SYNOPSYS®
Silicon to Software™

Tineola: Taking A Bite Out of Enterprise Blockchain

Attacking HyperLedger Fabric

Parsia Hakimian, Stark Riedesel

Defcon 26 – Aug 11, 2018

5 Courses

Our Team

Enterprise Blockchains

A Use Case

The Target – HyperLedger Fabric

Tineola

HyperLedger Fabric – Core Research Group

Parsia Hakimian
Senior Consultant

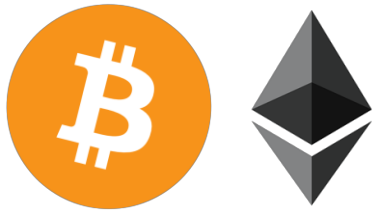
Stark Riedesel
Senior Consultant

Travis Biehn
Emerging Tech Lead

Koen Buyens
Principal Consultant

Enterprise Blockchain Terroir

Public Platforms



Enterprise Blockchain Enthusiasts

- Tech
- Auto & Aero
- Financial Services
- Accounting
- Healthcare
- Logistics
- Oil

Enterprise Platforms

Platform Desires Meet Reality

Promise

Immutability

Auditability

Tune-able Trust

Programmable

Challenge

Immutability

Mutability

Privacy

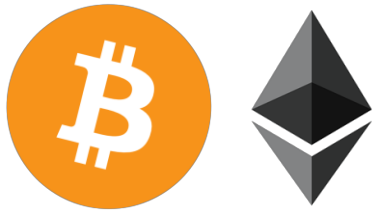
Correctness and Speed

Execution Environment

Platform Complexity

On The Chopping Block

Public Platforms



Enterprise Platforms

Enterprise Blockchain Enthusiasts

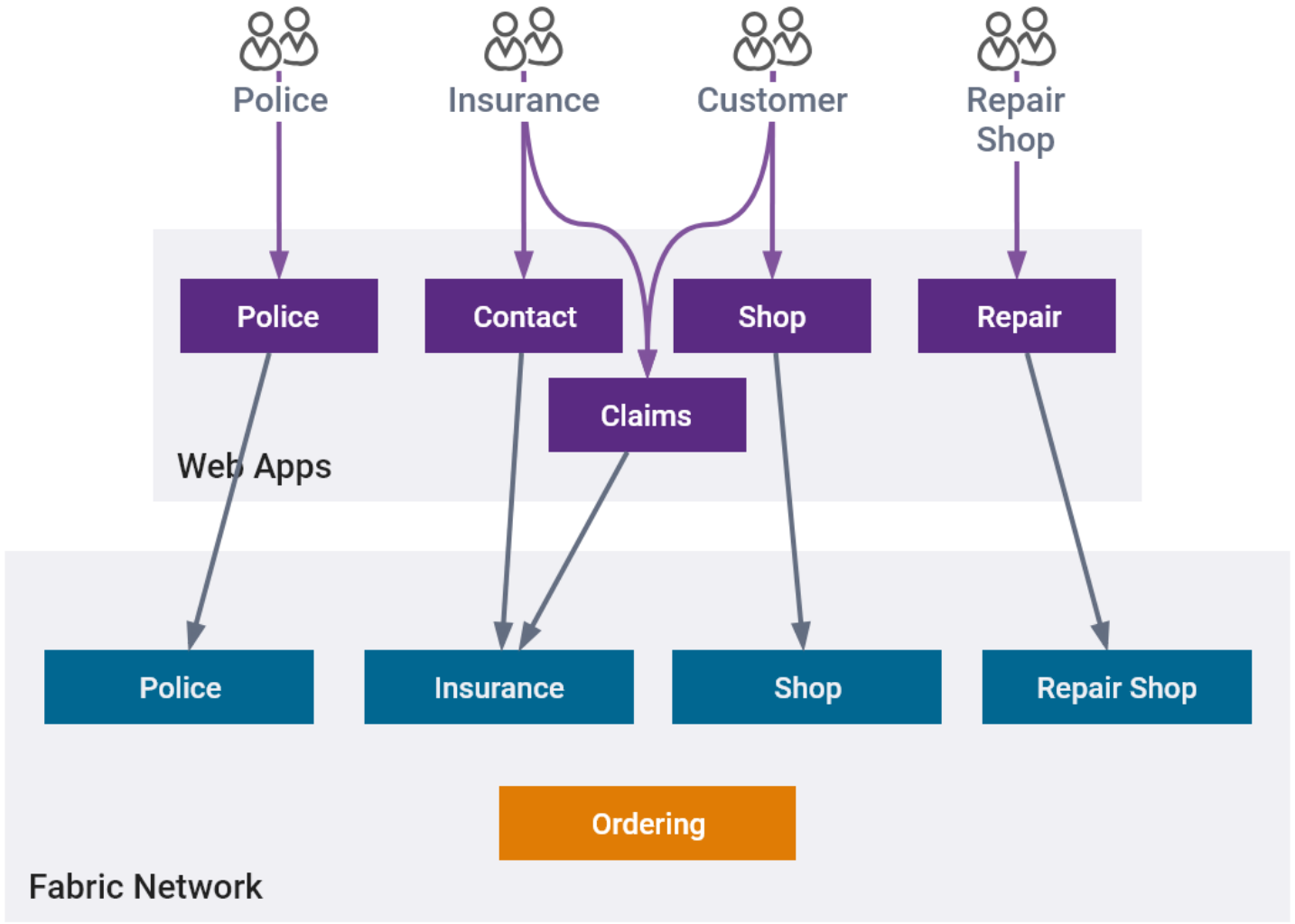
- Tech
- Auto & Aero
- Financial Services
- Accounting
- Healthcare
- Logistics
- Oil

