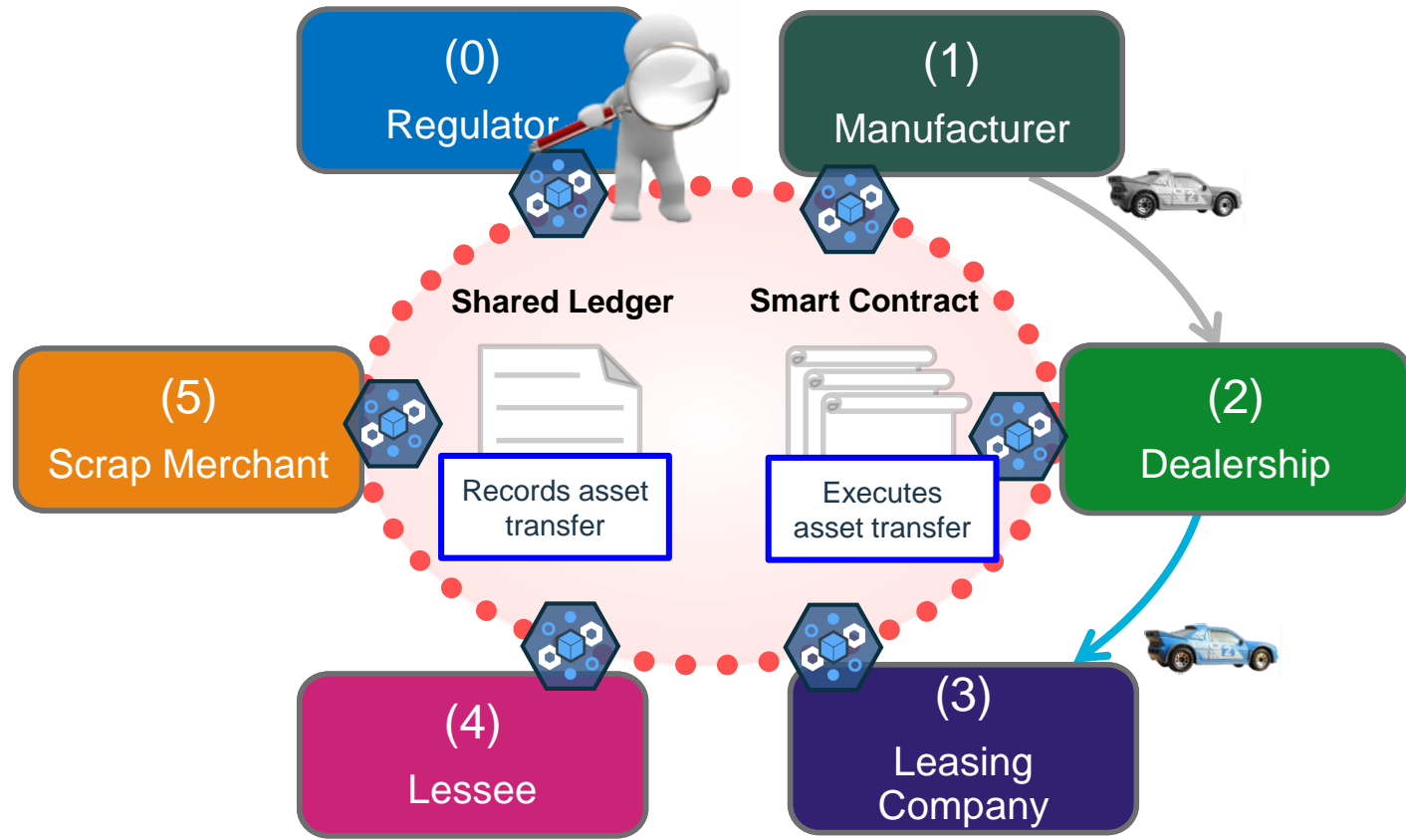
How do participants typically work today?



How could participants work with Blockchain 1/2?



How could participants work with Blockchain 2/2?



WATSON
BRASIL
DEVELOPER
SUMMIT
2017

Obrigado!

[Carlos L Rischioto] - [carlosr@br.ibm.com]

[Marcos Tadeu Brisola Vieira] - [marcos.vieira@br.ibm.com]

[Percival Silva de Lucena] - [plucena@br.ibm.com]

