



HYPERLEDGER

超级账本

www.hyperledger.org

郑嘉文

2020/09



HYPERLEDGER
GLOBAL FORUM





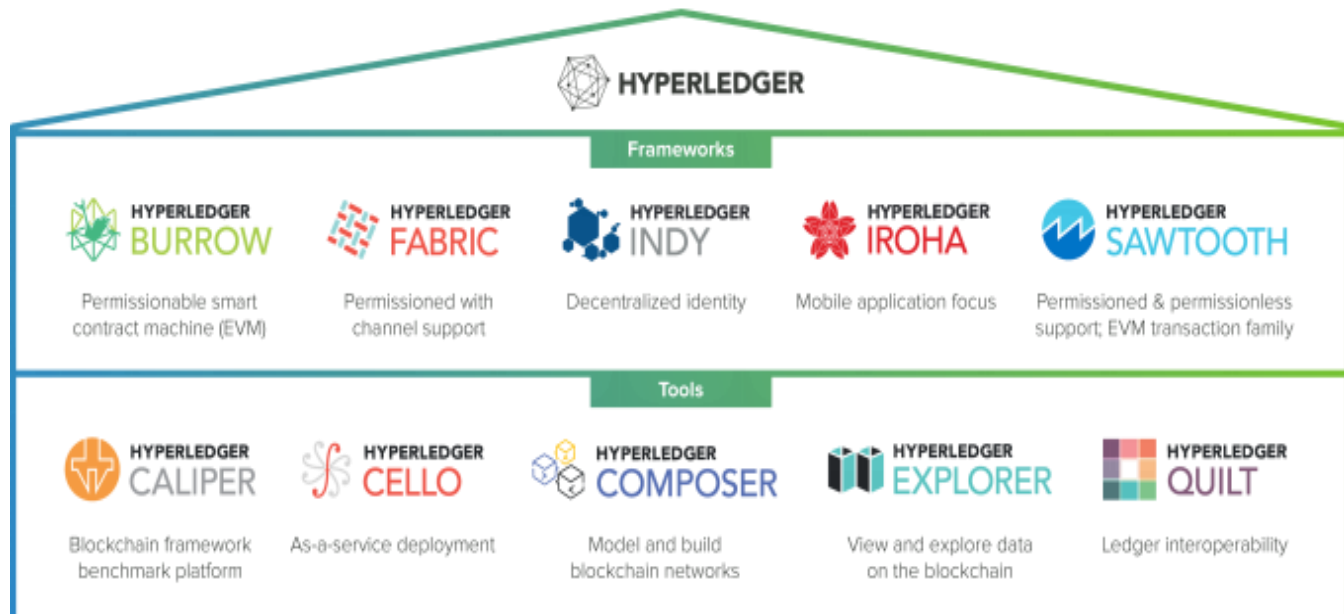
回顾

[illegible]

