



面向多元场景的共识机制 性能分析



目录



- 一致性与共识
- 现有区块链共识机制对比
- 区块链的性能指标与优化
- 北大数瑞随机共识

一致性(Consistency)与共识(Consensus)

数据一致性：数据在多份副本中存储时，各副本中的数据是一致的

事务一致性：操作序列在多个服务节点中执行的顺序是一致

从分布式账本的角度，一般通过事务的一致性来实现“数据一致性”

	定义
严格一致性	对x的任何读操作返回对x的最近写操作的结果
顺序一致性	所有进程以一定顺序执行，每一个进程的操作都以程序规定顺序出现
因果一致性	并发写操作在不同机器顺序可能不同
最终一致性	同步完成后，共享数据才一致

币交易的需求

实现并发/并行的基本需求

初始状态：x=20; y=20	
if (x>10){ y=5; }	if (y>10){ x=5; }
结果：x=5; y=5 一种符合顺序一致性的序列但不 满足 “币交易场景”	

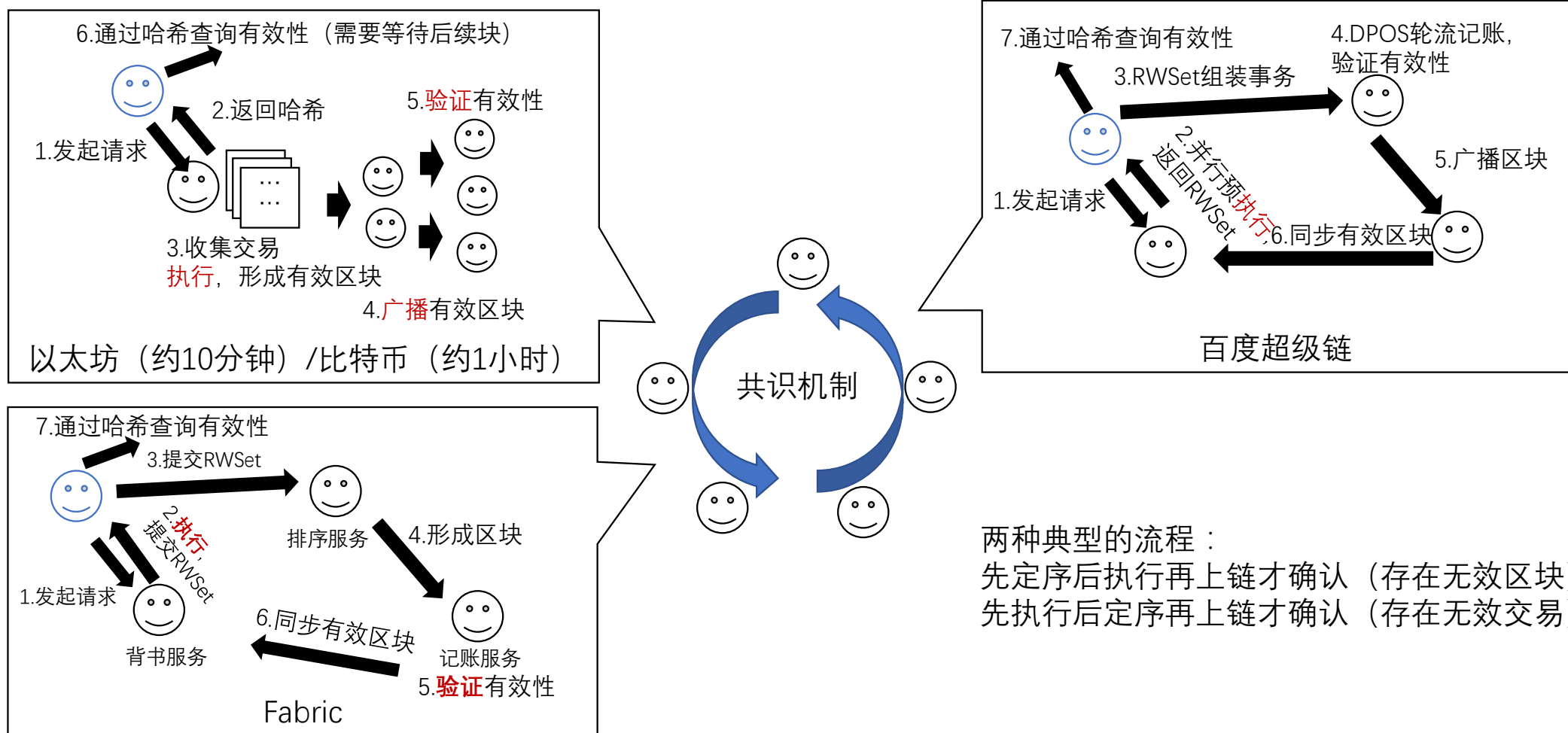
共识是描述进程间相互协作的过程，经过这个过程来实现某种一致性。



目录

- 一致性与共识
- 现有区块链共识机制对比
- 区块链的性能指标与优化
- 北大数瑞随机共识

典型的智能合约执行流程与开销



现有区块链平台共识过程的比较

通过多节点保存合约状态数据来实现信任。主要关注**执行结果的正确性**和**可见顺序的一致性**。

比特币	以太坊	Fabric(只考虑单Channel)	百度超级链 (DPOS)
支持并发发请求			
