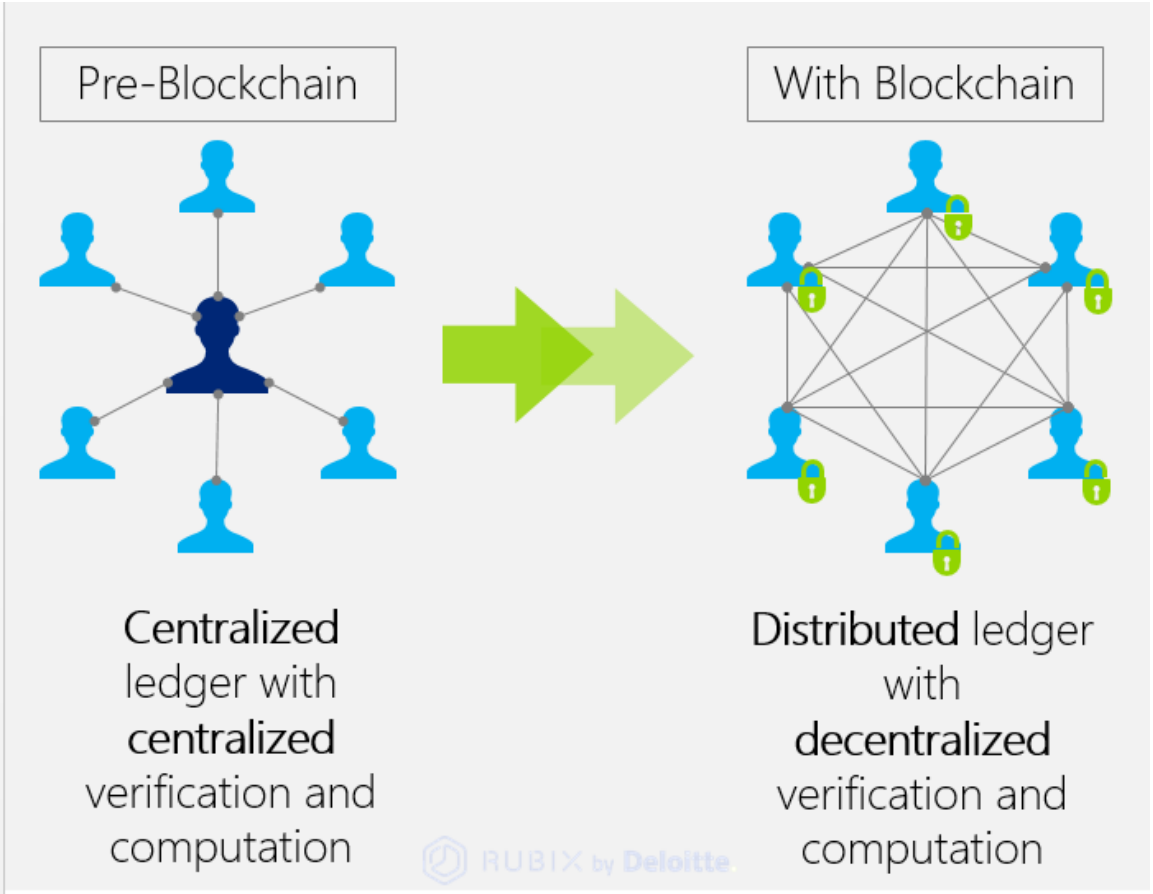
区块链应用开发简介

祝小翰 meter.io

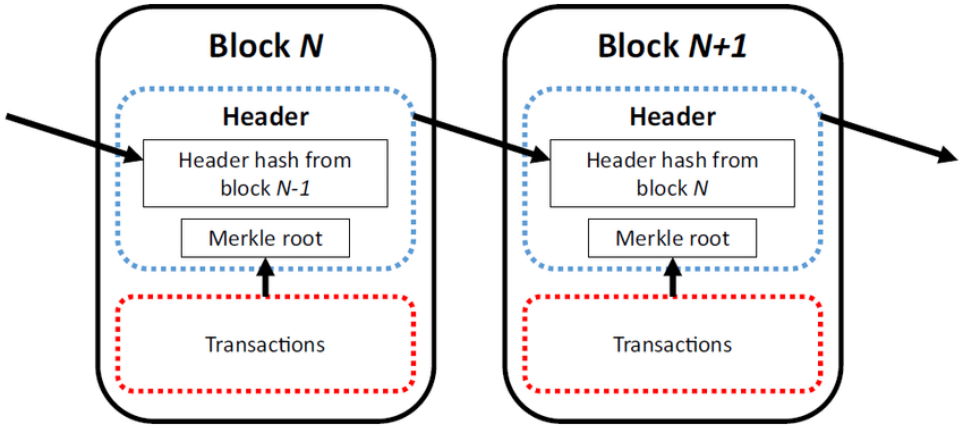
区块链基本原理

- 互联网应用模型
  - Client-Server架构
  - 高效，数据中心化存贮
  - 无条件信任运营者
- Web3
  - P2P架构
  - 每个节点同步数据，大量冗余备份
  - 任何一个节点都可以提供全部服务
  - 只需要对协议（代码）的信任



区块链的基本结构

- 链式数据结构
  - 分叉
- 状态一致性复制
  - 区块链的数据库就是状态机(State)，所谓共识就是如何保证所有的节点的状态持续一致
- 数据访问权限
  - 无需许可（但是一般有交易成本）
  - 读取完全公开透明
  - 通过加密算法对写入鉴权（只有对应的私钥才能改变一个账户的状态）



区块链共识算法

- 通过某种协议达到全网节点的数据一致
- 只影响交易的收录和排序
  - 交易本身安全性由加密算法保证
    - 即使共识失败，没有私钥也不能盗币
    - 双花是交易排序问题
- PoW和PoS不是共识算法
  - 防止垃圾节点占据全网多数
