LAYER 2

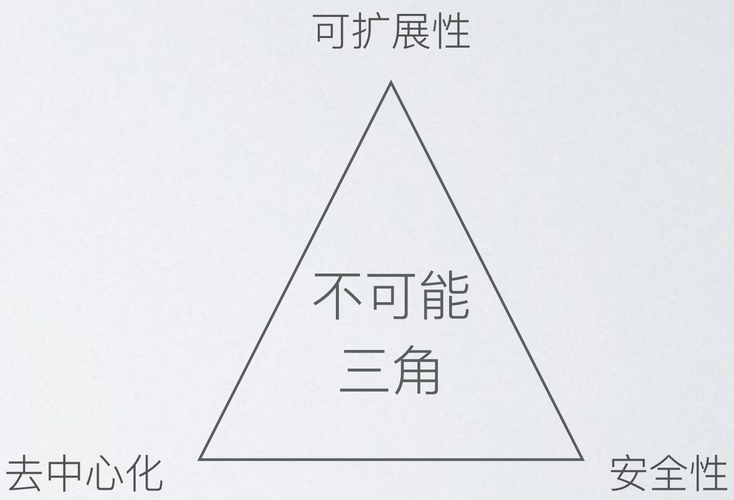
登链社区 - Tiny熊

要点

- 以太坊的扩展性问题（为什么需要 Layer2）
- 有哪些扩容方案
- Rollup
- 模块化区块链

为什么需要 Layer2

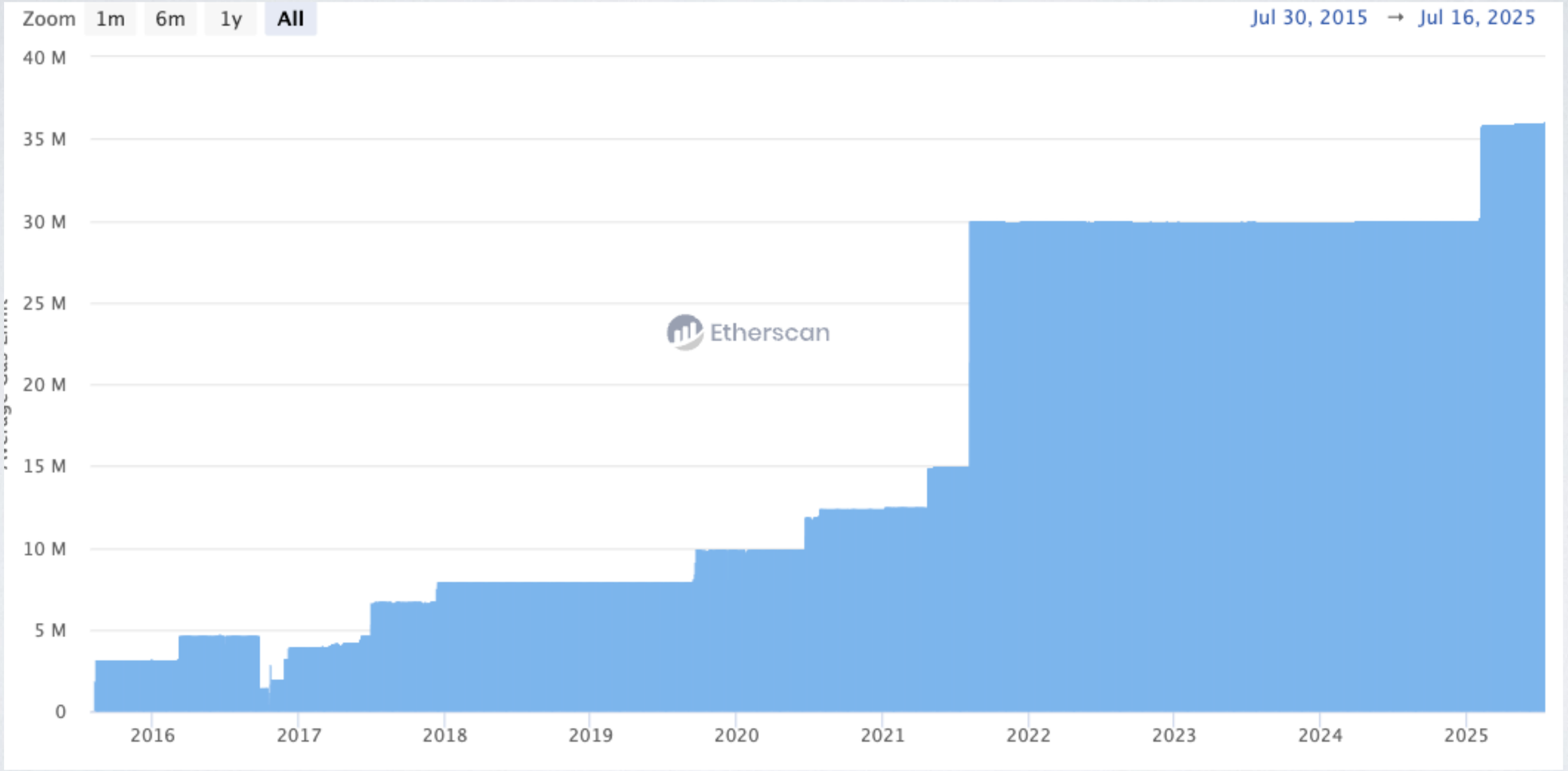
- 以太坊的扩展性问题：吞吐量低、Gas 费用高、确认时间长
- 以太坊吞吐量: 12 秒 4000 万+ gas
- Web2 支付系统：
  - VISA / 支付宝：20000 + TPS