超级账本

02

超级账本(Hyperledger)项目正是由Linux基金会主导推广的区块链开源项目



Fabric简介

03

超级账本中最具知名度的联盟链的代表项目



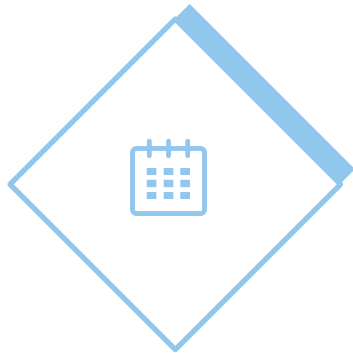
支持智能合约

Fabric有自己的智能合约—链码。基于 Docker实现



许可链/联盟链

Fabric是联盟链的代表



PBFT共识

Fabric没有采用PoW或者PoS共识。其共识效率要高于PoW或者PoS共识



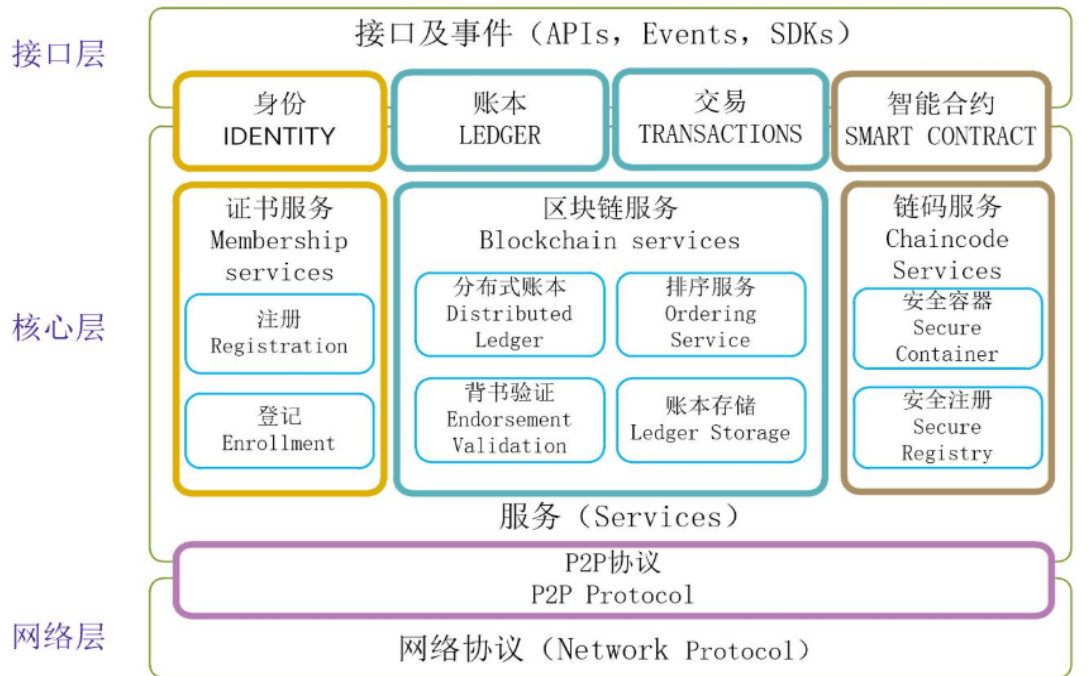
成员服务 (MSP)

