**“区块链+网络货运”技术架构体系研究报告**

**负 责 人：** 陈志立

**研 究 人 员：**庞雄韬、韦健、符捷、杨豪

**完 成 单 位：**华东师范大学 & 中交智运

**提 交 日 期：** 2022.10.01

目 录

1、功能架构 1

1.1 流程优化 1

1.2 物流追踪 2

1.3 物流供应链金融 2

2、业务架构 3

2.1物流单证 4

2.2 供应链金融 4

2.2.1 多中心数据维护机制 4

2.2.2 交易确权的真实性和时效性 5

2.2.3 交易真实性证明 5

2.2.4 共享账本应用 5

2.2.5 区块链合约的应用 5

2.3 数据存证平台 6

2.4 物流征信平台 6

3、数据架构 7

4、部署架构 11

4.1 部署内容 11

4.2 联盟链节点组织 11

4.3 分布式管理 12

5、技术架构 14

5.1 成员管理 14

5.2 节点管理 15

5.3 权限控制 16

5.4 Gossip协议 18

5.5 共识机制 19

5.6 身份隐私保护 20

5.6.1 限制输入 21

5.6.2 节点身份隐藏 22

5.7 金额隐私保护 23

5.7.1 累加器 24

5.7.2 同态算法 24

5.8 密钥生成 25

5.8.1 随机数生成 25

5.8.2 素数生成 26

5.8.3 组协商密钥 27

5.8.4 基于区块链的密钥管理系统 28

5.9 分片技术 28

5.10 跨链技术 29

5.10.1 公证人机制 29

5.10.2 侧链/中继链机制 30

5.10.3 哈希锁定 31

5.10.4 设计方案 31

5.11 国密算法 31

5.12 数据归档 33

6、区块链融合架构 35

本文拟通过对区块链关键技术攻关，构建起区块链在“网络货运”及其相关业务领域中的整体技术架构，提出创新性的思路，以合理解决系统规模、业务复杂度与性能、稳定性之间的矛盾。并根据业务量、数据量、访问量实际，在保证系统可靠性、鲁棒性、服务及时性的前提 下，合理规划系统建设成本、周期、复杂性、冗余度等要素，以千万级用户量及日活、千万级日交易量为指标，构造系统的区块链体系架构，提出区块链技术总体设计方案，包括业务架 构、数据架构、功能架构、部署架构、技术架构、区块链融合架构等。有助于带动中交智运在网络货运及其相关领域的核心技术创新、提升科技竞争优势，同时有助于加深中交中交智运对区块链技术的储备，为中交智运抢占万亿级物流市场奠定基础。

# 功能架构

项目采用联盟链方式的多层主子链区块链体系架构，建设“中交智运主链 + 货运物流子链、货运金融、其它应用子链”的货运物流区块链应用体系。主子链结构主要依托于主链的安全性共识，主链通过注入核心资产, 包括公共数据、标识资源和监管规则等，整个平台作为主链，安全性较高，对外提供基础服务。子链则通过主子链注册流程启动，主要针对不同业务场景独立设计, 并支持其独立执行共识, 实现数据安全隔离、高性能运行。流程优化、物流跟踪和物流金融可以作为中交智运的三个功能子链。



图1： 区块链在货运物流领域的应用场景

* 1. 流程优化

在流程优化方面，货运物流承运企业或货运司机和发货企业之间的运单等电子合同，是双方进行运费结算的重要凭证。而使用传统纸质单据，首先是运营成本高，结算的效率低，不符合当前物流信息化的高效便捷的要求。通过区块链技术、电子合同和电子签名等技术，实现运输凭证签收无纸化，并将单据的流转及电子合同签署、货物签收等过程记录，写入区块链存证，既保证了数据的可信不可篡改，不可伪造，实现承运过程中的信息流与单据流自动核验，为运单费用的计算，提供真实准确的运营数据基础。在对账环节，关于整个物流过程的关键信息（货量、金额、车型等）都在区块链上可信的保存，并根据物流业务逻辑，编写相应对应的智能合约，完成自动对账，同时将异常对账出现的相关信息上链，整个对账过程是高度智能化并且是高度信任的，解决物流平台的结算问题，减少纠纷的发生。

物流供应链中，企业与企业之间，个人与企业之间的信用签收凭证大部分还处在纸质单据与手写签名的阶段，这些纸质单据不仅作为运营凭证使用, 还作为结算凭证使用。传统纸质单据会造成已有纸化办公带来的成本上、管理上的浪费，由于传统内审、外审的要求，造成有纸化委托书的存在， 势必在材料成本和管理成本方面造成浪费。而通过区块链和电子签名技术的结合可以解决传统纸质单据签收不及时、易丢失、易篡改，管理成本高的问题，实现单据流与信息流合一。以物流快运配送的场景为例: 司机与承运商，承运方与货主之间的结算凭证是通过区块链上真实可靠的电子运输委托凭证，而不是传统的纸质委托书。

* 1. 物流追踪

在物流追踪方面，区块链商品溯源平台通过物联网和区块链技术实现商品从生产、加工、运输、 销售等全流程的透明化。区块链技术可以保证数据存放的真实可靠，物联网技术则可以保证数据在收集过程中的真实可信，消费者通过商品上的溯源码溯源商品信息。 供应链中的关键方参与区块链系统节点，如农业生产商、物流运营商，以及港口运营商和海关， 基于区块链的系统将在分布式网络上存储集装箱、文件和金融交易的数据，实现端到端的供应链全 程数字化，帮助企业监控和跟踪数以万计的船运集装箱记录。可以改善库存管理，减少资源浪费，缩 短货物在海运过程中所花费的时间，同时也可以提高贸易伙伴之间的信息透明度，实现高度安全的信息共享，消除欺诈与不守信行为。供应链生态系统中的每个参与者都能查看货物在供应链中的进度，了解集装箱已运输到何处。通过实时交换原始供应链事件和文档改善对集装箱在供应链中所处 位置的详细追踪。未经网络中其他方同意，任一方都不能修改、删除, 甚至附加任何记录。这种级别 的透明度有助于减少欺诈和错误，缩短产品在运输和海运过程中所花的时间，改善库存管理，最终减少浪费并降低成本。

* 1. 物流供应链金融

在物流供应链金融方面，可依托于区块链上可信全流程的存证数据，如：征信评级、应收账款、固产/动产等，向金融机构证明交易的真实性和票据的真实性，帮助金融机构完善中小型企业的 KYC 画像，解决中小型企业融资难融资贵的问题。银监会可作为监管节点参与到货运物流联盟链中，提前预判和规避可能存在的金融风险。

传统的仓单质押业务，以物流企业为中心建设的仓单业务系统，存在银行对仓单信息获取不及时的问题，可能出现内部人员在仓单上伪造银行解押信息, 给资金方造成损失的风险。

基于区块链构建的数字仓单可以使物流企业、经销商和银行对仓单的权属等状态达成共识，形成不可篡改的共享账本信息，区块链为不互信的各方创造了信任，同时结合物联网技术对接到质押监管系统可以有效避免人为造假行为。在仓库管理中, 物联网技术能够准确感知货物的重量、位置、轮廓、运动状态、管理权限等精确物流信息，是保障动产的有力手段，可以促进动产质押业务从现有的自发自主描述化的模式向系统确认的模式转变，实现监管公示力向公信力的延伸。

* 1. 基于权威 CA 去建立真实可靠的信用主体，通过数字身份和电子签章等技术，确保参与多方的身份真实，对每笔交易进行签名, 实现交易合法并且不可抵赖。
  2. 通过区块链和物联网技术确保质押物的真实性，使融资过程中的买方、卖方、金融机构，核心企业的四个参与方对一份真实可靠数据进行操作，减少信息不对称和信用摩擦成本，避免传统中心化金融平台的不足。
  3. 区块链的自治性能够规避道德犯错，通过智能合约的自动执行，避免人为操作时出现的执行错误，或者故意不履行合同等行为, 能够确保按照预先声明的合约在规定的时间去执行。
  4. 区块链可以保证信用主体累积的交易数据真实可靠，不可被篡改，包括主体的融资数据、还款数据，质押物的数据都可以作为后续买卖双方交易前的对融资风险评估依据。

# 业务架构

针对“网络货运”业务需求，结合现有平台和业务模式，充分考虑“网络货运”、“智慧物流”的发展方向，发挥区块链技术优势，深度挖掘区块链应用场景，规划易用、易维护、易扩展、兼容性好的应用框架，设计应用模式、功能模块和服务方式，提出满足需求且实用性强的应用方案。中交智运平台作为主链提供基础服务比如公共数据、标识资源和监管规则等，下面的业务功能可以看成是各个子链。

2.1 物流单证

区块链是一种不可篡改的分布式账本技术，有助于在无信任的多方之间达成可信和透明的交易。区块链技术应用于物流单证中具备较多优势，联盟链上链的单证数据可以实现全程追溯，实时监控物流单证的数据状态，有助于物流单证的溯源与防伪。同时，基于区块链技术和电子签名技术实现物流单证的无纸化，可利用区块链的共识机制和分布式架构等特性，关联包括保险机构、银行机构、司法鉴定中心等多方权威机构，进一步提升物流单证的公信力，提升认证结果的可信程度。区块链去中心化、不可篡改和可追溯等特性可以解决 PKI 体系存在的固有问题，如单一节点故障不会影响分布式记账整体运行，从而使电子签名具有了更高的连续性、可靠性和容错性。

将物流单证上的运价信息、履约信息编写成智能合约，并由相关方进行背书后发布到区块链网络，协议中明确了双方的权利和义务，开发人员将这些权利和义务以电子化的方式进行编程，代码中包含会触发合约自动执行的条件。比如，承运商福佑卡车的运输司机将一车整车货物按照中交智运的要求从 A 始发地运到 B 目的地，同时，这份智能合约中也规定了从 A 到 B 地的价格，随即系统则自动触发该笔交易的生成，参与方各自收到账单，财务按照相应账目进行月结对账付款即可完成整个付款流程，利用区块链分布式账本技术降低对账成本、降低结算周期。

2.2 供应链金融

### 2.2.1 多中心数据维护机制

基于区块链的多中心数据存储的思路，解决传统供应链整个产业链条的信息严重不对称问题。采用联盟链的方式，搭建包含供应链整个产业链条且共同认可的账本。核心企业、各级供应商和资金方可在账本内预设各方权限，如共享贸易流、资金流相关信息。从而实现信息流、资金流、贸易流的共享协同，解决传统供应链信息严重不对称的问题。

### 2.2.2 交易确权的真实性和时效性

通过基于加密数据的交易确权的区块链应用，可以实现交易确权凭证信息的上链操作。通过分布式存储和共享，提升交易确权操作的安全性。通过联盟链的形式，建立一套各方认可的规则合约， 减少交易背书和担保等中间环节，从而实现成本降低。区块链首先要实现的是数据的标准化和线上化，在此基础上供应链金融应用中可以打通债务方，尤其是核心企业的 ERP 实现实时确权，从源头上保证了效率和真实性问题，并实现确权凭证的开具、背书、审核、签收的全流程上链，各方均可查看无法篡改，提高了交易安全性，保证了可回溯性。

### 2.2.3 交易真实性证明

通过互联网、物联网和区块链技术的联合使用，结合供应链金融具体场景，可以交叉验证诸如主体信用、采购数据、物流数据、订单数据、仓储数据、贸易数据的可靠性。而区块链主要在其中承担整个链上交易的验证工作，记录不同数据、交易节点和时序关系及变更历史，提高整体交易网络的真实性。

### 2.2.4 共享账本应用

区块链技术可以通过分布式数据存储的共享账本将数据安全地同步各参与方。以应收账款融资为例，区块链技术可使整个供应链产业链条中的多级供应商或者经销商共享信用的传递。各方均可以通过区块链技术追溯确权凭证的开具、转让、拆分、融资过程，并在平台上展示，解决了资金方传统审核整个供应链条的企业贸易背景、主体信用的不可操作性，从而实现成本的降低。

