Hyperledger Fabric: a Distributed Operating System for Permissioned Blockchains. (2018)

Authors: Elli Androulaki, Artem Barger, et al.

Presented by Chi Chen

Why Fabric?

Permissioned blockchains require **every peer** to **execute every transaction**, **maintain a ledger** and **run consensus**.

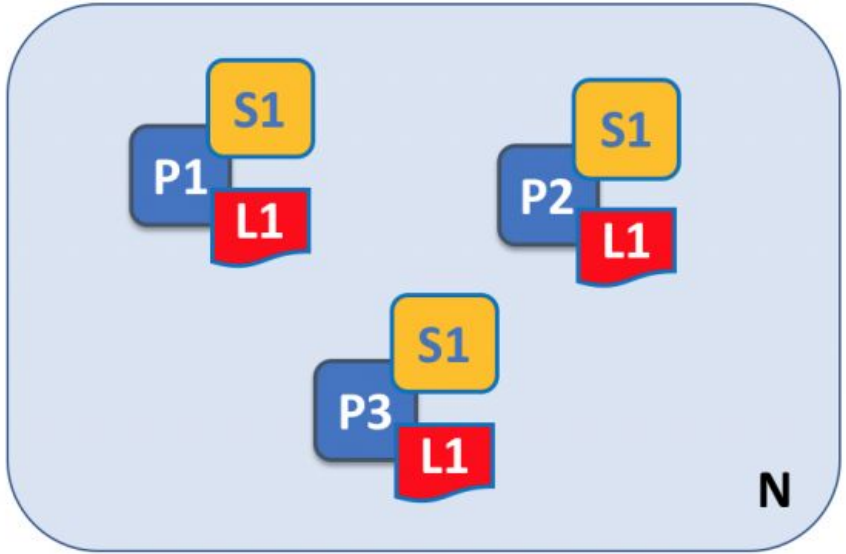
Can’t support **private** transactions and **confidential** contracts.

- **Modular**
- **Configurable**
- **Pluggable Consensus Protocols**

Components

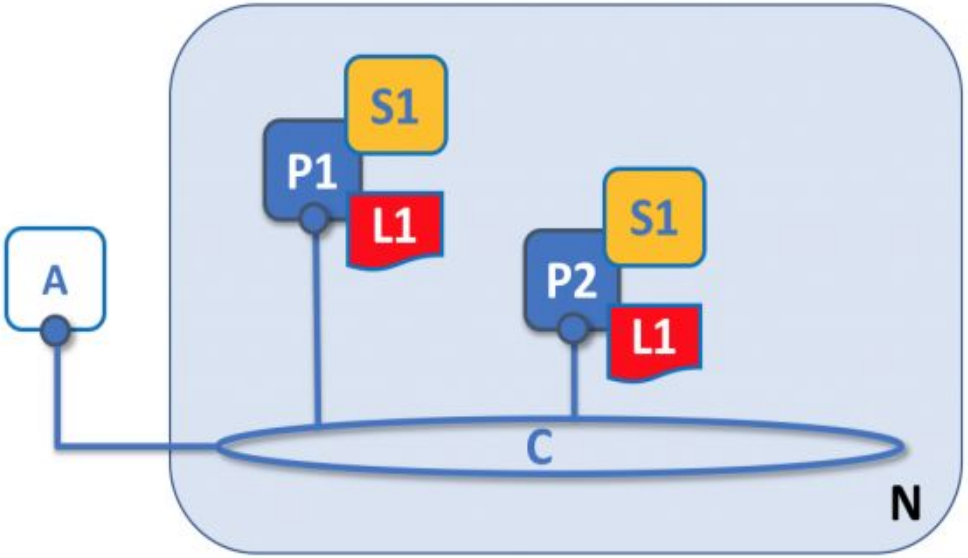
- **Shared ledger**: world state and blockchain
- **Smart contract**: chaincode, contains the business logic of the system
- **Client node**: a client application
- **Peer nodes**: host ledgers and smart contract, can be endorser
- **Orderer nodes**: default implementation is based on Apache Kafka
- **Channel**: a logical structure formed by a collection of peers
- **Membership Services Provider (MSP)**: certificate authority

