以太坊: 一种安全的去中心化的通用交易账本

戎佳磊
garyrong0905@gmail.com
rjl493456442

大纲

- 以太坊的发展历史及现状
- 以太坊中的区块链范式
- 以太坊中的关键概念
- 课程安排

1. 以太坊的发展历史及现状

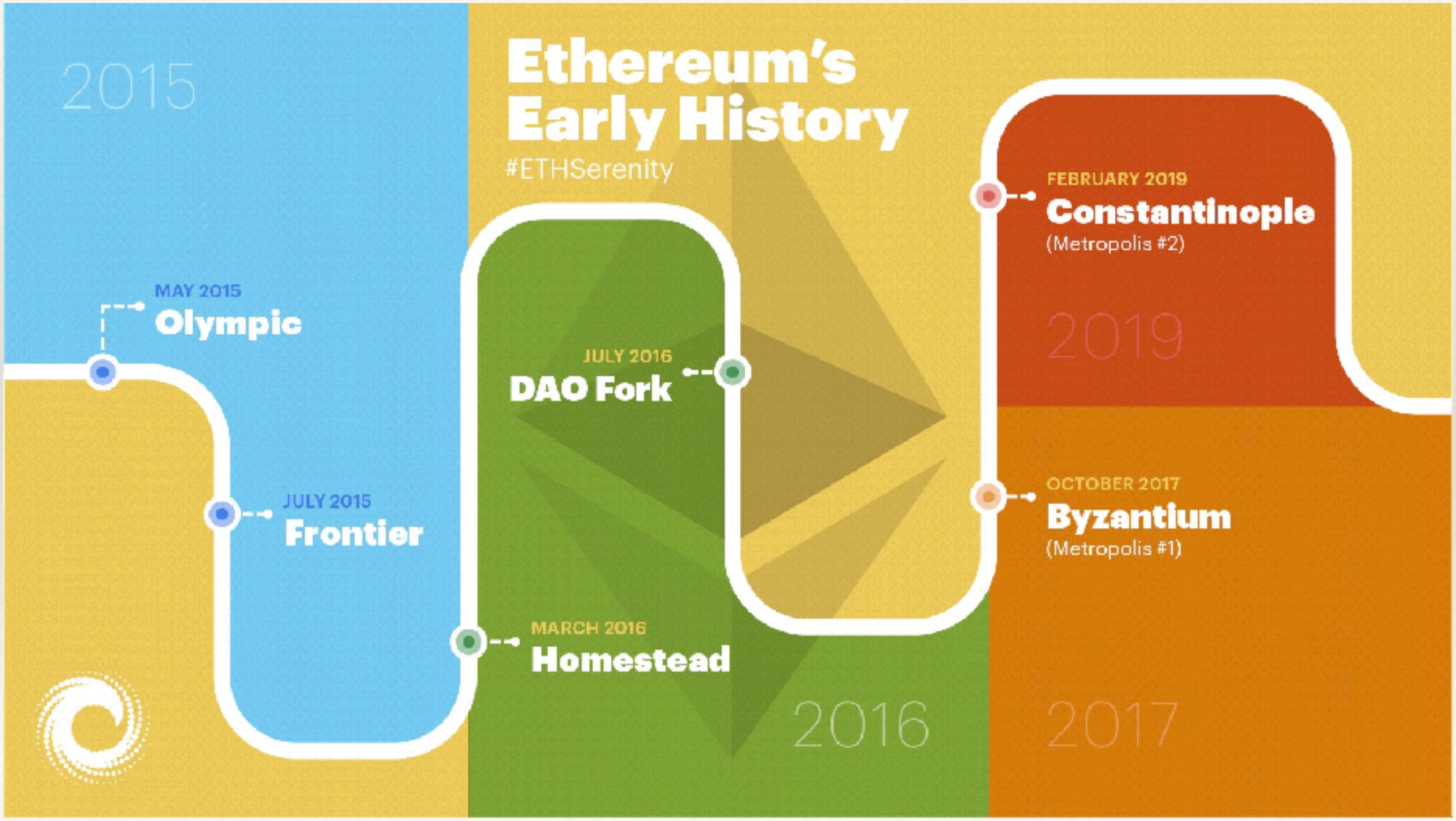
1. 以太坊发展的历史及现状

- 2013年年末，以太坊创始人Vitalik Buterin发布了以太坊初版白皮书
- 2014年2月，Gavin Wood和Jeffrey Wilcke加入以太坊
- 2014年7月24日起，以太坊进行了为期42天的以太币预售，一共募集到31,531个比特币，根据当时的比特币价格折合1843万美元
- 2014年11月份，以太坊在柏林举办了第一次小型开发者会议（DEVCON 0）

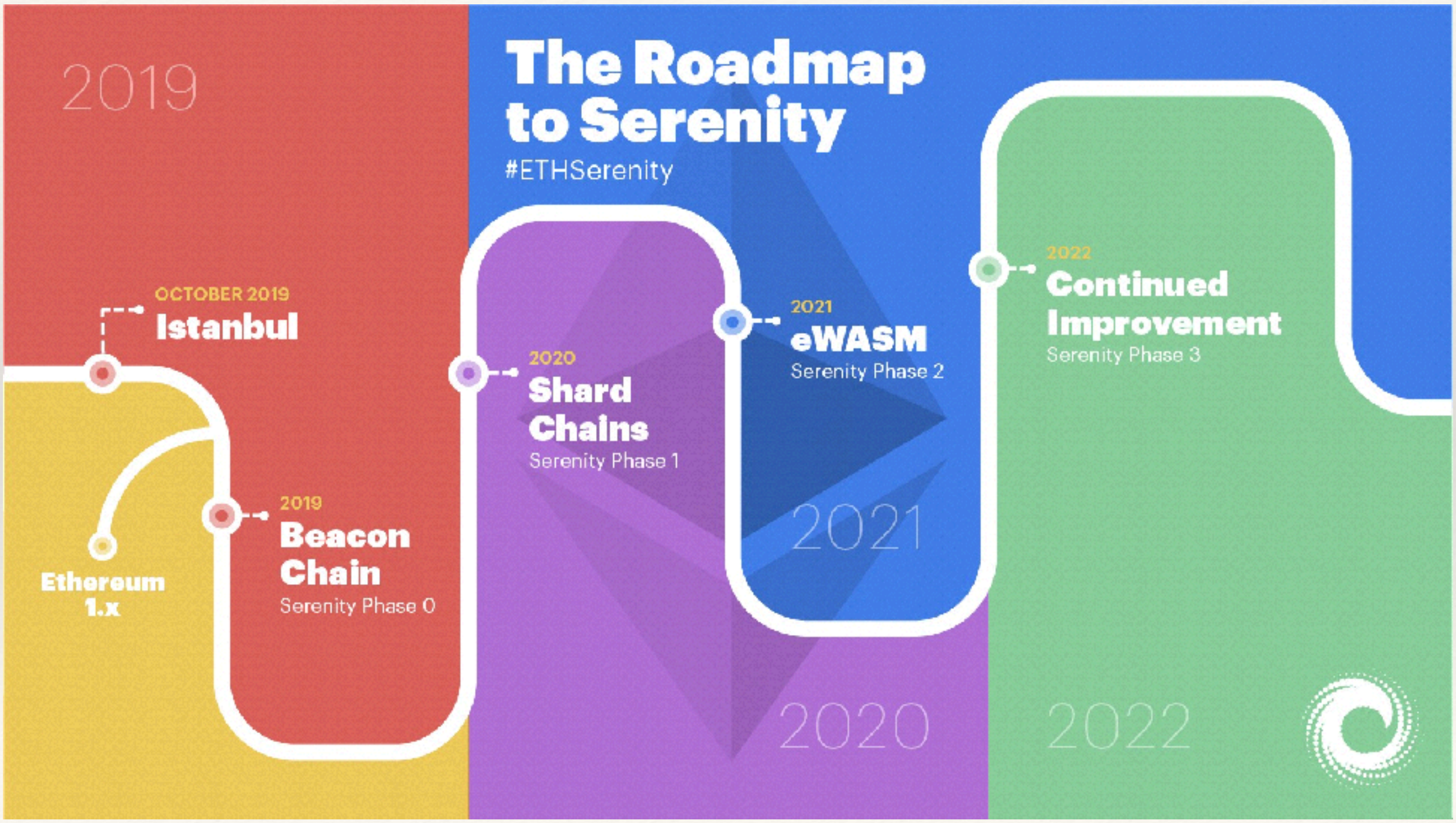
1. 以太坊发展的历史及现状

- 2015年5月份，团队发布了最后一个测试网络（POC9），代号为Olympic
- 2015年7月30日，以太坊发布了**Frontier(**前沿**)**版本，主网上线