  - PoW的过程需要时间，所以造成延迟
- 共识算法种类
  - 概率共识
    - 最长链 (中本聪共识)
    - 最多传播 (雪崩共识)
  - 确定性共识
    - pBFT
    - HotStuff

区块链共识算法

- 扩容的两个维度
  - 吞吐量 （交易费太高）
    - 大区块，降低去中心化程度（ETH2.0 PoS不能更好解决的问题）
  - 时延（交易确认太慢）
    - 替换PoW （ETH2.0 PoS能更好解决的问题）

没有人讲的问题

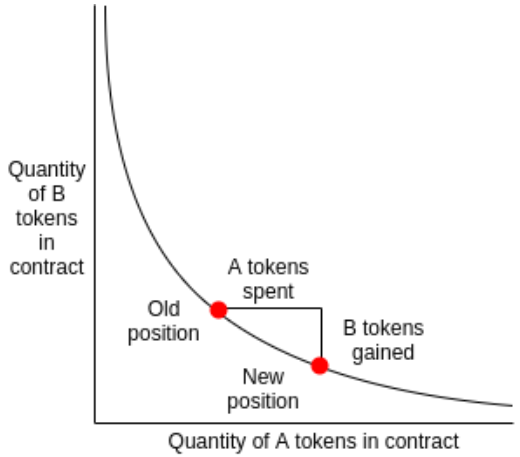
- EVM的处理瓶颈
  - EVM是单线程处理模型，当前最强CPU单线程处理能力在1000到2000tps之间
- 状态存储爆炸和I/O瓶颈
  - 高tps产生大量状态数据
  - BSC大约200tps，硬盘和I/O性能要求都已经到极限
    - AWS节点每月成本数千$

区块链应用

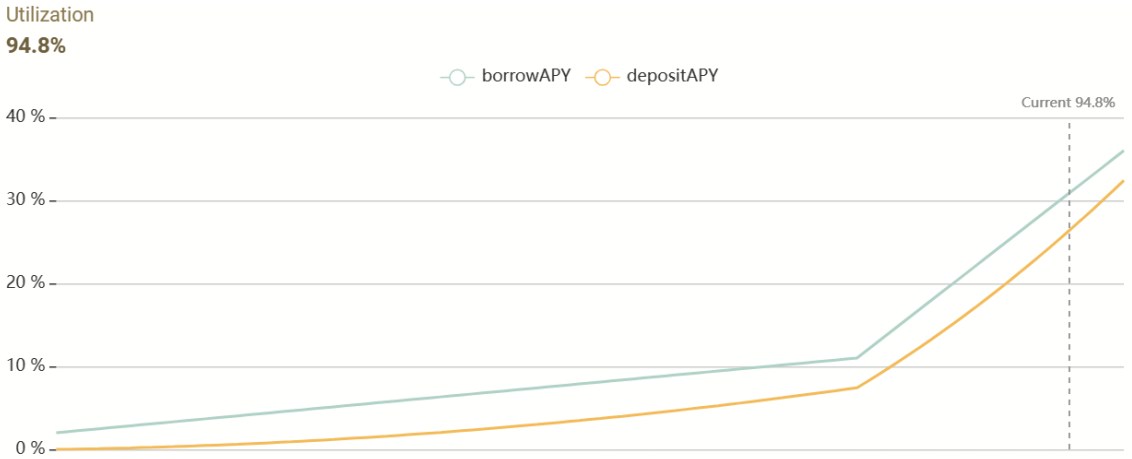
- 发币
  - ERC20
- DeFi
  - DEX，借贷，流动性挖矿
- NFT （ERC721，ERC1155）
  - 头像，游戏
  - 市场
  - 借贷