扩容方案

- 一层扩容：
  - 提升 Gas limit： 过去提升 gas limit 比较保守， 担忧状态增长和带宽需求不利于去中心化，未来将加快，参考： Vitalik 讨论帖 , EIP-7938
    - zk 技术的成熟，提升 gas 更有条件，验证成本更低。
  - 分片：每个分片链存部分链数据
- 二层扩容：把一些交易放到链外执行（减轻链的压力），实现扩容

扩容方案



https://etherscan.io/chart/gaslimit

二层扩容方案

- 侧链
- 状态通道
- Plasma
- Rollup
  - Optimistic Rollup
  - zkRollup

侧链

- 独立于以太坊主链的区块链，通常拥有自己的共识机制，通过跨链桥互通。
  - 侧链与 Alt Layer1 (如 BNB Chain) 并没有明显的界限。早期一些 altLayer1 将自己成为侧链， Polygon PoS 是最知名的侧链
- 通过跨链桥将资产转移到侧链上，在侧链上进行 DEFI 交易
- 以太坊没法保证资产在侧链的安全性（不继承以太坊的安全性）
  - 主链能否承载安全性关键指标：用户能否将资产安全带回到主链

状态通道

- 适用于高频双边交易（交互）场景
- 状态通道工作流程：
  - 开启通道：参与者将资金存入合约中，
  - 更新通道：参与者在链下通过双方签名确认交易信息（nonce、当前状态等）
  - 关闭通道：双方同意后，将最终签名提交到链上合约，在挑战期内无争议，资金根据最终状态分配并解锁。
- 典型项目：Raiden Network， Connext、Celer （后面两个现在重心调整为做跨链）
- 状态通道：安全性高， 但：无法与没有建立通道的人交互、要求参与方保持活跃（或者使用 Watchtowers）。