矿工节点弱一致 执行	矿工节点弱一致 执行	背书节点弱一致 执行	由有权节点，多核 执行
矿工节点有效区块广播	矿工节点有效区块广播	排序节点进行严格一致产块，后广播	产生区块后广播
矿工节点再次顺序 执行	其他矿工节点是顺序 执行	记账节点是顺序 验证 读写集	其他节点 验证
上链时间： 运气好约10分钟	上链时间： gas付高，约2分钟	上链时间： 发请求+执行+定序+广播+验证+查询有效性	上链时间： 发请求+定序+执行+广播+验证+查询有效性
确认时间：后续6个块	确认时间：后续40个块	确认时间：上链即确认	确认时间：上链即确认
信任来源：由于账本全网同步，6个区块以后被改概率极小	信任来源：由于账本全网同步，40个区块以后被改概率极小	信任来源：由于背书节点背书+排序节点定序+执行结果多个节点存储，被改概率极小	信任来源：由于通过投票选出了有权节点，结果被多个节点存储，被改概率极小

现有区块链平台的比较

对比项目	比特币	以太坊	Fabric(只考虑单Channel)	百度超级链
共识机制	PoW	PoW→PoS	可配置	可配置
是否可扩展	不可扩展	不可扩展	不可扩展	不可扩展
TPS	<10TPS	<100TPS	数千TPS	8.7万TPS
在何阶段并行	矿工并行产块	矿工并行产块	可并行生成读写集 校验有效性串行	可并行生成读写集
说明	通过控制难度来控制TPS，可能产生分叉，分叉则为 无效区块 ，10分钟产生一个区块	刚发布时是以PoW为共识，目标是PoS。可能产生分叉，分叉则为 无效区块 ，当前产块效率约10秒1个区块	读写集提供了一种弱一致性的执行； 当存在冲突时，会有 失效交易	

