Ethereum 生态及核心原理解析

何立宝

holybao@yqtc.co

深圳市优权天成科技有限公司

2017 年 8 月 19 日

内容提要

背景知识

- Blockchain Introduction
- Bitcoin Technology
- Ethereum Vision

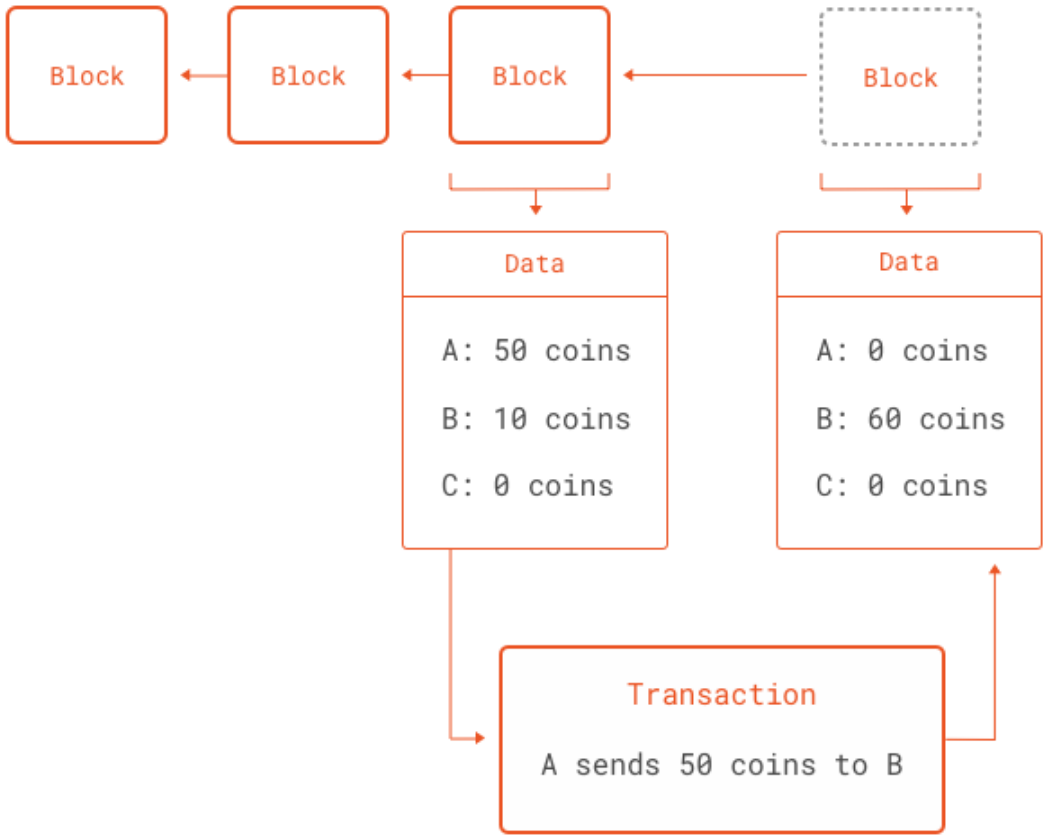
BLOCKCHAIN INTRODUCTION

- 狭义来讲，区块链是一种按照时间顺序将数据区块以顺序相连的方式组合成的一种链式数据结构，并以密码学方式保证的不可篡改和不可伪造的分布式账本
- 涉及领域：
  (1) 密码学
  (2) 博弈论/共识机制
  (3) P2P 技术
  (4) 分布式数据存储

BLOCKCHAIN OVERVIEW

- 区块链本质 $\Leftrightarrow$ 老子《道德经》中的“道”
  - 不同个体的理解有差异
  - 相同个体在不同阶段的理解有差异
  - 科学范畴 + 哲学范畴
- 典型范式:
  - 协议或合约的实施依赖可信第三方
  - 可信第三方用区块链代替
  - 区块链状态由协议或合约参与方协作达成共识