SYNOPSYS®

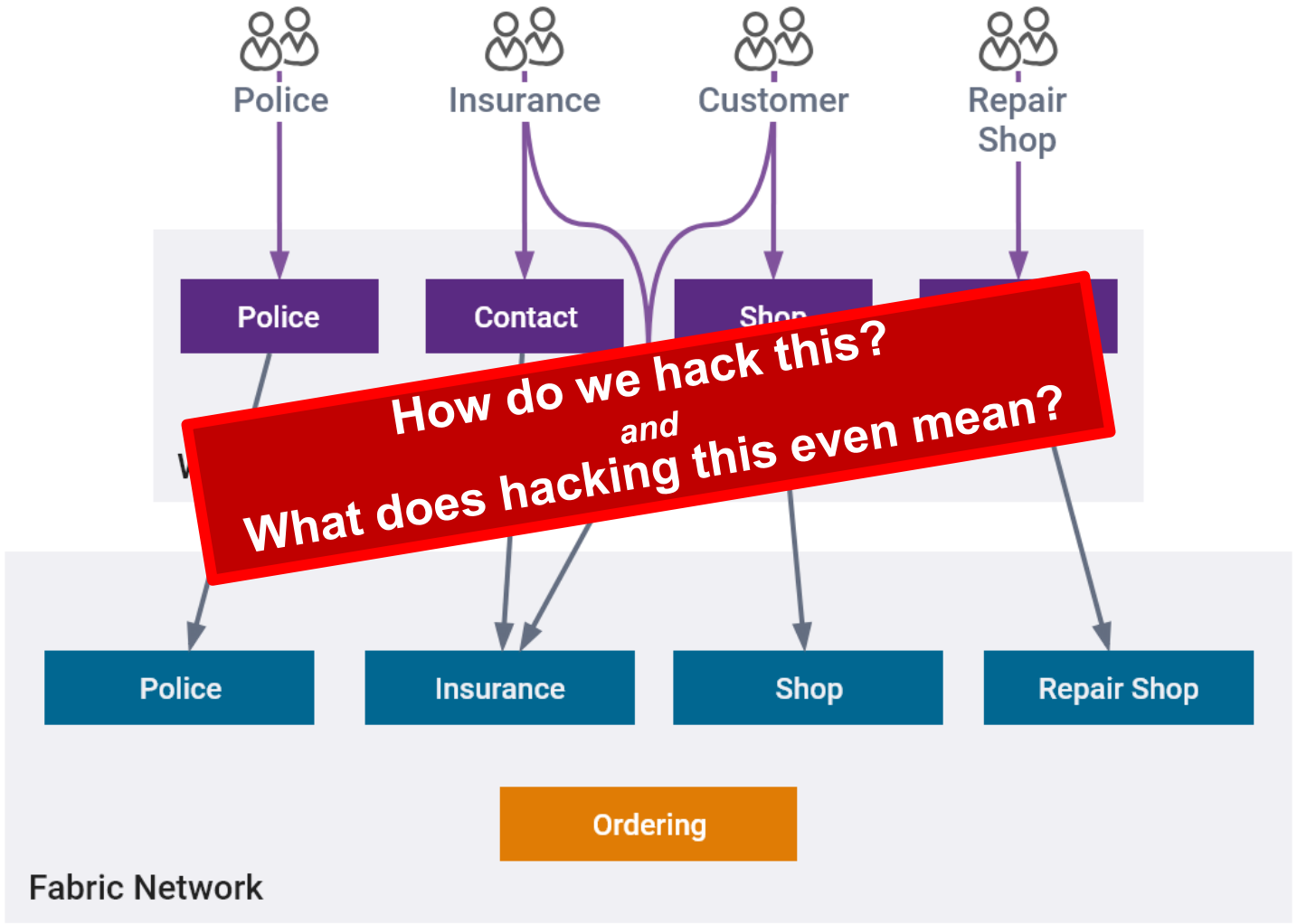
Build Blockchain Insurance App

Our Enterprise Application Strawman

Build Blockchain Insurance App



Build Blockchain Insurance App



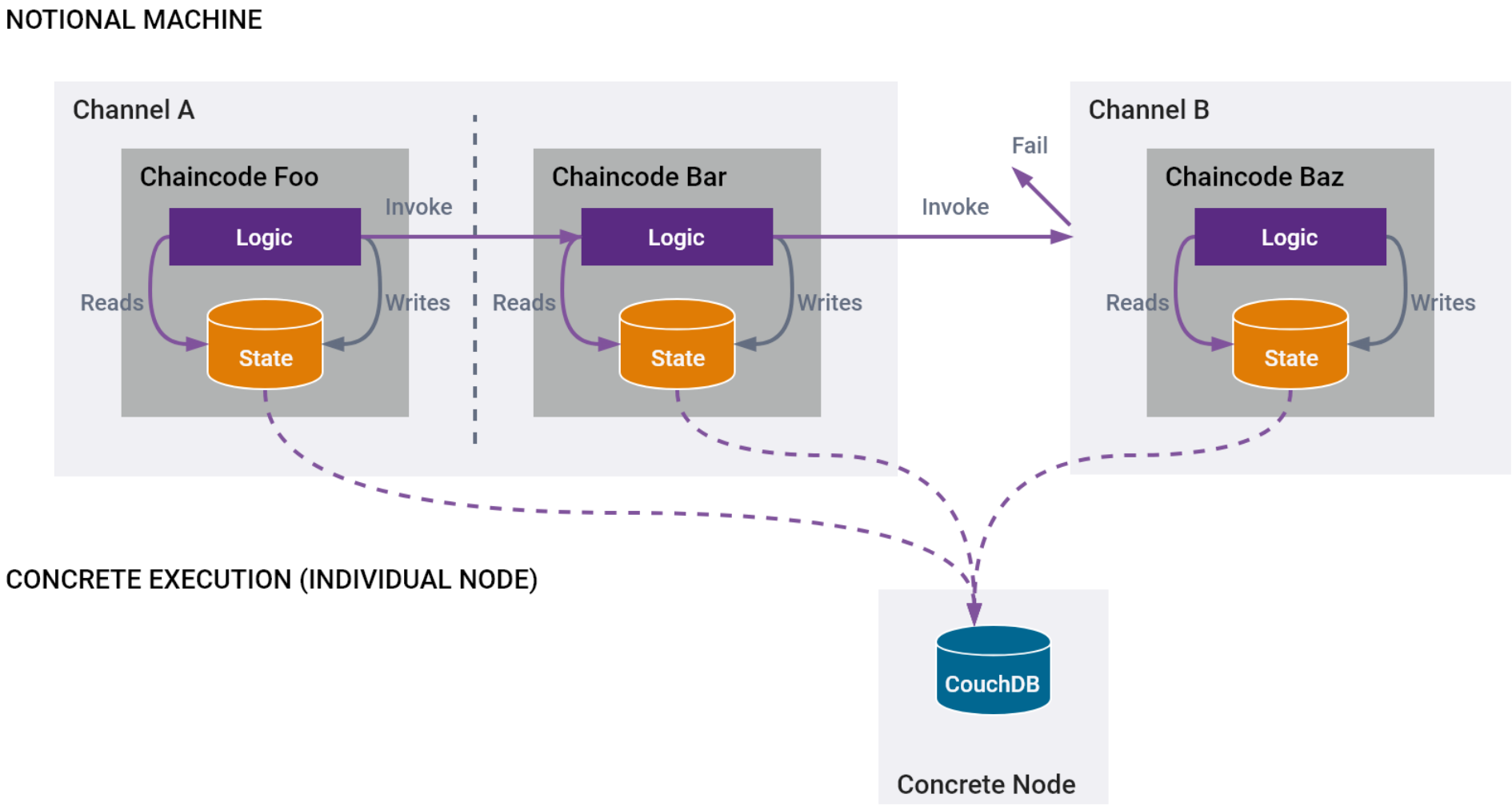
Meet HyperLedger Fabric

An Interesting New Machine

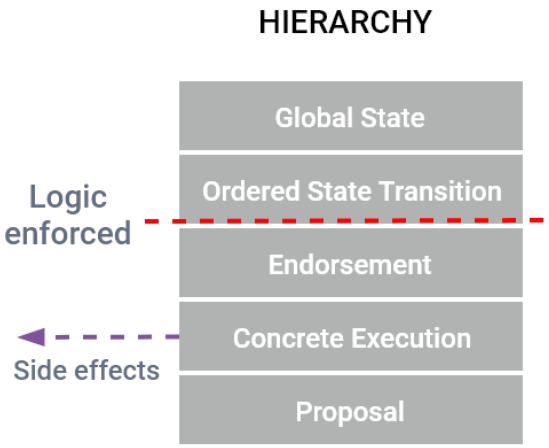
Chaincode: Fabrics' Smart Contracts

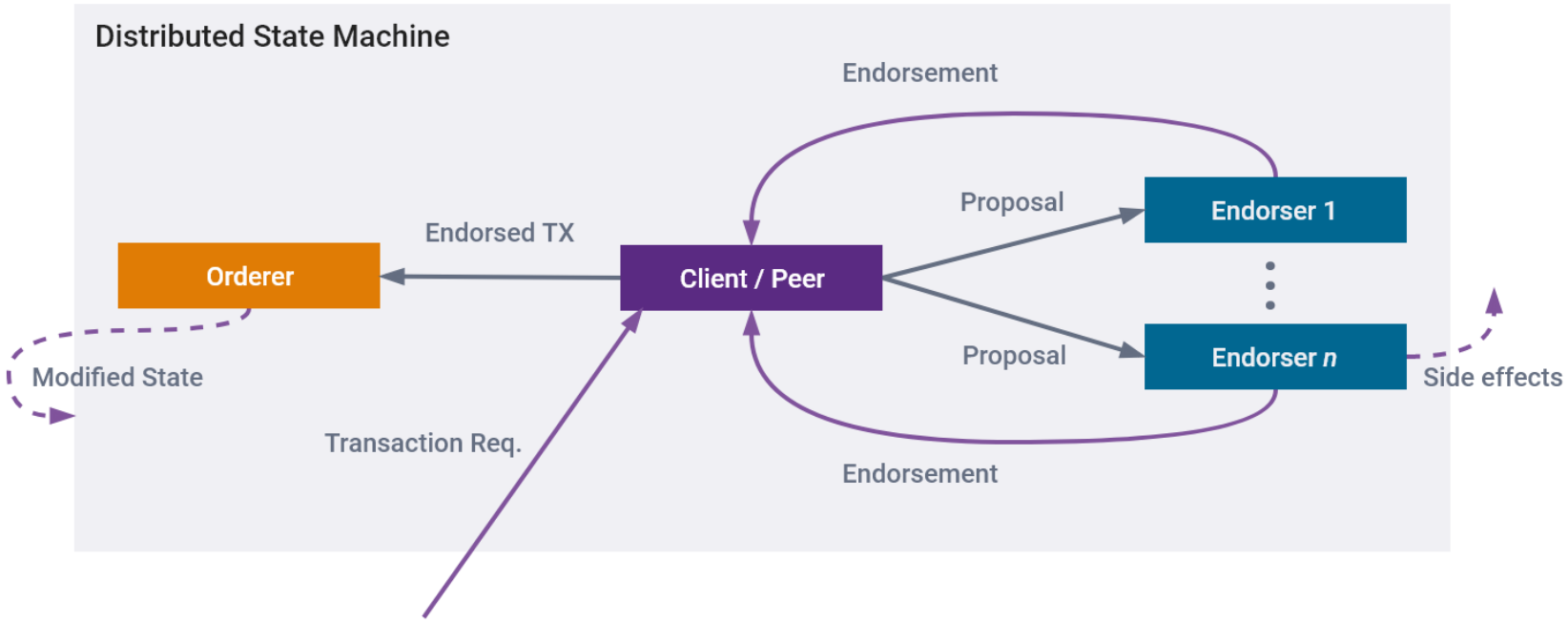
```
func (s *SmartContract) changeCarOwner(APIstub shim.ChaincodeStubInterface, args []string) sc.Response {

        if len(args) != 2 {
                return shim.Error("Incorrect number of arguments. Expecting 2")
        }

        carAsBytes, _ := APIstub.GetState(args[0])
        car := Car{}

        json.Unmarshal(carAsBytes, &car)
        car.Owner = args[1]

        carAsBytes, _ = json.Marshal(car)
        APIstub.PutState(args[0], carAsBytes)

        return shim.Success(nil)
}
```