### 2.2.5 区块链合约的应用

智能合约是一种供应链金融业务执行的自动化工具，可以通过预先设定好的规则和条款，准确、高效、自动地执行合同缔约各方所达成的契约，以此来降低人为因素的干扰。通过智能合约技术，可以解决合同执行过程中各方违约的问题，以此来提升合同执行效率及条款执行的准确率。

2.3 数据存证平台

通过打造区块链存证服务，将货物和订单全程追溯信息、电子存货仓单的追溯信息和全生命周期关键节点、用户的入库、出库、过户等关键操作、入库单、出库单、过户单、合同等重要单据上链、并提供货运物流数据的主动、被动核验功能，更进一步的功能演进则是提供物流电子单据服务和物流交易清、结算，包括电子单据的证据化存证、电子单据自动流转、交易自动结算、多方交易自动分润、电子单据质押、虚拟货仓管理等业务，以及面向司机群体的小贷金融、保险、信用支付等服务。

2.4 物流征信平台

打造基于区块链的物流征信信息平台，为供应链参与方（个体、组织、智能设备）提供多角色的分布式可信身份服务。基于国家/行业/团体联盟的征信评级标准，构建物流快递征信评级体系。利用数字钱包作为激励载体，构建个人征信类数据资产，并在区块链网络进行确权流通。物流 + 区块链技术应用联盟的企业共建行业标准，基于区块链技术搭建去中心化的可信服务体系，建立诚信阳光的物流供应链协同环境。

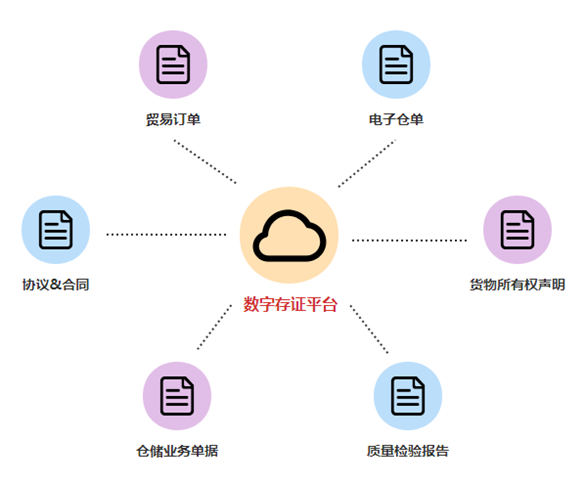


图2： 货运物流数字存证平台

作为发起方联合行业内的物流企业基于区块链技术搭建征信联盟链，参与方作为区块链节点加入网络。各个节点既作为数据提供方，也作为数据使用方，各个商家的原始数据均保存在自己的中心数据库，只从中提取少量非敏感摘要信息，通过区块链广播，保存在区块链中。当某一商家对另一商家的信用数据有查询需求时，首先查询自己所在节点中公开透明的摘要信息，通过区块链转发查询请求到数据提供方，在获得工程师授权，并收到数据查询方支付的费用后，从自己本地的数据库中提取详细的明文信息给查询方。

利用区块链技术为每个参与主体构建一个区块链数字身份，将这个数字身份关联到 CA 证书， 这样数字身份在参与社会活动时具备法律效应。利用信用钱包将数字身份关联的属性进行定义，利用权威机构进行背书，例如：张三定义一张身份证，通过权威机构进行认证后，将认证信息加密后写入区块链存证。当第三方需要验证张三身份时可以通过授权的方式进行验证。同样，从业资格证也可以利用相同的手段去建立。

# 数据架构

“区块链 + 智慧运输”系统需要实现 3000 万以上个人信息、1300 万以上货运车辆信息和 600 万以上企业信息的管理和存证，并且实现对 3000 万以上电子证照的线上核验、出示和鉴证信息管理。系统用户基数大，信息量多，必须采用可扩展模型的数据架构，方可支持海量数据存储，并且需要提高数据的 IO 读取能力，解决数据读取速度，才能支撑高并发用户数。针对这些问题，在研究系统数据架构时，提出平行数据库设计，结合区块链技术，搭建可扩展的数据架构模型，并设计一套数据归档的架构用来进行数据的备份和归档。单链系统吞吐量配置建议不低于 3000TPS，区块链的数据管理存储能力建议达到 10PB。鞍点

设计可扩展的“区块链 + 平行数据库”的数据架构，即在区块链框架内，以区块链为总数据存储资源。其设计思想为，作为数据资源的总管理层，对本地数据进行分类管理。并且对重要的数据， 可根据实际需要上链存储。平行数据库，作为区块链下的数据存储仓库，与区块链框架内的状态库、历史记录库等一样，作为区块链的存储资源，同时，作为数据存储，向外提供数据服务。该设计有利于数据的分类存储，不同的数据类型，在区块链的统一管理下，存储在不同的平行数据库中；有利于数据的分级存储，区块链存储数据的管理信息，平行数据库存储数据资源，对于重要数据，可以直接上链存储，达到数据存储与安全需求的平衡；有利于数据存储资源的调配，当数据的管理层架设在区块链上，可以灵活配置平行数据库，可根据需求增减数据库搭建的数量，实现数据库的可扩展性。

我们认为链下的数据在上链前需要进行确认，一旦上链则认为是真实可信的数据。对于链上链下的数据一致性，我们以链上的数据为主来验证链下的数据，因为链上的数据一旦上链成功是无法篡改的。平行数据库采用分布式的形式，对于平行数据库和区块链数据的一致性，可以利用链上的数据为标准来保持平行数据库的一致，通过链上的数据和平行数据库存储的 hash 进行对照来验证一致性，从而保持多方之间的数据一致性。具体的保持数据一致性的方法可以采用基于区块链的数据核验方法，我们提供两种数据核验的方式如下：

1. 被动核验，即平行数据库通过服务接口，调用数据核验功能，查询并验证自己所需的数据是否一致，区块链平台提供服务智能合约，从区块链账本中数据库中调取所查验数据的哈希值，并计算所查验数据的哈希值作为比对分析，数据一致则通过验证，如果不一致则用区块链上的数据覆盖平行数据库中当前核验的数据。
2. 主动核验，即区块链平台在一定的时间段（比如夜间数据处理业务压力较小的时间段），主动对当前平行数据库中的数据进行核验服务，数据核验通过计算待验证数据的哈希值，与从链上取得的待核验数据的哈希值进行比对，如果一致则通过核验，如果不一致则用区块链上的数据覆盖平行数据库中当前核验的数据。

在设计可扩展的“区块链 + 平行数据库”的数据架构的同时，研究使用高效的大容量缓存，提升数据 IO 读取能力，支撑海量数据的读取和高并发用户数。缓存的目的是提升系统访问速度和增大系统处理的容量，可谓是抗高并发流量的银弹。缓存主要作用体现：预读取、写入缓冲和临时存储。在数据架构中，缓存是设计的一种重要手段，对性能提升有着特别显著的效果。使用缓存时，三个关键因素需要关注，影响缓存的有效性、使用效率和实现效果，即：缓存键集合大小、缓存空间大小和使用寿命。高效的大容量缓存的使用，在物理条件下解决了这些制约。大容量的缓存虽然可以提高数据的命中率，但并不意味着缓存越大就越出众，缓存的应用存在一个算法的问题，即便缓存。容量很大，没有一个高效率的算法，无法有效发挥大容量缓存的优势。使用高效的大容量缓存，重点在于使得算法和缓存容量相辅相成，充分发挥资源优势。

同时辅以数据归档的架构技术，数据归档是指将不再经常使用的数据移到另一个单独的存储设备来进行长期的过程。数据存档由旧数据组成，但它是以后参考所必须且很重要的数据，其数据必须遵从规则来保存。数据存档应具有索引和搜索功能，这样数据可以很容易地找到。

传统数据库的归档技术已经相当成熟，但区块链由于其去中心化，不可篡改的特性，目前在数据归档功能在区块链中并没有提供较好的支持。然而在某些场景下，尤其是大型企业级联盟链下，往往对于链上的数据存储可扩展性要求非常高。随着链上存储业务的冗余，很多不常用的交易依然存储在链上，使得单个节点的存储开销很大，同时降低了交易查询的效率。在我们的调研工作中，发现目前主流的联盟链系统大多都没有完善的数据归档机制，如 Fabric 中暂未支持数据归档操作；长安链中虽支持数据归档操作，但对于大容量数据的归档恢复效率不高。因此，结合我司的网络货运物流场景，我们提出在区块链系统上构建支持大规模可查询可恢复的高效数据归档方案。数据归档的技术具体应用在每个子链上。结合业务场景明确以下数据归档需求：数据可灵活性地转移到独立设备存储；归档数据应确保隐私安全性，可保证可查询性；归档数据可以再恢复到区块链节点中。

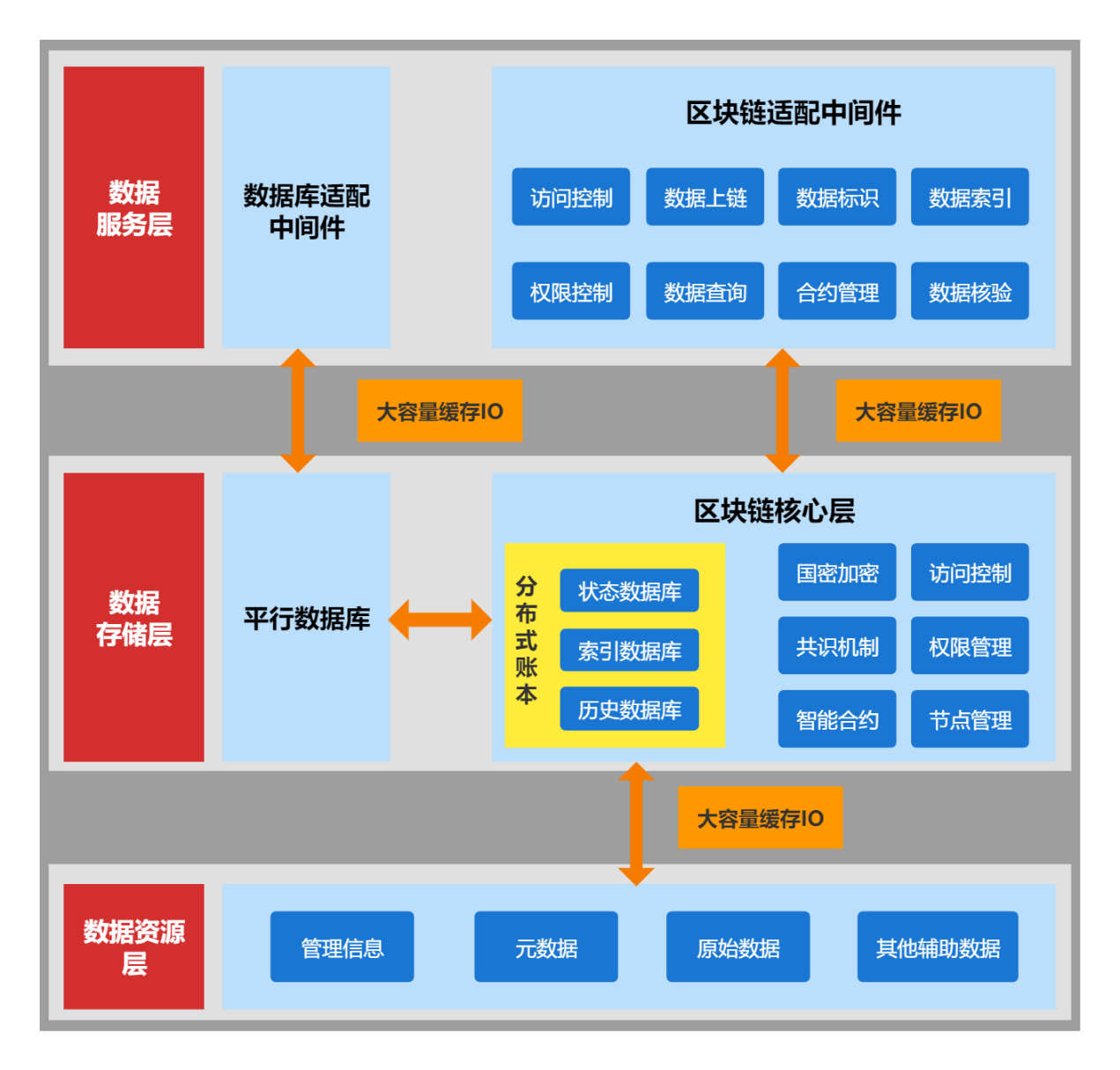


图3： 平行数据库架构

# 4、部署架构

4.1 部署内容

中交智运区块链平台的部署内容和规模，需要根据项目建设过程中的实际参与单位和业务规模具体制定。作为区块链基础设施建设项目，建议选用高性能服务器用于记账节点（全量节点）；配备一定数量的 Kafka 服务器和 ZooKeeper 服务器用于提升节点的稳定性和吞吐量，同时配备多台客服端服务器用于提供对外的应用服务。根据参与联盟组织的单位数量，利用云计算平台部署相应数量的服务器用于参与单位使用。