Blockchain Overview



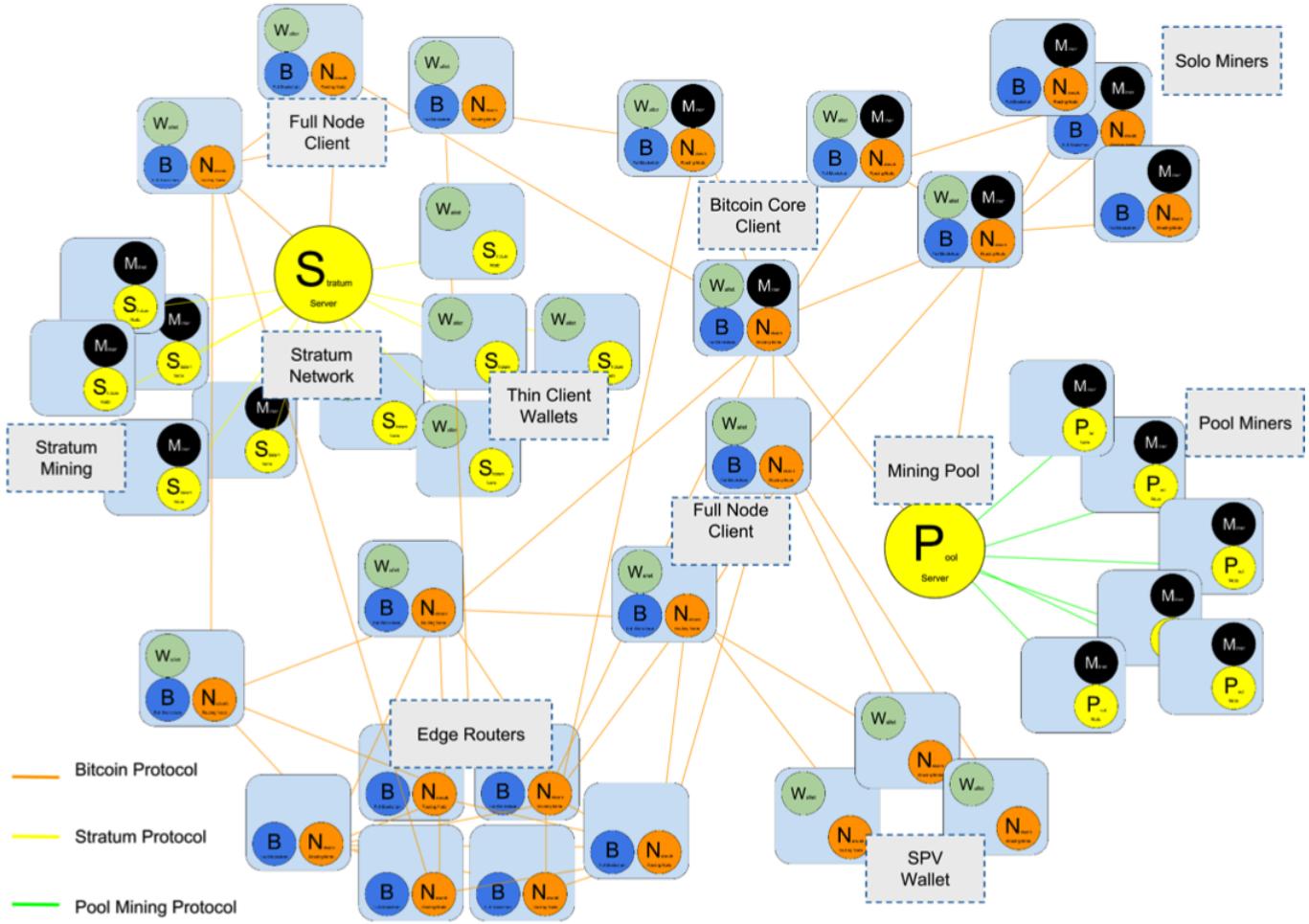
BITCOIN TECHNOLOGY

- 核心原理：UTXO model + POW
- 状态转换函数 APPLY(S,TX) -> S'：
  (1) 交易的每个输入
    - 引用的 UTXO 不存在 S 中，返回错误提示
    - 签名与 UXTO 所有者签名不一致，返回错误提示