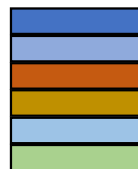


目录

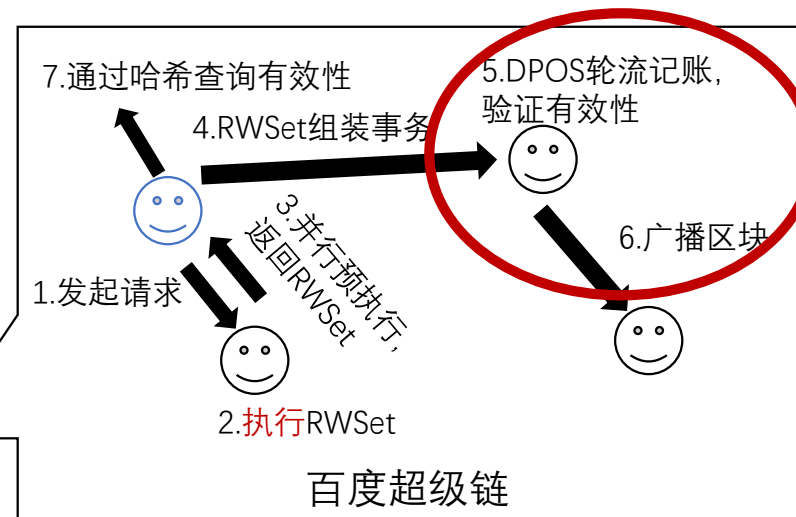
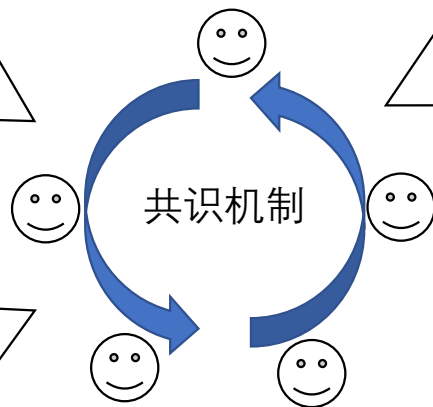
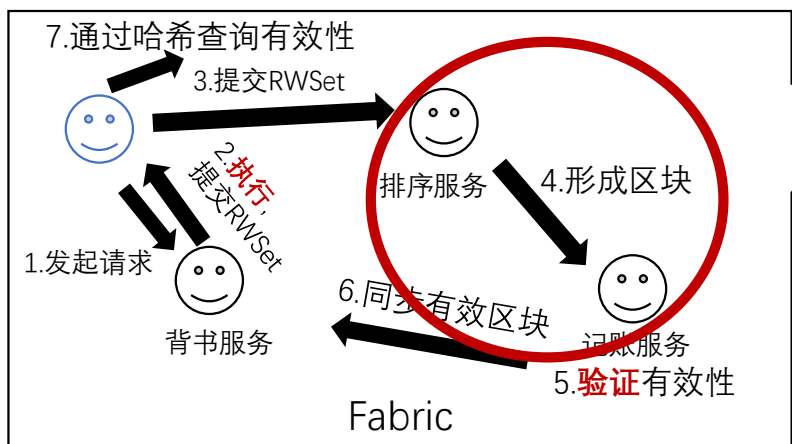
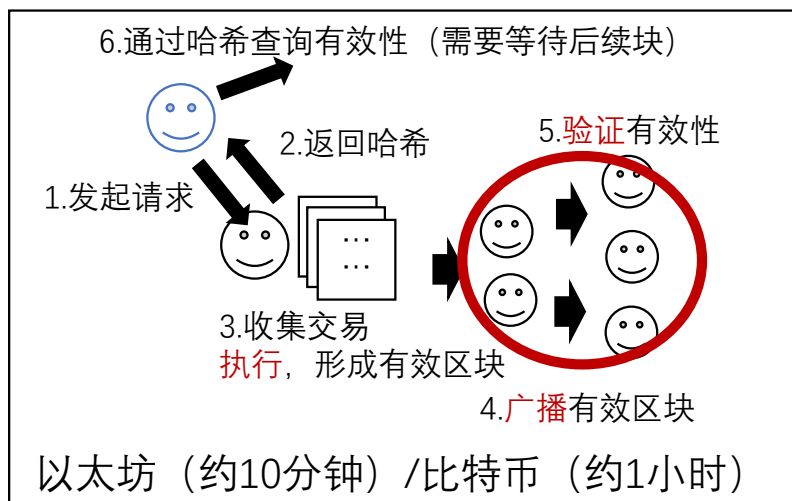