Bonding Curve – DeFi的基石

- DEX (Uniswap)
  - X*Y=K

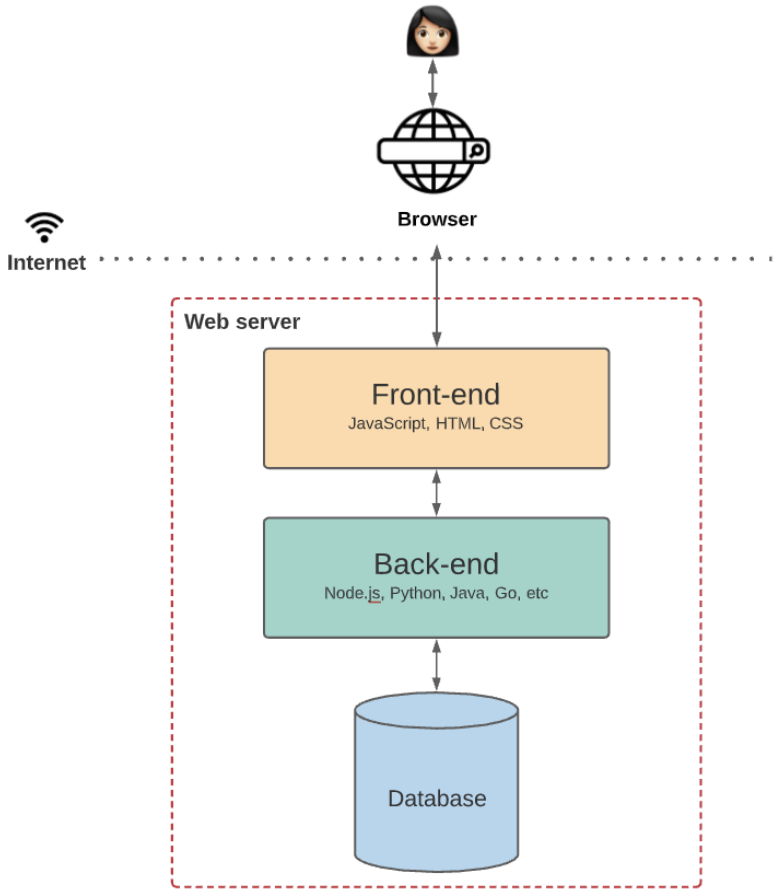


- 借贷 (compound, aave)
  - R=R_base+utilization*slope
- 流动性越多交易体验越好



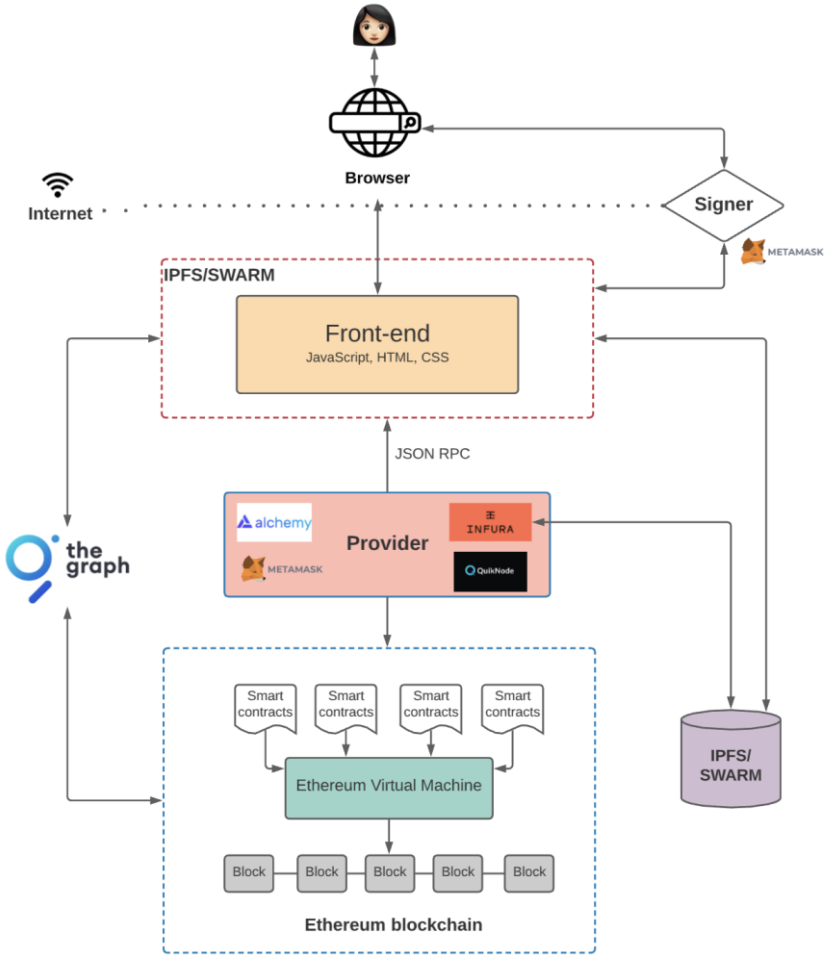
Web3应用开发架构

- 传统web2应用架构



Web3

- 前端
  - 钱包插件
    - 浏览器Plugin
    - Wallet Connect
- 后端
  - 智能合约（数据库+简单计算）
  - 数据流（The Graph）
- 存储
  - 中心化: AWS S3
  - 去中心化: IPFS



区块链开发例子-部署一个ERC20代币

- 准备开发环境
- 选择区块链
  - Chainlist.org (RPC 配置）
  - 浏览器，gas代币（水龙头）
  - https://docs.meter.io/developer-documentation/introduction
- 代币合约Github
  - https://github.com/meterio/tokenERC20
- 合约常用开发环境
  - Nodejs，npm，hardhat

Rollup (Optimistic)

- 交易在二层网络打包出块
  - 二层网络中心化处理
    - 可以过滤交易，决定打包顺序
- 压缩版交易数据打包在以太坊主网
- 用户可以在一层网络提交异议，要求交易在一层网络处理
  - 不是所有号称的rollup都可以要求交易在一层网络重放
    - Merkel checkpoint是不够的