Security Model

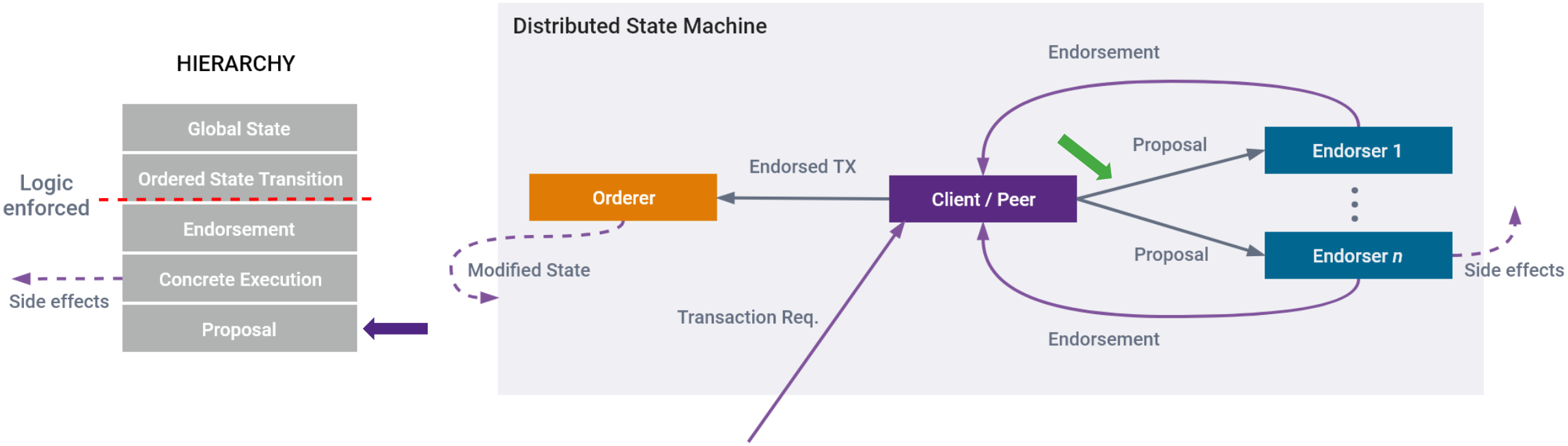


HyperLedger Machine



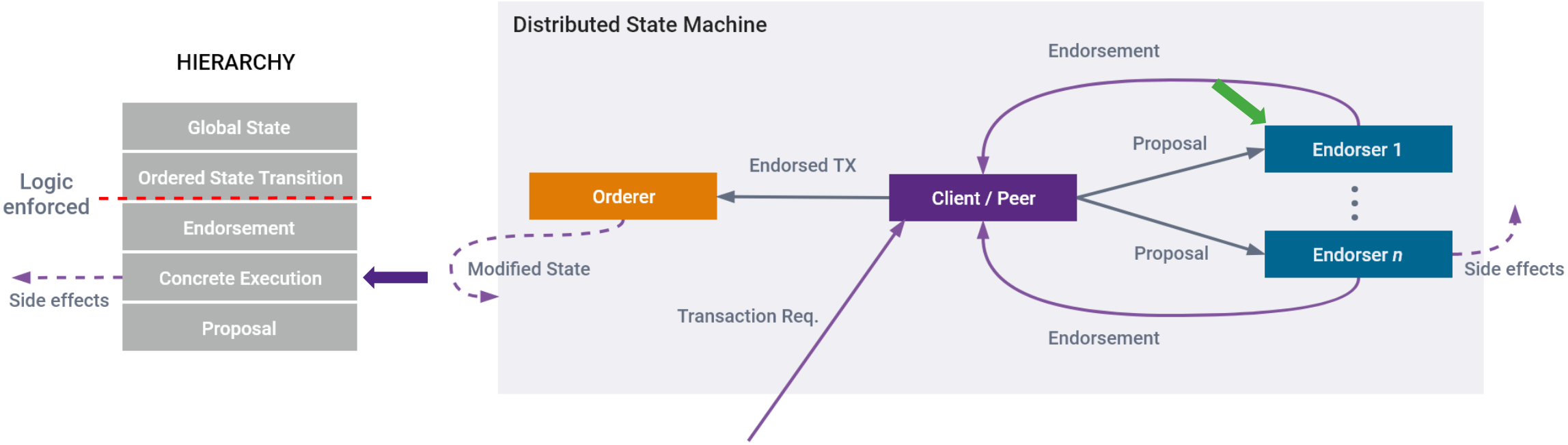


HyperLedger Machine – Proposal

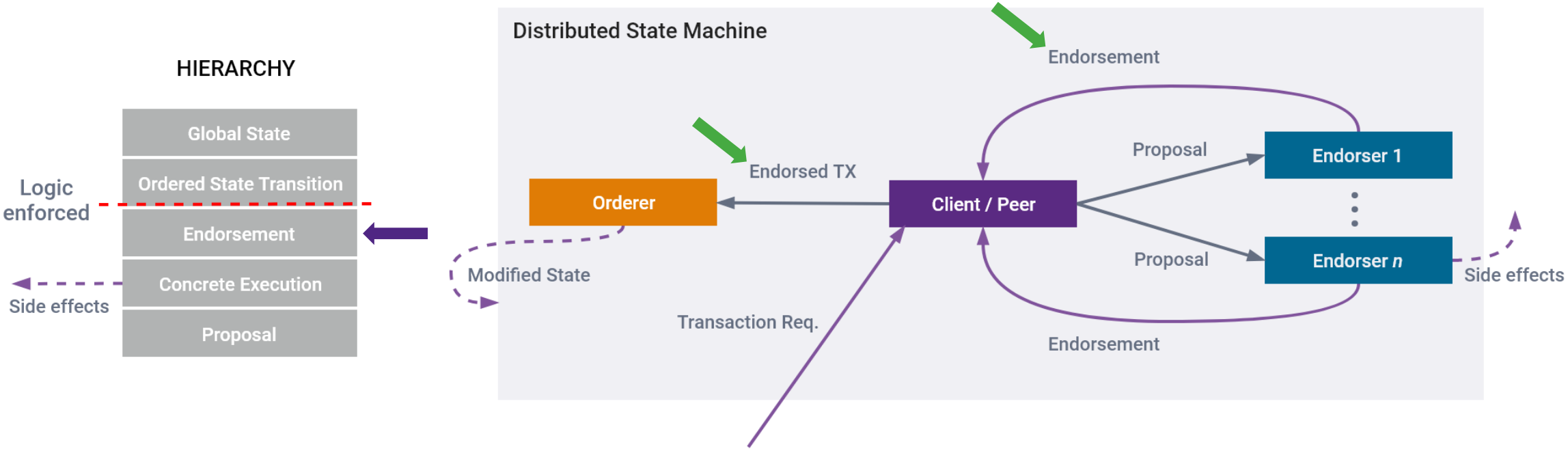


SYNOPSYS®

HyperLedger Machine – Concrete Execution