PAYMENT CHANNELS

1. Using a smart contract, lock the state. Open Payment Channel

3. When state changes are submitted, final balance from total payments submitted on to the chain

2. Off-chain, fast transactions

.123

.0005

.03

.0004

.09

.0007

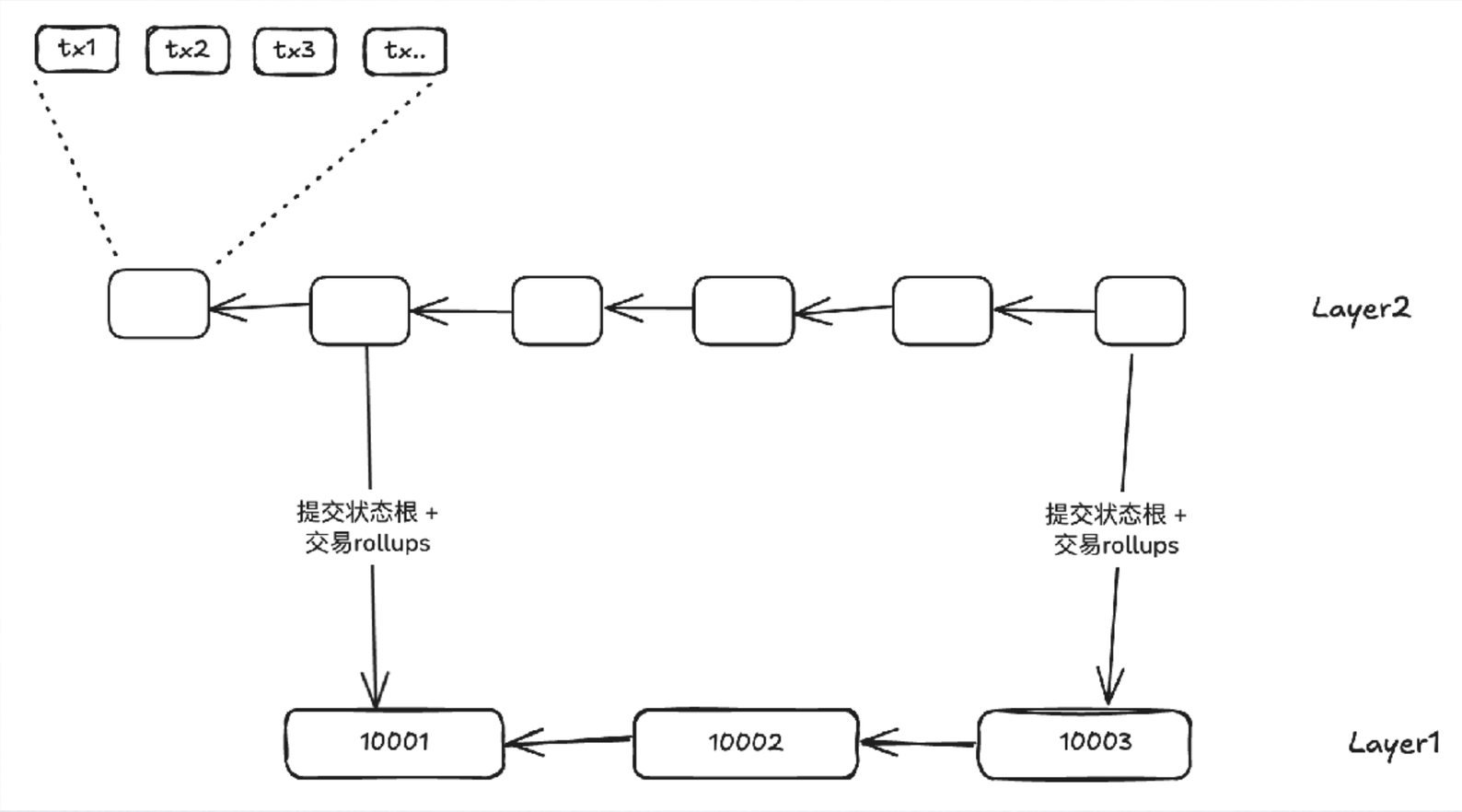
Plasma 方案

- 创建一个颗子树链，交易在子树链上进行，向主链提交的子树链上一个批次交易后的根哈希。
- 由于主链没有有效的子链交易数据，无法验证子链运营商作恶。欺诈证明实现难以实现。

最近有一个撞名的 Plasma 公链(比特币侧链），稳定币结算链，支持零 Gas 交易 USDT 转账

Rollup 方案

- Rollup 不仅仅将二层链执行后的状态树提交到主链上，同时将所有的交易“卷起”（rollup）打包发送到主链。
- Rollup 可以有更快的确认时间（通常 2-3 秒）
- 主链有二层链提交的压缩的所有交易，因此可以重放来验证二层链的交易的正确性，防止二层链运营者做恶。