  (2) 所有 UTXO 输入面额总值不小于所有 UTXO 输出面额总值，返回错误提示

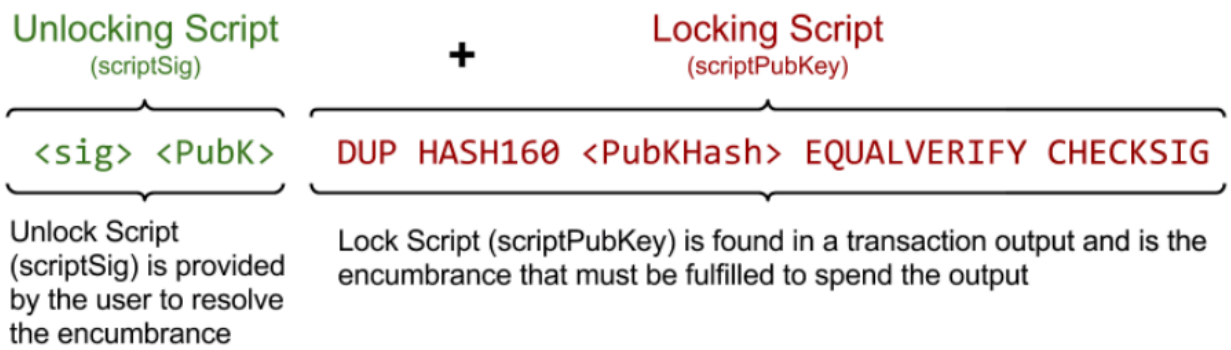
  (3) 返回新状态 S'，新状态 S 中移除了所有的输入 UTXO，增加了所有的输出 UTXO
- 节点功能组件：
  (1) Wallet
  (2) Minner
  (3) Full Blockchain Database
  (4) Network Routing

BITCOIN TECHNOLOGY



BITCOIN TECHNOLOGY

- 交易类型:
  (1) Pay to Public Key Hash (P2PKH)
  (2) Pay-to-Public-Key
  (3) Multi-Signature
  (4) Data Output (OP_RETURN)
  (5) Pay to Script Hash (P2SH)



BITCOIN TECHNOLOGY

- 四个局限性：
  (1) 缺少图灵完备性
  (2) Value-blindness：UTXO 脚本不能提供账户额度精细控制
  (3) 缺少状态：UTXO 只有已花费或者未花费状态
  (4) Blocchain-blindness：UTXO 看不到区块链数据