1. 以太坊发展的历史

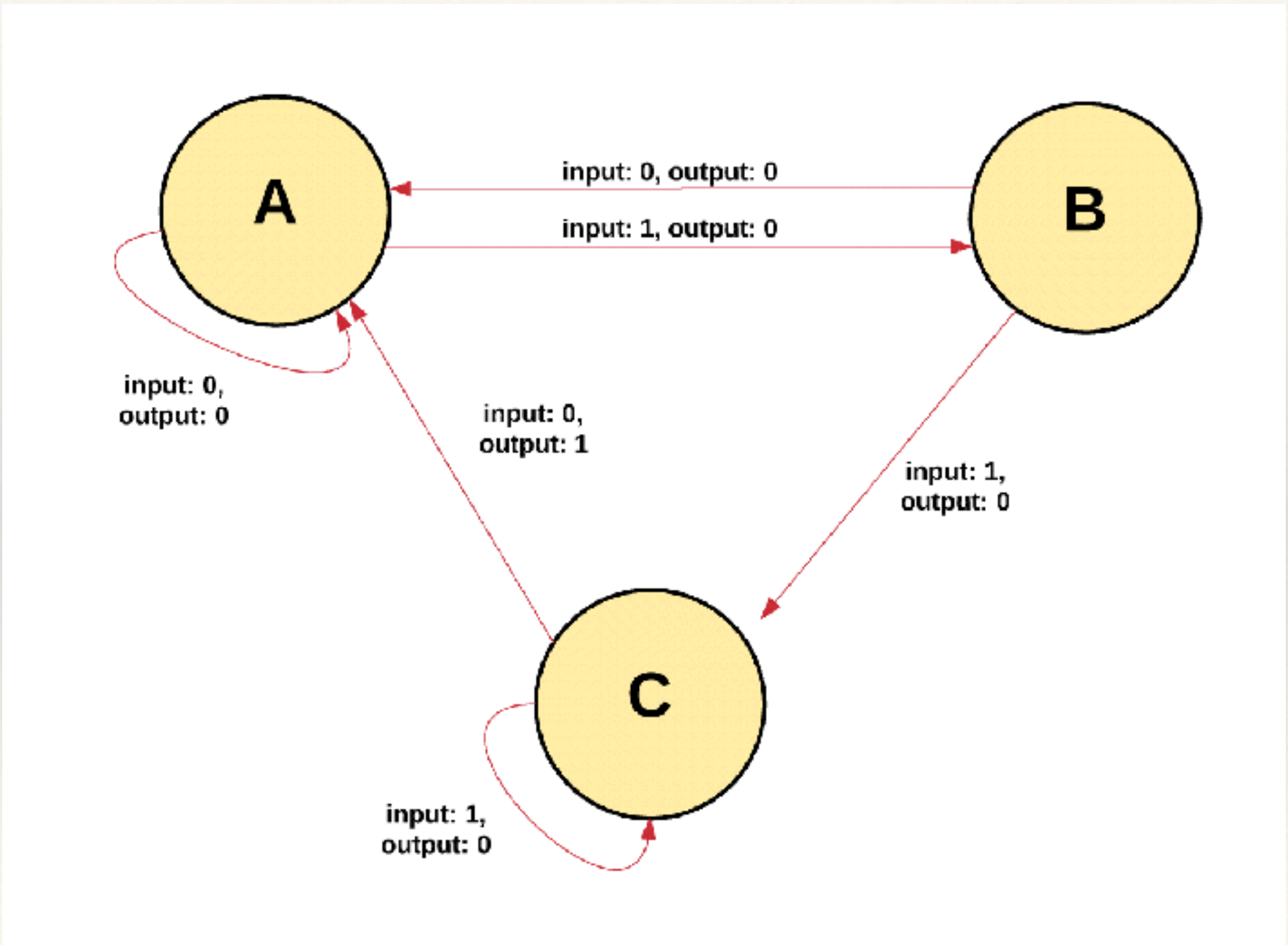

2015
Ethereum's Early History
#ETHSerenity
MAY 2015
Olympic
JULY 2015
Frontier
JULY 2016
DAO Fork
MARCH 2016
Homestead
FEBRUARY 2019
Constantinople
(Metropolis #2)
2019
OCTOBER 2017
Byzantium
(Metropolis #1)
2016
2017

1. 以太坊未来的规划

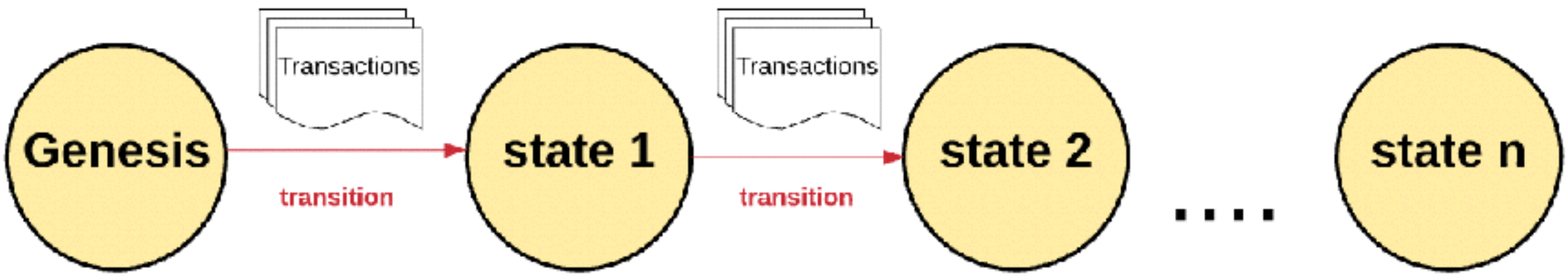

2019
OCTOBER 2019
Istanbul
Ethereum 1.x
2019
Beacon Chain
Serenity Phase 0
The Roadmap to Serenity
#ETHSerenity
2020
Shard Chains
Serenity Phase 1
2020
2021
eWASM
Serenity Phase 2
2021
2022
Continued Improvement
Serenity Phase 3
2022