Rollup 分类

- 如果交易在 Layer1 重放交易，无法实现扩容效果…
- 乐观（Optimistic） Rollups：乐观的假设所有的交易都是正确执行的，只有在被挑战时才验证。
  - 当前市场的主流选择
  - 代表项目：Arbitrum One、Base、Optimism、Unichain、HashKey Chain
- ZK Rollups：通过 ZK（零知识证明）算法交易的有效性证明，证明提交到主链，通过合约验证。
  - 代表项目：Starknet、ZkSync-Era、Linea、Scroll、PolygonZkEvm、Tikao

https://l2beat.com/scaling/summary

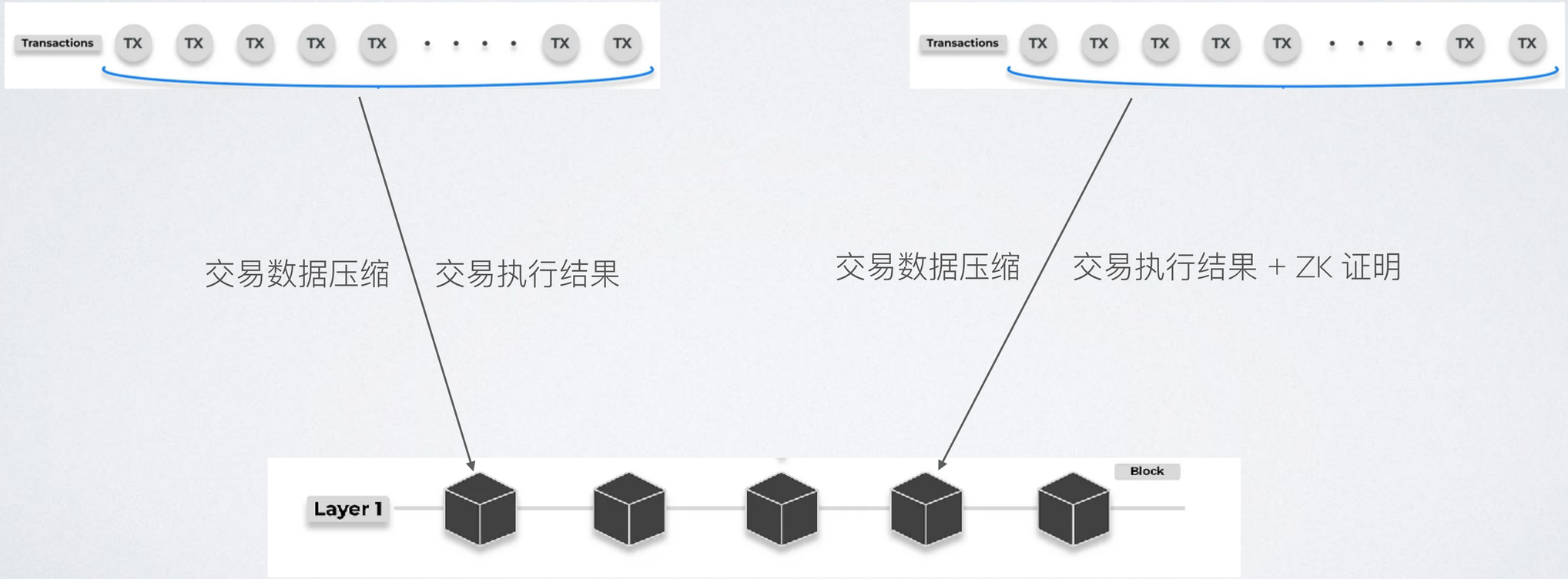
零知识证明

- 零知识证明（ZKP / ZK）：允许一方（证明者）向另一方（验证者）证明他们知道某个特定的信息，而不需要透露任何关于这个信息的具体内容。
- ZKP的两大特性：零知识 和 简洁性 - 验证效率高(证明复杂)
- 两种证明方式：
  - 交互式：多轮问答
  - 非交互式证明： 只需要提交一次证明， 验证方自行验证 （不同的方案：zk-SNARK、zk-STARK、Bulletproofs …）
- 三个主要应用方向：计算证明、隐私证明（Aleo、Zcash）、数据压缩（FileCoin…）
  - ZK Rollups （计算证明）：借助 ZKP 技术，节点可以将大量的计算外包给链下节点，链上只需要验证链下提交的计算结果和计算证明就可以知道计算是否正确