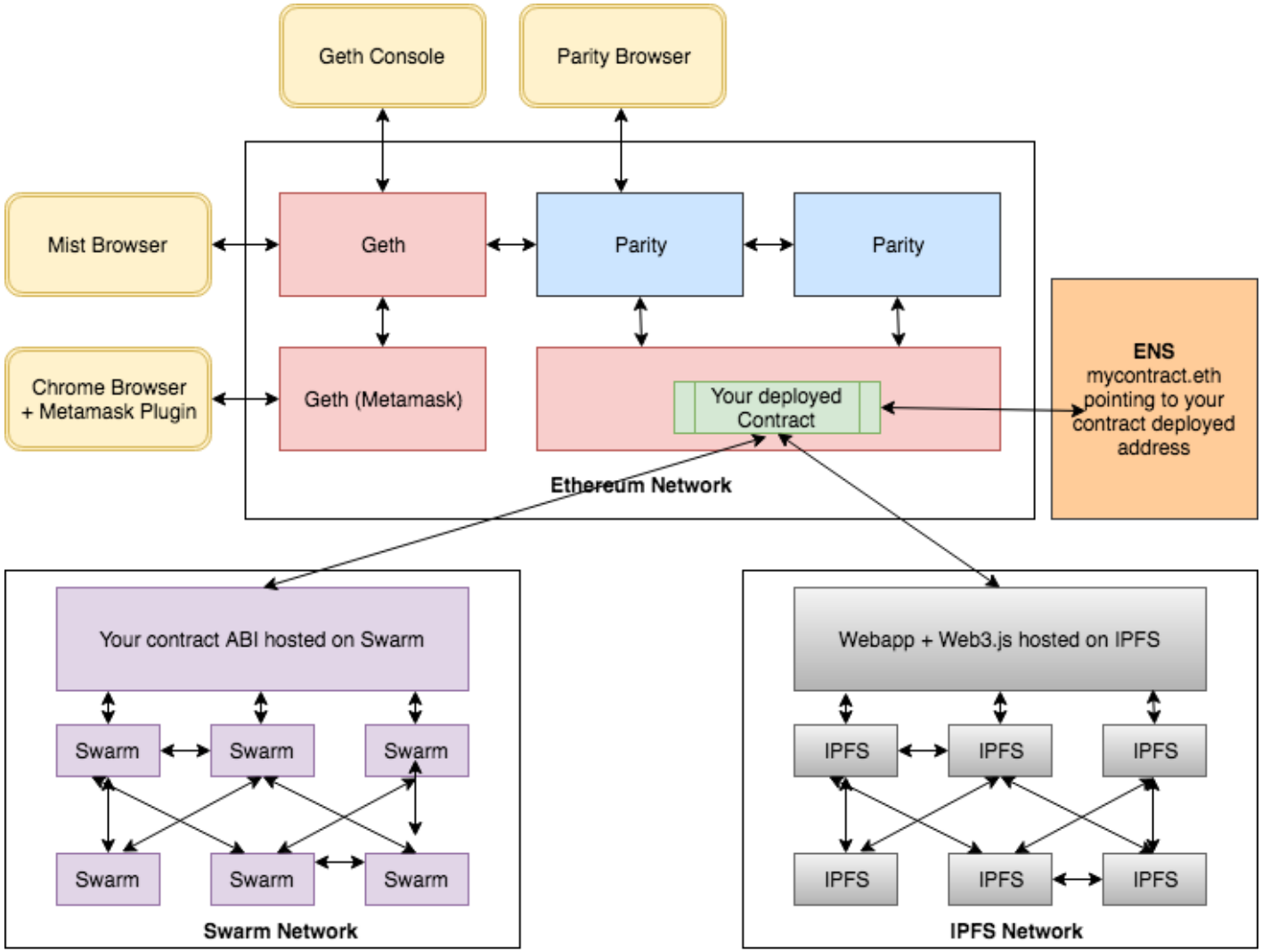
ETHEREUM VISION

- 目标：解决 Bitcoin 的四个局限性，提供一个带有内置的成熟的图灵完备语言的区块链，用这种语言可以创建合约来编码任意状态转换功能，用户只要简单地用几行代码来实现逻辑，就能够创建 Bitcoin、Litecoin 等已知区块链系统以及许多目前还想象不到的其它系统
- 定位：下一代智能合约和去中心化平台

共识机制

- Architecture Overview
- Some Related Technology
- Enterprise Ethereum Alliance

ARCHITECTURE OVERVIEW



Architecture Overview

1. Geth：Go 语言编写的官方客户端软件，通过 Geth Console 和 Mist Browser 访问
2. Parity：Rust 语言编写的非官方客户端
3. Web3.js：Ethereum JSON-RPC 封装而成的 JavaScript 库
4. Solidity：最流行的智能合约编程语言
5. Truffle/Embark/Meteor：开发 DApps 的常用管理框架
6. Metamask：与节点进行交互的 Chrome 插件
7. ENS：以太坊域名服务
8. Swarm：去中心化的内容存储和分发服务
9. IPFS：去中心化存储系统，与以太坊系统集成
10. Whisper：DApps 通信协议