- 一致性与共识
- 现有区块链共识机制对比
- 区块链的性能指标与优化
- 北大数瑞随机共识

区块链的性能指标（写入TPS）



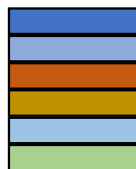
☺ 无冲突交易：顺序不影响交易是否成功



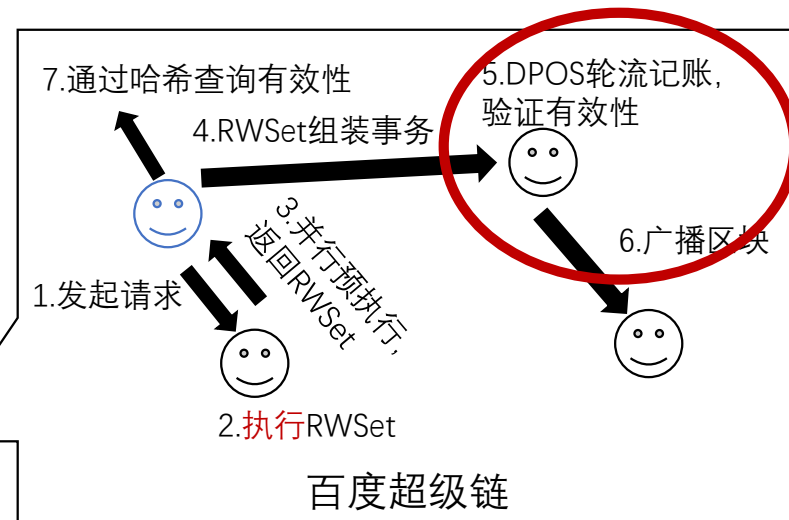
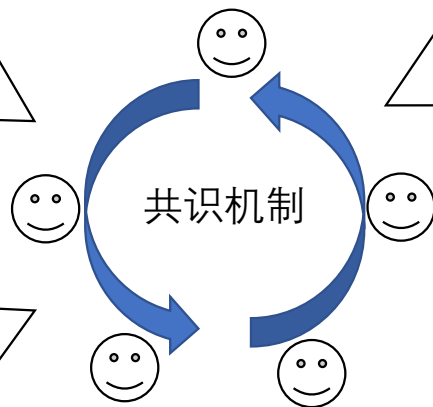
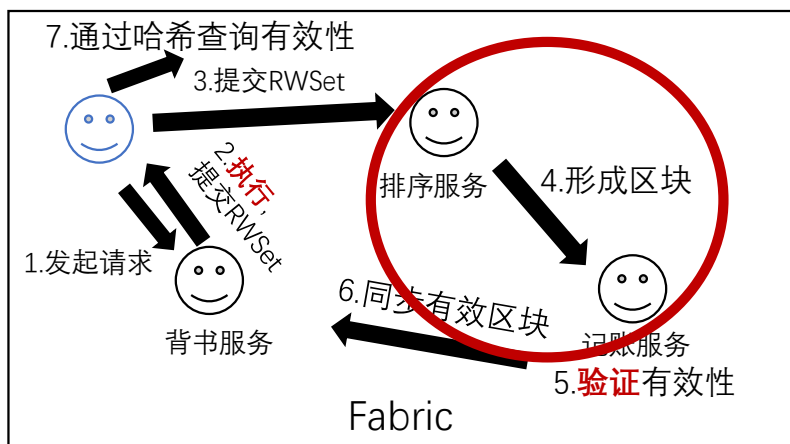
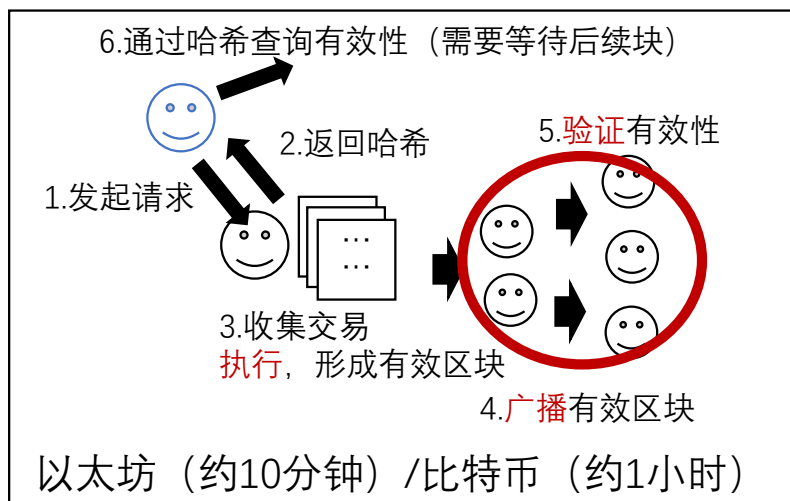
一些不合适的比较：