在架构设计中，采用主子链模型对业务进行分割。其中，主链用于初始化数据以及管理子链节点，子链用于对应的业务，不同子链可进行跨链交易。通过主子链模型对业务进行分类处理，可以减少业务建设成本，提高并行能力，并且联盟链节点可以减少存储冗余数据，增加数据安全性。

在 Fabric 中，多个节点通过商议配置信息建立通道，在一个通道上可以安装多个链码，多个通道之间可以通过共有节点进行跨链访问。因此可以使用 Fabric 模拟主子链模型，其中使用一个通道作为主链，其余通道分别处理对应业务。

4.2 联盟链节点组织

Fabric 是目前流行的联盟链架构，使用容器化技术管理节点，并能够承载不同语言编写的智能合约。Fabric 网络下主要有如下四种类型的节点：Client 节点，用于提供应用访问区块链网络的入口，负责系统与区块链之间的数据交互；Peer 节点，包含了 Anchor 锚节点、Endorser 背书节点、Committer 记账节点，锚节点负责组织间交互，背书节点负责对交易请求验证和签名，记账节点负责保存区块链信息和更新世界状态，一个 Peer 节点可身兼多种角色，需要注意锚节点一个组织只能有一个；Order 节点，负责对用户的交易请求进行排序，打包成区块并分发至其他组织的锚节点； CA 节点，负责鉴权相关工作，可由第三方机构实现。

本项目中，联盟链由多家组织组成，其中每个组织均存在一定数量的节点，不同组织通过锚节点进行交互。各组织架构设计示例如下：

Client 依托于某个 Peer 节点执行区块链操作，由于联盟链中发起请求方均属于某组织，Client 以组织的方式进行管理，在发起交易请求时，首先在向联盟链发起背书请求，收集到足够的背书后， 将交易信息发送给 Order 节点，随后进入排队和确认流程。

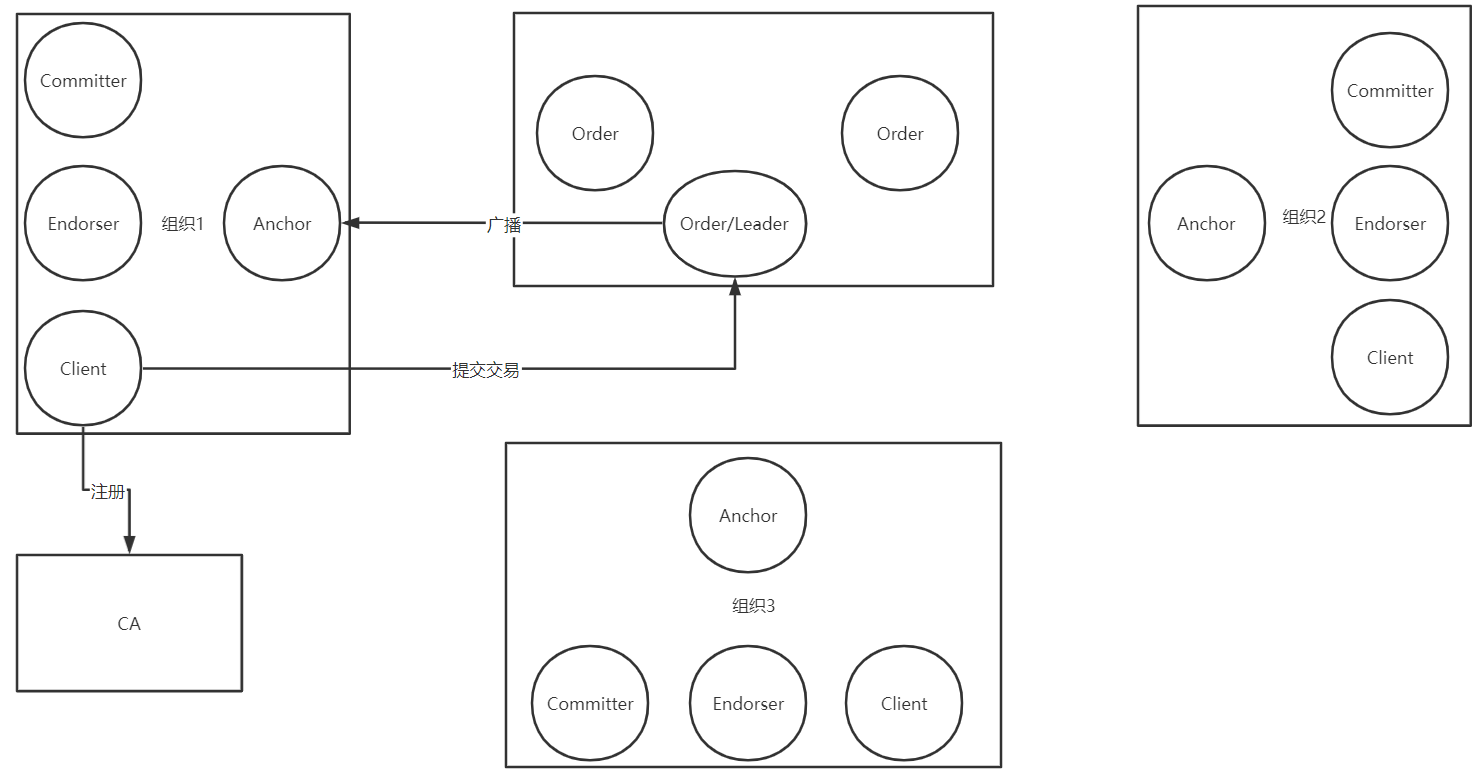


图4： 联盟链节点组织

4.3 分布式管理

随着组织内节点数量增多，可使用 K8S+Zookeeper 进行运维管理，容器化消息传递服务、数据库、服务处理端口，以组织为单位向客户提供具有高可用、高可扩展性的服务。

如下图所示，用户通过客户端向组织发起交易请求，组织内通过 Nginx/Tomcat 集群实现服务接口功能完成负载均衡与请求转发，随后根据服务类型进行业务处理，由区块链服务模块进行区块链业务调用，实现信息上链与发起交易功能，由数据服务模块对数据进行持久化处理，便于更进一步的信息管理与备份。整体通过 Zookeeper 集群实现架构的高可用性。

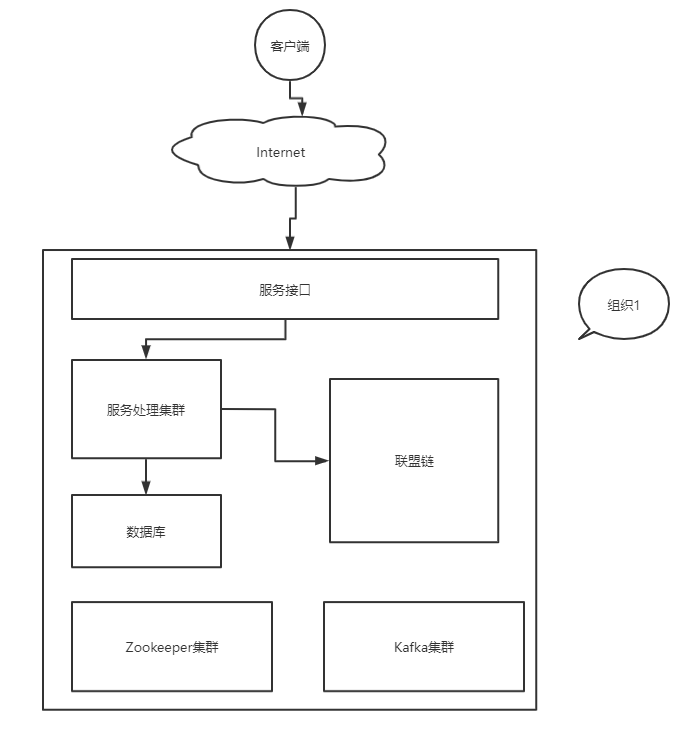


图5： 集群部署

# 5、技术架构

5.1 成员管理

用户注册和登录系统后，获取到用户注册证书（ECert），其他所有的操作都需要与用户证书关联的私钥进行签名，消息接收方首先会进行签名验证，才进行后续的消息处理。网络节点同样会用到颁发的证书，比如系统启动和网络节点管理等都会对用户身份进行认证和授权。管理员和用户可以进行账户信息查询。用户能够对自身涉及的事务操作处理，以及接口服务管理。

CA 节点（Fabric-CA）负责对 Fabric 网络中的成员身份进行管理，采用数字证书机制来实现对身份的鉴别和权限控制，CA 节点则实现 PKI 服务，主要负责对身份证书进行管理，包括生成、撤销等。通过第三方的可信任机构–认证中心 CA(CertificateAuthority)，把机构的公钥和机构的其他标识信息绑定在一起，验证网络上用户的身份。CA 节点可以提前签发身份证书，发送给对应的成员实体，这些实体在部署证书后即可访问网络中的各项资源。后续访问过程中，实体无须再次向 CA 节点进行请求。因此，CA 节点的处理过程跟网络中交易的处理过程是完全解耦的，不会造成性能瓶颈。

身份认证体系目前主要针对机构。在特定业务背景下，经个人用户授权后，由机构代理其个人用户在链上进行交易。一个机构拥有一个到多个区块链节点，节点接入到联盟链的网络中，代表其所属机构在链上进行活动，包括同步数据、参与共识、发起交易等，机构和节点是一对多的关系，为了达成机构级别的安全控制和审计监管，首先需要对机构进行身份认证和准入控制。

平台支持采用 CA 证书实现机构身份认证的机制与流程如下：

1. 机构将自己的机构基本信息以及所属节点的信息，以及 CA 公钥证书提交给联盟链运营管理者，统一进行准入审核；
2. 机构准入审核被通过后，联盟链管理员将该机构的节点信息，CA 公钥证书向全联盟链广播，告知全部参与机构，将新机构的新节点接入到网络，链上已活动的节点应允许新节点进行连接和进行握手；
3. 节点互相连接认证时，节点之间会采用 CA 证书里的私钥对自己的握手信息进行签名，发送给对方，接收方根据发送方的握手信息查询 CA 公钥证书，并使用公钥证书进行签名验证，以判断是哪个机构的节点发起的连接，是否允许继续通信；
4. 证书管理服务定期检测机构的证书状态，判定证书是否有效，有无过期、被吊销等情况。如证书已经失效，则该机构的节点再尝试连接到网络时会被拒绝。