2. 以太坊中的区块链范式

2. 以太坊中的区块链范式

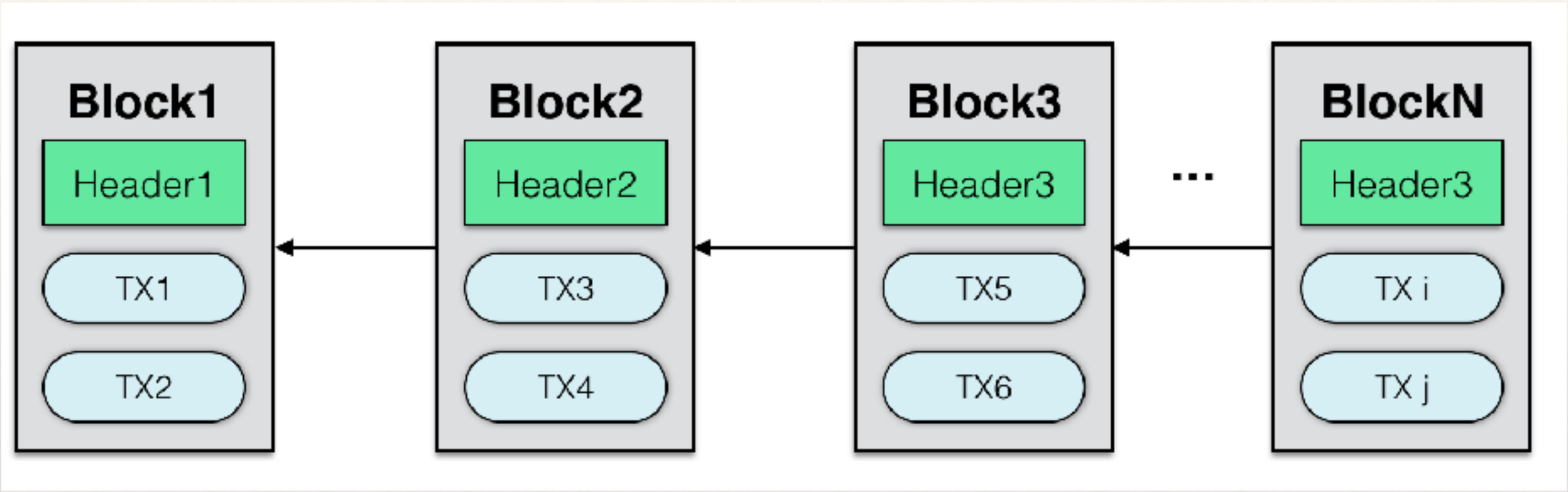


2. 以太坊中的区块链范式



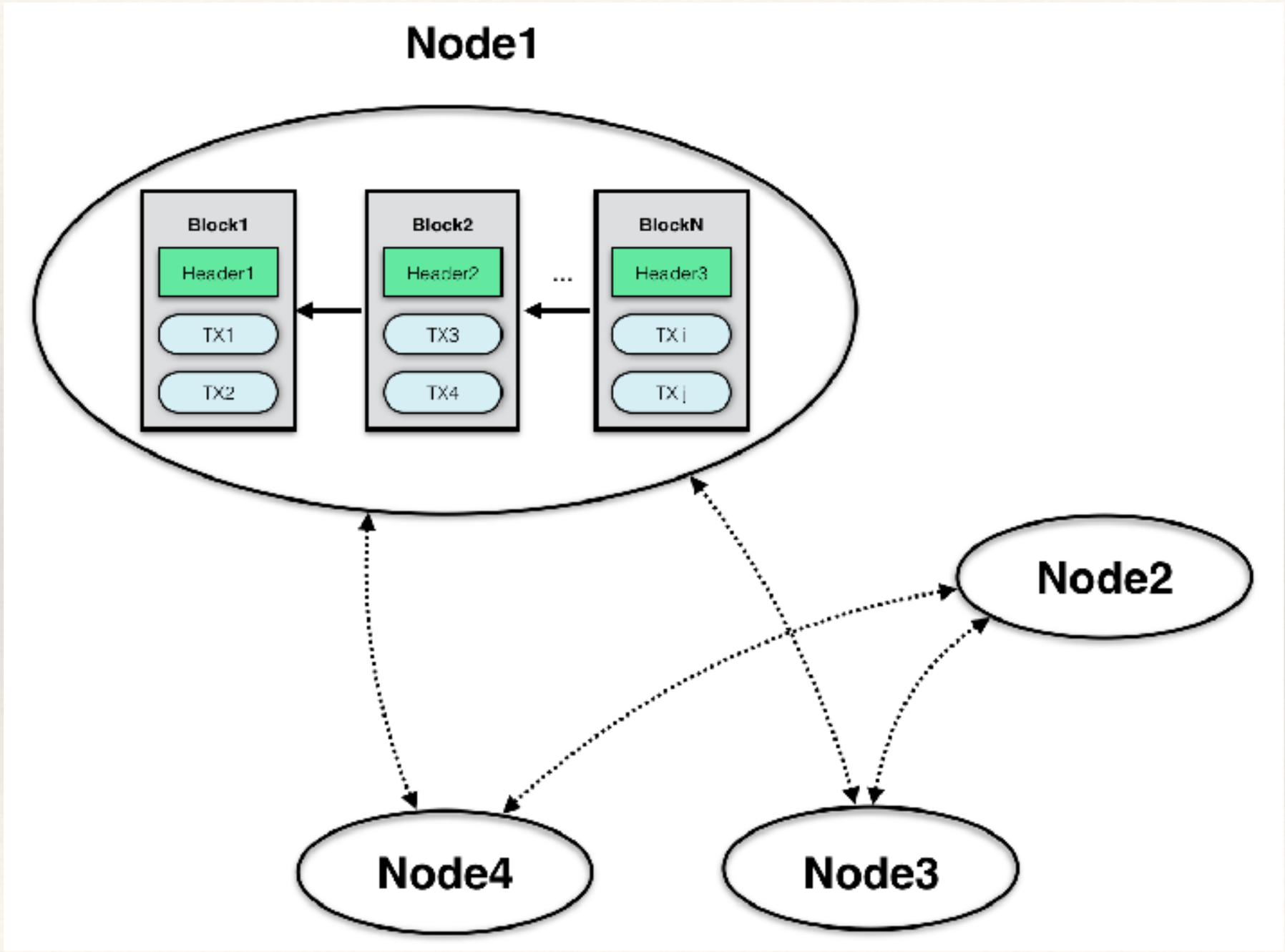
- 以太坊本质上是一个基于交易的状态机
- 以太坊有一个初始状态我们称为**Genesis**
- 状态转换的最小单元是交易(原子性、一致性)
- 每次执行一条或者多条交易后发生状态转换
- 任何一个最新的状态用来代表以太坊的当前状态