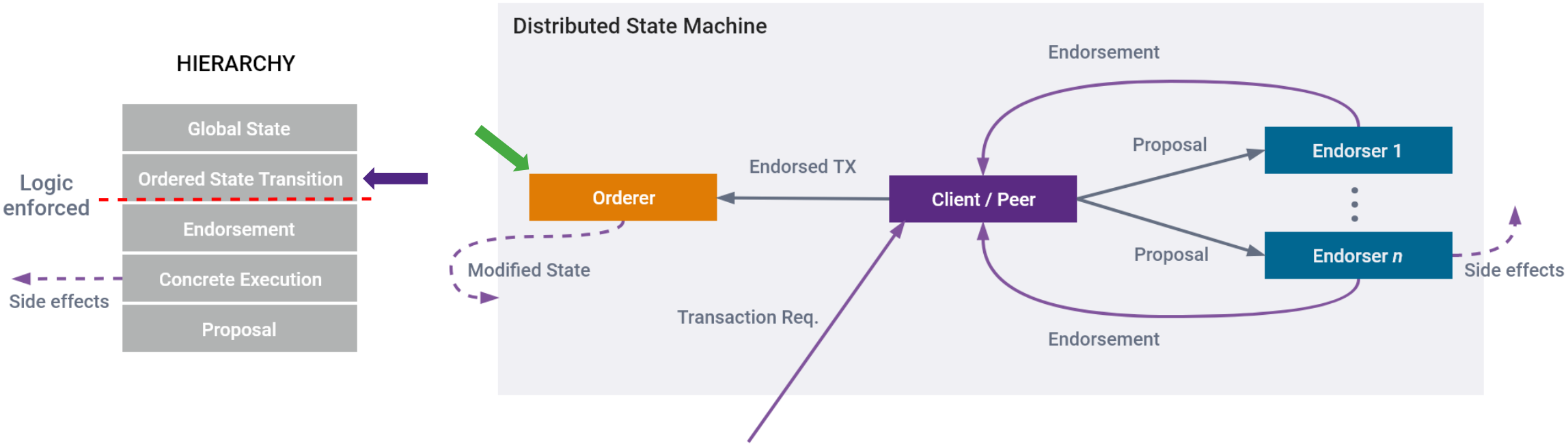


HyperLedger Machine – Endorsement

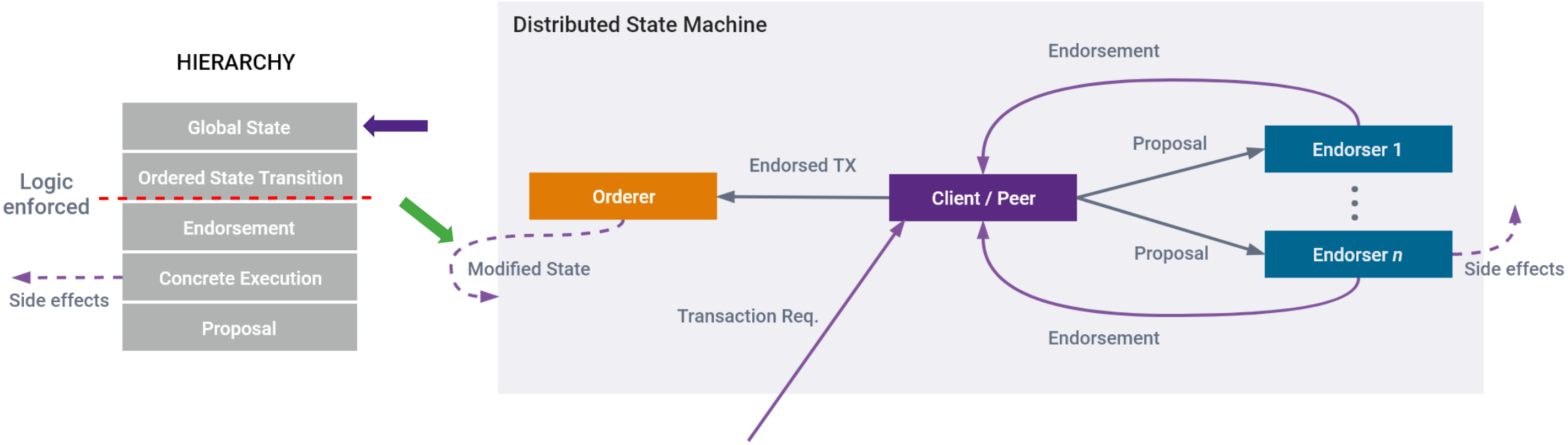


SYNOPSYS®

HyperLedger Machine – State Transition

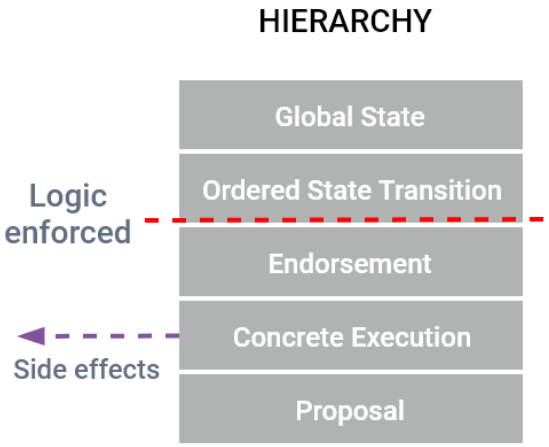


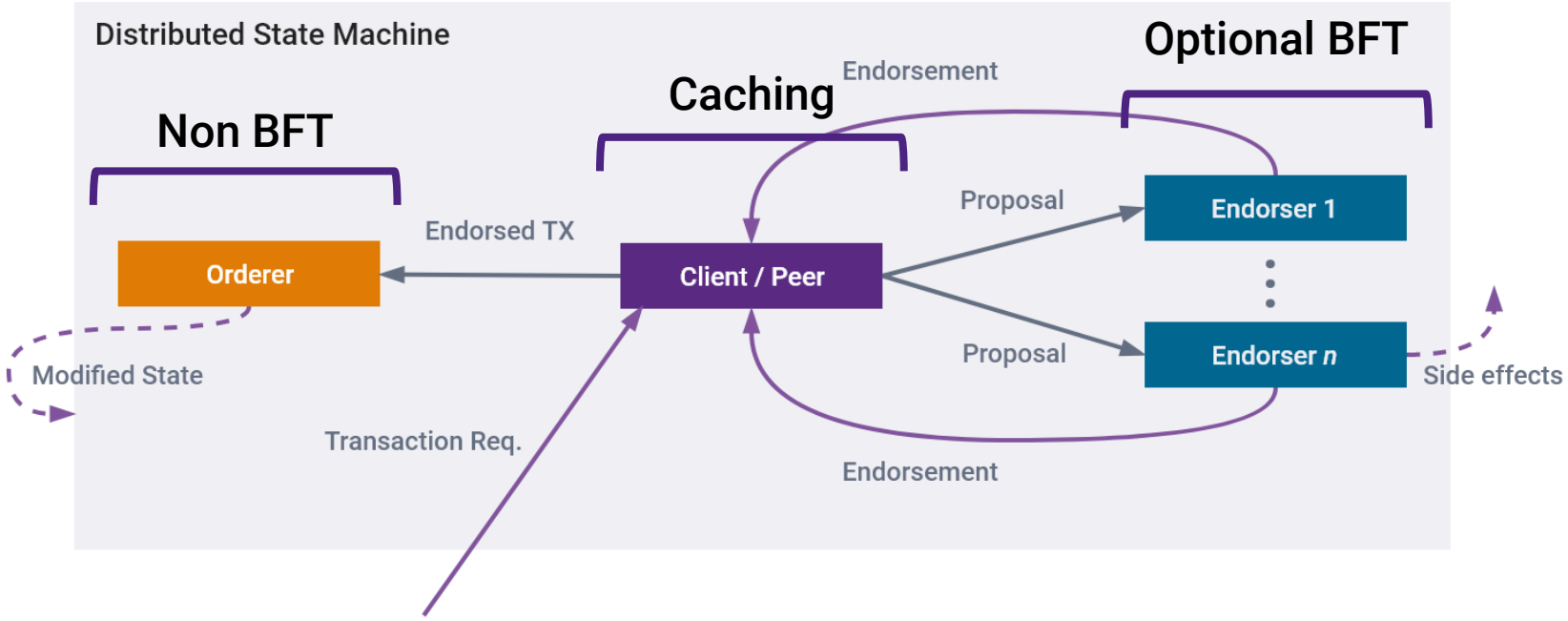