5.2 节点管理

Fabric 提出了 MSP 的概念，抽象代表了一个身份验证的实体，基于它可以实现对不同资源进行基于身份证书的权限验证。从节点层面来说，联盟链中权限管理主要可分为网络准入、组织准入和节点资源访问权限。

节点加入联盟链网络主要包括以下步骤：

1. 首先，基于 PKI 的分层模型允许想要加入网络的用户向 Fabric-CA 提出请求获取数字证书，提供个人身份信息；
2. Fabric-CA 确认用户是否在网络中已注册或存在于证书撤销列表中（CertificateRevocationList， 简称 CRL）；
3. 第二步确认无误后，Fabirc-CA 返回用户注册的数字证书，数字证书上附有用户各项属性值及用户公钥；
4. 用户可使用 CA 颁发的数字证书向对应联盟链网络申请准入，若联盟链网络认可该身份，则用户加入联盟链网络。

数字证书是能证明证书持有者身份且与持有者相关的属性的文档。最常见的证书类型是符合

X.509 的证书，它允许在其结构中编码一些用于身份识别的信息。证书还包含更多信息，其中证书持有者的公钥是在他的证书分发的，而他的私钥不是。私钥用于持有者进行签名认证，必须保密。同时，证书内容将通过 CA 进行签名，如果发生篡改将使证书无效。联盟链网络信任证书颁发者，即证书授权中心（CA），那么该证书就可以证明持有者身份。只要 CA 安全地保存 CA 私钥，任何拥有数字证书的人都可以确定有关的证书信息没有被篡改。

节点进入联盟链网络后需要加入相应组织，特殊情况下可能节点自身作为组织成员。加入组织同样需要获得组织信任的 root-CA 或 intermediate-CA 的认证，然后在本地 MSP 则将节点写入组织列表，并基于其角色或属性分配权限，最后共识更新组织内所有节点及相关通道的 MSP。

节点进入联盟链网络后需要加入相应组织，特殊情况下可能节点自身作为组织成员。加入组织同样需要获得组织信任的 root-CA 或 intermediate-CA 的认证，然后在本地 MSP 则将节点写入组织列表，并基于其角色或属性分配权限，最后共识更新组织内所有节点及相关通道的 MSP。具体步骤如下：

1. 申请节点向请求加入组织的管理员节点提出交易，申请加入组织；
2. 组织内的管理员节点，审核申请节点身份信息，判断是否合法；
3. 若第二步审核通过，组织信任的 root-CA 或 intermediate-CA 将为申请节点分配数字证书；
4. 组织 MSP 更新写入申请节点的数字证书。申请节点的数字证书数字证书将被打包为交易进行组织内广播，组织内的其他节点达成共识更新本地 MSP。

5.3 权限控制

权限管理是区块链技术的关键部分，尤其是对数据访问有更多要求的许可链。Fabric 是一个认证性的网络，所以区块链参与者需要一种向网络中其他参与者证实自己身份的机制从而在网络中进行交易。Fabric 中的不同参与者，包括 Peer 节点、排序节点、客户端等，每一个参与者都具有封装在 X.509 数字证书中的数字身份，这些数字身份确定了它们对资源的确切权限。同时，要使身份可以被验证，它必须来自可信任的权威机构。成员服务提供者（MSP）是 Fabric 中可以信任的权威机构，一个 MSP 是定义管理组织有效身份的组件。总的来说，Fabric 中默认 MSP 使用 X.509 证书来作为身份，并采用传统的公钥基础设施（PKI）分层模型进行数字证书的分发管理。在权限管理模块中，MSP 从节点、组织、联盟等多层次对联盟链进行权限管理。

如下图所示，Fabric 对联盟链进行模块化分层管理，分别包括网络层，交易层，业务层，其中交易层进一步模块化为共识机制模块和权限管理模块。在权限管理模块中，MSP 从节点、组织、联盟等多层次对联盟链进行权限管理。

Fabric 中组织之间通过通道隔离进行资源访问的权限管理，组织内部也可以进一步划分组织单元进行组织内部的资源访问的权限管理。Fabric 网络提供的是联盟链服务，联盟是由多个组织构成， 组织中的成员提供了节点服务来维护网络，同时通过数字证书来进行权限管理。通道概念可以理解为一个通道对应一个联盟。一个通道上可以由多个组织，每个组织有多个成员，也就是说，成员提供节点服务。需要注意的是，一般情况下，组织加入某个通道 A 并不代表组织内所有节点都加入通道 A。一个组织中的成员可以根据网络中流量的要求，提供不同数量的节点服务来满足网络中的访问量要求。

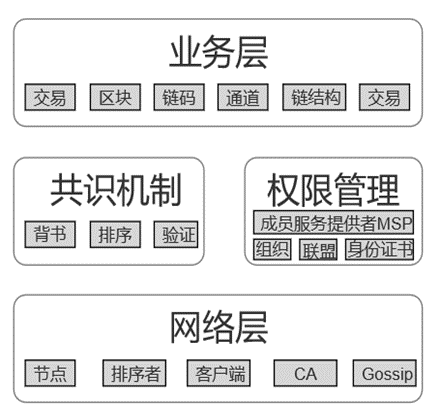


图6： fabric逻辑模块

组织之间能够通过通道隔离进行资源访问的权限管理。如上图所示，三家物流公司可分别视作三个组织 ORG1、ORG2、ORG3，共创建三个通道，以实现不同的业务逻辑，其中 ORG1、ORG2 和 ORG3 共同处于 Channel1，意味着 Channel1 内链上数据由三个组织共同管理。Fabric 中存在数个 Channel，每个组织又可以根据业务需求加入不同的 Channel，具有丰富灵活的可扩展性。

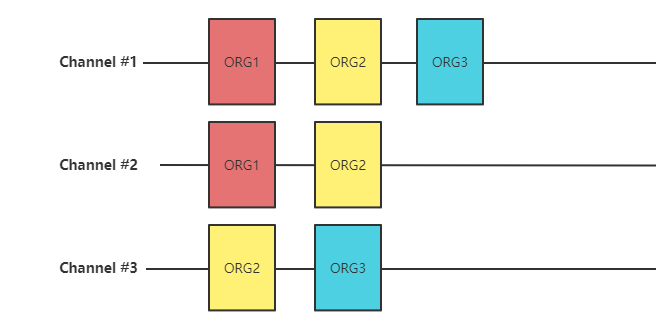


图7： 通道隔离

同时，组织内部进一步可以划分问组织单元，如下图所示，ORG1 划分了组织单元的货运部门、技术部门、市场部门，Channel 中共享货运账单，则通道 A 的 MSP 将 MSP-org1.Transportation 加入到 Channel 的 MSP 成员列表中，即 A 的货运部门节点才能进行通道 1 中账本的查看。

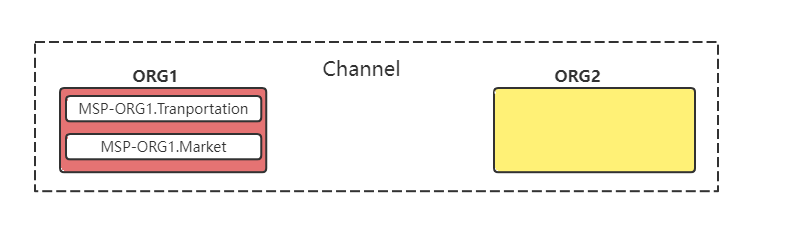


图8： 组织单元隔离

5.4 Gossip协议

区块链节点组成了一个 P2P（peertopeer）的网络。P2P 网络的节点可以通过直接交换来共享资源和服务，节点之间彼此处于对等的地位，并不依赖于集中的服务节点来进行资源调度。网络中的每一个节点既能充当网络服务的请求者，又能响应其它节点的请求，提供网络资源和服务，以达到资源共享的目的。P2P 技术主要用于网络节点之间的健康检测和账本同步。

为了优化区块链网络性能、安全性和可扩展性，可以把工作节点分解为执行交易（背书和提交） 节点和交易排序节点。这种解耦网络操作的方式需要一个安全、可靠、可扩展的数据分发协议来保证数据的完整性和一致性。为了满足这些要求，应用了 Gossip 数据分发协议。

节点利用 Gossip 来以一种可扩展的方式广播账本和通道数据。Gossip 出来消息是连续的，并且通道上的每个节点都在不断地接收当前来自多个节点的账本中已达成一致性的数据。每个通过 Gos- sip 传输的消息都会被签名，因此由拜占庭节点发送的伪造的消息将会很容易地被识别出来，而且可以防止将消息分发到不希望发送的目标处。节点因为受到延迟、网络分区或者其他原因的影响导致缺少部分区块的情况，最终将通过联系已拥有这些缺失的区块的节点的方式，与当前账本状态进行同步。

基于 Gossip 的数据传播协议在 Fabric 网络上执行三个主要功能：

1. 通过不断识别可用的成员节点并最终监测节点离线状态的方式，对节点的发现和通道中的成员进行管理
2. 通过通道中的所有节点来分发账本数据。任何数据未同步的节点都可以通过通道中其他节点来标识缺失的区块，并通过复制正确的数据来进行同步
3. 通过允许点对点状态传输更新账本数据，使新加入连接的节点快速得到同步

5.5 共识机制

本本在区块链中、为了获得系统给予的奖励，区块链网络中的每个节点都试图向链上添加区块，然而如果每个节点都添加区块，那么一笔交易可能被包含在多个区块中，同时每个节点添加的区块也不完全相同，造成节点之间维护的区块链数据不一致，最终导致整个系统失效。为了解决这个问题， 区块链系统使用各种共识算法使得网络中互不信任的节点一起工作，并对添加到链上的数据达成一致性意见。

在超级账本中的基本共识流程有以下三步：

1. 背书过程：背书（endorsement）是指背书节点对收到的来自客户端的请求（交易提案）按照自身的逻辑进行检查，以决策是否予以支持，并对请求提案和造成的状态变更（读写集）添加数字签名。当调用某个链码时，需要获得一定条件的背书才被认为合法，获得来自一定数量的参与成员的一致同意，或者指定的某个成员个体的支持。这些规则由链码的背书策略来指定。背书策略内容可以使用多种规则自由组合，并在链码进行实例化（instantiate）的时候指定。

2. 排序服务：排序服务（orderingservice）通常是由排序节点组成的集群来提供，对一段时间内的一批交易达成一个网络内全局一致的顺序。排序服务采用可拔插的架构，除了用于测试的 solo 模式，后端还可以接入包括 Kafka、Raft 在内的 CFT 类型后端，或者支持第三方实现的 BFT 类型后端。排序服务节点支持横向扩展，以提高整个网络的吞吐率。

3. 验证过程：验证（validation）是对排序后的一批交易进行提交到账本之前最终检查的过程。验 证过程包括检查交易结构自身完整性，交易所带背书签名是否满足预设的背书策略，并且交易的读写集是否满足多版本并发控制的相关要求。确认前的验证可以避免交易并发时的状态更新冲突，确保交易发生后所有节点看到的结果都是一致的。为了满足日千万级交易量的共识需求， 可以采用基于 Raft 的排序服务来优化提高整体共识性能。

Raft 是一种基于状态机负责的共识机制，其为故障容错协议。Raft 的核心由日志复制与领导者 选举两部分构成，节点的状态将会根据不同的条件在领导者 (Leader)、跟随者 (Follower)、候选者(Candidate) 之间变迁。系统在任意时刻只拥有一个领导者，并与所有跟随者之间维持周期性的心跳消息。当跟随者超时仍未收到领导者的心跳消息时，将会转变为候选者状态。候选者将会向其他节点发送投票请求消息，申请成为新的领导者，如果此候选者在超时前收到超过半数节点的确认消息， 则转换为领导者。节点状态的详细变迁过程如图所示：