2. 以太坊中的区块链范式

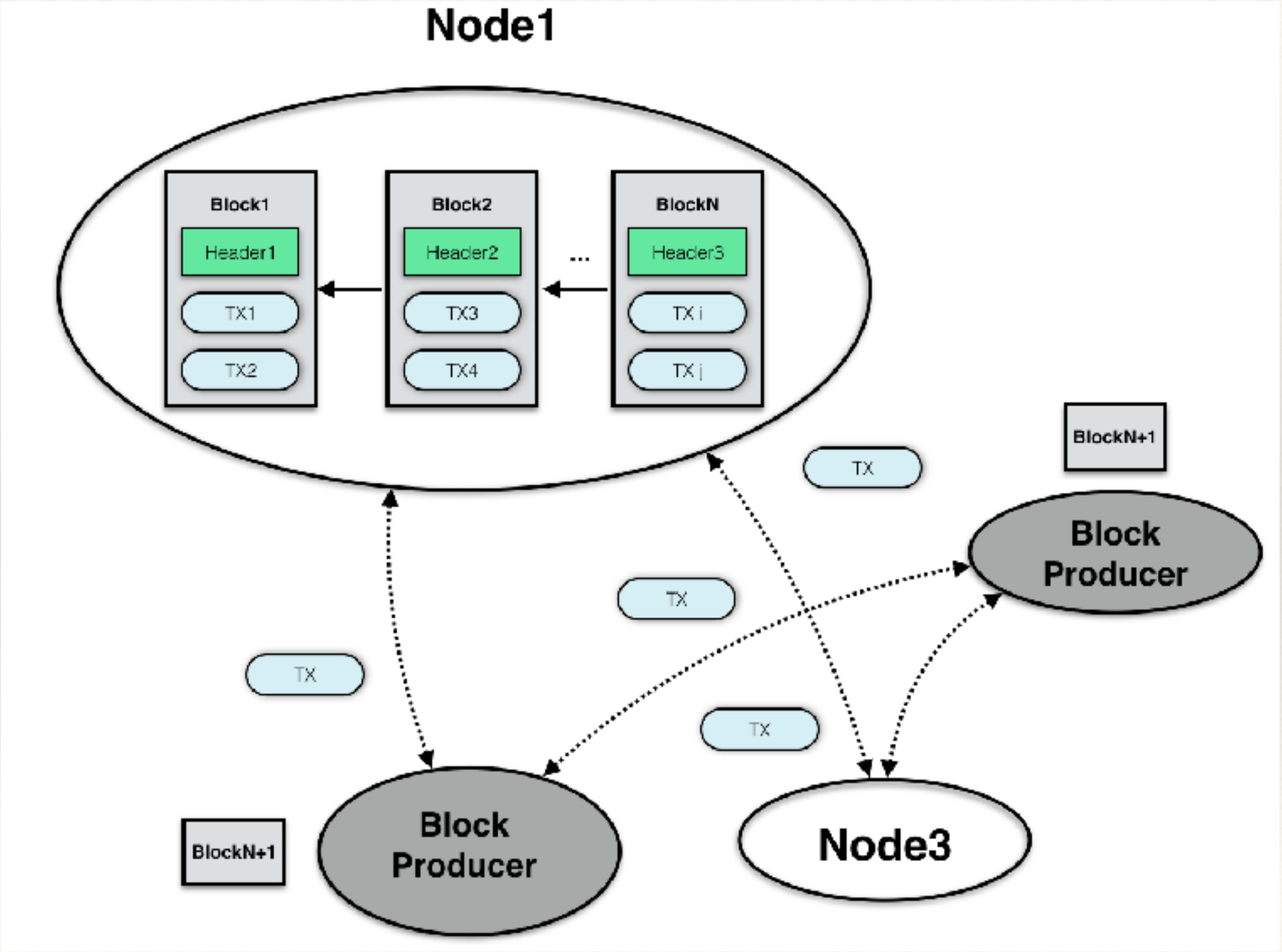


- 每次状态转换所执行的交易被组合成一个区块
- 每一个子区块都通过唯一的方式指向父区块
- 每一个区块都有一个唯一标识

2. 以太坊中的区块链范式



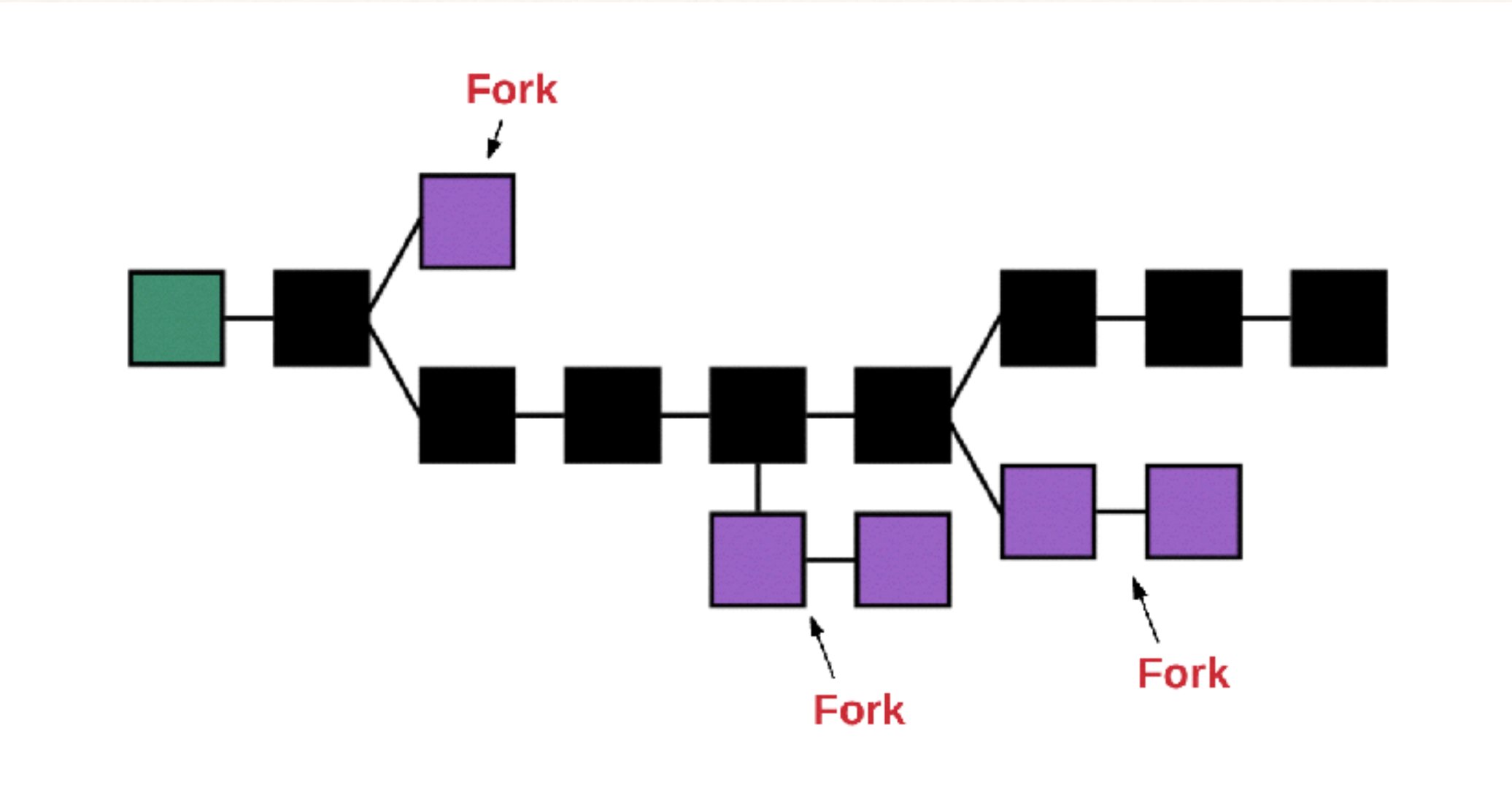
2. 以太坊中的区块链范式



2. 以太坊中的区块链范式

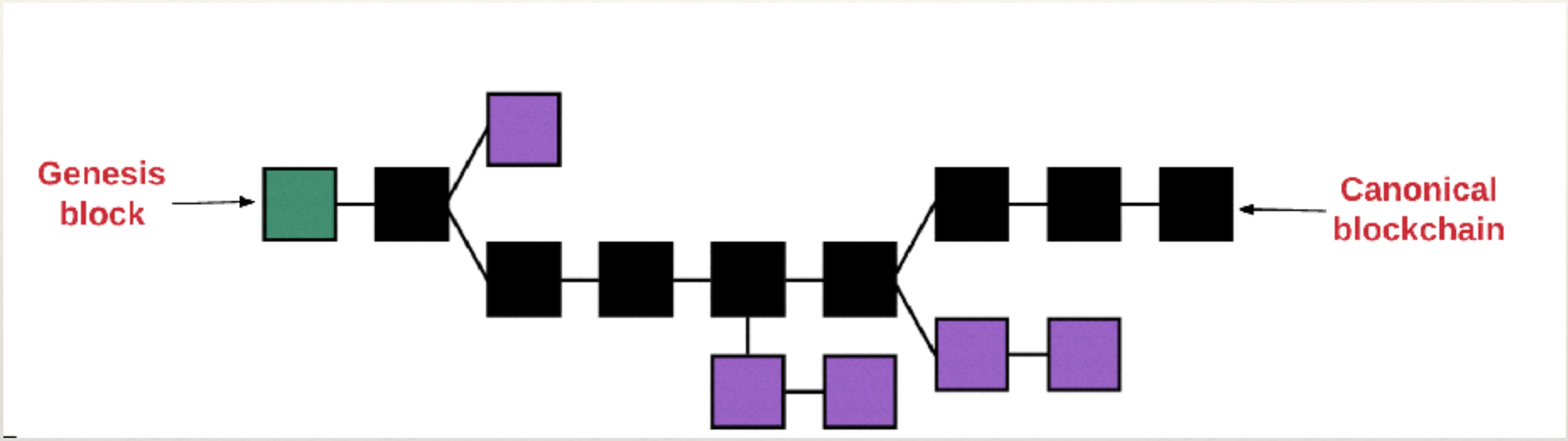
- 所有能够被Block Producer所接收的交易都必须是合法的
  - 密码学合法
  - 状态转移结果合法
  - 等等...
- 网络中可以有任意多个Block Producer
- 任意Block Producer都可以在任何状态的基础上接收合法交易创建区块（确保了网络的活性）
- 创建区块需要满足一定的规则（指定，计算量证明，惩奖规则）
- 创建区块有对应的经济奖励