HyperLedger Machine – New Global State



SYNOPSYS®

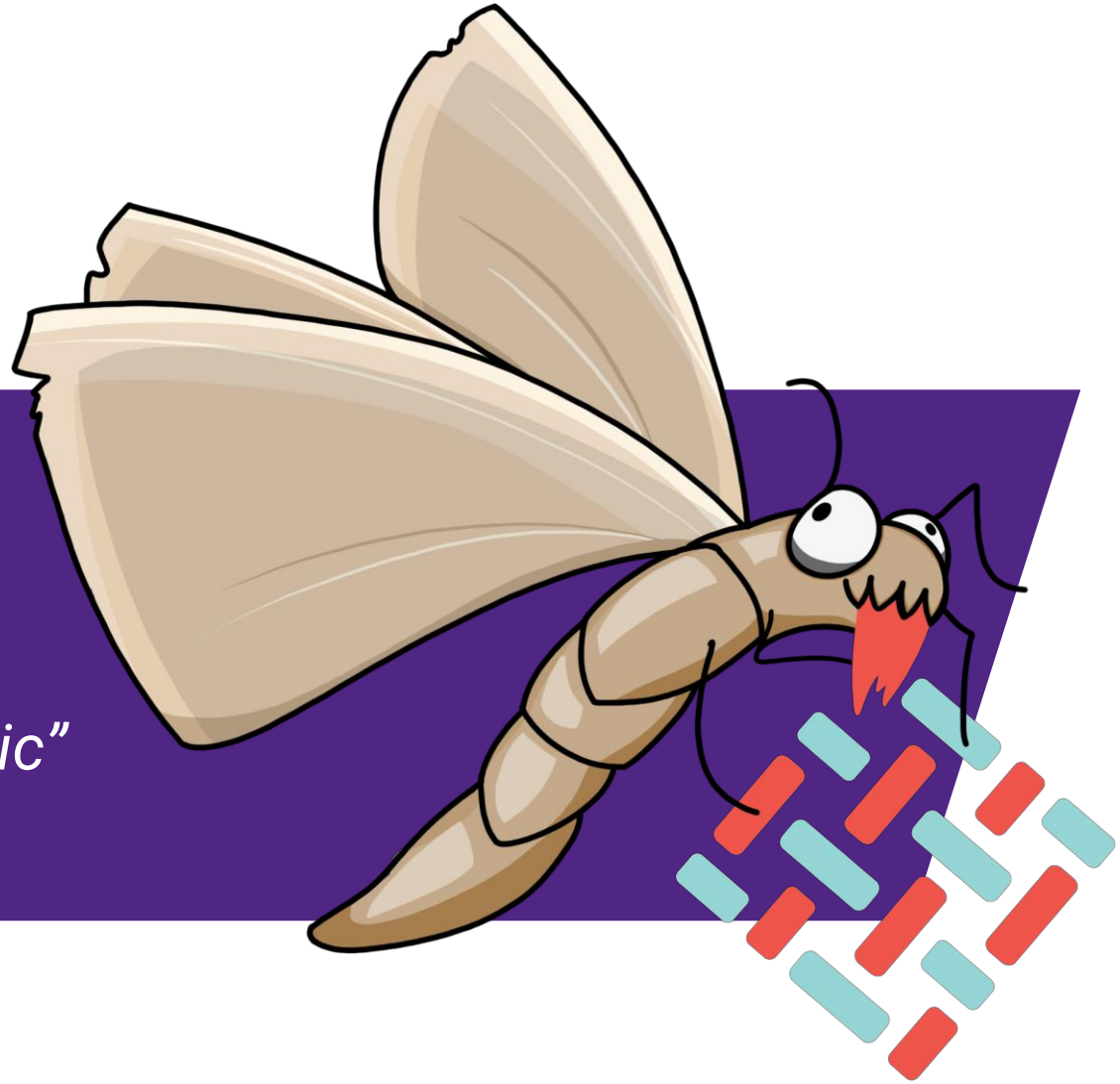
HyperLedger Machine – Suspect



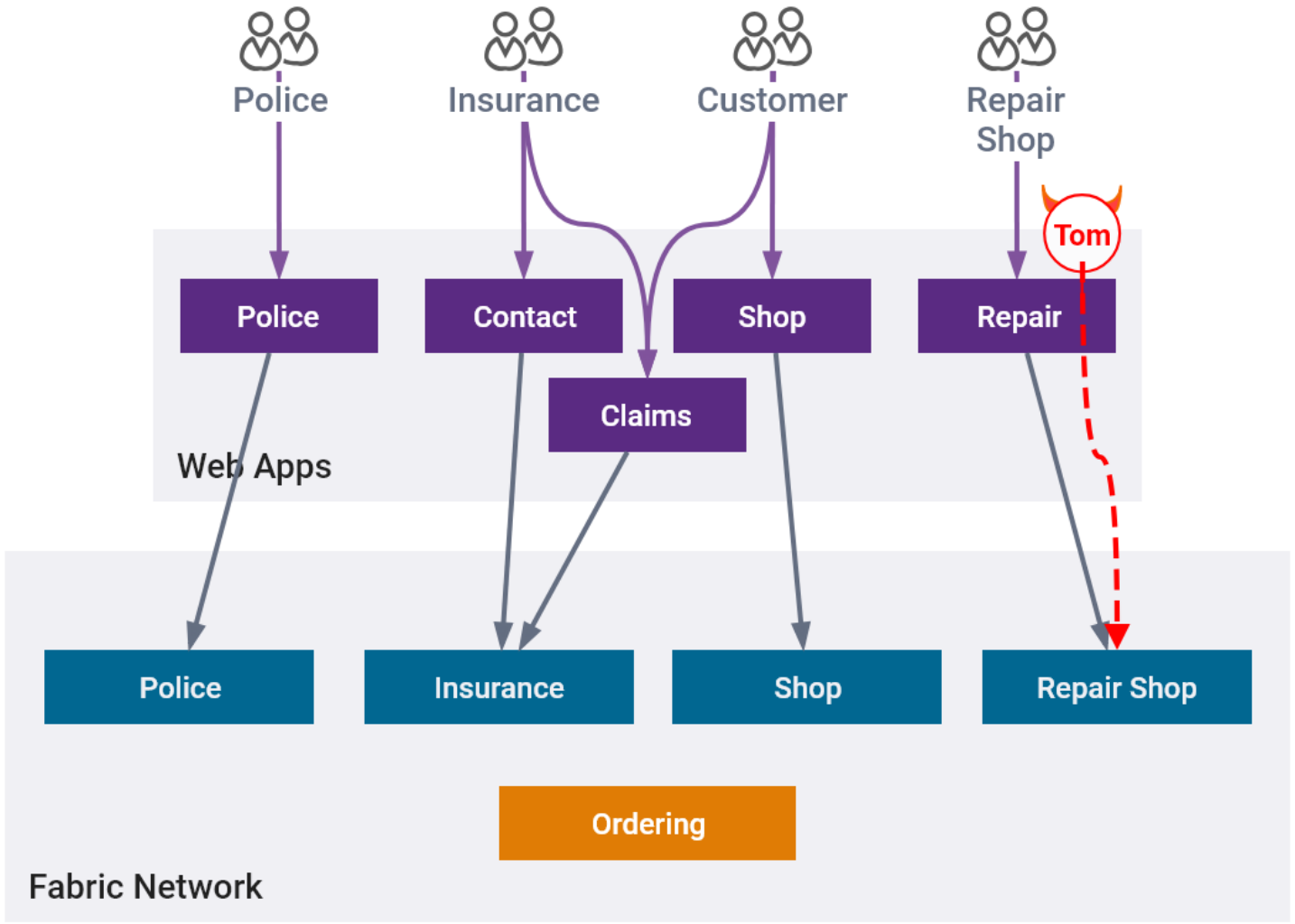


Tineola

“A Tool to Interface With HyperLedger Fabric”