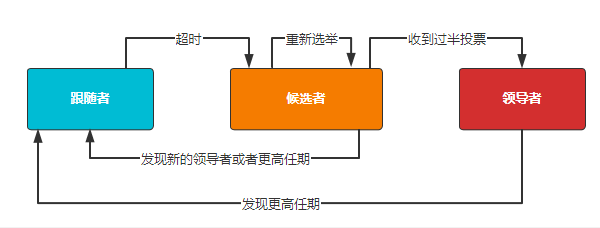


图9： Raft选举机制

在日志复制阶段，所有的交易均由领导者打包生成区块并向所有的跟随者广播，在收到超过一半的跟随者回复后，领导者将会向所有节点发送确认信息，此时该区块将会被提交上链。Raft 算法的通信复杂度为 O(N)，共识效率较高，具有良好的扩展性。Raft 算法在系统内不超过一半的节点宕机时仍能正常工作。

在超级账本中，可以使用基于 Raft 的排序服务替代之前的 Kafka 排序服务。每个排序节点都有其自己的 Raft 复制状态机来提交日志。客户端利用 BroadcastRPC 发送交易提议。Raft 排序节点基于共识生成新的区块，当对等节点发送 DeliverRPC 时，将区块发送给对等节点。

Raft 排序节点的工作流程如下：

1. 交易（例如提议、配置更新）应当自动路由到通道的当前主导节点；
2. 主导节点检查交易验证的配置序列号是否与当前配置序列号一致，如果不一致的话则执行验证， 并在验证失败后驳回交易。通过验证后，主导节点将收到的交易传入区块切割模块的 Ordered 方法，创建候选区块；
3. 如果产生了新的区块，主导排序节点将其应用于本地的 Raft 有限状态机（FSM）；
4. 有限状态机将尝试复制到足够数量的排序节点，以便提交区块；
5. 区块被写入接收节点的本地账本；
6. 每个通道都会运行 Raft 协议的单独实例。换句话说，有 N 个通道的网络，就有 N 个 Raft 集群，每个 Raft 集群都有自己的主导排序节点。

5.6 身份隐私保护

本节旨在确立参与方准入规范以及保护项目中合法参与方的身份隐私，使得项目可以确认并管理并保护合法参与方并及时排除恶意参与方。本货运项目通过联盟链组织，参与方在加入时需要经过严格的身份验证，以此确立参与方隐私保护的基础。身份验证环节需要有人工介入，用来确认参与方的身份、来源、权限等。在审核通过后，会为其分配可接入节点。示意如下:

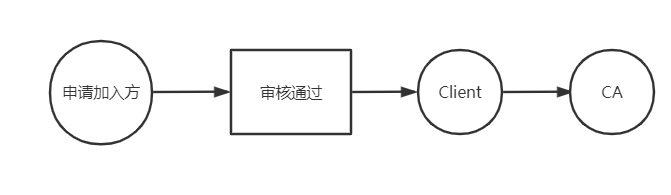


图10： 身份保护

### 5.6.1 限制输入

本限制接入技术目前多是一种策略，即对区块链中的节点进行验证授权，通过严格的验证机制大大降低恶意参与方出现的可能性，常常用在私有链或联盟链中。在比特币系统中, 为了防止恶意节点进行拒绝服务攻击，节点设置了黑名单机制，当从某一节点接受到不合法消息后，会降低该节点信誉度，降低到 0 后加入黑名单。然而，目前对于恶意节点自动检测相关的技术并不多，在区块链的运营中，多是人工设置策略进行管理，本节通过介绍相关论文提出一种相对可行的技术。

在项目运行过程中，可以通过恶意节点检测辨别异常操作，恶意节点检测可以通过业务逻辑设置，并使用机器学习辅助决策，通过聚类实现自动分类节点，识别非合法节点的行为模式。在网络货运项目中，用户的正常操作包括三单的存取、交易溯源、清账等操作，可以通过收集正常操作的行为模式，定义行为模板，然后通过聚类等方式识别异常操作，从而达到异常节点检测的目的。

在进行聚类时，需要对操作相似性进行衡量，比较典型的序列相似性度量方法如下：欧几里得距离算法（Euclideandistance），要求两个序列具有相同长度，使用 L2 范数（norm）计算距离；动态时间扭曲算法（DynamicTimeWarping/DTW），不要求长度一致，并且可以处理时移（TimeShifting）问题，这也是本技术最终选用的方法；实序列距离编辑算法（EditDistanceonRealsequence/EDR），利用了杠杆差（LeveragesGap）和不匹配惩罚，并且可以处理噪音和时移；最长公共子序列算法（Longest- CommonSubSequences/LCSS），对于序列噪音引入了门限，用于发现两个序列中的最长公共子序 列。

聚类算法目前也具有多个可选项，根据类别介绍如下：基于分区的聚类算法，即将未标记数据分成 n 份，每份决出中心，不过对于中心的定义根据算法会有所不同，例如 k-means 以平均值为数据中心，而 k-medoids 以集中点为中心；基于密度的聚类算法，从一个元组开始，根据设定的密度阈值吸收周围元组，其中比较著名的如 DBSCAN，其核心要点是先找到密度较高的点，然后把相近的高密度点逐步相连，在 2015 年的论文《DBSCANRevisited:Mis-Claim,Un-Fixability,andApproximation》中对其进行了改进，分析了 DBSCAN 在高维空间的限制并给出了解决方案；基于分层的聚类算法，对数据集进行层次分解，根据分解策略又可以分为凝聚分层聚类（自上而下）和分裂分层（自下而上）聚类，凝聚分层是指从各样本点开始，根据设定规则合并直到达到设定条件，分裂分层是指在初始状态下合并所有样本点，根据设定准则逐步分裂直到达到要求，分层聚类算法比较著名的有 Chameleon， BIRCH 等；基于网格的聚类算法，将数据集网格化，调整分辨率并识别高密度网格单元，能够获得密度聚类的优点并减少一定的计算复杂度，典型算法如 STING，CLIQUE 等。

### 5.6.2 节点身份隐藏

本在交易环节，可以通过现有的节点隐藏技术保护参与者的身份隐私，在需要隐藏身份的业务场景下，如匿名竞价、匿名担保，通过筛选，本文推荐两个较为成熟的实现方案：一次性地址和环签名。

在大多数采用地址作为标识符的密码货币项目中，地址都是由公钥压缩编码得出，虽然无法通过地址反推出用户信息，但是仍然可以通过追踪交易记录和利用社工手段猜测到用户身份。门罗币提出了一种一次性地址方案，在每次交易时，发送方生成临时公私钥对，其中临时公钥作为接收方地址，而接收方可以通过对应的临时私钥消费。

在网络货运中，以匿名竞价为例，多个出价方使用一次性地址向供货方发送交易和出价信息，在经过混淆网络后，供货方可以使用临时私钥获得各方的出价信息，并且无法得知具体出价人。一次性地址方案的技术细节如下。假设 A 为出价方，B 为供货方，其技术细节如下：

1. B 维持一公私钥对 (k, K), K = kG，G 为椭圆曲线的基点 (生成元)。
2. A 希望发送一笔交易给 B。A 构造一对公私钥，(KO = rG + K, kO = r + k)，r 是 A 生成的随机数；公钥由 A 生成，作为 B 的临时地址，私钥由 A 的 r 和 B 的 k 组合而来，A 将 r 发送给 B，使得 B 可以获得私钥。
3. A 设法秘密传递 r。利用 DH 算法共享 N ←rkG，A 发送 R←rG 给 B，此时 N = r ∗ K，B
   1. 计算得到 N = k ∗ R，再计算 r = N /kG。
4. A 公开发送信息。发送 KO←Hn(rK)G + K，使用 Hash 函数将椭圆曲线上的二维点映射成标量；发送 R←rG
5. B 验证一次性地址。计算 KO ←Hn(kR)G + K，与 KO 对比，若一致，说明是自己的一次性地址，恢复一次性地址对应的私钥 kO←Hn(kR) + k。
6. 代理验证。生成两对密钥 (kv, Kv), (ks, Ks)，将 view 密钥交由代理人，每当出现一次 UTXO， 由代理人验证，通过后再由自己生成一次性私钥；A 发送 KO←Hn(rKv)G + Ks 和 R←rG；B 验证恢复 KO←Hn(kvR)G + Ks 和 kO←Hn(kvR) + ks。

在网络货运中，存在多方进行联合签名的场景，比如在供货链中存在多级供货商，每级会有多个供货商，当消费者要求接收到的货物需要有多级供货商联合签名，但是处于隐私考虑，供货商只需要证明自己属于该级供货商但是不需要泄露具体身份，此时可以使用环签名解决。数字签名类似于纸质合同上的签名，用来证明被签名的数字内容的完整性和来源，其流程是先对数字内容进行摘要以加快签名速度，然后对摘要进行签名，使用推荐的商用签名算法可以达到数学可验证的强安全性。如果签名者不止一方，则称为多重签名，即只需要收集指定数量的签名即可认为有效。群签名由 Chaum 和 Heyst 于 1991 年提出，群体中任意成员都可以以匿名的方式代表群体对信息签名，并且可以通过群公钥验证，群管理者可以管理群的公钥和私钥，利用群私钥对群用户生成的群签名进行追踪，并获得签署者身份。环签名可以看作简化的群签名，与群签名相比，不再需要管理者，并且其他人仅可以验证环签名的有效性，而无法获知签名者身份。

5.7 金额隐私保护

本网络货运项目中产生的各类数据，包括交易信息，各类凭据等，是区块链隐私保护的核心对象。除了一般的密码学加密实现数据保护外，目前数据保护的研究方向在于，隐藏交易相关数据的基础上提供一定的验证信息。

### 5.7.1 累加器

本网络货运项目可使用累加器来实现身份证明。累加器是一种单项成员散列函数，密码学累加器最早由 Benaloh 和 Mare 于 1993 年提出，当时是为了解决区块链上的数据可访问性问题。累加器通过设计函数，将数据隐藏在累加器值中，同时支持证明数据是否存在于累加器。在元素加入或删除时，根据 commitment 与 membershipproofs 是否需要重新生成分为静态累加器和动态累加器。通常的实现方式有 RSA（如 BP97、CL02）、bilinearmaps（DT08、CKS09）、Merklehashtrees（Mer88、 CHKO08）等。需要注意的是，累加器需要有一个中心节点，因此适合组织向其成员颁发证明等存 在中心的场景中。以 RSA 累加器为例说明其构造过程。其流程如下：

* + 1. 累加器Accumulator创建。Setup环节选择质数g作为基底，再选两个大质数p，q，计算N=p\*q；加入元素，以添加a为例，计算root=g^a mod N；删除元素，删除a，计算root=root/g^a mod N。
    2. 成员证明。假设当前root=g^(a\_1 a\_2 a\_3 ) mod N，要证明a\_2存在与本累加器中，提供证明w=root/a\_2 =g^(a\_1 a\_3 ) mod N。
    3. 成员验证。验证者得到root，w，a\_2，计算root=w^(a\_2 ) mod N是否成立聚合证明。提供w=g^(a\_2 a\_3 )用于验证a\_2 a\_3
    4. 非成员证明。假设集合有三个元素a\_1 a\_2 a\_3，要证明a\_4不在集合内，相当于证明a\_4不是π=a\_1 a\_2 a\_3的素因子，即π，a\_4互质，只需找到一组(a，b)，使得aa\_4+bπ=1，生成证明w=(g^a，b)。在验证时，需要验证等式g^(a\_4 a) root^b=g。