Rollup (Optimistic)

- 优点
  - 采用以太坊一层网络确保安全性
    - 进出二层网络的资金安全
- 缺点
  - 争议处理造成用户取款需要长时间等待
    - 采用快速跨链桥放弃安全性优点
  - 交易成本高
    - 压缩交易打包到以太坊一层成本高
    - 争夺一层存储和I/O资源
  - 单个交易的大小可能小于一层网络

不同Layer 1的特点

- 网络模型
  - 强同步网络假设
    - 网络节点之间的通信永远良好，时间在预设范围内高度同步
      - Cosmos，Solana
  - 最终同步网络假设
    - 网络节点之间的通信可能中断，但最终能联通并达到同步
      - HotStuff (Meter), Avalanche, ETH2.0
      - PoW自带时间同步
- 出块节点选择
  - 非核心共识问题，提高区中心化，降低过滤和影响交易顺序的可能
- 数据结构
  - EVM只能支持链式结构 (DAG will not work)

不同BFT共识的复杂度

Protocol	Correct Leader	Leader Failure	f Leader Failures	Responsive
DLS	$O(n^4)$	$O(n^4)$	$O(n^4)$	
PBFT	$O(n^2)$	$O(n^3)$	$O(fn^3)$	✓
SBFT	$O(n)$	$O(n^2)$	$O(fn^2)$	✓
Tendermint/Casper	$O(n^2)$	$O(n^2)$	$O(fn^2)$	
Tendermint/Casper*	$O(n)$	$O(n)$	$O(fn)$	
HotStuff	$O(n)$	$O(n)$	$O(fn)$	✓