Peer nodes



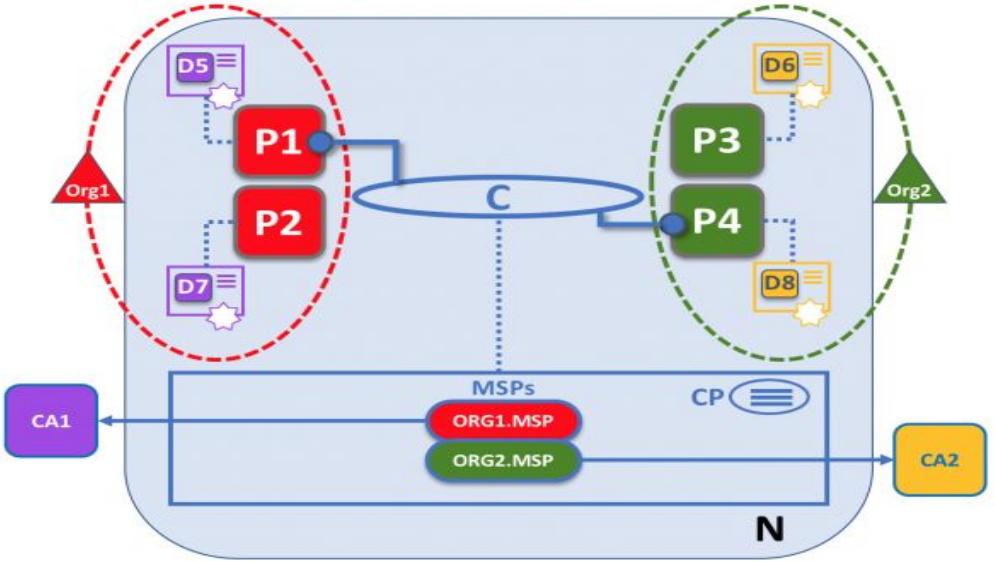
N	Blockchain network
P	Peer node
S	Smart contract (aka chaincode)
L	Ledger