### 5.7.2 同态算法

本对于需要隐藏交易金额的场合，可以同态算法实现，使用授权节点不需要知道具体的交易金额， 只需要验证输入输出相等即可。以同态加法为例，流程如下：

1. 使用同态加法隐藏金额。C=xG+aH，G，H为椭圆曲线生成元，x为随机数，a为金额。
2. 构造Σ=(C\_1^in+C\_2^in)-(C\_1^out+C\_2^out+C\_3^out)=((x\_1+x\_2)-(y\_1+y\_2+y\_3))G+((a\_1+a\_2)-(b\_1+b\_2+b\_3))H。
3. 对((x\_1+x\_2)-(y\_1+y\_2+y\_3))承诺。私钥签名k←((x\_1+x\_2)-(y\_1+y\_2+y\_3))，公钥验证Σ←((C\_1^in+C\_2^in)-(C\_1^out+C\_2^out+C\_3^out))，令H=H\_p (G)，消除利用H=γG得到γ的可能。对总数额签名，当签名验证通过，可以证明总数额没有变化，如果验证不通过，因为无法得到γ，发送方无法构造可验证的签名。
4. 构造中使用的x，a通过秘密传递给接收人，便于接收人将其花出去。

5.8 密钥生成

### 5.8.1 随机数生成

本随机数生成是多数生成密钥技术的基础，随机数具有随机性、不可预测性、不可重现性三种性质。随机数分为真随机数和伪随机数。真随机数是通过物理现象得到的，比如掷硬币、光波等，随机数具有完全的随机性、不可预测性和不可重现性，但是当前仍存在生成成本过高、速率过慢，不便于生产。伪随机数是通过可重复的计算方式产生的，目前可以在达到密码学安全要求的情况下实现低生成成本和高效率。

在本项目中，可以通过设计伪随机数生成算法完成大多数密钥的生成工作，对于核心数据，可根据需要委托第三方机构提供合适的真随机数生成系统。

伪随机数生成可以使用密码法伪随机数生成器，该生成器利用已有的加密算法生成难以预测的伪随机数列，依托于已有加密算法的安全性，具有较强的实用性。

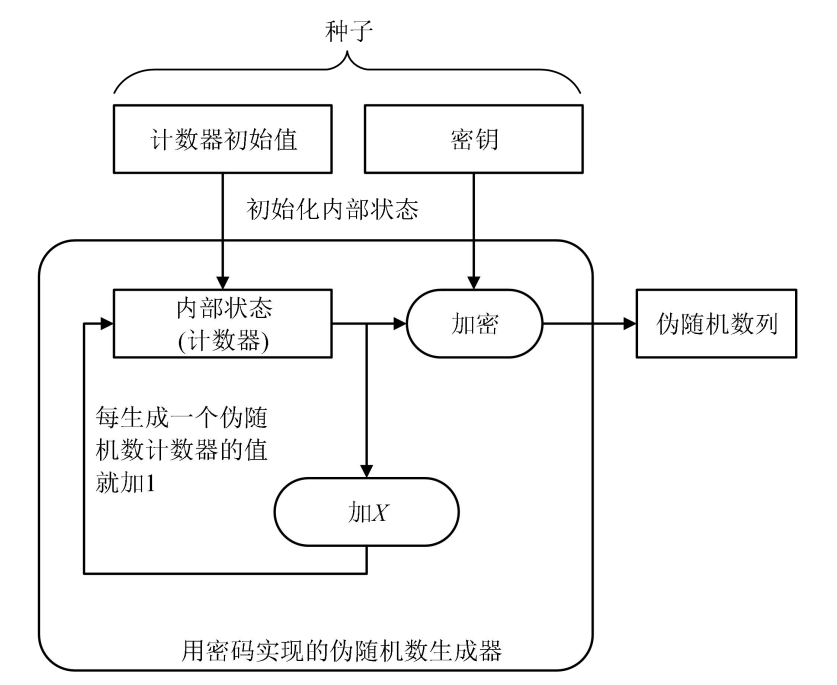


图11： 伪随机数生成机制

如图，使用初始值初始化内部状态计数器，随后使用密钥对计数器输出做加密生成伪随机数，计数器每产生一次随机数便进行自增，用于下一次随机数生成。在本伪随机数生成器中，计数器初始值和密钥共同作为种子，加密算法可以选用 AES265 等对称加密算法。

### 5.8.2 素数生成

大素数的生成及测试是密码学领域中的一个重要课题，产生素数的方法有确定性素数产生方法和概率性素数产生方法。确定性素数产生方法需要消耗大量的计算，难以实用化，但能保证产生的一定是素数。概率性素数产生方法可以在实用化的基础上产生符合安全要求的伪素数。

AKS 算法是一个确定性的素数判定算法。该算法成功将确定性素数判定算法降低到了多项式复杂度，但是消耗计算量仍然非常大。算法流程如下：

1. 对于待判断正整数 n>1
2. 如果 n 是某个数的整数次幂，那么直接返回“合数”
3. 找到最小的 r，使得 ordr(n) > log2n
4. 如果存在 a ≤ r 使得 gcd(a, n) 1orn，返回“合数”
5. 如果 n ≤ r，返回“质数”
6. 对于 a ∈ [1,ξ(r)logn]，判断 (X + a) = (xn + a)(mod(xr − 1), n) 是否成立，如果不成立返回“合数”
7. 否则返回“质数”

AKS 算法可以被用于检测一般的给定数字是否是素数，且并未依赖任何未证明猜想，是一个通用的确定性素数判定算法。

Miller-Rabin 素数测试算法是一种概率性测试方法，可以大概率地测试一个数是否为素数，同时兼顾了时间复杂度。算法流程如下：

1. 对于 0、1、2 和偶数直接判断
2. 对于待测试数 x，设 s、奇数 t，满足 2s t = x − 1
3. 取一个较小地素数 a，计算 a^tmodx, a^(2∗t)modx, a^2\*2∗tmodx, …, a^2\*s∗tSmodx, 判断结果是否全为 1，或者非最后一个数的情况下出现 p − 1
4. 如果都满足，则认为 x 为素数

Miller-Rabin 素数测试算法依赖于费马小定理和二次探测定理，可以通过选取特定的 a 以及选取不同的 a 多次尝试，提高检测准确率。

本货运项目可根据实际业务量决定素数生成算法。

### 5.8.3 组协商密钥

本对于需要协商加密的场合，如多方交易、多方签名等，可以使用组密钥协商技术获得一个共同的密钥。日前比较常用的有 DH 密钥协商算法，能够以较轻的代价实现密钥配送，不过易收到中间人攻击，除此之外还可以选用 BD 组密钥协商协议，BD 协议基于 DDH 假设实现安全性，可以实现多节点组密钥协商。BD 协议基于 DDH 假设实现安全性，可以实现多节点组密钥协商，步骤如下：

1. 各参与者随机选取，广播。
2. 各参与者收到其他参与者的广播后，计算并广播。
3. 各参与方计算密钥。，其中，各参与者绕成一个圈。表示i向前移动一位；表示绕了一圈，一直移动到初始值的前一位，由于参与者围成一个圈，，最终

对于 BD-GKA，如果缺少任一参与方，计算链都会中断，所以鲁棒性低。Stanislaw 等人于 Flex- ibleRobustGroupKeyAgreement 一文中对鲁棒性和计算复杂度都进行了优化。在本项目中，可以根据实际场合的安全性要求进行协商算法选用。

### 5.8.4 基于区块链的密钥管理系统

本我们提供了一个基于区块链的密钥管理系统可供参考。该系统管理密钥的生成，传输，启用，吊销，使用，更新，备份，恢复。管理内容包括：用户通过秘密分享向管理部门注册密钥；用户基于注册密钥生成临时会话密钥用于通信，并使用注册密钥签名以证明有效性；管理部门通过密钥哈希实施有效期管理；用户使用累加器证明所有权，管理部门通过阈值签名向申请更新方或申请恢复方提供资格证明。采用上述方式，可以实现联盟链中完整的密钥生命周期管理，降低联盟链用户之间加密通信的复杂度，并且具有较高的安全性。

5.9 分片技术

传统数据库解决数据存储扩展的有效方式是分片，将数据库分割为多个部分并分别存放在不同的服务器，从而达到横向扩展能力，区块链也可将其视为一种数据库，于是也可以采取这种分而治之的思想模式。在分片系统中的节点和相应的交易都会被划分到不同分片中从而提升了扩展性和处理性能，因为每个节点无需处理全部交易，只需要处理它分配到的一部分交易，并且可以和在其他分片中的节点并行执行多线程处理交易。由于对网络进行分片后达到了并行处理的效果，节点无需排队等待验证，而是可以同时验证多个区块，极大地提高了吞吐量和效率。当前分片有如下三种含义：

1. 网络分片，网络分片是指将全网分为不同的片区，每个片区由一个对应的委员会处理，每个委员会内部成员大部分时间只需内部通信，每个片区内部的其他客户端、节点大部分时间可以通过与该分片内委员会通信获得目前区块链的状态。

2. 交易分片，交易分片是指每个分片委员会只负责处理其对应的交易，如根据交易的 ID 判断其对应的分片，交易 ID 最末位数字如果是 i，则由 i 号分片委员会处理该交易，对交易运行委员会内分布式一致性算法，验证该交易的合法性，决定该交易是否被添加到区块链中。计算分片使不同的交易以并行的形式被不同的委员会处理，当网络中节点数量增多时，可以增加更多的委员会，这样不同的交易能够以并行的形式被不同的委员会同时处理，交易处理性能随着网络中节点数量的增多而增加，进而实现了交易处理的可扩展性。

3. 状态分片，状态分片是指不同分片委员会将处理后的交易分片存储，每个分片委员会只负责处理本分片对应的交易，将交易放到本分片专属的交易区块链上。交易区块链用于存储本分片产生的交易历史或当前分片的未花费交易池信息。存储分片将整个区块链系统的交易数据或未花费的交易输岀 (unspenttransactionoutput，UTXO) 数据分片存储，降低了节点的存储负担。

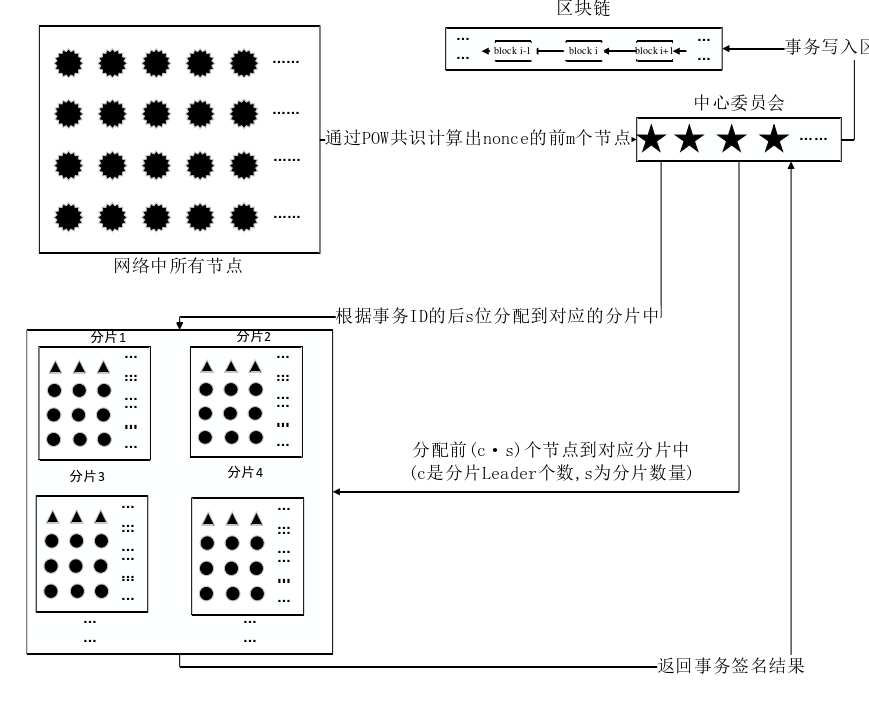


图11： 分片机制

5.10 跨链技术

本跨链本质是将一条链上的消息安全可信地转移到另一条链上并在该链上达到预期效果。公证人机制、侧链/中继机制、哈希锁定机制三种技术各有侧重，公证人机制跨链方式比较简单和单一，只支持资产交换；侧链/中继机制更加稳定和易于扩展，可以实现资产转移，数据交换，或者其他更加复杂的需求；哈希锁定机制只适合偏资产或者关键数据的交换，在支付领域应用较多。在这些跨链技术的基础上，目前已涌现出非常多的跨链项目，致力于推动区块链网络的发展和完善，实现区块链互联互通。目前主流的区块链跨链技术主要包括公证人机制、中继机制和哈希锁定。