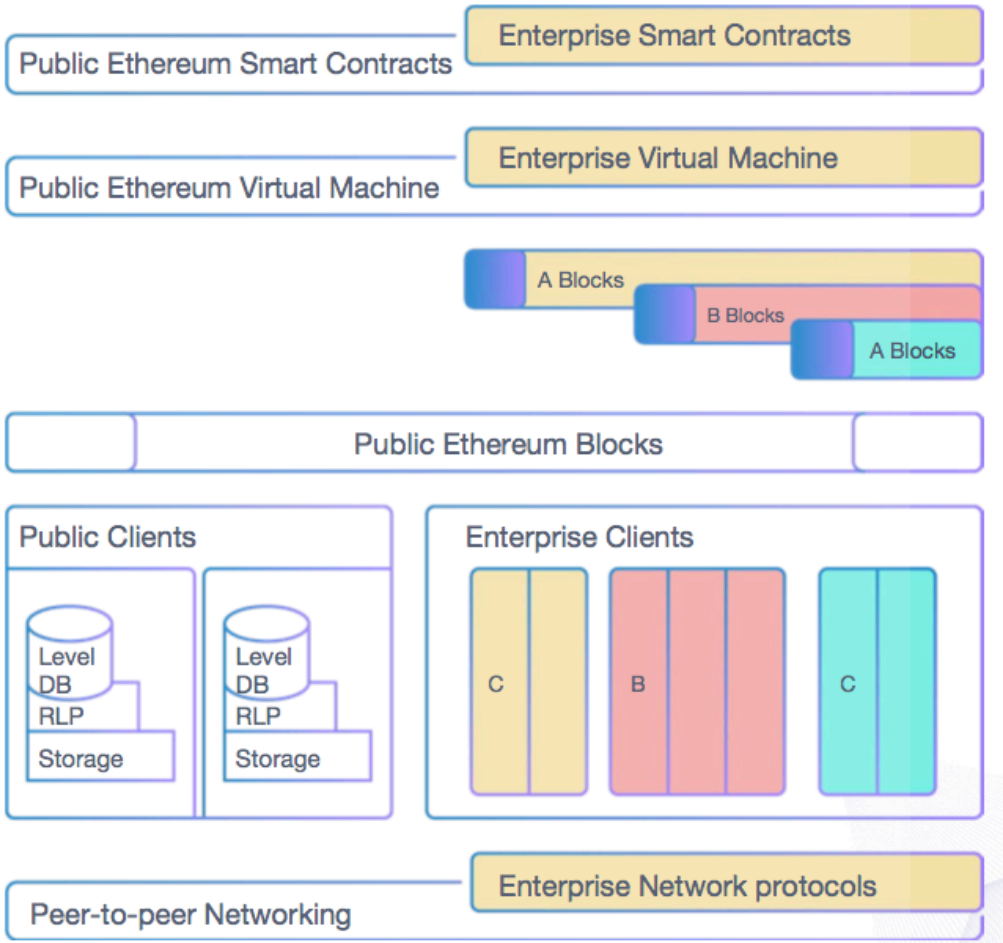
Some Related Technology

- 钱包相关：
  - 私钥形态
    (1) Private Key
    (2) Keystore && Password
    (3) Mnemonic Code
  - 分层确定性钱包：BIP32 + BIP39
  - 钱包 SDK：Consensys/eth-lightwallet、MyEtherWallet 等
- Ethash/Casper
- GHOST
- Trie: Modified Merkle Patricia Tree
- RLP: Recursive Length Prefix
- Smart Contract + EVM

