# 路线图 ——翻译者：金燕琪

### 一、主要内容：

我们正在制定一种混合共识机制，它的运行主要遵循以下几点：

1. 已被许可的BFT是在基于POW共识机制的网络中的几个节点上运行的。
2. BFT委员会是一个轮值委员会，能够有效地防止腐败现象的发生。
3. BFT委员会负责交易验证，而POW节点只负责根据我们得出和重新定义在以太坊黄皮书中提到的一些规则来选择或者选举委员会成员！
4. 据猜测，新许可的虚拟机（VM）灵感是源自以太坊虚拟机（EVM）。但是它又有不同于EVM的区块状态和事物执行流程。
5. 当前，POW链中的无许可以太坊虚拟机和新的许可虚拟机共存（这种虚拟机被称为初链虚拟机-TVM）。
6. 根据当前的讨论，初链虚拟机（TVM）将成为验证任何交易的一种方式，而EVM需要重新工作，不是为了共识而采取行动，而是轻量级钱包交易的BFT选举。
7. 这种激励模式，需要重新开展工作，使其基于TVM，也就是我们仍然以某种方式奖励POW链中的矿工。
8. 我们最终会支持分享BFT委员会的节点，提高可扩展性。
9. 补偿基础结构，如果节点规格不一致性（比如在节点池中不同的CPU /内存/网络带宽等）最终都会将成为共识的一部分，加快交易。
10. 因此，智能合约的执行只会发生在TVM（BFT节点），我们确实认为在混合设置中应该从POW节点支持部署，从而引发关于状态复制的问题。

注意：我们可能需要改变稳固性以及以太坊目前的轻钱包，这正是我们把它作为框架的选择。

### **二、时间进度计划：**

#TODO:在时间线中添加规范或者伪代码的细节

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 章节 | 时间(起点) | 阶段 | 详细 | 备注 |
| Q1/18 |  | Project.init() | V<3 Python | V also<3 GoLang |
| Q1/18 |  | 白皮书发布 |  | <https://www.truechain.pro/EnTruechain.pdf> |
| Q2/18 |  | 黄皮书第一草案 | 混合共识文档收录在 arXiv上 | <https://arxiv.org/abs/1805.01457> |
|  |  |  | 基本的TVM | EVM (POW) +TVM(BFT)==>PBFT+POW在单个节点上 |
|  |  | 激励计划 | 为新的虚拟机组合制定激励计划，考虑一个不同的Txn费用模型 |
|  | 黄皮书TIP-1 |  |  |
|  |  | 更新arXiv上的论文 |  |
|  |  | 混合共识伪代码发布 | |
|  |  | 去耦和耦合/重新排列以太坊来适应共识 |  |
|  |  | 调整以太坊Gas的经济模式 |  |
|  | 代码 | 实施混合共识代码库 | |
|  | 根据时间的重要性和依赖，重新排序，很快又进一步细分 |  | 初始化节点： 1. forkVirtualNodeBFT() 2. SpawnBFTnode() |
|  |  |  | 验证： 1. onChainValidation() a. validateFromPOW() b. FetchTxnHistory() 2. reElect() 3. checkTxnOrder() 4. updateSnailchain() |
|  |  |  | 对于视图更改和节点损坏的检测/更改 1. ViewChange() 2. complainToSnailChain() |
|  |  |  | 特征： 1. keygen() 2. SignLog() 3. Keccak256/ECDSA wrappers |
|  |  |  | 日志： 1. dailyLogOutput() 2. CreateLOG() 3. createTxTuple() |
|  |  |  | 交易池: 1. mempoolSubprotocol() a. propose() b. query() c. others.. 2. DailyOffchainProtocol() 3. |
|  |  |  | 选择和 VRF: 1. SeedSelectorVRF() 2. proposeStakeTransactions() |
|  |  |  | 简要说明： |
|  |  |  | 1. createSnapShot() |
|  |  |  | 交流： 1. gossipTxn() |
|  |  |  | 分享和补偿基础措施 |
|  |  |  | 基本的测试：集成测试、功能测试、单元测试 |
|  |  | 工程再造 |  |
|  |  | 基础设施监控/建设 |  |
|  |  | 安全审查 |  |
|  |  | 服务器安全强化 |  |
|  |  | 轻量级钱包再造 |  |
|  |  | 前端 |  |
|  |  | 发布设计文档，用于智能合约执行 | 公开在github /其他地方 |
|  |  | 自定义Dapps |  |
|  | 实验版本-r1-v0.? | |  |
|  |  | 开发者文档达成共识 | |
|  |  |  |  |
|  | 规模实验：建立在公有、私有、云 |  |  |
|  |  | 在全球各地区，建立内部规模实验室 |  |
|  | 仿真和测试 |  |  |
|  |  | 实验准备 |  |
|  |  | 模拟基本服务器负荷/成本 |  |
|  |  | 调整，修复bug，性能改进 |  |
|  | 实验版本-r2-v0.? | |  |
|  |  | 测试网 |  |
|  |  | 调整，修复bug，性能改进 |  |
|  | 实验版本-r3-v0.? | |  |
| Q3/18 |  |  | 在以太坊Dapps dev |  |
|  |  |  | 主要网络 |  |
|  | 特征请求、增强功能(RFES)和方法(TrueChain)，改进建议 |  |  |
|  | X86 平台 |  |  |
|  | 一些未来的发展规划 |  |  |