Channel



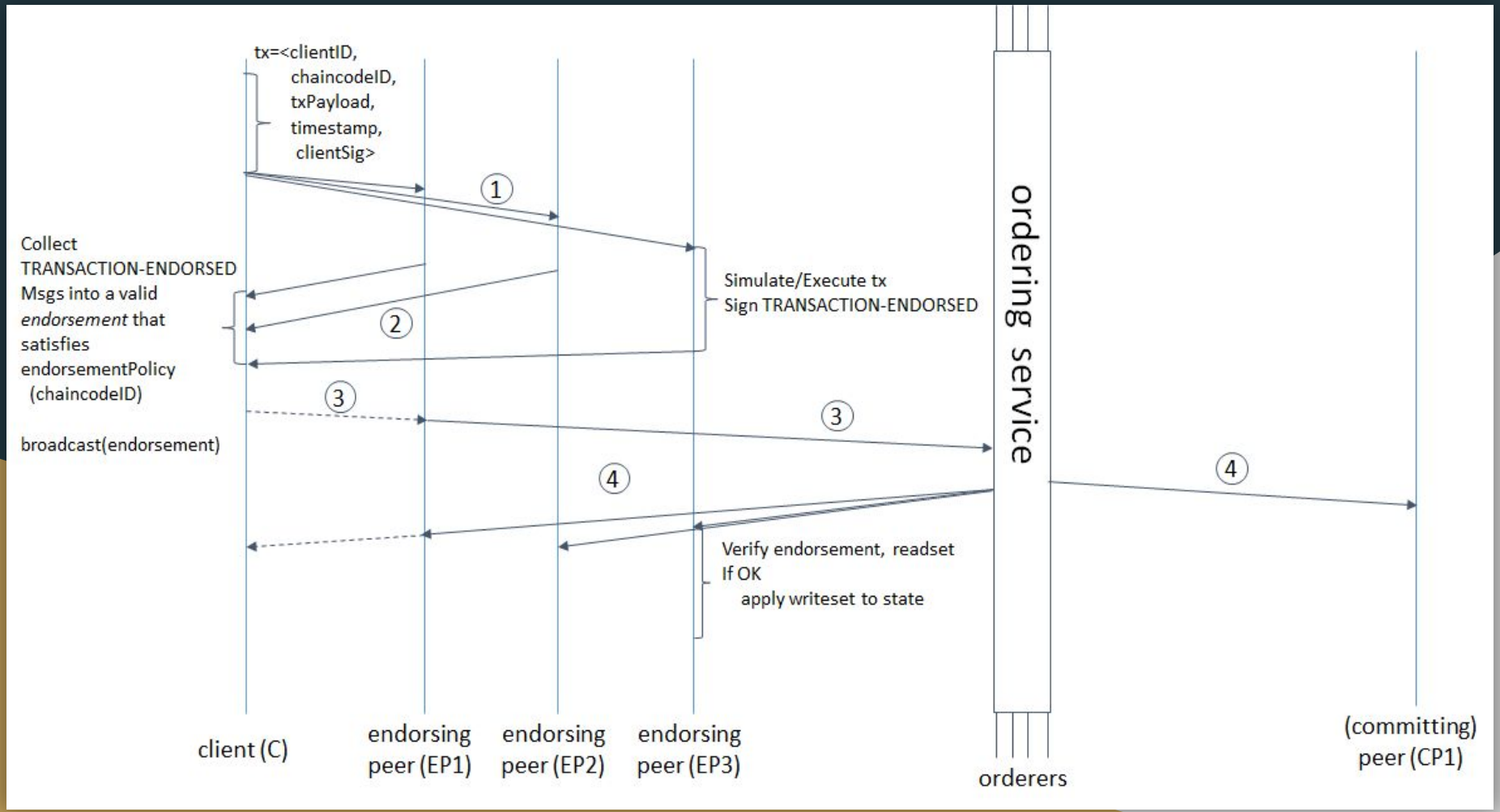
N	Blockchain Network	L	Ledger
C	Channel	A	Application
P	Peer	PA C	Principal PA (e.g. A, P1) communicates via channel C.
S	Chaincode		