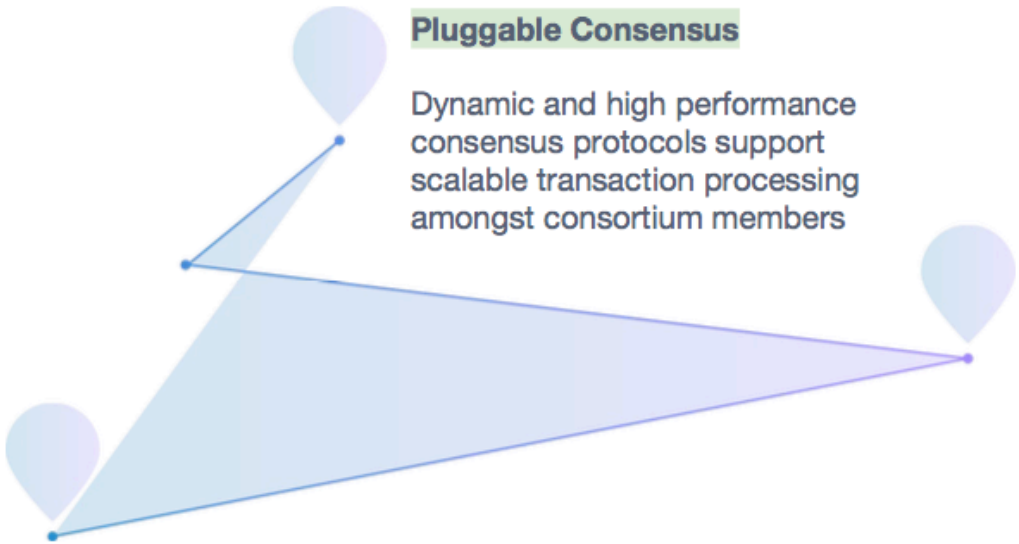
Enterprise Ethereum Alliance

- EEA：最大的区块链联盟，加盟企业数已达 150
- 技术目标：
  (1) 公有链的超集
  (2) 涵盖企业的附加需求：保密性、可扩展性、可授权
  (3) 项目落地
  (4) 专注规范，不制造产品
  (5) 区块链的管理
- 终极愿景：
  (1) 实现模块化，可根据场景满足所有公有链或者私有链的用例
  (2) 尽可能趋同公链和企业链的路线图
  (3) 挖掘潜在优势：数据反馈、数据管理、基础设施

ENTERPRISE ETHEREUM ALLIANCE



ENTERPRISE ETHEREUM ALLIANCE



Pluggable Consensus

Dynamic and high performance consensus protocols support scalable transaction processing amongst consortium members

Configurable Privacy and Security

A modular reference implementation supports development and comparison of trustless consortium paradigms

Benchmarking

Understanding features and performance of components helps to inform and integrate relevant enterprise use cases