乐观Rollup vs zkRollup

乐观的认可 Layer2 执行的结果，除非被挑战

使用 ZK 验证 Layer2 执行的结果



乐观Rollup vs zkRollup

	乐观 Rollup	ZK Rollup
优势	EVM 兼容性好 更低的 Rollup 成本	安全性更高 更快的提款时间
不足	从 Layer2 提款到 Layer1 需等待数天挑战期	较难实现等效 EVM
主要链	Base、Arbitrum、OP Mainnet、HashKey Chain	ZKsync、Starknet、Scroll Linea

乐观 Rollup 框架

- 主要两个实现技术方向：
- OP-Stack：Optimism 团队开发，模块化、免费开源（MIT）开发框架，允许开发者轻松创建自定义的 L2 。
  - 被大多数 layer2 项目选择： Optimism、Base、Unichain、HashKey Chain
  - SuperChain：通过 OP-Stack 实现多个 L2 链的标准化和**互联**，形成一个统一的“超级链”网络
- Arbitrum Orbit （ Nitro 核心技术栈 + Stylus 增强模块 ）：开源（BSL），轻松创建自定义的 L2 L3
  - 欺诈证明支持更好，支持多虚拟（ EVM 与 WASM ）， 不够友好开源授权协议，采用偏少
  - 使用项目：Arbitrum One、Arbitrum Nova

zk Rollup 框架

- AggLayer CDK（原 PolygonCDK）： CDK - Chain Development Kit， 可多技术栈的自定义Layer2链，实现更高的互操作性和安全性
  - zkEVM 特点：在 zk 电路中解释 EVM 字节码，兼容性更好
  - 使用项目：Polygon zkEVM 、Merlin Chain（比特币 Layer2）、X-Layer2
- ZK Stack：zkEVM，ZKsync 团队开发，构建 Elastic Network
  - 将 Solidity、Vyper 等代码通过 zksolc 和 zkvyper 转换为 zk 友好电路，效率更好
- StarkNet（StarkEx）： zkVM - CairoVM

不同类型的 ZK–EVM

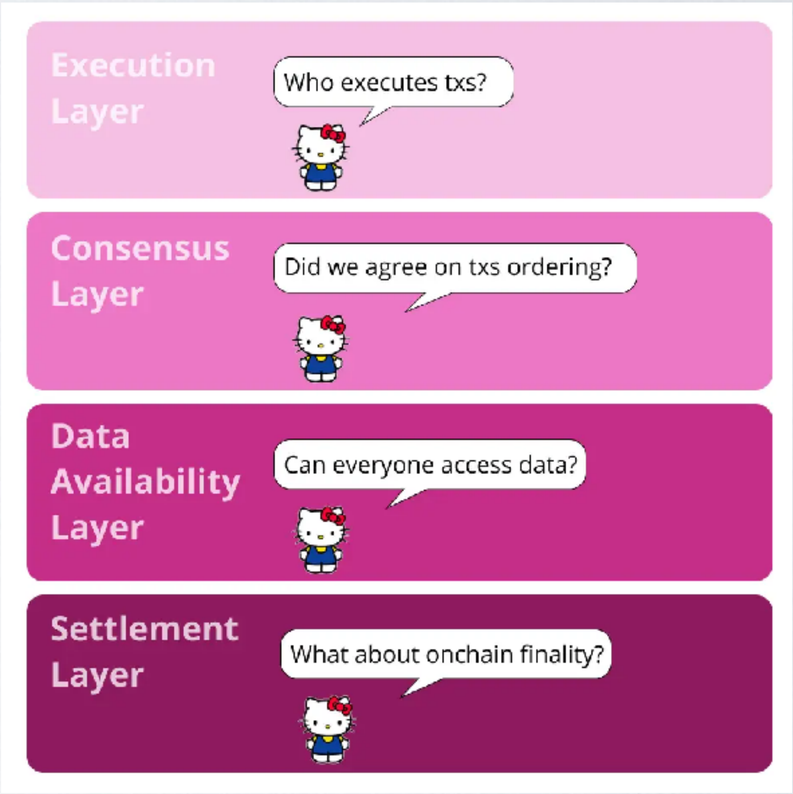
完全以太坊等效　　EVM 等效　　兼容
高层语言复用

	Type 1	Type 2	Type 2.5	Type 3	Type 4
Source code					
Bytecode				删除某些字节码支持	
Precompiles					
Gas pricing			增加 gas 成本 来优化证明生成		Solidity 直接编译为 ZK 电路
State trees					