所有的参与者都有自己的ID，由成员服务管理

Fabric简介

04

技术架构

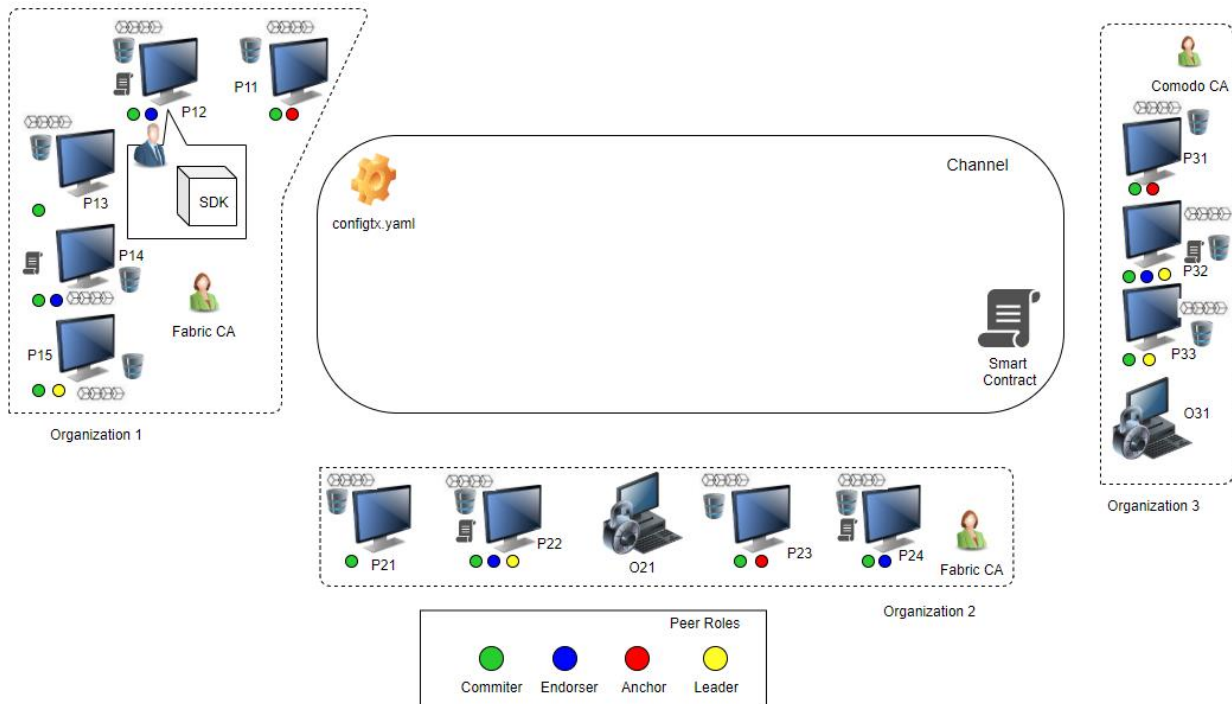


深入浅出区块链

| Fabric简介 |

05

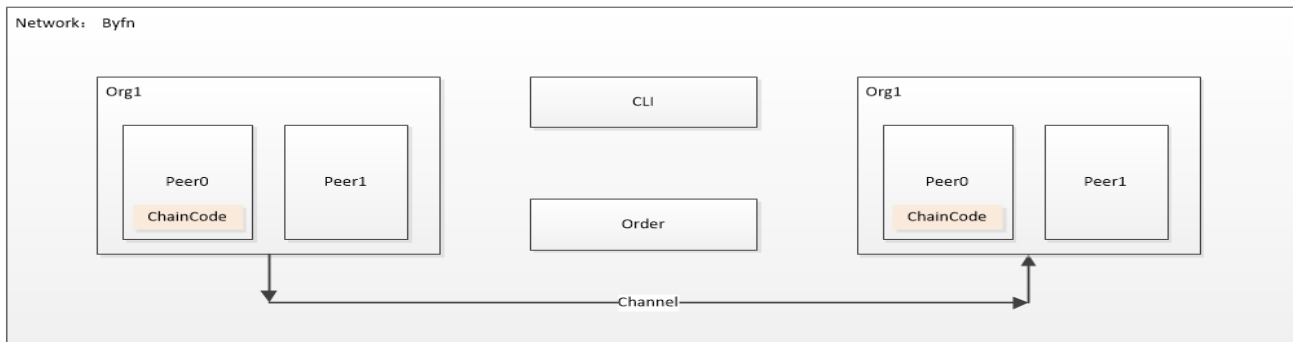
典型网络架构



Fabric网络成员

06

组织 - 节点



背书节点

一种特殊类型的
提交节点，比提
交节点多一个背
书交易功能

提交节点

提交节点提交接
收到的来自排序
服务的块到提交
节点本身所在链

锚定节点

只有被授权的节
点 - 我们称之为
锚定节点，才能
做到跨组织通信

领导节点

领导节点负责把
来自排序服务的
消息传播到同组
织的其他节点

排序节点和命令行节点



排序节点 (Orderer)

排序节点在Hyperledger Fabric系统架构中处于核心角色地位，管理着系统通道与所有应用通道，负责通道创建、通道配置更新等操作，并处理客户端提交的交易消息请求，对交易排序并按规则打包成新区块，提交账本并维护通道的账本数据，提供Broadcast交易广播服务、Orderer共识排序服务、Deliver区块分发服务等。