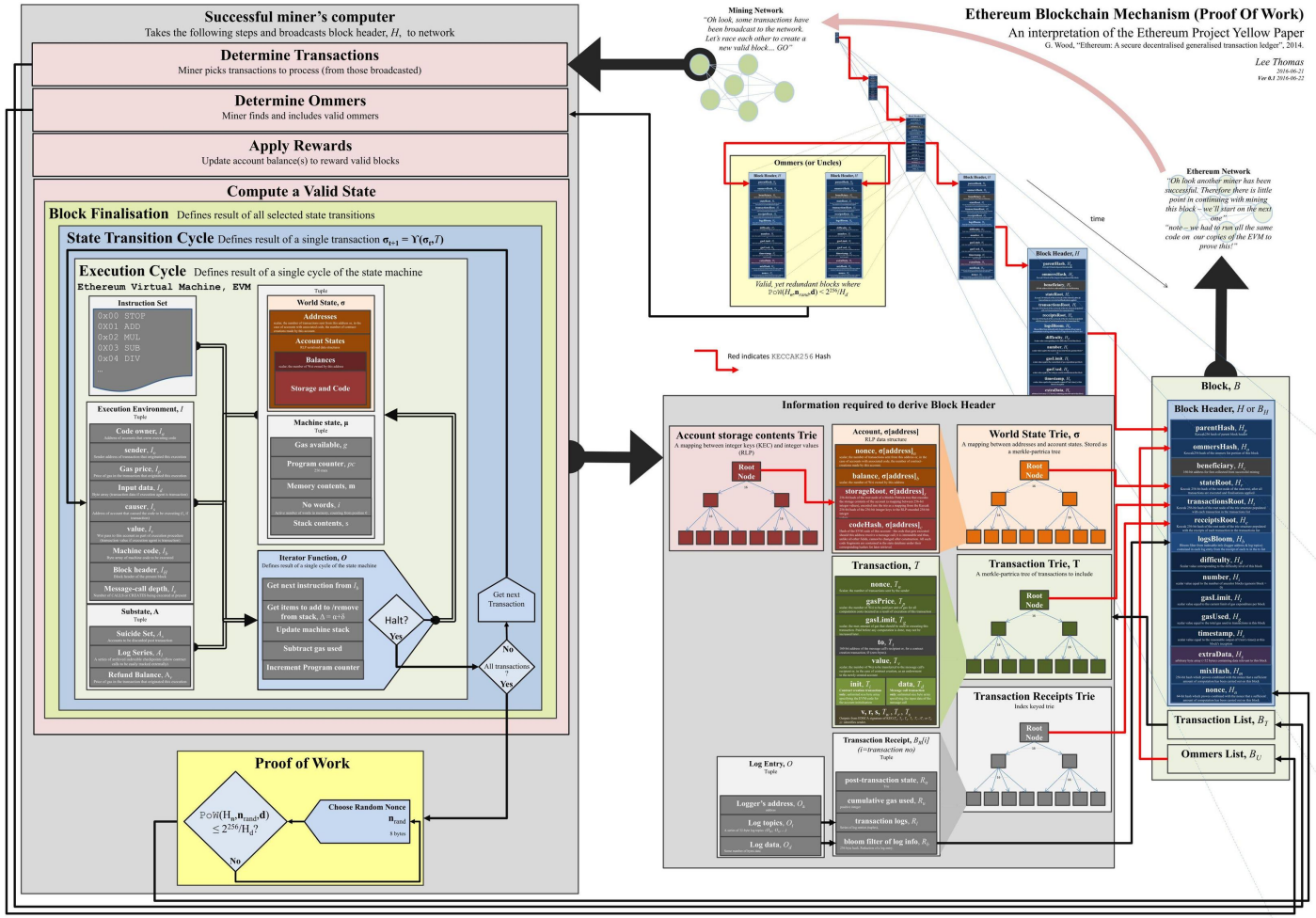
协议剖析

- Protocol Overview
- Block Architecture
- Account Model
- Transaction
- Reward

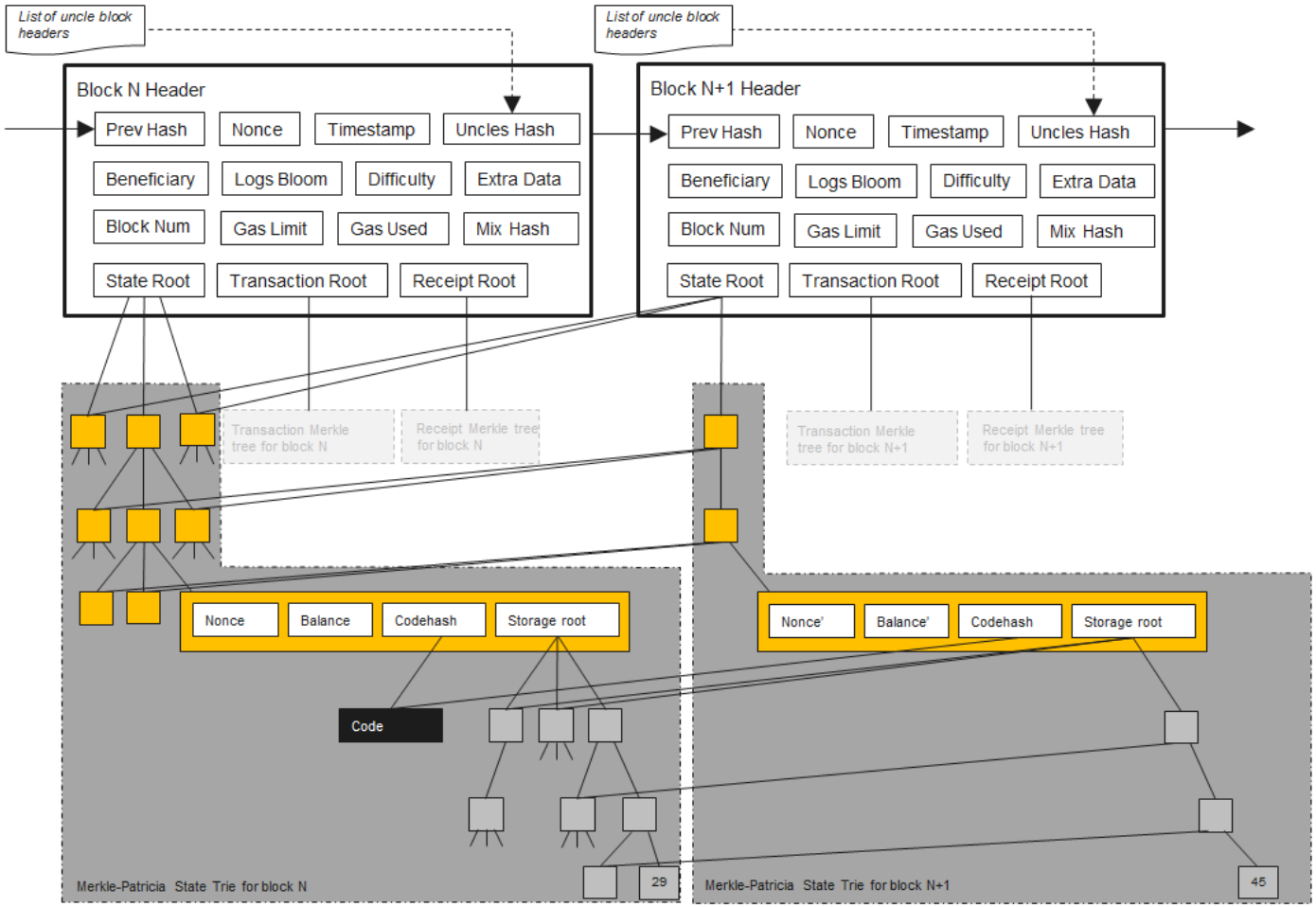
Protocol Overview

1. Block Architecture
2. Account Model
3. Minning Logic
   (1) Determine Transactions
   (2) Determine Ommers
   (3) Apply Rewards
   (4) Compute a Valid State

PROTOCOL OVERVIEW



BLOCK ARCHITECTURE



BLOCK ARCHITECTURE

- Blockchain Data = Chain Data + State Data
  - Chain Data: the list of blocks forming the chain
  - State Data: the result of each transactions's state transition
- Chain Data/区块结构： $B = (B_H, B_T, B_U)$
  (1) B_H： 区块头
  (2) B_T： 区块交易列表/状态转移函数集合
  (3) B_U： 叔块列表
- State Data/导出数据：
  - 账户数据
  - Transaction Receipts

Account Model

- 账户分类：
  - 外部账户 => 对应业务参与方，通过私钥控制
  - 合约账户 => 对应业务逻辑，只能通过外部账户控制
- 账户状态：
  - nonce：账户发起的消息调用总数，防重放攻击
  - balance：账户余额
  - storageRoot：合约代码执行状态
  - codeHash：合约账户关联的 EVM 代码

TRANSACTION

- 符号约定：
  (1) T_n: nonce
  (2) T_p: gasPrice