(Q1)1、2、3月

(Q2）4、5、6月

(Q3)7、8、9月

(Q4)10、11、12月

核心共识：

在PBFT链funcs：

阶段一：

*更新: 进一步细分*

|  |  |
| --- | --- |
| 方法 | 功能 |
| dailyLogOutput() | 将日志输出到done（\_，\_）然后发送到非成员节点 |
| CreateLOG() | 根据所有其他节点的日志为每日日志创建元组，跟踪签名日志由非委员会成员负责 |
| mempoolSubprotocol() | 使用Union集合来跟踪传入交易,支持查询方法返回确认的交易. |
| DailyOffchainProtocol() | 进行有条件的选举委员会成员, 基于节点是否是BFT成员。 |
| keygen() |  |
| forkVirtualNodeBFT() |  |
|  | 一个历史日志 |
| complainToSnailChain() | 触发器re\_elect() |
| ViewChange() | 触发viewchange，并反过来，触发complainToSnailChain（） |
| createTxTuple() | （R，l，tx）R - 当前日期，l是随机数（当天txn的序列） |
| SeedSelectorVRF() |  |
| gossipTx() | 正直的委员会成员然后将签名的元组闲聊到网络上 |
| createSnapShot() | 创建一个世界状态快照 |

阶段二：

1、时间戳验证（初链黄皮书图一）

2、create\_shard() and speculative\_transaction()（初链黄皮书图二）

POW链中的funcs

阶段1：fPOW（果实链）

* re\_elect() - POW挖掘功能，使用[Theta,LRU,stake\_in/out]( snailchain内部块的数量)，基于以下之一：

○由于腐败导致视图更改

○ 物理时间戳限制引发的第K天触发

○ 节点应该是DailyBFT委员会的过期Tstamp时间间隔会员

○ update\_snailchain()从N个PBFT节点接收done（\_，\_）散列，并由每个节点执行，然后将链上的节点写入到snailchain总账中。

钱包工程

模拟

确定以下内容：

1. 分片大小的上下限
2. Lambda安全参数边界
3. Theta手动参数

测试系列

1. 检查一致性和活性
2. 检查安全性
3. 腐败检查

智能合约重新设计/可靠性

TVM / WASM

文档

附录

符号

|  |  |
| --- | --- |
| 变量 | 意义 |
| tx | 交易 |
| l | 每个BFT实例内的事务的序列号 |
| LOG | 每个节点输出的完全有序的日志，LOG总是按顺序填充 |
| log | 一个BFT实例的日志，称为日志 |
| log[l : l′] | 在日志中编号为l到l'的交易 |
| log[: l] | log[1 : l] |
| λ | 安全参数 |
| α | 对方的哈希片段 |
| δ | 网络最大实际时延 |
| Δ | 网络延迟的上界（通常松散） |
| csize | 委员会大小，我们的协议集Csisie: =λ |
| th | th := ⌈csize/3⌉,一个阈值 |
| lower(R), upper(R) | lower(R) := (R − 1)csize + 1, upper(R) := R · csize |
| chain | 底层SnayLink协议中的节点局部链 |
| chain[: −λ] | 除了节点的局部链的最后一个块 |
| MinersOf(chain[s : t]) | 在链[s:t]中挖掘每个块的公共密钥。可能有几个公钥属于同一个节点 |
| {msg}pk−1 | 签名消息MSG，其验证密钥是PK |
| Tbft | 基础BFT方案的活性参数 |