Fabric目前有3种排序机制：

Solo

KAFKA

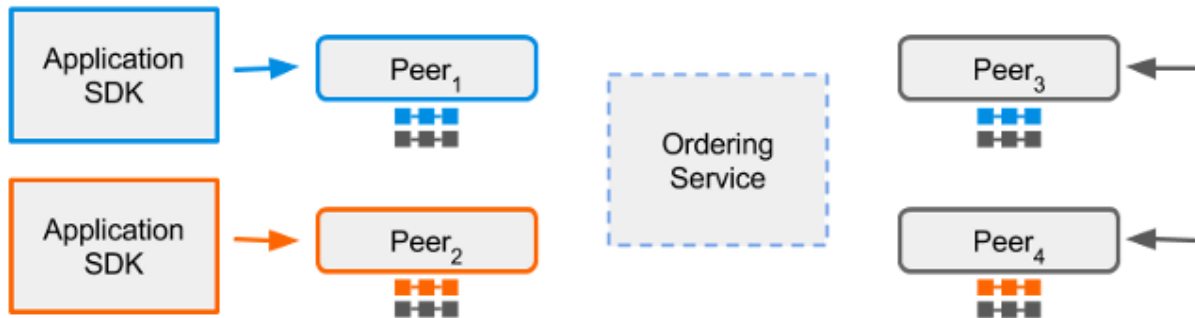
SBFT



命令行节点 (CLI)

主要用于调试和监视各节点状态

通道 Channel



上图中蓝色的账本由P1和P3来维护，而橙色的账本由P2和P4来维护，但是黑色的账本是由P1,P2,P3和P4共同维护的。



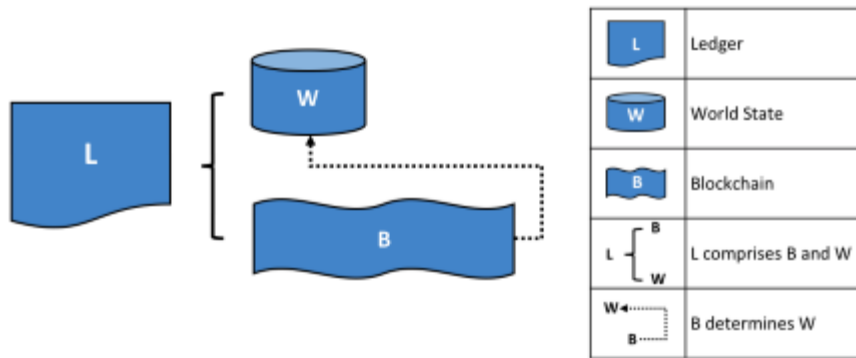
通道 (Channel)

一个 fabric网络可以有多个通道。通道允许不同的组织能够利用同样的网络组建区块链，同时和其他区块链隔离。只有通道的成员节点才能看到通道里的交易。只有通道里的成员才参与共识过程。一个节点可以保有多个账本，加入多个通道。

Fabric账本结构

09

账本结构



L = 账本

W = 世界状态 (World State)

每个链码都有自己的世界状态和区块链。世界状态是一个保存有账本当前状态的数据库。状态是一个键值对。当交易被提交，并且提交节点认为其是合法交易，交易首先提交到世界状态，而后再修改账本。这意味着任何时候从区块链能生成世界状态。

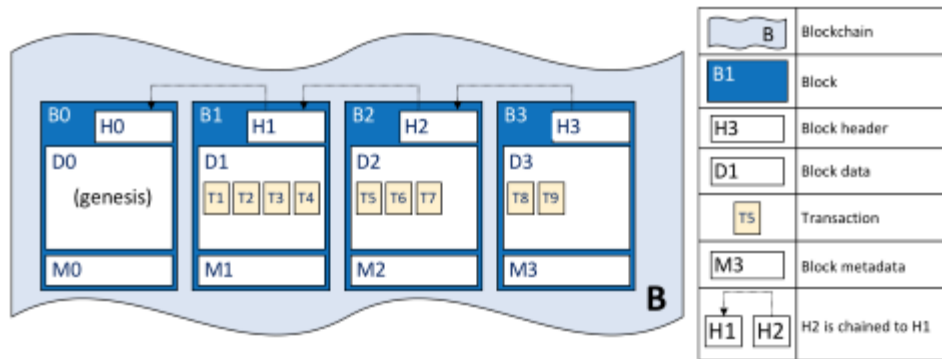
B = 区块链 (Blockchain)