- Kafka消息队列 vs BFT定序算法
- 无冲突交易的性能 vs 有冲突交易性能 vs 实际运行的TPS

区块链的性能指标（写入TPS）



☺ 无冲突交易：顺序不影响交易是否成功

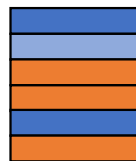
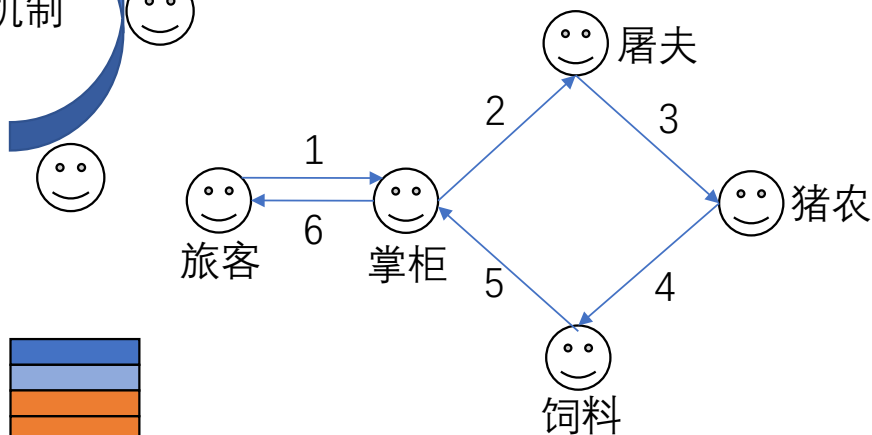
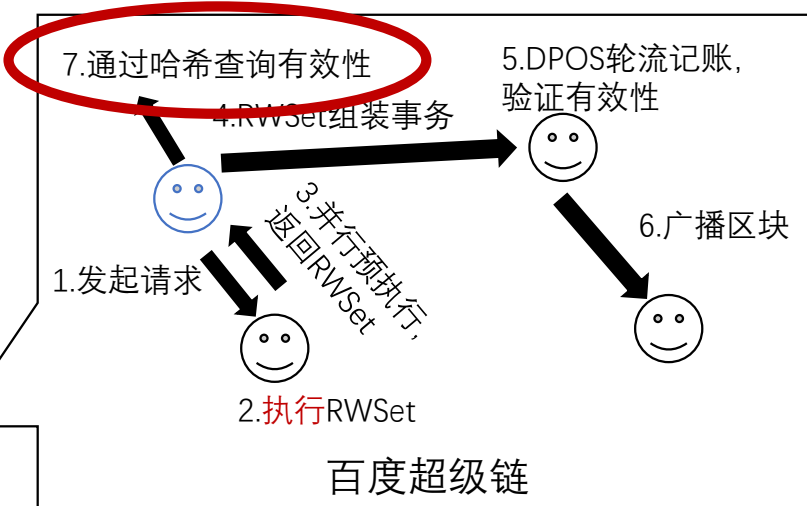
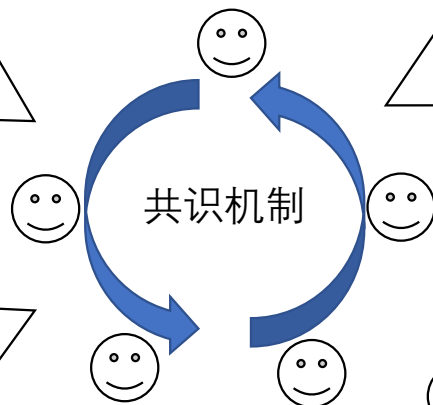
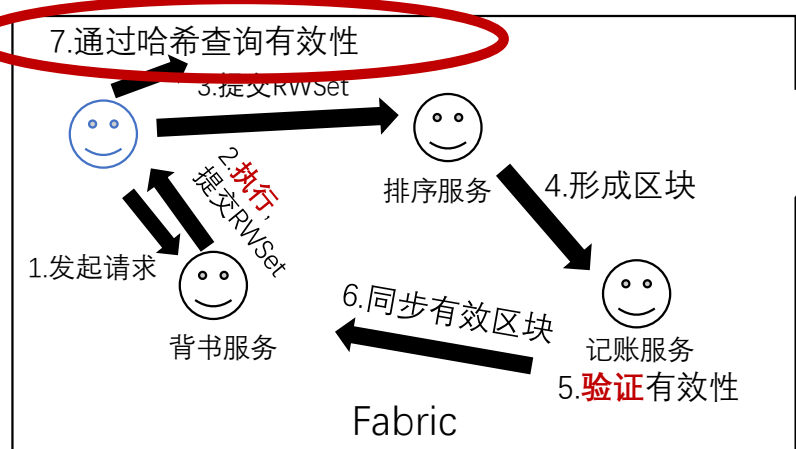
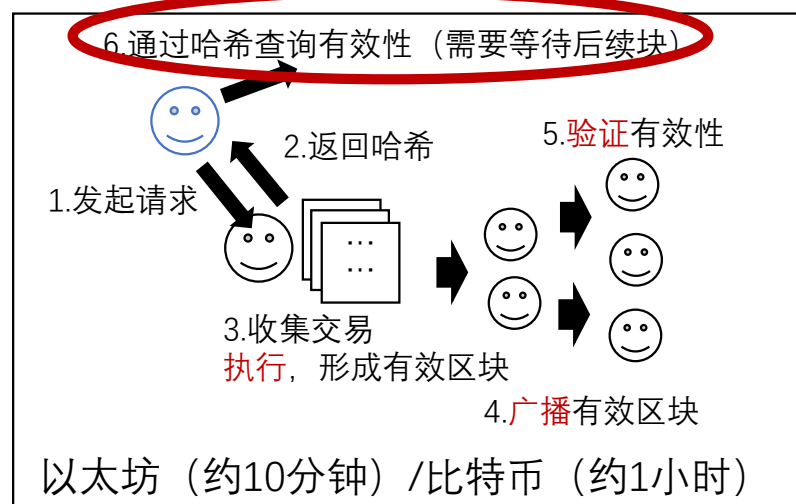


优化思路：在**无冲突交易**场景下

- 并行地计算智能合约
- 更高效的排序算法
- 并行/并发地有效性校验
- 更快的同步策略

区块链的性能指标 (TPS)