2. 以太坊中的区块链范式



区块链是一个具有共享状态的单例交易状态机！！

2. 以太坊中的区块链范式

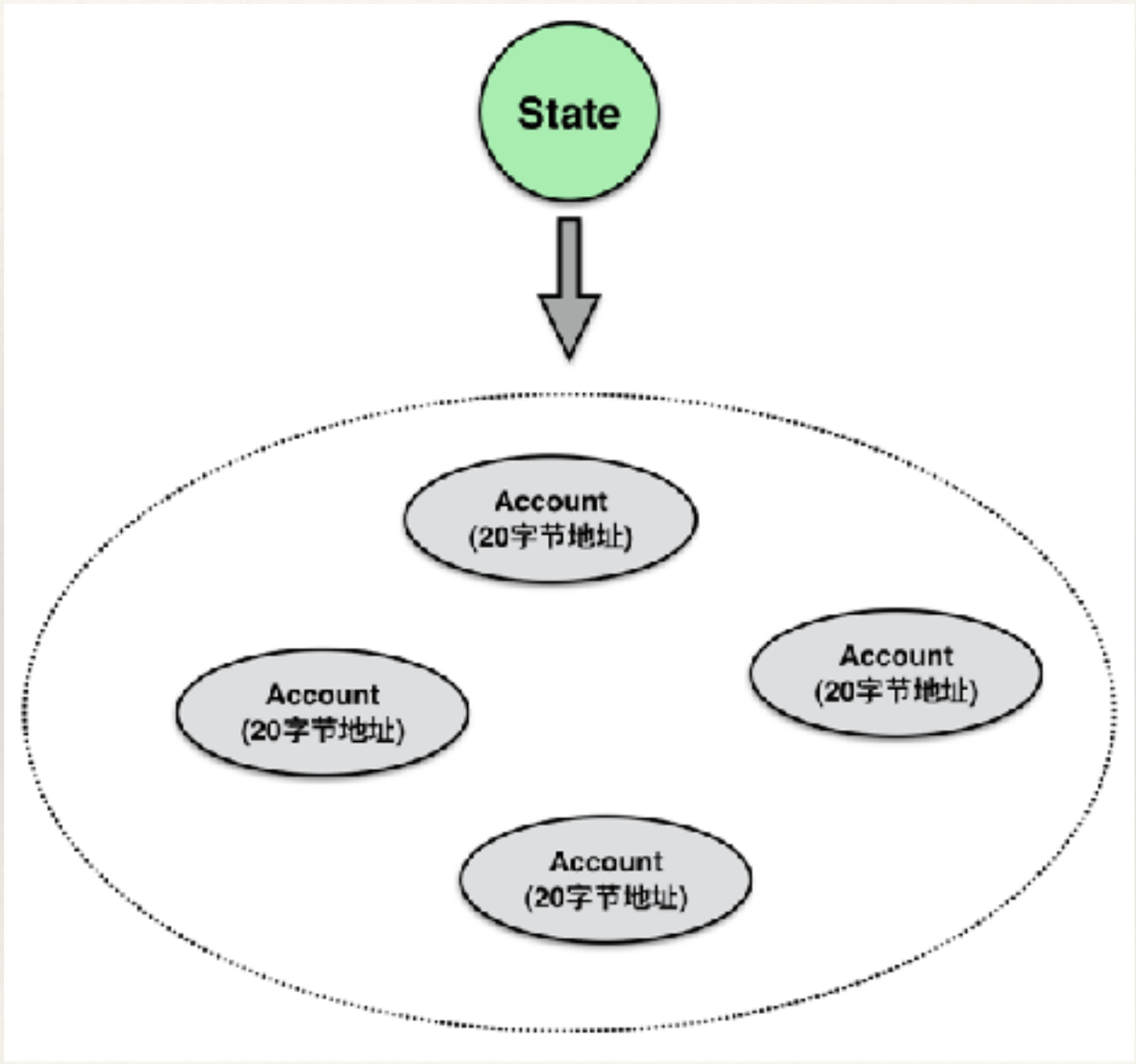


- 分叉选择公式
- 选择一条唯一的“区块多叉树”的路径作为区块链

3. 以太坊中的关键概念

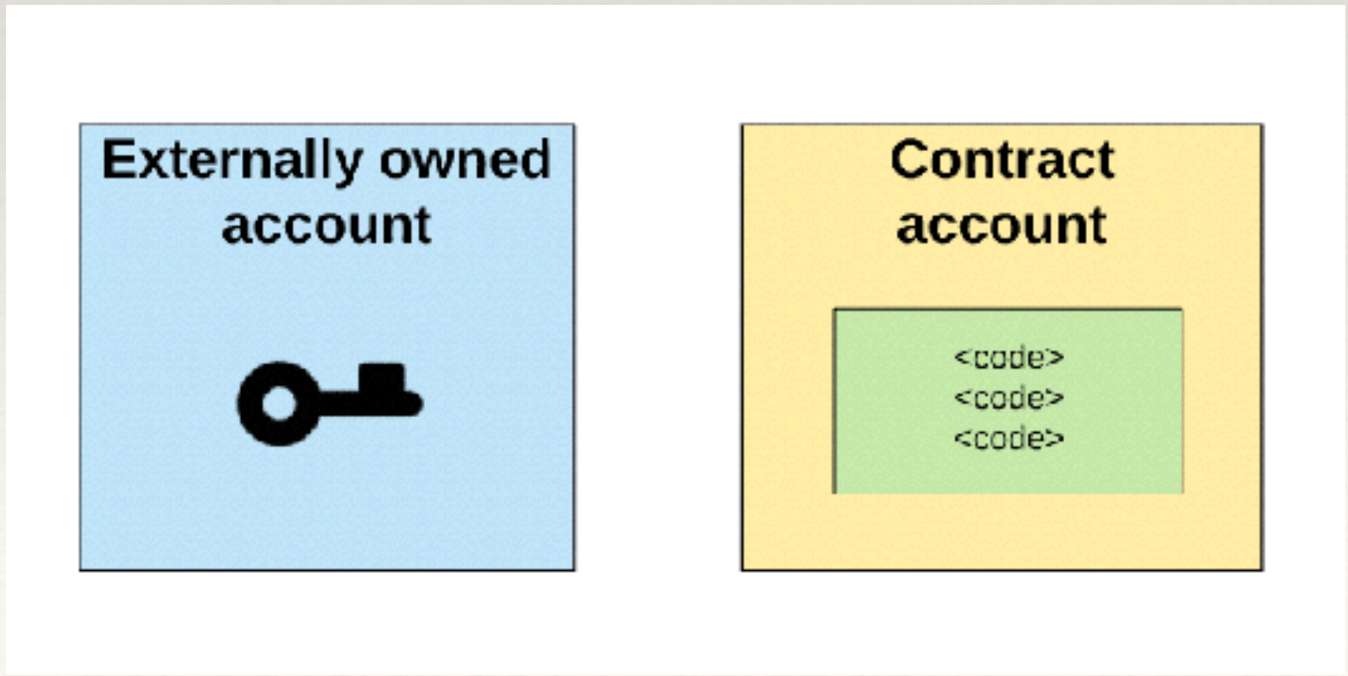
3. 以太坊中的关键概念（账户）

- “世界状态”是由一系列“账户”所组成
- 每一个账户都有一个唯一的地址
- 地址的产生规则分为两类
  - 由账户对应的公钥计算所得
  - 由部署者的信息计算所得



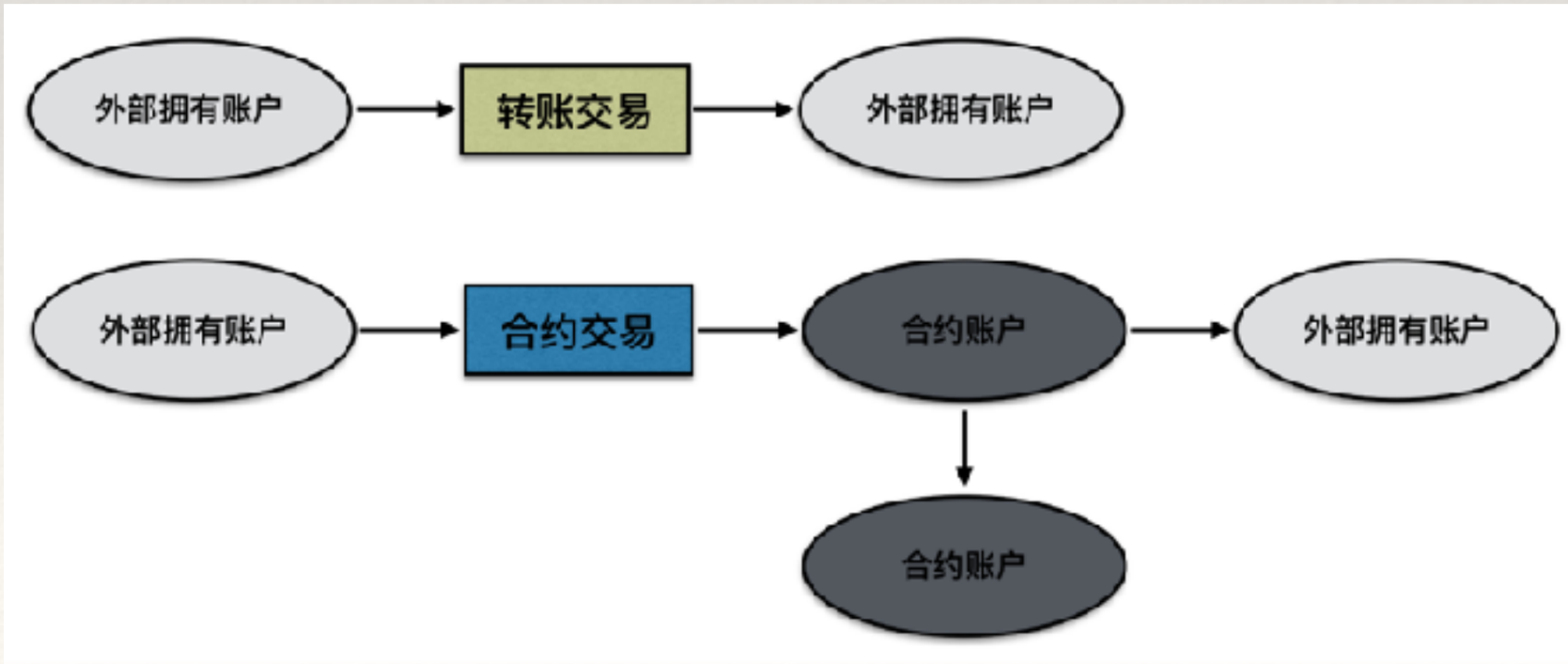
3. 以太坊中的关键概念（账户）

- 账户的类别分为两类：“合约账户”及“用户账户”
  - 用户账户是受到唯一的私钥文件所控制，且没有对应的运行码
  - 合约账户是受唯一的运行码控制



3. 以太坊中的关键概念（账户）

- 用户账户可以主动发起交易：转账交易或合约交易
  - 转账交易完成内置数字资产Ether的转移
  - 合约交易触发接收账户的运行码进行运行
- 合约账户只能通过“合约交易”或“消息调用”的方式被触发运行代码
- 合约账户在代码运行期间可以完成任意复杂度的操作：修改存储空间数据项，发送“消息调用”给其他账户等