导致当前世界状态变化的所有变化的交易日志。每个块都包含一系列的交易，每个交易代表一个对世界状态查询/修改

Fabric账本结构

10

链结构



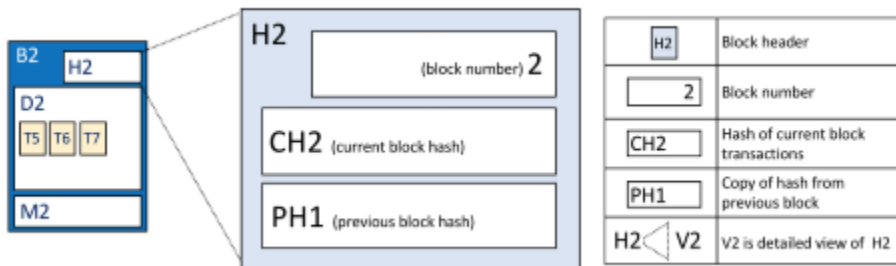
B = 区块链

B0, B1, B2, B3 ... Bn = 块

每个块包含有块头(H), 数据(D), 和元数据(M)。而D中又包含交易(T)。

每个块头都指向前一块。

块结构



H = 区块头部

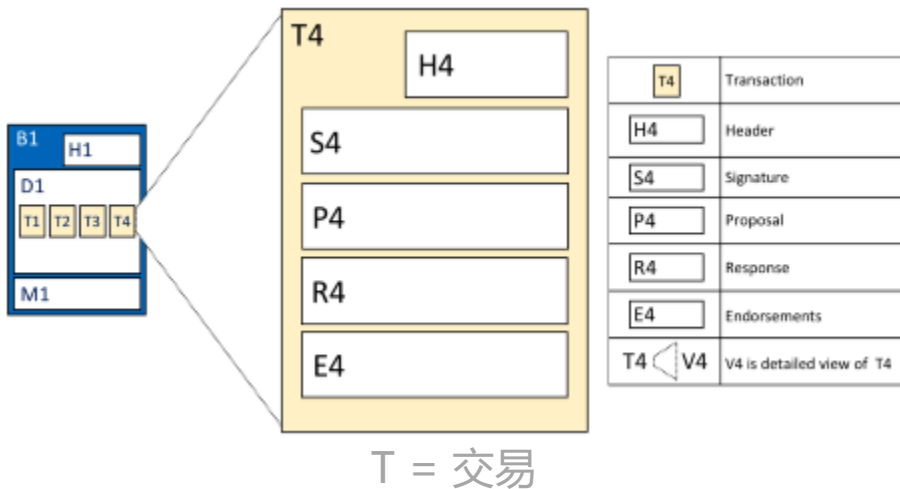
块头(H) 包含3个域:

块号 (Block number)