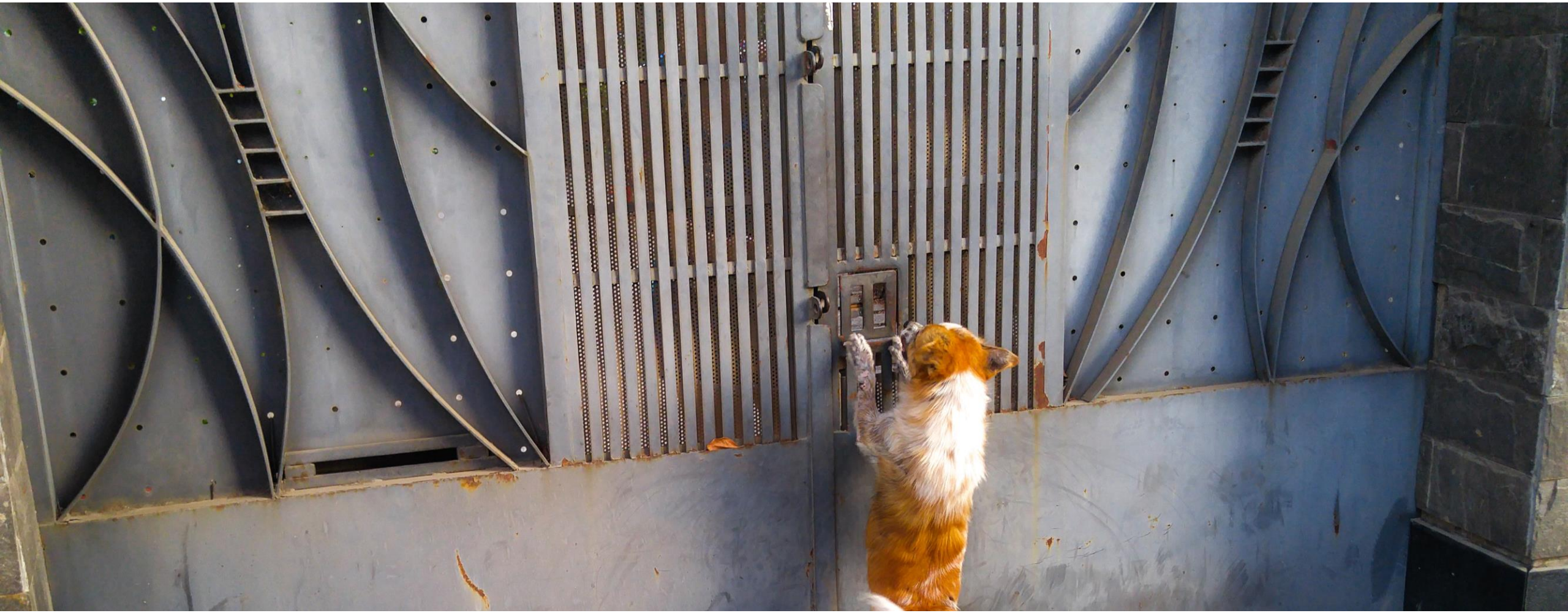


Appetizers



Enumeration

Invoking Chaincode

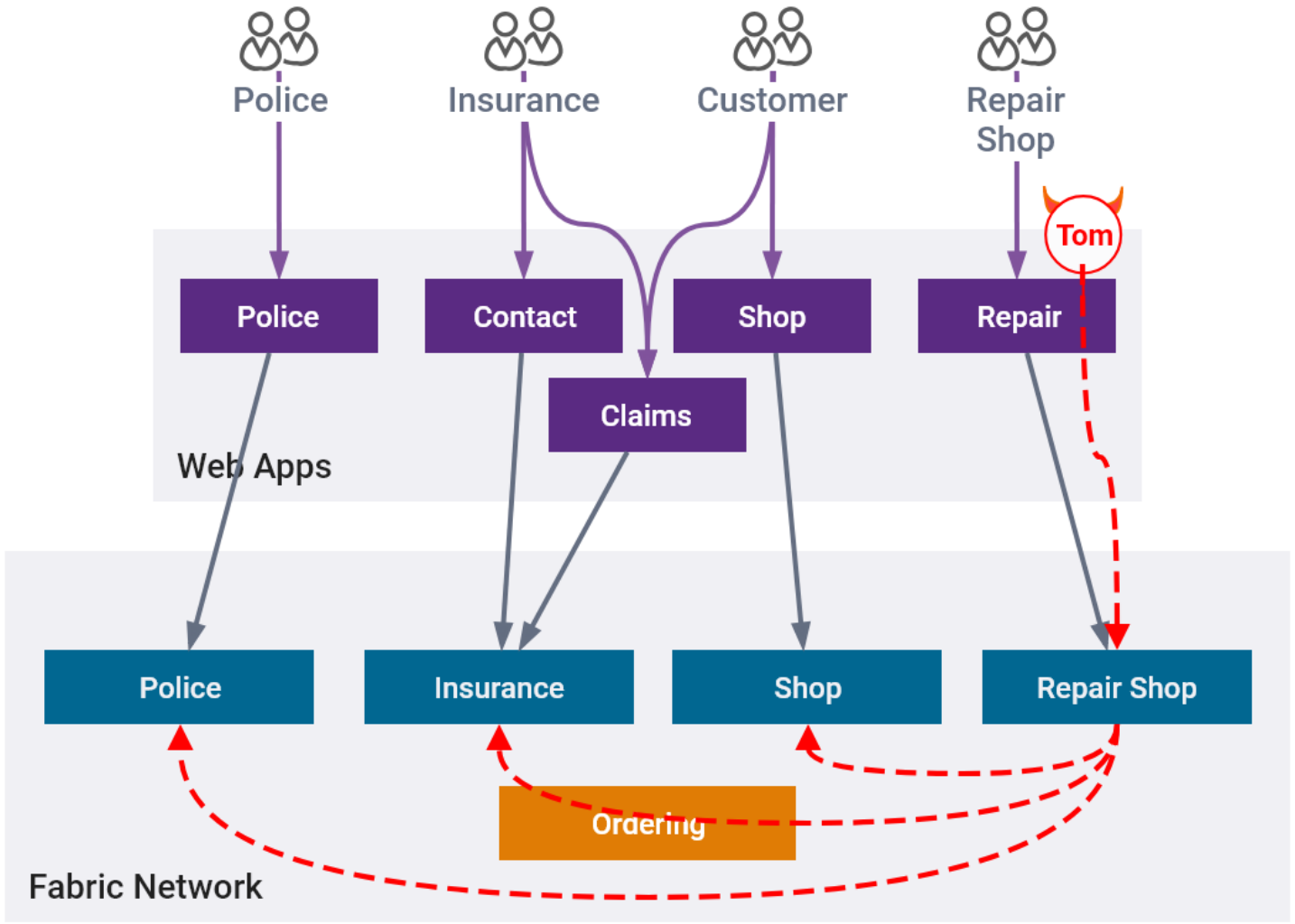


Fuzzing

Simple Injection

```
A = args[0]

// Get the state from the ledger
Avalbytes, err := stub.GetState(A)
if err != nil {
        jsonResp := "{\"Error\":\"Failed to get state for " + A + "\"}"
        return shim.Error(jsonResp)
}

if Avalbytes == nil {
        jsonResp := "{\"Error\":\"Nil amount for " + A + "\"}"