MSP

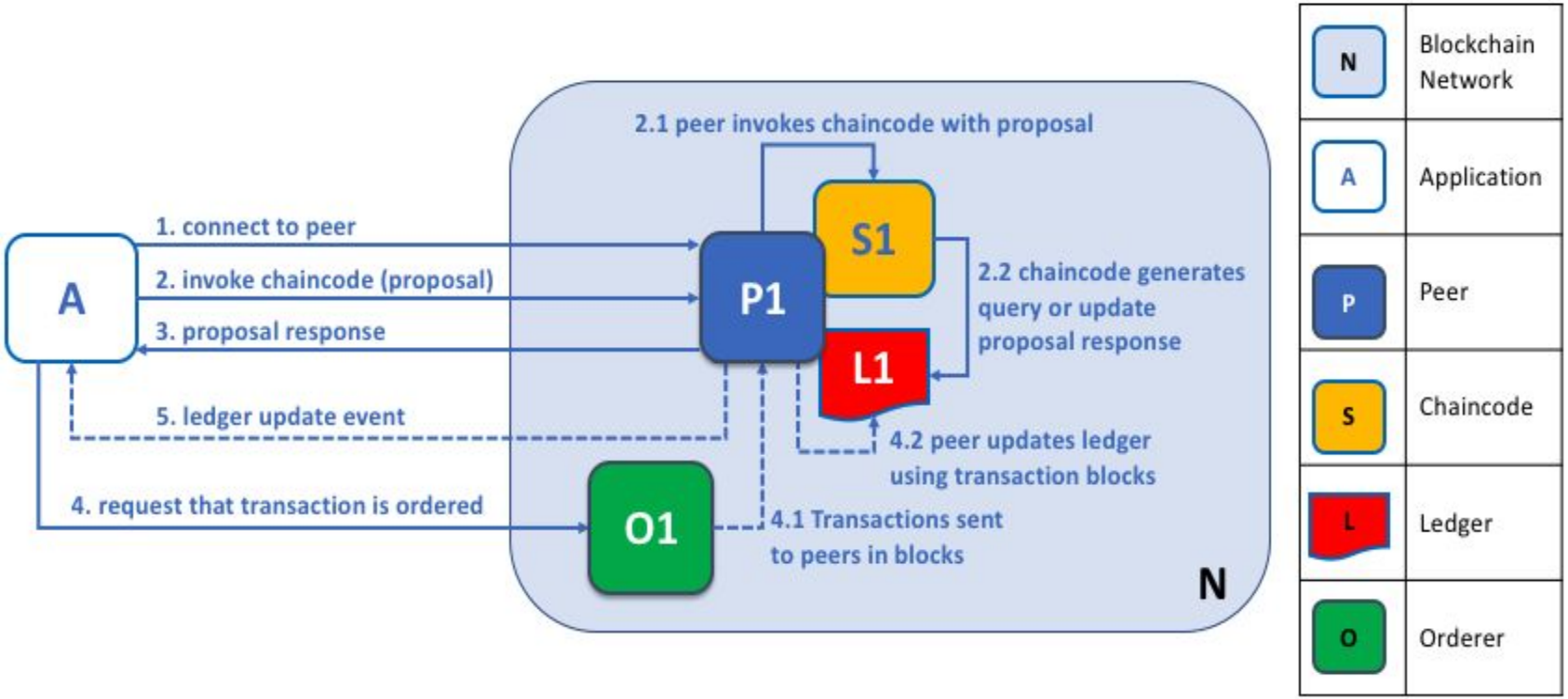


N	Blockchain Network	P	Peer
C	Channel	Org	Organization
I	Identity	PA / C	Principal PA (e.g. P1,P4) communicates via channel C.
CP	Channel policy		
CA	Certificate Authority	MSP	Membership Service Provider
A1 / R / P1 / P2	Organization R owns application A1 and peers P1, P2.		
C / CP	Channel C subject to policy CP.	MSP1 / MSP2 / CP	Channel policy CP contains MSPs: MSP1 and MSP2.
MSP1 → CA1	MSP1 selects the Certificate Authority CA1 to provide certificates for it.		