当前块哈希 (Current Block Hash) : 块中所有交易的哈希

前一块的哈希 (Previous Block Hash) : 前面一个块的哈希

交易结构



头 (Header, H) : 包含交易的基本元数据。比如,链码的名字和版本。

签名 (Signature, S) : 包含客户端的加密签名的合法性: 没有篡改以及要求使用私钥签名。

提案 (Proposal, P) : 给智能合约提供编码后的输入参数。通过提供当前世界状态和提案的输入参数, 就将决定新的世界状态。

响应 (Response, R) : 智能合约的输出, 包含读/写指令集合之前和之后的世界状态。

背书 (Endorsement, E) : 满足背书策略的一系列签名交易的响应。

系统内置链码

系统链码名称	功能介绍	支持外链调用否
配置系统链码(Configuration System Chaincode, CSCC)	负责账本和链的配置管理	是
背书管理系统链码(Endorsement System Chaincode, ESCC)	负责背书签名过程	否
生命周期系统链码(Lifecycle System Chaincode, LSCC)	负责管理用户链码的生命周期	是
查询系统链码(Query System Chaincode, QSCC)	负责提供账本和链的信息查询功能, 包括区块和交易等	是
验证系统链码(Verification System Chaincode, VSCC)	交易提交前根据背书策略进行检查, 并对读写集合的版本进行验证	否

Fabric的系统链码功能主要有:

- 负责 Fabric 节点自身的处理逻辑, 包括系统配置、背书、校验等工作
- 系统链码仅支持Go语言, 在Peer节点启动时会自动完成注册和部署

用户链码

用户链码 是由应用程序开发人员根据不同场景需求及成员制定的相关规则，使用 Golang（或Java，Nodejs）语言编写的基于操作区块链分布式账本的状态的业务处理逻辑代码，运行在链码容器中，通过 Fabric 提供的接口与账本状态进行交互。

用户链码创建使用实例：

安装链码

```
peer chaincode install -p [可执行文件的所在目录路径] -n [实例] -v [版本号]
```

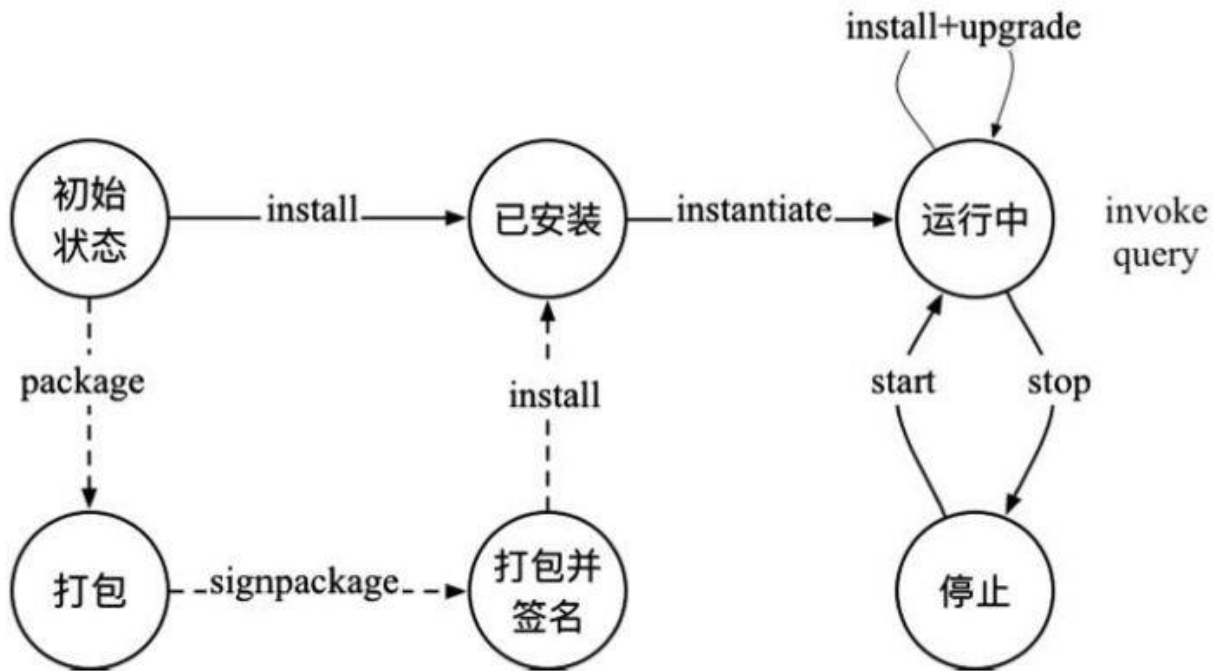
实例化链码

```
peer chaincode instantiate -n [实例] -v [版本号] -c '{"Args":["函数","参数","参数"]}' -C [通道]
```