### 5.10.1 公证人机制

本公证人机制是一种最简单实现跨链的方法。通过引入一个或多个可信实体作为“公证人”，完成数据收集和交易验证。在跨链过程中，可信实体向一条链声明另一条链上发生了某事件，为跨链行为做背书。这些可信实体既可以不断监听链上事件，达到条件后自动地做出响应；又可以在收到请求时才对某些消息进行签名。根据公证人类型主要分成两类：多签名公证人机制、分布式签名公证人机制。

多签名公证人机制，顾名思义，是由多个公证人组成的公证人小组，组内每个人都有一个属于自己的私钥，跨链时需要一定数量的签名达成共识，交易才能完成。这种方式与单签名公证人机制

比起来更加安全和稳定，但同时存在一定局限性，需要跨链双方链上都要支持多重签名功能，有一定的实现难度。

分布式签名公证人机制实现起来相对比较复杂，该机制借鉴安全多方计算的思想，基本思路是整个系统只产生一个私钥，然后将私钥拆散成多个碎片，并进行加密处理后随机分配给各个公证人，公证人之间互相不知道对方的密钥。跨链时只有达到一定比例的公证人签名才能复原出完整私钥，验证完成跨链交易。这种方式更加安全，当少数公证人遭受攻击时，不会对系统稳定性造成影响。

### 5.10.2 侧链/中继链机制

本侧链是一条区块链（主链）将另一条区块链作为自身的延伸（侧链），主链维护一个资产分类 账本，并连接到侧链，侧链是通过跨链通信协议连接到主链的独立系统。这里侧链不一定是次要的， 两条链可以成为彼此的侧链。最常见的是主链与侧链之间资产转移：当用户想从主链向侧链转移资产时，首先在主链上向一个特殊地址发送 X 个令牌，这些资金即被锁定在主链上，同时相应数量通证被创建在侧链上，于是完成跨链资产转移，用户就可以使用侧链上的令牌。中继机制是侧链与公证人机制的一种融合。与公证人机制相比，中继机制用中继链代替原先的公证人，接管公证人的任务，是一种去中心化的公证机制。中继链是一条功能完备的区块链，可以读取并验证连接到其上的区块链数据信息，中继机制可以说是一种更加直接地实现跨链的方法。中继机制可以在不依赖可信第三方的情况下完成交易验证，是一种更加灵活、更易于扩展的跨链技术，实现形式多种多样，如Cosmos 中的 Hub、Polkadot 中的中继链等都充当中继的角色，一些侧链的实现也采用了中继模式。

### 5.10.3 哈希锁定

本哈希锁定最初是作为集中式交换的替代品出现的，支持跨链原子操作，即允许一个用户向另一个用户发送一定数量的加密货币，交换另一个区块链上持有的加密货币。哈希锁定主要利用哈希锁和时间锁技术，不需要两条链之间过多交互，不需要第三方公证人存在，通过交易者承诺在超时前向另一方提供加密证明来进行交易。

### 5.10.4 设计方案

本针对本项目的实现，我们决定采用侧链方案来完成项目的跨链需求。现有的方案在应用之前，介于应用场景不同，以及公链与联盟链的架构不同，还需要再进行一些针对 Fabric 的优化。首先我们来看 Fabric 的跨链方式。

fabric 的启动过程：创建证书，生成创世区块，通道配置交易块，创建通道，节点加入通道，安装链码，实例化链码，链码的调用。这个是完整的生命周期。一个节点上可以安装多个 chaincode， 且每个 chaincode 是一个账本。同一个通道中，所有的节点安装的是相同的 chaincode, 所以每个节点都有完整的数据，不存在跨链之说。综上，跨链是指跨 channel，因为不同的 channel 拥有不同的账本，跨链的本质是把一个链上的数据转移到另外一条链上。跨链既可以在上层来做，也可以在chaincode 层进行。

所以放在 Farbric 上，我们需要建立一条专门用于跨链消息转发的 channel，作为我们的”中继链“，暂且称呼为 Xchannel。应用的所有的跨链交易都通过 Xchannel 进行转发，接收。从而应用了中继链方案。

5.11 国密算法

本国密算法是国家通用密码算法的简称，是国家密码管理局制定的自主可控的国产算法。国密算法和商用算法一样，可分为对称密码算法、非对称密码算法、哈希密码算法三类，包括 SM1、SM2、SM3、SM4、SM7、SM9、祖冲之密码算法（ZUC) 等。



图13：国密算法

密码技术是 Fabric 平台实现的核心技术之一，是 Fabric 上层应用开发的基础支撑技术。基于密码技术，Fabric 实现了一种基于证书的身份管理与一种“执行-排序-验证”并行执行的新交易模式，保证了对链上的每一步操作进行严格的身份与权限验证，进而保证该平台上数据的安全与不可篡改。但目前原生的 Fabric 平台仅支持通用的国际密码算法，缺乏对国家密码算法（简称国密算法） 支持。同时，我国政府在技术监管和自主可控方面对从事区块链系统建设、服务运营的机构提出了国密算法规范要求 [6]，因此，将我国自主研发的国产密码算法嵌入 Fabric 平台显得尤为必要。目前，国密算法 SM2、SM3 的可证明安全性已经达到了密码算法的最高安全级别，其实现效率相当或略高于高级标准的密码算法，相较于比特币、超级账本等现有区块链架构选用的国际通用密码算法ECC、SHA-256，拥有安全、稳定、高效等优势。目前各主流区块链公司都有尝试将国密应用在区 块链上，这些工作主要集中在：

1. 用 SM3 代替 SHA-256 等作为密码杂凑 Hash 算法；
2. 用 SM2 替代 ECDSA 签名算法；
3. 用 SM2 证书代替 RSA、ECDSA 证书。

Fabric 国密改造项目主要包括三个部分：Fabric、Fabric-CA 和 FabricSDK。根据改造需求分析，Fabric 国密改造方案由三个部分构成：

1. 国密算法基础库收集与改造
2. Fabric 本体改造
3. FabricSDK 改造

限于篇幅，本节只给出国密改造的一般性思路，目前 Fabric 国密改造的具体方案可以参照Fabric 国密项目组（https://github.com/Hyperledger-TWGC/fabric-gm-wiki/wiki）。密算法嵌入设计思路如下图所示，主要工作有以下 4 个部分：

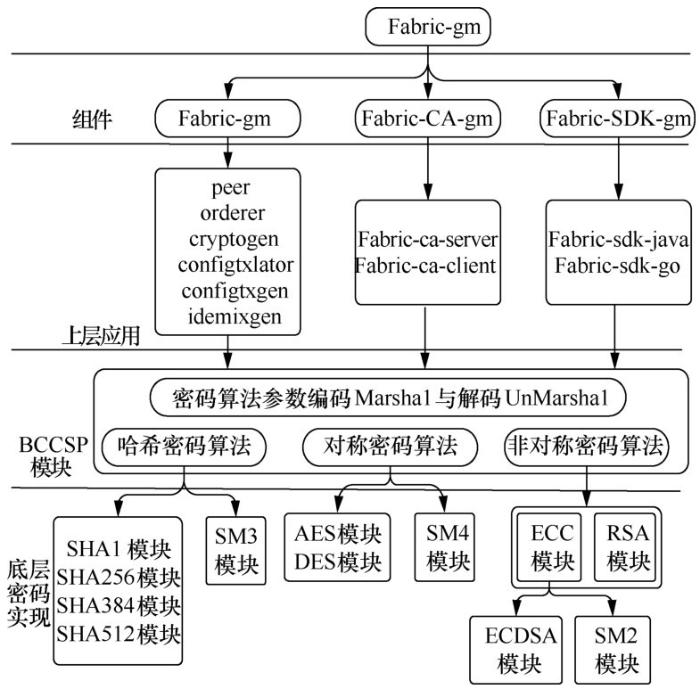


图14：国密算法改造示意图

1. 基于 Go 标准的底层国密算法嵌入：可依据 SM 系列国密算法的标准规范文件，设计并实现基于 Go 标准的国密算法，包括常用的加解密算法（SM4、SM2）、哈希算法（SM3）以及签名验签算法（SM2）等。在这部分工作中，本文对开源实现（指同济区块链研究院对国密 SM2、SM3 和 SM4 算法的开源实现）的底层国密算法实例 [19] 进行了正确性验证，并在 BCCSP 模块的国密实例中采用了该算法实例。
2. BCCSP 模块的国密算法接口嵌入：分别在 BCCSP 模块的哈希算法子模块、对称密码算法子模块和非对称密码算法子模块下，添加相关的国密 SM3 模块、SM4 模块和 SM2 模块的调用接口。
3. 上层应用的国密算法接口嵌入：在上层程序与应用中，添加相关的国密算法接口调用，将原哈希、签名验签等密码算法接口与国密 SM3、SM2 算法接口关联，以实现上层应用对国密算法调用的支持。
4. 对嵌入国密算法及相关调用接口后的 Fabric 平台源码进行编译，形成支持国密算法的可执行文件，并打包成 Fabric-gm 镜像文件，部署支持国密算法的 Fabric-gm 平台，通过基于国密接口的加解密、签名验签及哈希对比等实验验证所嵌入的国密算法的正确性、可用性和有效性

5.12 数据归档

本结合我司的网络货运物流场景，我们提出在区块链系统上构建支持大规模可查询可恢复的高效数据归档方案。首先结合业务场景明确以下数据归档需求：

1. 数据可灵活性地转移到独立设备存储；定时归档：最大归档高度，归档标识，归档区块区间，归档触发条件归档内容转移：归档的内容从链上转移到链下数据库，链上内容删除。
2. 归档数据应确保隐私安全性，可保证可查询性；Block。超过最大归档高度等。
3. 归档数据可以再恢复到区块链接点中。

结合上述数据归档需求，区块链数据归档流程可设计如下：

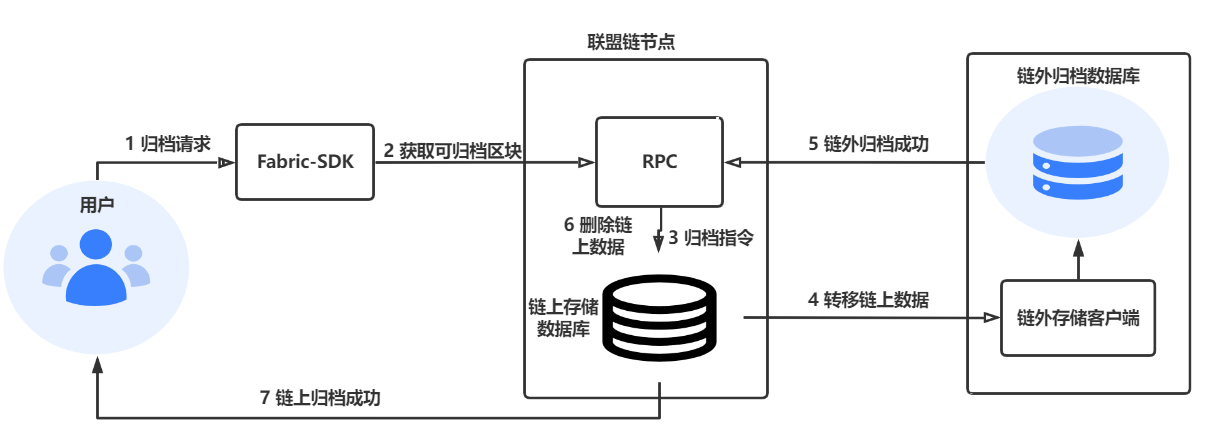


图15：数据归档流程图

1. 用户端向 Fabric-SDK 发起归档请求，归档请求中包含用户签名及各种归档参数，包括最大归档高度，待归档区块区间等定时归档：最大归档高度，归档标识，归档区块区间，归档触发条件归档内容转移：归档的内容从链上转移到链下数据库，链上内容删除
2. Fabric-SDK 通过联盟链节点 RPC 获取可归档区块；
3. RPC 解析归档请求，向链上存储数据库发起归档至零；
4. 节点向链外存储数据库转移待归档数据，链外归档数据库对待归档数据进行存储；
5. 链外存储数据库存储归档数据完成后，向联盟链节点 RPC 发送链外归档请求信息；
6. 联盟节点 RPC 收到链外归档成功后，移出链上数据，同时生成并链接虚拟创世块；
7. 归档完成后，联盟链节点通知用户链上归档成功。