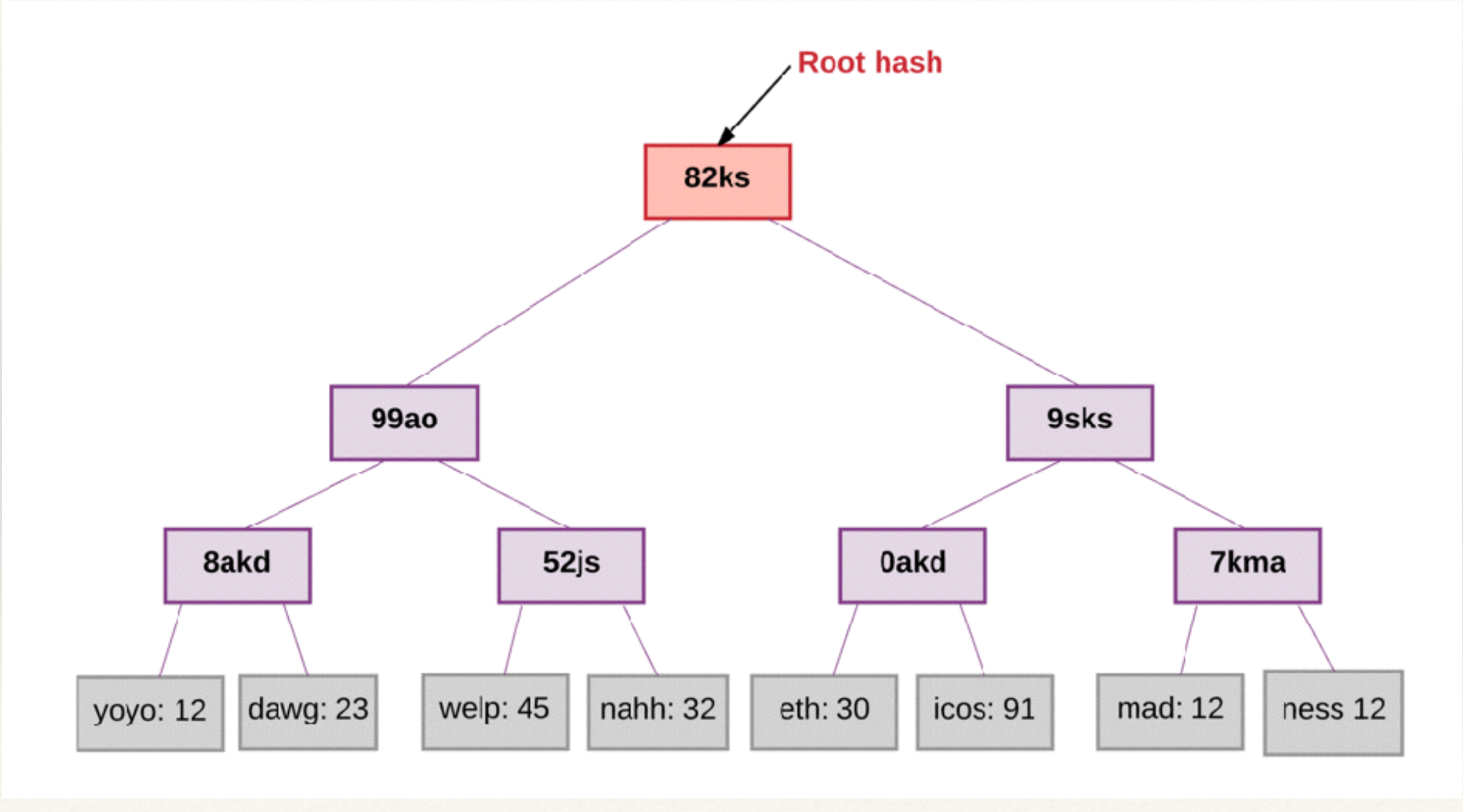
3. 以太坊中的关键概念（账户）

- 无论是“用户账户”或者是“合约账户”，其在底层虚拟机的视角来看是一致的
- 每一个账户都有一个“balance”字段，表示其拥有的加密资产数量
- 每一个账户都有一个key为32字节，value为32字节的“永久”的存储空间（对于用户账户来说为空）

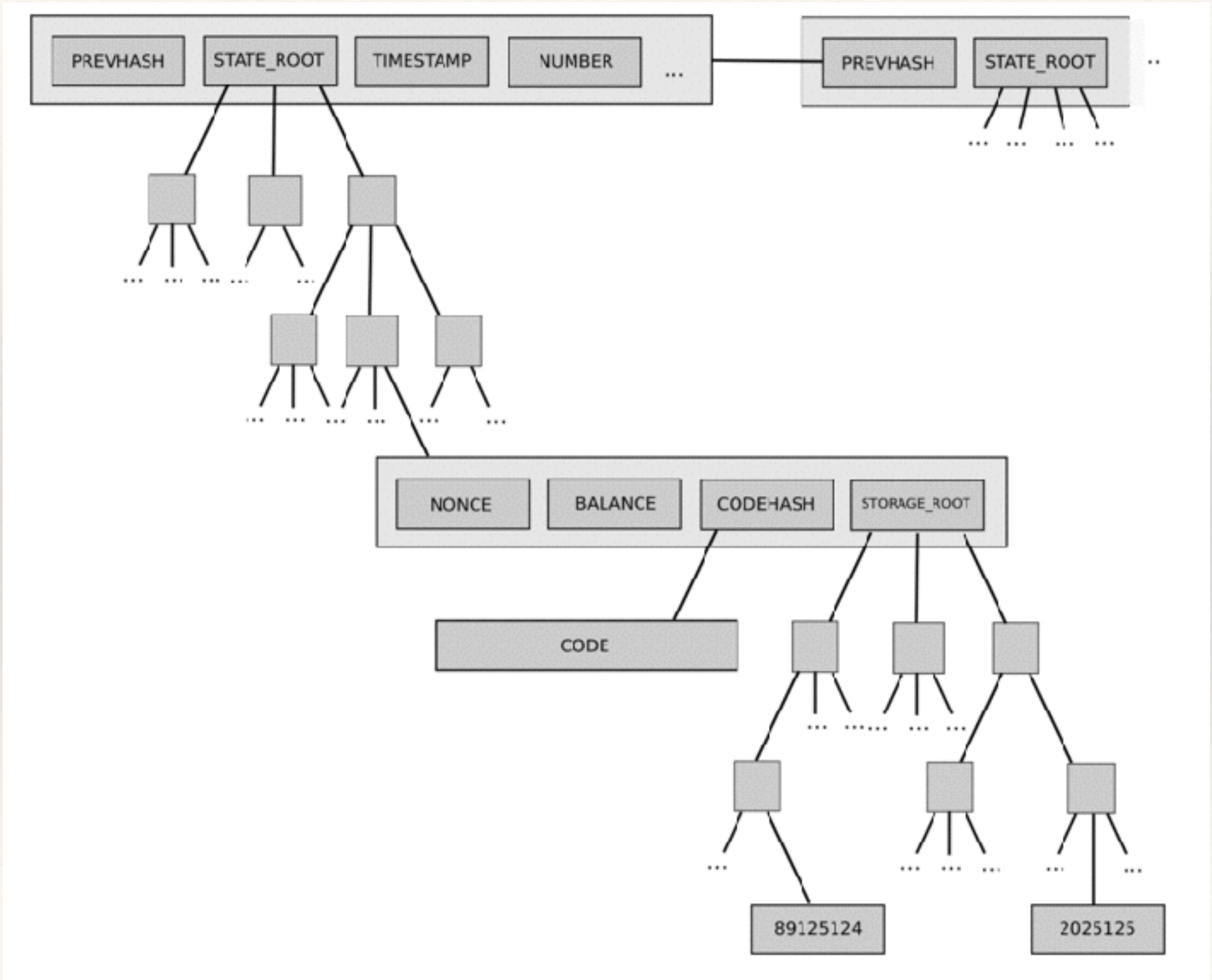
3. 以太坊中的关键概念（账户）

属性	解释
Nonce	已经发起交易的笔数/已经发起的消息调用次数
Balance	内建数字资产Ether的余额
Storage Root	合约账户存储空间的哈希标识
Code Hash	合约账户运行码的哈希标识

3. 以太坊中的关键概念（Merkle Tree）

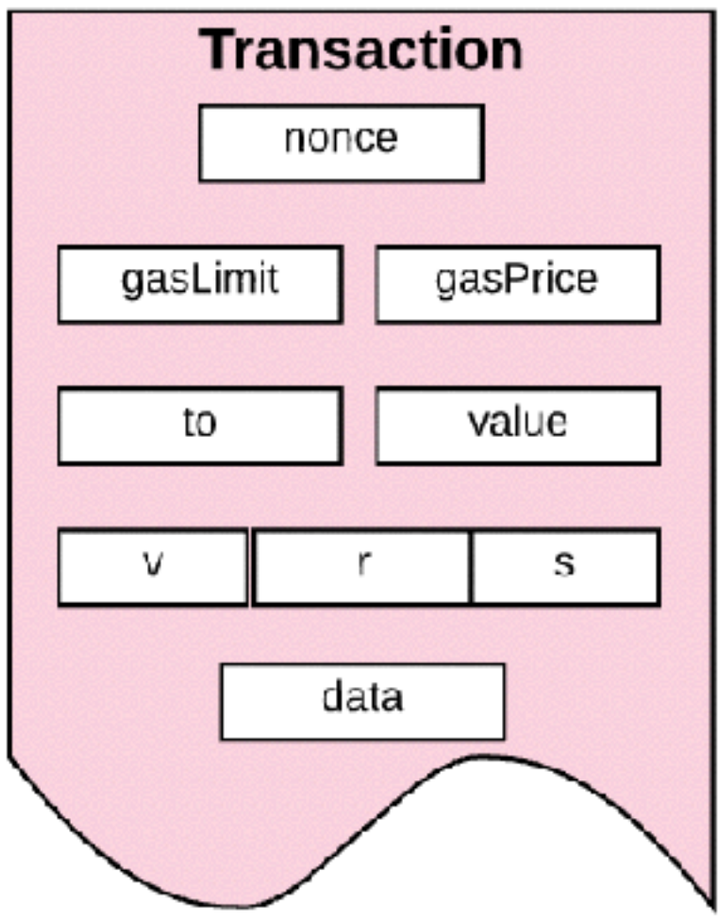


以太坊中的关键概念（世界状态）



3. 以太坊中的关键概念（交易）

- 交易是一组由密码学算法签署的由外部拥有账户产生的指令（转账，调用合约），将其序列化后提交给区块链网络。
- Nonce: 用户账户已经发起的交易笔数
- GasLimit: 该交易所预购的Gas值
- GasPrice: 每单元Gas的费用
- V,R,S: ECDSA签名