调用链码

```
peer chaincode invoke -n [实例] -c '{"Args":["函数","参数","参数"]}' -C [通道]
```

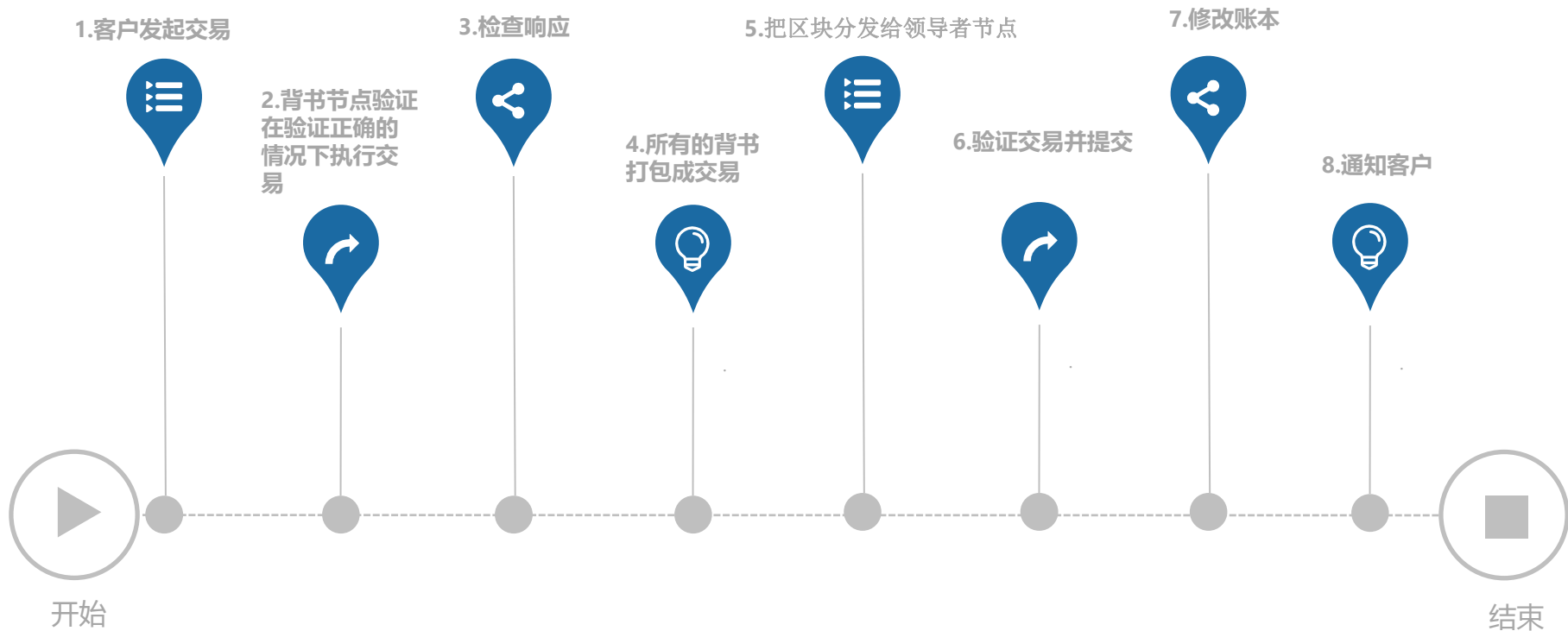
链码的生命周期



| Fabric 链码交互 |

16

应用角度视图





既见君子，云胡不喜

电话 13240946967

邮箱 zy731@hotmail.com

微信 gavinzheng731

博客 <https://my.oschina.net/gavinzheng731/>





谢谢聆听

.....

Q&A

郑嘉文