  (3) T_g: gasLimit
  (4) T_t: to
  (5) T_v: value
  (6) T_w, T_r, T_s：交易签名相关参数 v, r, s
  (7) init: 合约账户初始化时附带的 EVM 代码
  (8) data: 消息调用输入参数
- 交易类型：
  (1) 合约创建：$L_T(T) = (T_n, T_p, T_g, T_t, T_v, T_\mathrm{i}, T_w, T_r, T_s)$, $T_t = \varnothing$
  (2) 消息调用：$L_T(T) = (T_n, T_p, T_g, T_t, T_v, T_\mathrm{d}, T_w, T_r, T_s)$, $T_t \neq \varnothing$

REWARD

1. 区块奖励：$R_b = 5$
2. 叔块奖励：$(U_i - B_{H_i}) * R_b/8$，即叔块与当前块高度差的 5/8
3. 区块引用奖励：$||B_U|| * R_b/32$，即引用区块数的 5/32

前研课题

- Efficiency and Security
- Oracle

EFFICIENCY AND SECURITY

- 效率:
  - EVM 代码执行效率低，“伪”图灵完备
  - State Database 数据快速膨胀，分区?
  - Plasma: Scalable Autonomous Smart Contracts
- 安全:
  - 合约代码安全，包括制定工程规范、提供缺陷扫描工具等
  - 隐私保护的智能合约
  - 许可链以及联盟链成员管理等

ORACLE

1. Oraclize
2. RealityKeys

 Nakamoto S. Bitcoin: A peer-to-peer electronic cash system[J]. 2008

 Antonopoulos, Andreas M. Mastering Bitcoin: unlocking digital cryptocurrencies. " O'Reilly Media, Inc.", 2014

 Ethereum White Paper

 Ethereum Yellow Papaer