有冲突交易场景下，
写后读的延迟导致的性能下降



有冲突交易的例子：请问旅客要上楼看多久？



目录

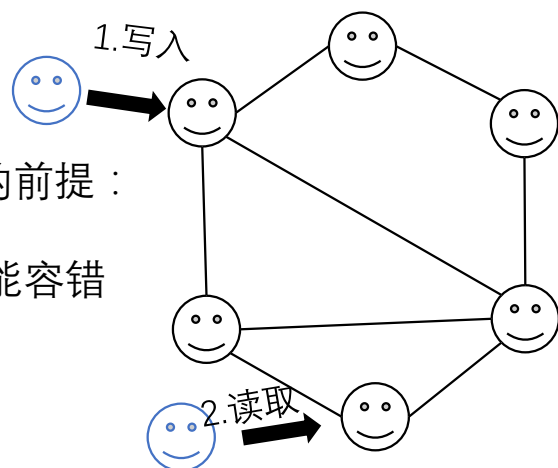


- 一致性与共识
- 现有区块链共识机制对比
- 区块链的性能指标与优化
- 北大数瑞随机共识

北大数瑞随机共识（无冲突交易→存证场景）

可信存证场景的前提：

- 篡改难
- 数据不丢、能容错

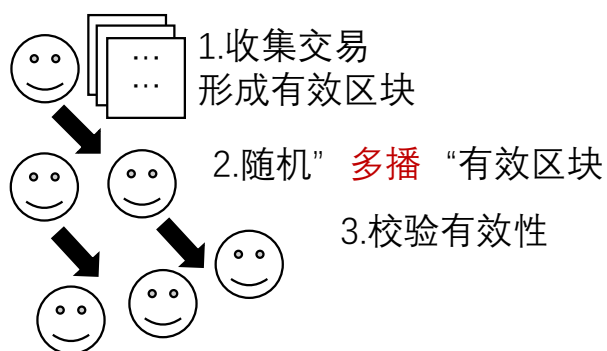


思路：定序，每节点都存一份
权衡的重点：

- 需要“分片”机制
- 需要解决区块同步的“写入”与“读取”的效率

北大数瑞随机共识机制的重点：

- 着重在突破不同分片间（或是跨链）的**读取**
- 如何实现**错误的检测与恢复**



北大数瑞随机共识：写入过程示例

北大数瑞随机共识：

可配置项：

- 以区块为单位进行分片/以时间为单位进行分片
- 节点分片数量
- 节点分片的阶段

写入：可扩展，防篡改

互联网环境100节点实测100000+TPS

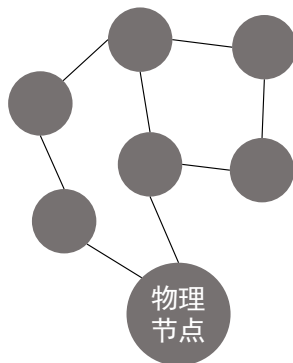
读取：秒级查询与统计

- 线性扩展：单笔存证开销 $K*(L+1)$ ，与N无关（1000TPS/阿里云标准虚机）
- 防篡改：区块随机分布，增大攻击难度（后继5个区块时，随机控制90%的节点，篡改成功率0.06%）

设：Y为区块覆盖的节点数量期望
M为待篡改区块的后继区块数量
J为随机攻击并成功控制的节点数量
P为篡改成功率

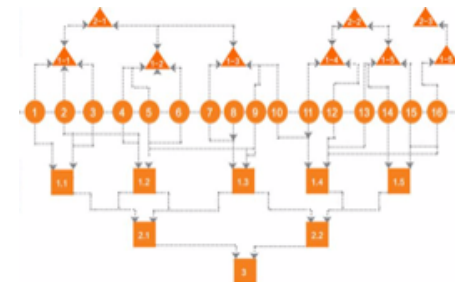
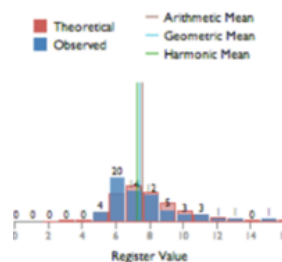
$$K * (M + 1) * (L + 1) = \sum_{i=0}^{Y-1} \frac{N}{N-i} \approx N * \ln \frac{N}{N-Y+1}$$

$$P = \frac{C_{N-Y}^{J-Y}}{C_N^J}$$



网络延迟50ms
单点查询50ms
网络带宽10Mbps
N:节点总数
K:见证节点数
L:共识节点数

- 高效查询：开销 $\log N$ 跳（10000节点仅需5跳）
- 高可用：改进KAD生成树，可容忍N/3个节点失效
- 高效统计：LogLogCounting，传输压缩率 $\frac{\log(\log D)}{D}$ （GB级原始数据，压缩后统计数据为KB级）



北大数瑞随机共识（有冲突交易→可信计算场景）

智能合约定义的演变：

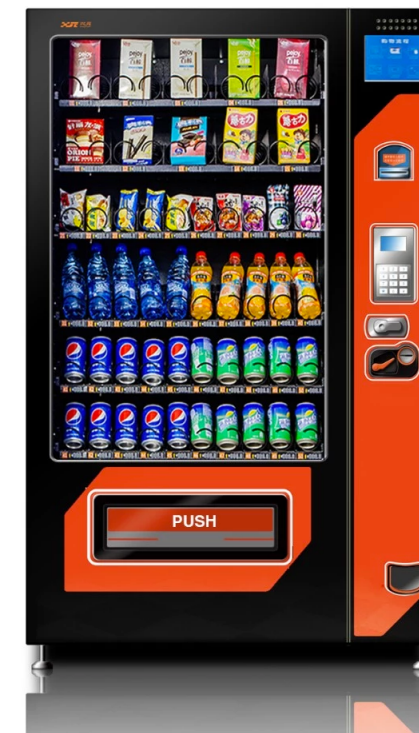
Nick Szabo: a computerized transaction algorithm, which performs the terms of the contract. (1994)

Gideon Greenspan: A smart contract is a piece of code which is stored on an Blockchain, triggered by Blockchain transactions, and which reads and writes data in that Blockchain' s database (2016)

...