性能更好

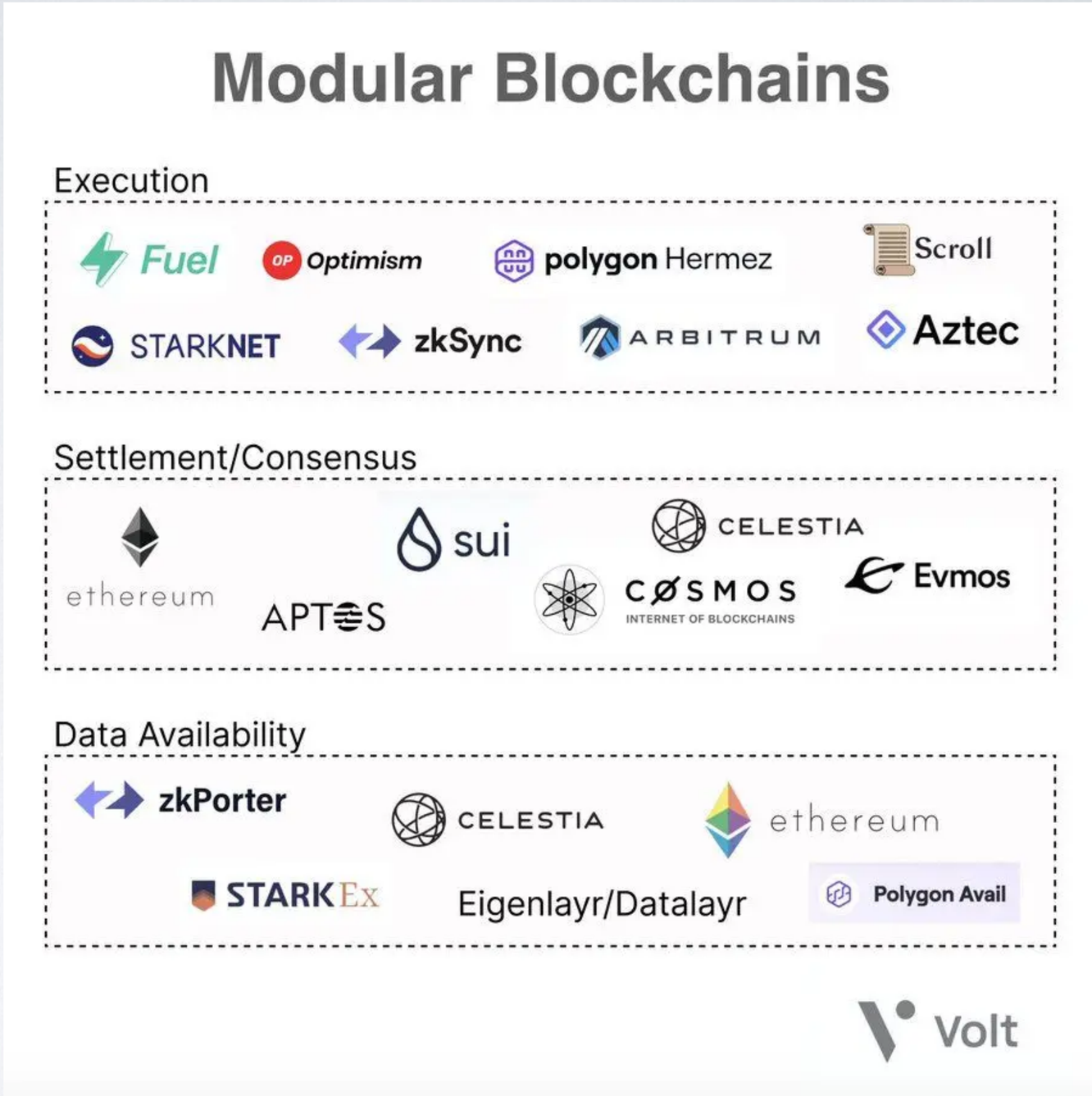
模块化区块链（Modular）

- Rollup 发展的同时，发展出了模块化（Cosmos 生态较早提出）的概念，将 4 大核心功能分为 4 层：
  - 执行层：交易在哪里执行
  - 数据可用（DA） 层：交易数据发布到哪里
  - 共识层：数据如何达成共识（Layer2 排序）
  - 结算层：最终的状态数据在哪里
- 功能模块化，便可对某些模块替换