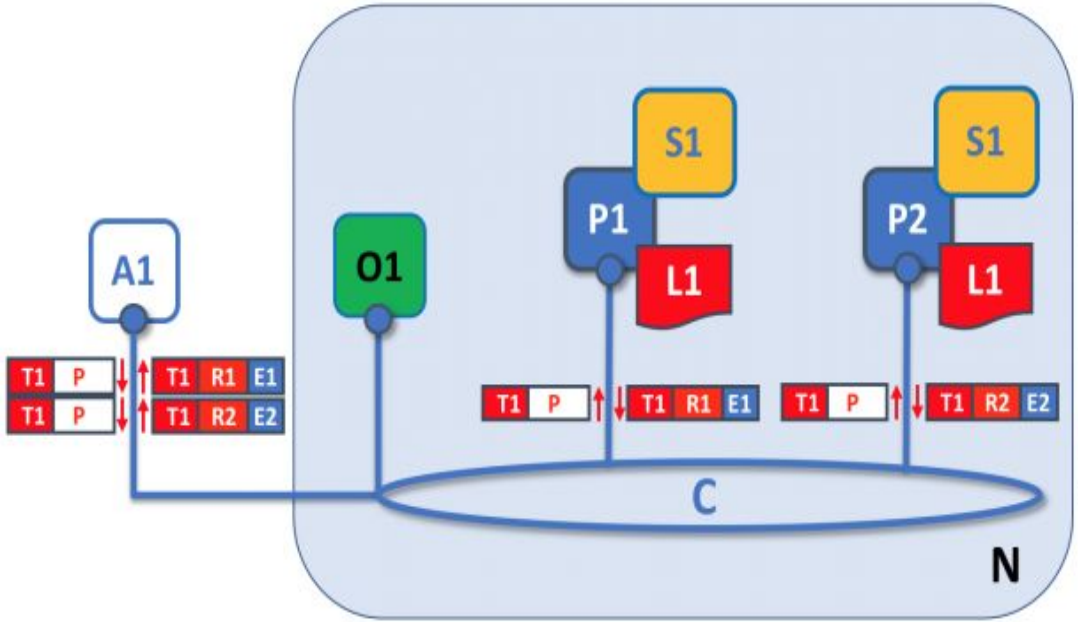
PA / D	Principal P has identity D


tx=<clientID,
chaincodeID,
txPayload,
timestamp,
clientSig>
1
Collect
TRANSACTION-ENDORSED
Msgs into a valid
endorsement that
satisfies
endorsementPolicy
(chaincodeID)
2
Simulate/Execute tx
Sign TRANSACTION-ENDORSED
3
broadcast(endorsement)
3
4
4
Verify endorsement, readset
If OK
apply writeset to state
ordering service
client (C)
endorsing
peer (EP1)
endorsing
peer (EP2)
endorsing
peer (EP3)
orderers
(committing)
peer (CP1)

Steps

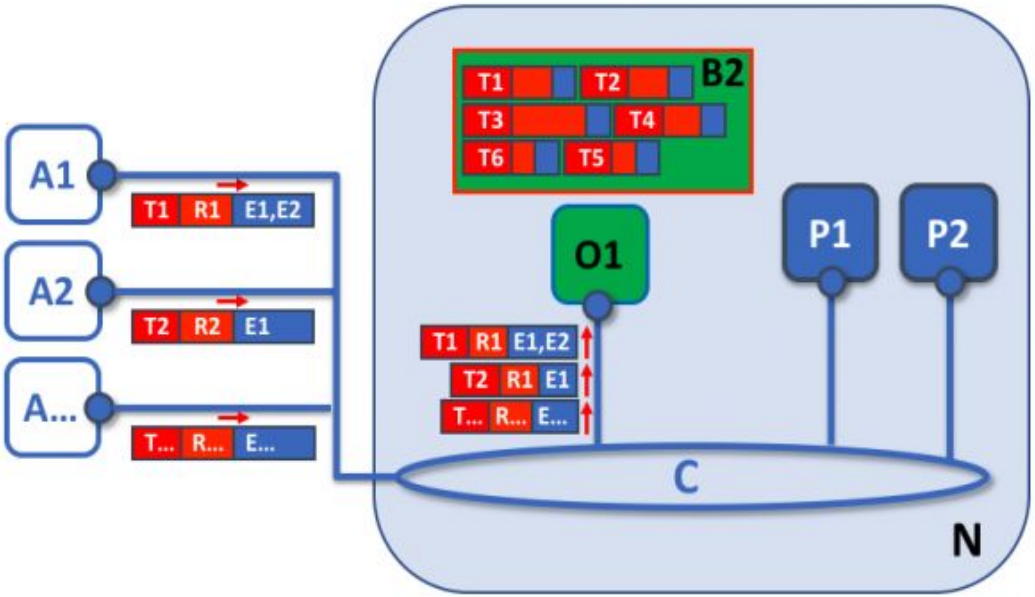

2.1 peer invokes chaincode with proposal
1. connect to peer
2. invoke chaincode (proposal)
3. proposal response
A
P1
S1
L1
2.2 chaincode generates query or update proposal response
5. ledger update event
4. request that transaction is ordered
O1
4.1 Transactions sent to peers in blocks
4.2 peer updates ledger using transaction blocks
N
N
Blockchain Network
A
Application
P
Peer
S
Chaincode
L
Ledger
O
Orderer