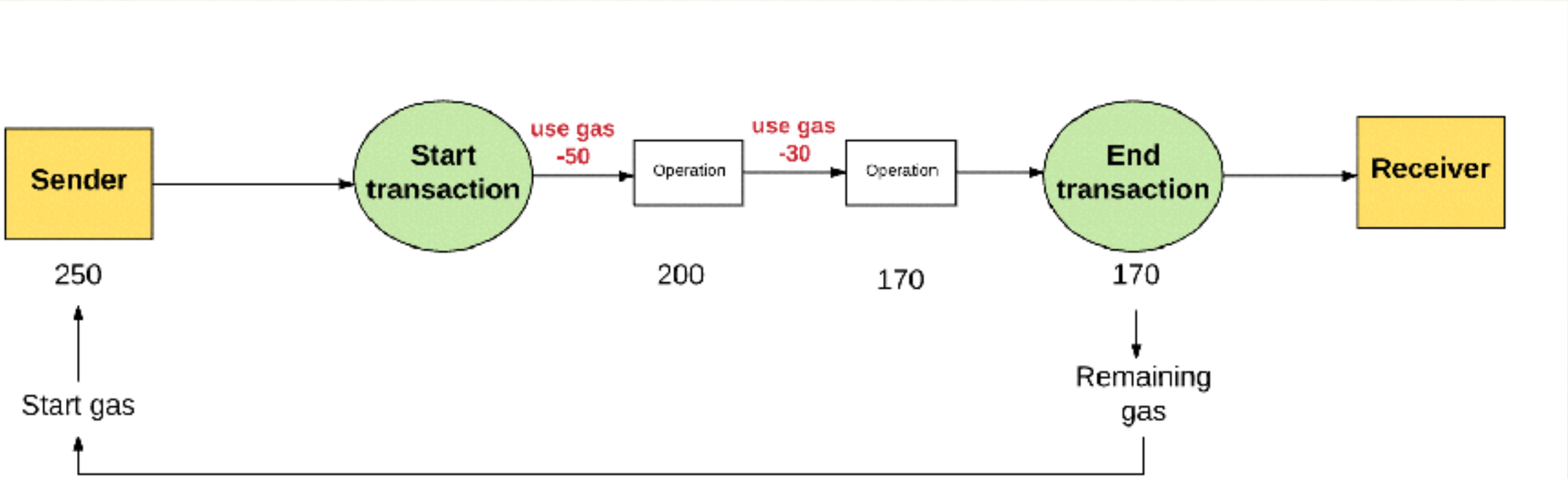
3. 以太坊中的关键概念（交易）

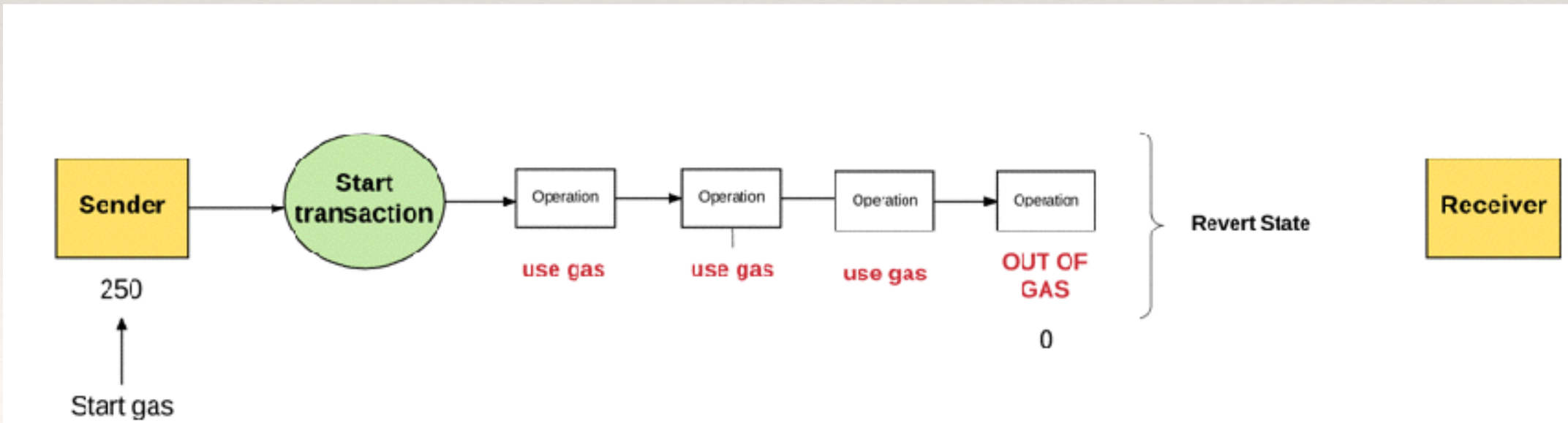
- 交易是用户可以更改区块链世界状态的唯一途径
- 交易必须具有“原子性”
- 通过非对称签名算法，可以保证：
  - 交易体的内容不会被“中间人”篡改
  - 交易只能够由“拥有发起者私钥”的用户发起（确保加密资产的安全性）

3. 以太坊中的关键概念（Gas）

- 在以太坊上，任何引起状态转移的操作都是需要收费的
  - 数学运算
  - 状态存储
- Gas是用来计量以太坊系统资源使用情况的最小计量单位
  - 状态转移中所有的动作都有一个复杂度的衡量公式
  - Add操作花费3个Gas，SStore操作花费20000个Gas
- 一次交易执行过程，累积消耗Gas超过发送者预付的总量，交易执行失败

3. 以太坊中的关键概念（Gas）



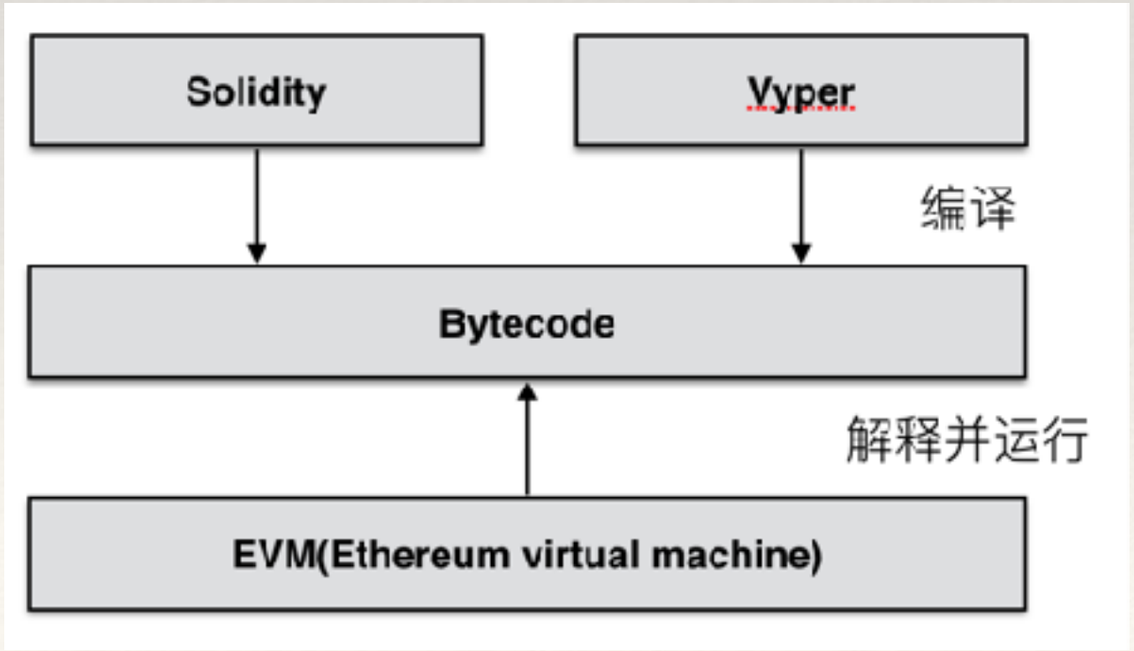


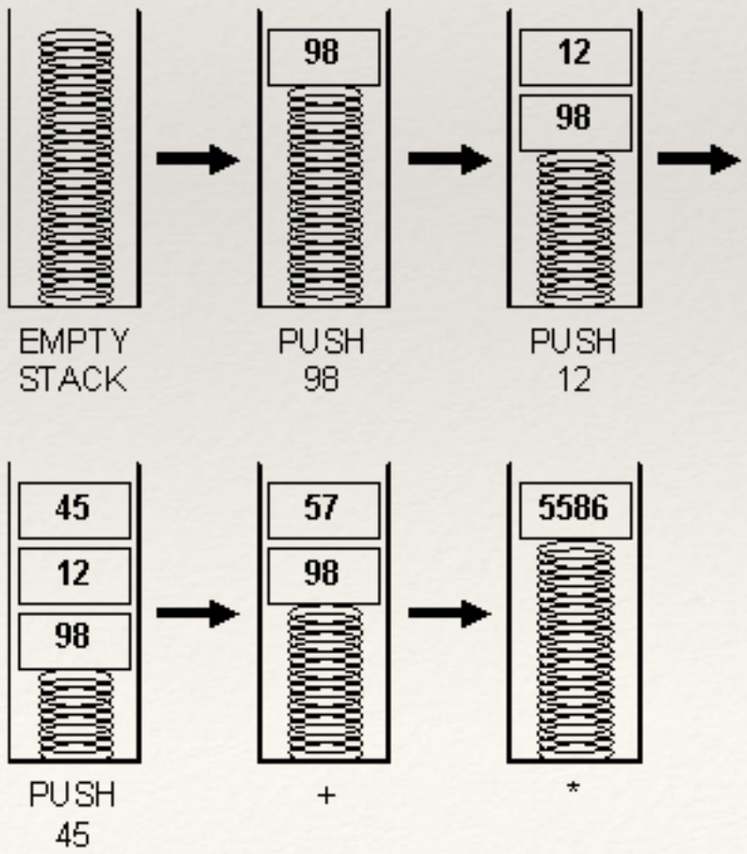
3. 以太坊中的关键概念（Fee）

- GasPrice表示发送者预付的Gas价格
- Fee = Gas * GasPrice
- 发送者必须有足够多Ether余额来支付交易费用
- 交易所产生的手续费作为Block Producer的经济激励