典型的 Layer2 相当于充当了以太坊的执行层

在不同的模块
可以使用不同的技术
出现了各种不同的项目

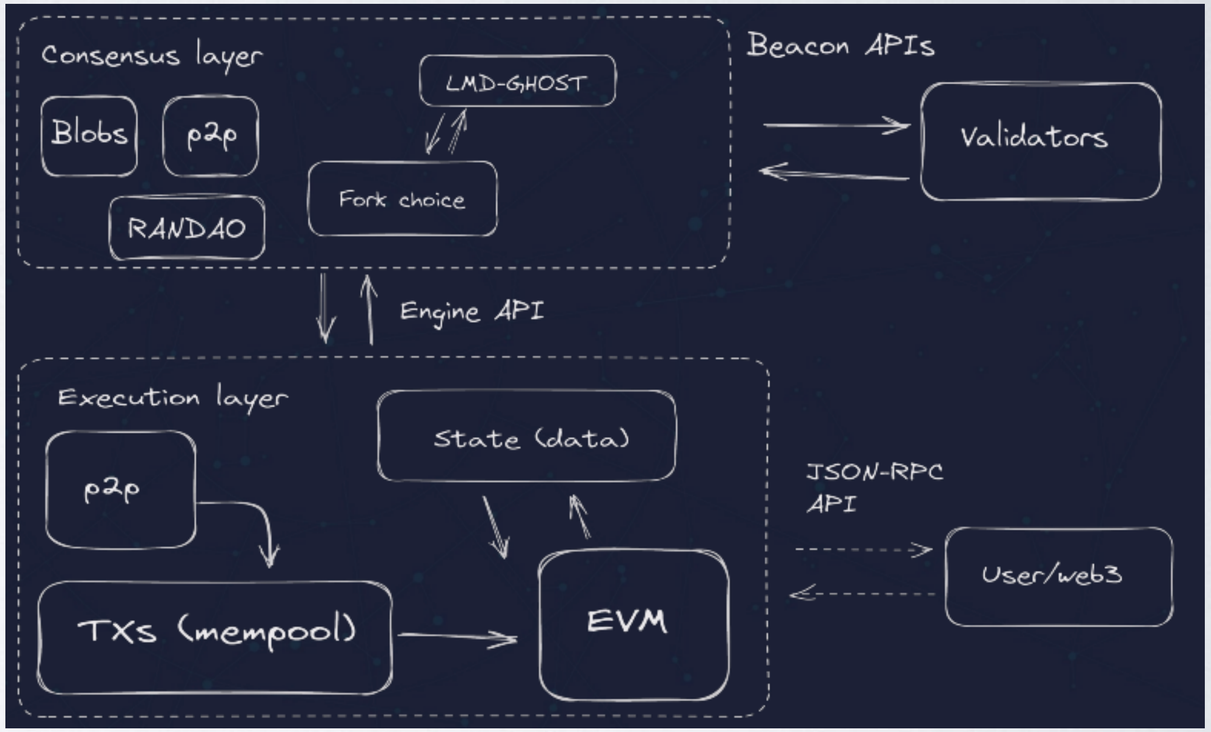


不同的执行层

- EVM 作为执行层，有一些不足：
  - 无法并行执行、一些指令 ZK 证明效率低（ KECCAK256 等）
  - 但 EVM 生态强大多数项目（不仅仅 Layer2 ）依旧选择兼容 EVM： Arbitrum、Base、Op-Mainnet
  - Vitalik 有想法将 EVM 切换为 RISC-V 虚拟机 (CPU 指令集架构： CISC 、RISC： ARM、RISC-V)
- 执行层使用不同的虚拟机：
  - zkEVM： 智能合约在 zkEVM 中运行后, 会生成一个 zk 证明, 它可以证明运行状态转换的有效性，典型项目有： Taiko、Scroll、Linea、ZkSync、PolygonzkEVM 等。 不同的实现，与 EVM 的兼容性有差异（以太坊也投入在L1 使用 ZK 证明）
  - zkVM: 证明任意程序运行的有效性，典型的项目：RISC ZERO 、SP1 （RISC-V） 、Jolt(RISC-V)、Starknet(CairoVM) 、 ZKM （MIPS） 、更多
  - SVM: 使用 Solana 虚拟机，更好的并行性能， 典型的项目： SOON、Eclipse
  - UTXO 模型虚拟机（FuelVM） : 每笔交易操作的是独立的 UTXO，不会相互冲突，实现并行执行
  - WASM：通用的虚拟机，支持通用的编程语言，如 Rust、JavaScript 、C++ 等， 典型的项目： Arbitrum Stylus

以太坊作为 DA

- Data Availability: DA
- “正统” layer2 将 交易数据 rollup 以太坊， 如： Batch Txs
  - rollup 压缩数据作为一个普通交易的 input data（calldata）
  - Arbitrum rollup Tx，Op Mainet rollup Tx
- Rollup 压缩数据作为 blob - Binary Large Object （EIP-4844 提出）
  - 在**共识层（信标链）**开辟的空间
  - 引入了一个 Tx Type3 交易类型，发起 blob 交易
  - 如：Base Rollup 、Blobscan



EIP4844 – Blob

- 每个 blob 大小：4096 * 32 bytes =128 K，例如：Blobscan
- 每个区块可最多 16 个blob， 共 2 M