Execution Phase



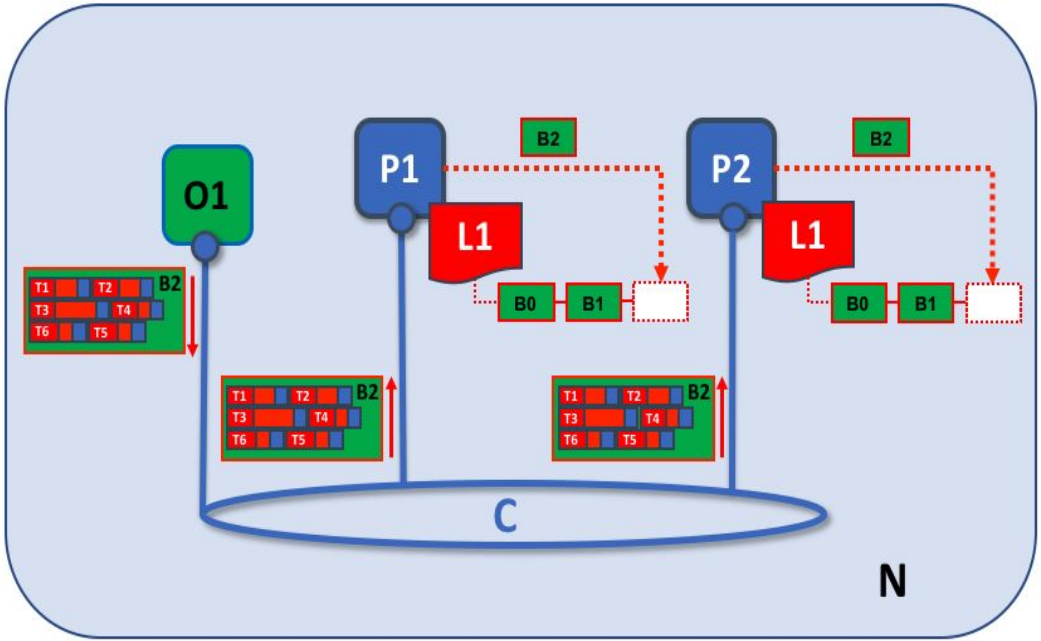
N	Blockchain Network	S	Chaincode
C	Channel	O	Orderer
P	Peer	L	Ledger
T1 P	Transaction T proposal P	T1 R2 E2	Transaction T1, response R2 endorsed with E2
T T C	Ledger transaction T1 flows on channel C	PA C	Principal PA (P1,P2) communicates via channel C.

Ordering Phase



N	Blockchain Network	P	Peer
B1	Block B1	O	Orderer
T1 R2a E2	Transaction T1, response R2a endorsed with E2	C	Channel
T1 T2 T3 B1	Block B1 contains transactions T1, T2, T3...		
T1 T1 C	Ledger transaction T1 flows on channel C	PA C	Principal PA (P1,P2) communicates via channel C.

Validation Phase



N	Blockchain Network	P	Peer
C	Channel	O	Orderer
L	Ledger	B	Block B
L1 B0 B1	Ledger L1 has blockchain with blocks B0, B1	T1 T2 T3 B1	Block B1 contains transactions T1, T2, T3...
B1 C	Block B1 flows on channel C	PA C	Principal PA (P1, P2) communicates via channel C.