归档具体原理如下图所示，首先联盟链节点定位待归档数据块区间 [m,m’]，归档区间的基本单位是数据块，但在大规模数据场景下也可设定一定数据块的集合切片作为基本归档单位。然后，节点构造虚拟创世块，虚拟创世块中包含待归档数据的索引和由待归档数据块的区块头 ID 组织成的MerkleTree，虚拟创世块提供两方面作用，一方面上述索引提供了已归档数据块中交易内容的快速可查询性，而不单纯依赖归档数据库的键值查询，另一方面 MerkleTree 中存有所有归档区块的区块头，这确保了即使归档数据被归档到链外数据库，数据有效性不会受到安全性影响。最后，虚拟创世块被链接到上述归档区块的位置上。需要指出的是，区块链本身依然具有不可篡改性，Fabric 也不例外，此处为了保证安全高效地对链上数据归档，引入了虚拟创世块，并链接到原来的归档数据块位置，此处似乎破坏了区块链的不可篡改性，但实际上，虚拟创世块在生成的过程中，引入了归档区块的下一区块头作为哈希依据，这保证了后向链接性；同时，虚拟创世块的生成依然需要联盟节点的共识，这也就保证了后向链接性，故即使虚拟创世块被链接到已归档区块的位置，区块链的不可篡改性依然没有被破坏，即归档并不会对联盟链本身的安全性并未造成损害。归档后的链上数据由高速数据库转移到其他数据库进行整合，并采用密码技术进行隐私保护。归档后的数据依旧可可提供全链可查询性，查询操作无需通过区块链模块完成，具体而言，当用户需要查询过去交易数据，首先系统会先根据查询请求判断该交易数据所在区块是否已归档，如果未归档，即交易数据依旧存在链上，系统通过区块链模块直接获取交易数据。如果由于数据累积，导致过去的交易数据已归档，系统将通过链下归档数据库获取交易数据。

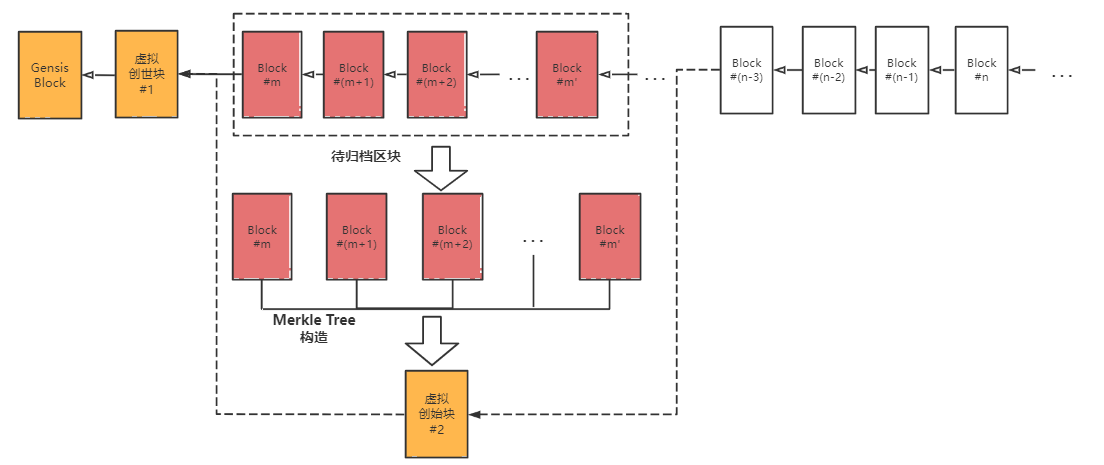


图16：数据归档原理示意图

归档后的链上数据由高速数据库转移到其他数据库进行整合，并采用密码技术进行隐私保护。归档后的数据依旧可可提供全链可查询性，查询操作无需通过区块链模块完成，具体而言，当用户需要查询过去交易数据，首先系统会先根据查询请求判断该交易数据所在区块是否已归档，如果未归档，即交易数据依旧存在链上，系统通过区块链模块直接获取交易数据。如果由于数据累积，导致过去的交易数据已归档，系统将通过链下归档数据库获取交易数据。

# 6、区块链融合架构

本区块链技术的现有的大部分架构都基于公有链来进行设计，基本需要基于大量的节点资源，这些节点资源分散繁多，管理比较困难。而针对本项目中的技术选型来说，在联盟链场景下，我们可以对其进行优化，利用云计算技术来实现节点资源的有效管理。通过将云计算技术和区块链技术进行深度融合，实现对区块链节点、框架、网络等虚拟化的表达，实现节点资源动态扩容、远程运维， 最终实现了区块链系统的快速部署、快捷运维和敏捷开发等目标。通过构建虚拟全节点，在不影响系统正常运行的基础上，通过配置监管策略，自动完成系统安全性合规性的穿透式柔性监管，为区块链行业应用提供安全和技术保障，为区块链技术的发展提供强有力的援助和支持。实现区块链快速应用的主要关键技术包括：

* 1. 将区块链技术与云计算技术深度融合，支持以 VM、Docker、Kubernetes 集群等方式实现区块链节点、框架、网络的全虚拟化表达，支持动态扩容、远程运维，从而达成快速部署、快捷运维和敏捷开发的目标。对于货运物流业来说，在货运物流平台上有多少个参与方，就会有多少个节点，这些节点加入到货运物流区块链网络中，对整个网络都是一个非常大的挑战，我们需要花费大量的时间成本去对每一个节点进行监控和维护。此外，这些节点在区块链网络中，会慢慢的产生大量的数据和日志，并且数据的增量是不可预计的，在这样的情况下，本地资源永远是有限的，只有云资源才可以不断的扩展，将区块链与云计算融合，借助云平台的容错可扩展、网络的多链路负载、计算资源的动态调整，我们可以针对区块链中的节点能够更好的实现节点的服务，从一定角度来讲也节省了节点的运行成本，提高了整个系统之间交互的效率
  2. 支持区块链节点的追踪与可视化：该项技术研究可以为货运物流平台的运维人员提供非常便利且高效的区块链网络和节点运维服务，它将各项监控指标和追踪内容抽离成标准的监管模块， 之后将标准监管模块放置于每一个区块链节点中，标准监管模块每隔一段时间自动搜集指标数据，搜集完全后通过 WebAPI 服务提供给前端可视化界面进行展示，运维人员能够根据每一节点的指标变化对该节点的运行现状作出评判。
  3. 支持联盟链穿透式监管：联盟链穿透式监管的主要原则是将联盟链网络中的每一个环节产生的真实可信的数据串联起来实时的上传到监管节点，以便于提高监管节点的效率，降低监管节点的成本。联盟链网络中每一个环节产生的数据都是大量的，随着时间的流逝，这些数据会变得异常庞大，因此，结合货运物流的行业实际进行区块链式的分析，研究什么数据需要上传到监管节点，要以怎样的方式上传到监管节点。
  4. 实现区块链环境的一键式快速部署与应用，且安全、合规地部署区块链应用，简化区块链部署、运维和管理等环节：研究区块链环境的一键式部署是实现区块链商业化，促进区块链落地应用发展的重要内容之一，基于 Docker 容器技术的发展与 Shell 脚本编程，将区块链网络的各节点抽离成标准节点模板，之后将标准的节点模板镜像化容器化，放置于安全的可控的仓库，当需要部署网络时，只需要配置脚本需求，就可以安全高效的部署区块链网络。

HyperledgerFabric 区块链系统采用容器式运行的方式，其系统中的智能合约称为 Chaincode。Chaincode 分为特殊的系统 Chaincode 和交易 Chaincode，前者负责区块链的管理，后者负责保存状态和账本数据，并执行交易。

Chaincode 生命周期分为 5 个阶段：打包、安装、实例化、升级、删除。Chaincode 通过 API

与区块链中的各种节点进行交互，同时也可以通过 API 对 Chaincode 的生命周期进行管理。

打包过程包括创建包和包的签名，源码按照部署规范格式打包，签名主要用于检查和确认 Chain- code 所有者，可以在创建包的同时进行签名，一次签名的包用执行 install 交易，多次签名包围多个所有者一次签名。安装过程需指定 CDS 包的路径，发送一条 SignedProposal 消息给 peer 节点的生命周期系统节点调用 LSCC 上的 install 方法完成 Chaincode 安装。实例化过程调用 LSCC，在channel（Ledger 上包含特定 peer 节点的私有链）上启动一个 Chaincode 容器，实现 Chaincode 与channel 的绑定。实例化交易执行过程中，验证 Chaincode 的实例化策略，以确保实例化交易执行的合法性。实例化成功后，Chaincode 即处于激活状态，时刻监听并接受交易请求。升级过程类似于实例化过程，即修改新的 Chaincode 版本并与 channel 绑定。为保证升级该 Chaincode 的合法性，该过程须验证当前版本的实例化策略。删除过程只需删除对应的容器，同时删除每个安装的 Chaincode 的背书节点上的 SingedCDS。

编写 Chaincode 智能合约时需要实现 Chiancode 接口，以响应传来的交易消息，Init 和 Invoke是两个必须接口，分别实现智能合约的部署（实例化、升级等）和交易调用。

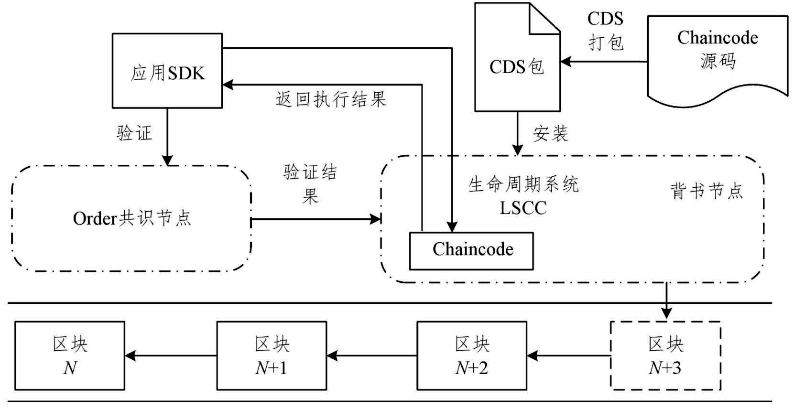


图17：chaincode的部署和执行流程图

Chaincode 在超级账本上安装部署及运行流程如上图所示，具体过程如下：

1. Chaincode 源码以 CDS 规范签名生成 CDS 包；
2. 通过生命周期系统 Chaincode，将 CDS 包安装在同一通道内的背书节点上，生成运行在节点上的 Chaincode；
3. 应用程序通过 SDK 发送请求到背书节点；
4. 节点通过 Chaincode 执行交易并将执行结果返回给应用程序；
5. 应用程序搜集结果，将结果发送给 Order 共识服务节点；
6. 共识服务节点执行共识过程并生成区块验证结果；
7. 背书节点各自验证交易并提交到超级账本区块中。