智能合约的特点：

- 1) 支持有状态的计算
- 2) 可以自动执行
- 3) 结果被参与各方认可
- 4) 无需可信第三方

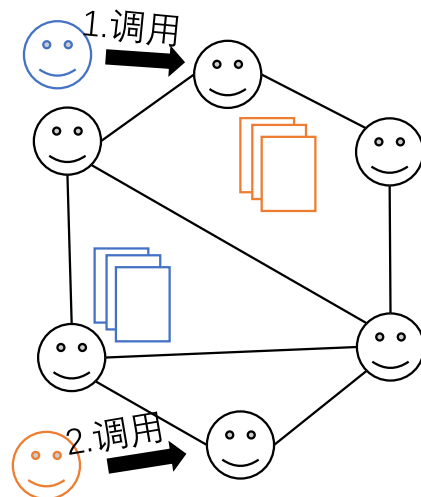


自动售货机是可自动执行的合约
但不是智能合约

北大数瑞随机共识（有冲突交易→可信计算场景）

可信计算场景的前提：

- 篡改难
- 合约状态不丢、能容错



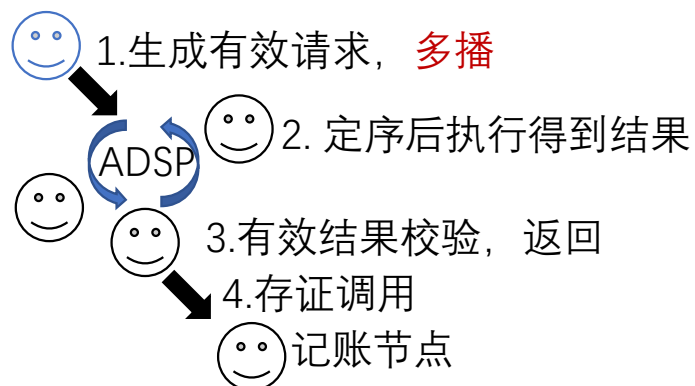
思路：定序，分片

权衡的重点：

- 需要“分片”机制
- 需要解决合约与合约之间、调用与调用之间的一致性

北大数瑞随机共识机制的重点：

- 着重在突破不同分片间（或是跨链）的**调用**
- 如何实现**错误的检测与恢复**



北大数瑞随机共识：合约调用示例

北大数瑞随机共识：

分片策略：以合约为单位

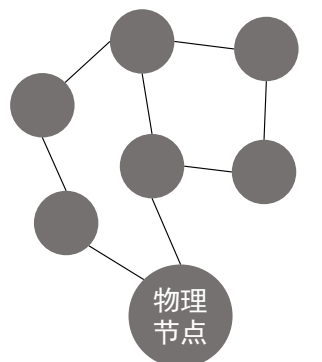
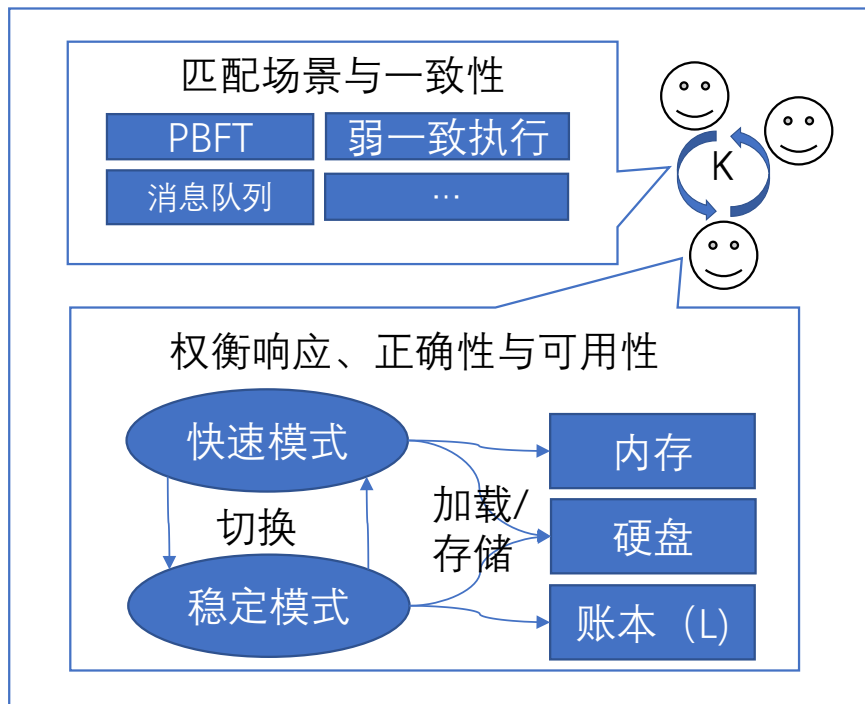
可配置项：