- blob 的生命周期为 18 天（ 4096 epoch），18 天后节点可删除，使得 blob 存储成本更低（calldata 永久存在交易中），blob 单独计费， 提交 Blob 交易时，同时提交以下两个数据，以便验证 blob 的正确：
  - KZGCommitment： blob 数据对应多项式的承诺（用一个短小的值代表整个 blob 数据）
  - KZGProof： 用于 blob 数据的正确性（验证数据来自符合承诺的 blob ）
- blob 无法被 EVM 访问的， 只能访问到当前交易的 BLOBHASH （KZG 承诺的 Hash）

不同的 DA

- 也有一些项目寻求更低的成本考量（但数据不可用的风险更高），选择 rollup 到其他地方
  - EigenDA（EigenLayer EigenCloud 提供的 DA 服务）： Mantle 、Fuel
  - Celestia （公链）：Manta、Eclipse
  - Avail（公链）： Lens
  - Arbitrum AnyTrust： Arbitrum Nova
    - 允许链选择 数据委员会（Arbitrum DAC： Data Availability Committee ）

https://l2beat.com/data-availability/summary

Rollup 变体方案

- 如果 Rollup 不将交易数据发布到以太坊，这些方案被称为：
- Validium： 只将状态根和证明提交到主链，交易数据提交到外部（zk-Rollup 变体）
  - 项目： DyDx（StarkEx）
- Optimium： 只将状态根提交到主链（Optimistic Rollup 变体），如果数据不可用，则欺诈证明无法进行
  - 项目：Arbitrum Nova

Layer2 的问题

- 各 Layer2 去中心化及安全性程度并不一样
  - Stage 0： 完全的辅助轮，由 operators 运行
  - Stage 1： 具备证明系统，可去中心化提交证明，无需 operators 时，用户可退出
  - Stage 2 : rollup完全由智能合约管理，证明系统是无需许可的
- Layer2 割裂、互联互通性不够好
- 流动性被分散
- 以太坊没有足够捕获 layer2 价值

Rollup 新动向

- Based Rollup: Rollup 网络的排序发生在 rollup 所基于的 L1 上。
- Native Rollup: 加入一个 EXECUTE 预编译来验证 Rollup 交易
- Ultrasound Rollup： Based Rollup + Native Rollup

作业（选择题）

- 理解以太坊 Layer2
- https://decert.me/quests/d8d09ca5-005b-4bcc-9510-90070afe770b