3. 以太坊中的关键概念（EVM）

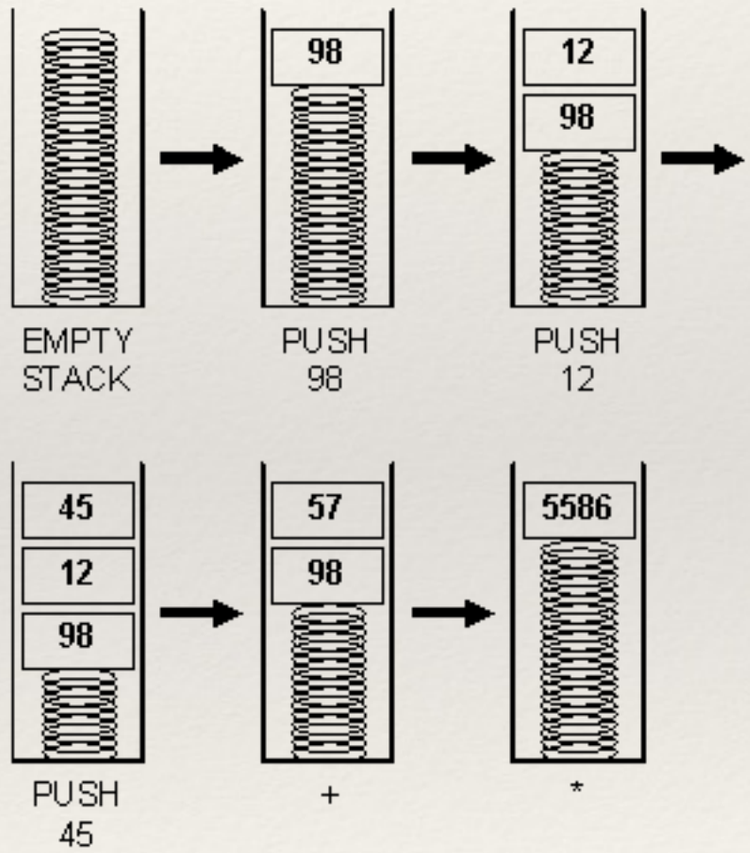
- EVM(Ethereum Virtual Machine) 是指用来解释跟执行合约账户字节码的解释器
- EVM基于栈的解释器
- EE(Execution environment) 是EVM的执行环境，包含环境参数以及存储空间读写函数
- EVM是拥有完全隔离的执行环境，合约无法访问宿主机的“网络”，“文件系统”等系统资源





3. 以太坊中的关键概念（EVM）

- EVM可以进行数据存放的区域有三个:
  - 合约的“永久存储空间”
  - 临时的“堆空间”
  - 临时的“栈空间”



3. 以太坊中的关键概念（指令集）

- EVM有一个固定的指令集
- 指令集包含：算数运算，比特运算，逻辑运算，跳转指令，状态读取、存储指令等
- 所有指令的运算必须是确定性的（例如高精度的浮点运算是不支持的）

3. 以太坊中的关键概念（日志）

- 对于以太坊上部署的智能合约来说，外部拥有账户所发起的交易，是链下世界对链上世界的输入
- 智能合约必须也需要某种途径把链上世界的信息传递出去 - 日志
- 合约编码者可以在智能合约中定义Event
- 当智能合约运行过程中执行该语句，便会产生一个虚拟机日志，并且将其存储在回执中

3. 以太坊中的关键概念（日志）

Transaction Receipt Event Logs

124

Address 0xdac17f958d2ee523a2206206994597c13d831ec7

Name `Transfer (index_topic_1 address from, index_topic_2 address to, uint256 value)`

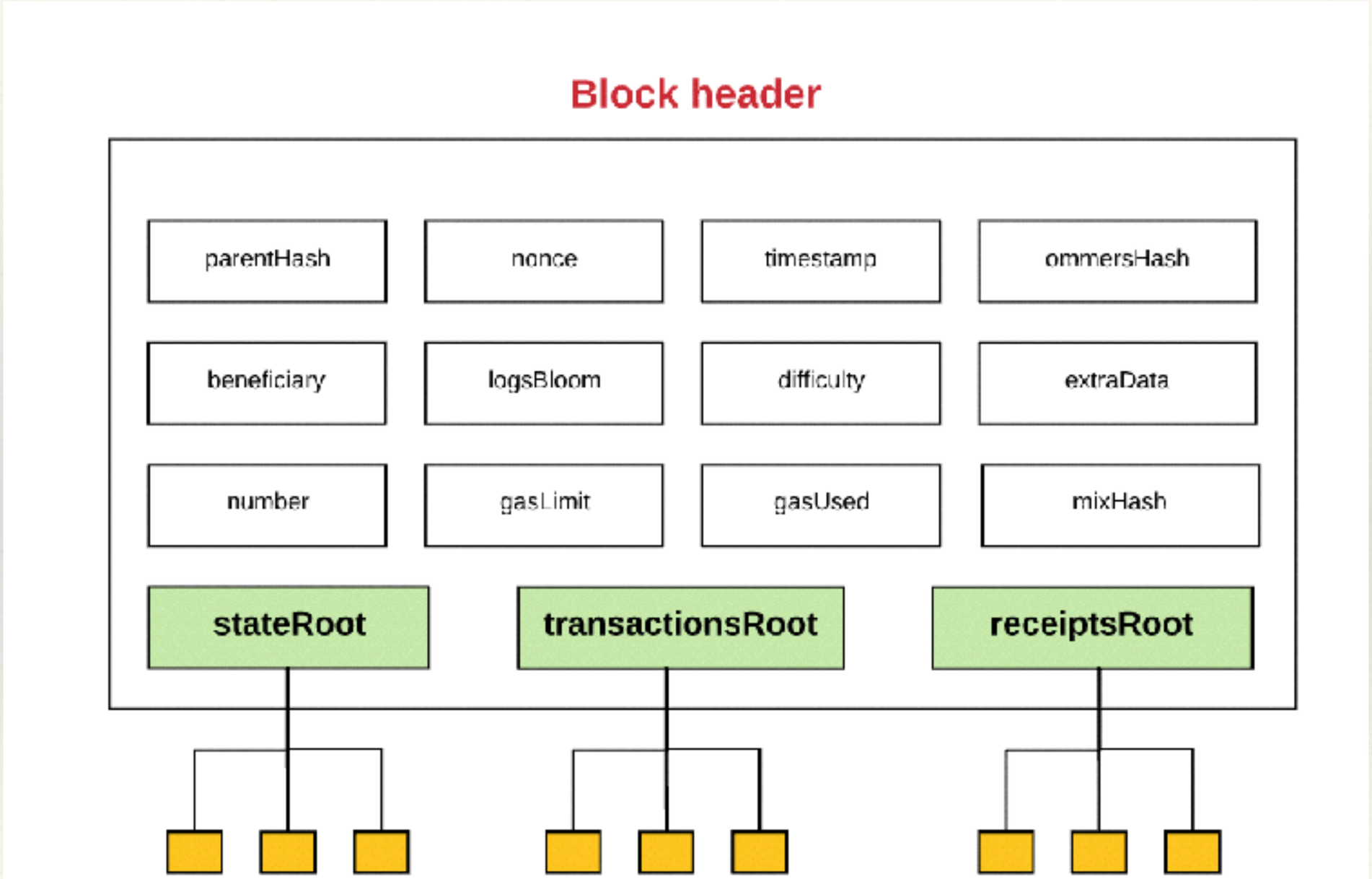
Topics
0 `0xddf252ad1be2c89b69c2b068fc378daa952ba7f163c4a11628f55a4df523b3ef`
1 Hex → `0x000000000000000000000000e1dfa1b5988953433059ca7364fcddab3be9953c`
2 Hex → `0x000000000000000000000000084c7dfae5bcfca7a99c61f54bf4afb88d54032d1`

Data Hex → `0000000000000000000000000000000000000000000000000098410d0`

3. 以太坊中的关键概念（回执）

属性	描述
Status	交易执行状态
CumulativeGasUsed	累积使用的Gas值
Bloom	交易日志的布隆过滤器信息
Logs	交易执行过程中所产生的日志集

3. 以太坊中的关键概念（区块）



3. 以太坊中的关键概念（Ethash）

- 以太坊Proof-of-Work算法的具体实现：ethash
- ethash是memory-hard的PoW实现
- ethash算法由两个子类算法组成：Dagger-Hashimoto
  - Dagger：产生用于PoW计算的数据集
  - Hashimoto：PoW运算规则

课程安排

- 以太坊技术原理：
  - 点对点网络协议栈devp2p
  - 共识算法
  - 以太坊存储结构
  - 以太坊虚拟机
  - 同步算法和轻节点协议

课程安排

- 智能合约及开发：
- Solidity基本语法
- Solidity高级知识
- Solidity最佳编码实践及常见攻击

Thanks