- 以智能合约为单位多种配置(BFT/KafKa)
- 以合约为单位配置一致性策略（严格一致/顺序一致/弱一致）

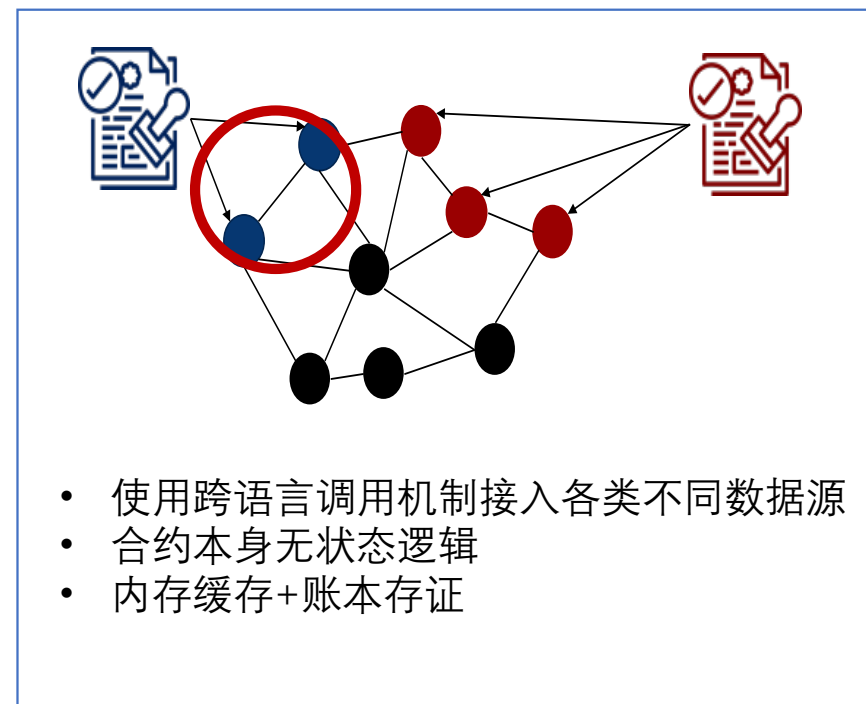
节点之间的交互

互联网3节点（分片大小）全冲突BFT定序50TPS
互联网3节点（分片大小）全冲突消息队列200TPS

节点与外部数据的交互



网络延迟<50ms
单点查询50ms
网络带宽10Mbps
N:节点总数
K:见证节点数
L:共识节点数



智能合约执行过程的比较

	比特币	以太坊	Fabric (只考虑单Channel)	百度超级链 (DPOS)	BDWare
请求阶段	支持并发发请求				
产块阶段	矿工节点弱一致 执行	矿工节点弱一致 执行	背书节点弱一致 执行	由有权节点，多核 执行	由子网定序（类BFT，以交易为单位定序）
广播阶段	矿工节点有效区块广播	矿工节点有效区块广播	排序节点进行严格一致产块，后广播	产生区块后广播	分片全部 执行
校验阶段	矿工节点再次顺序 执行	其他矿工节点是顺序 执行	记账节点是顺序 验证 读写集	其他节点 验证	记录存证账本
上链时间	运气好约10分钟	gas付高，约2分钟	发请求+执行+定序+广播+验证+查询有效性	发请求+定序+执行+广播+验证+查询有效性	发请求+定序+执行+返回结果校验+图式账本响应时间
确认时间	后续6个块	后续40个块	上链即确认	上链即确认	返回即有效
信任来源	由于账本全网同步，6个区块以后被改概率极小	由于账本全网同步，40个区块以后被改概率极小	由于背书节点背书+排序节点定序+执行结果多个节点存储，被改概率极小	由于通过投票选出了有权节点，结果被多个节点存储，被改概率极小	多个节点 共同定序并执行了合约，多节点签名，被改概率极小



谢谢聆听

请您批评指正！

联系人：蔡华谦
bdware@pku.edu.cn