        return shim.Error(jsonResp)
}

jsonResp := "{\"Name\":\"" + A + "\",\"Amount\":\"" + string(Avalbytes) + "\"}"
fmt.Printf("Query Response:%s\n", jsonResp)
return shim.Success(Avalbytes)
```

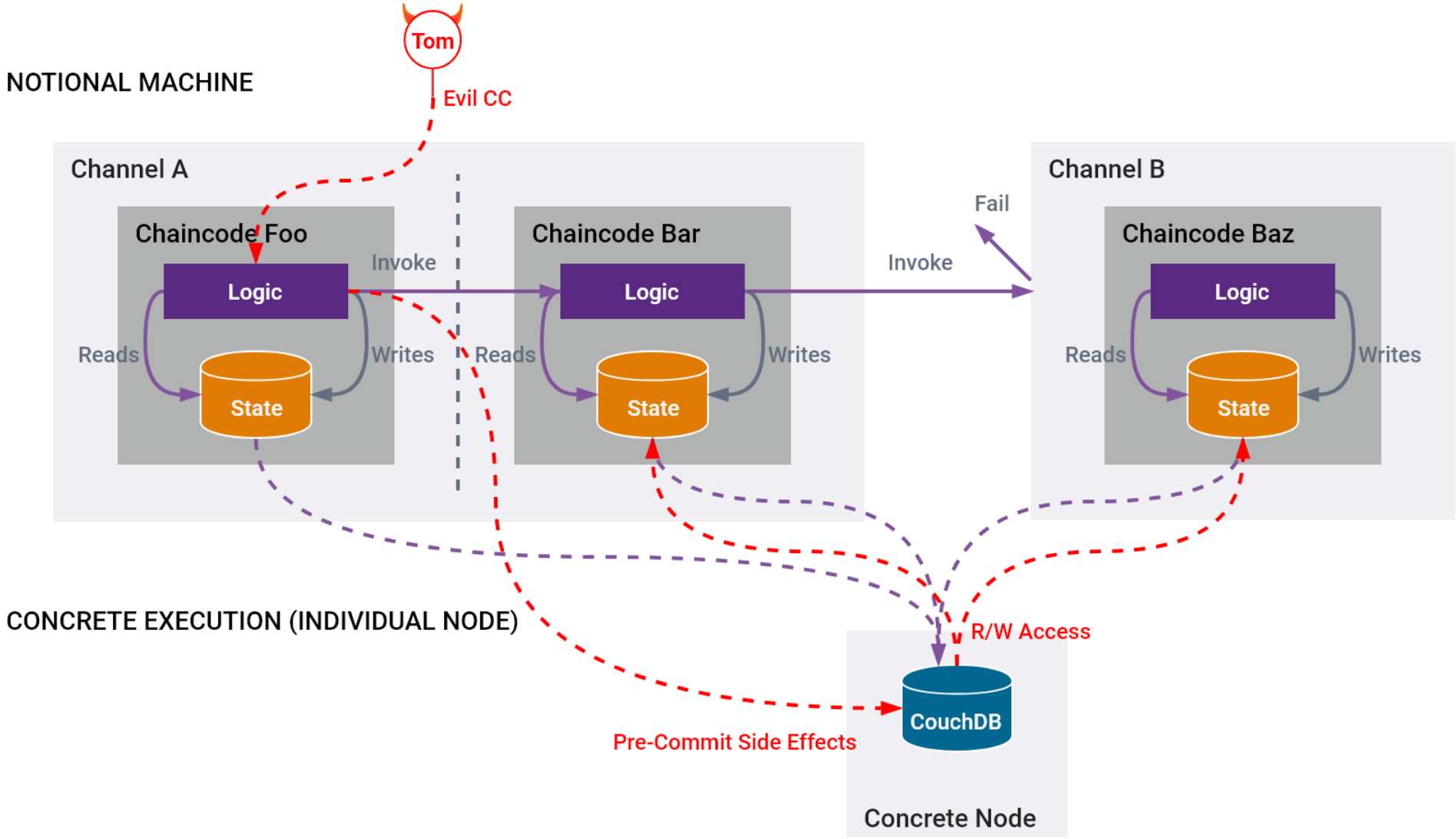
Entrée



Pivoting

Direct DB Manipulation – Hierarchy Abuse

Pre-Commit Side Effects: Problems



Get Your Own Taste

Follow and PR: https://github.com/tineola/tineola

SYNOPSYS®
Silicon to Software™

Thank You


SYNOPSYS®
Silicon to Software™