Web3 时代“开源”的意义

A new open source deal for Web3

演讲嘉宾：Boris Mann

fission

BORIS MANN

- 15 years ago, worked on building out the Drupal community, built the first commercial company around Drupal, helped form Drupal Association
- Last year, focused on Ethereum community, helping out with the Ethereum Magicians, Core Devs
- Fission is building developer tools: decentralized Firebase with user controlled data

fission

Fission 致力于为开发者创造“由用户掌控数据的去中心化 Firebase”。

fission

DEFINITION OF OPEN SOURCE

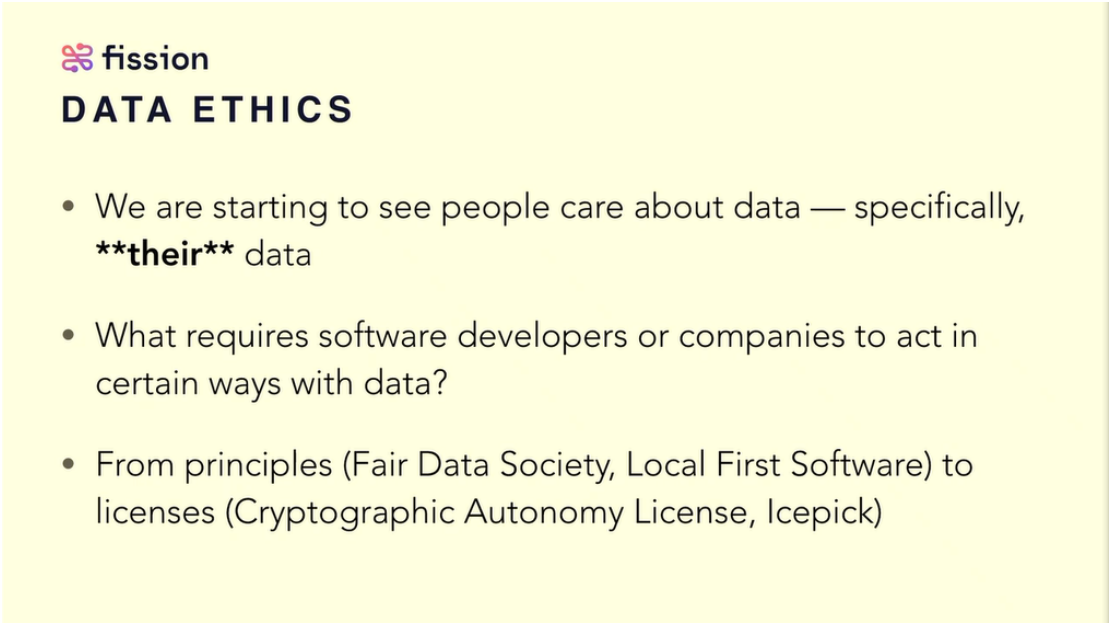
1. Legal innovation of licenses
2. Way of working together on code, collaboratively, asynchronously, remotely
3. Ideology of code re-use & sharing

“开源”到底指什么？

1. 许可（license）的合法化创新
2. 异步、远程的代码合作方式

3. 重用和分享代码的思想观念

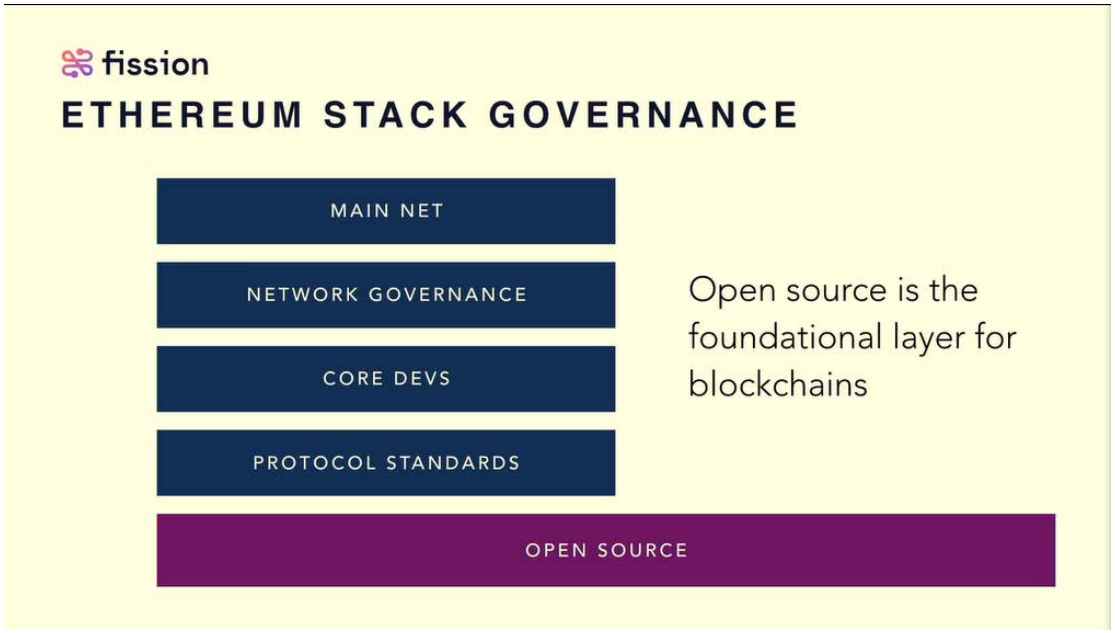
Boris 将“开源”比作免费领养一只未经驯养的小狗，开源代码需要经过自己后期的培育和训练才能发挥出最优功能。



用户越来越在乎自己的数据或代码的所有权和用途。这不再是一个相信口号和自觉遵守原则的时代（例如谷歌的“Don’t be evil”），因此在开发软件的时候需要遵守一定的许可。目前出现了一些新的许可，例如 Crypto Autonomy License 和 Icepick。

关于区块链、Web3 和开源的思考：

1. 区块链和 Web3 的用户和建设者需要怎样关注开源？
2. 关于“开源”我们能学到什么？
3. 开源面临怎样的挑战，是否能从区块链和 Web3 的工具中寻求到解决方案？
4. 我们为什么要关注开源及其发展？



开源代码层是区块链的基础，在这之上才能建立协议标准，无开源就无区块链。

fission

DAPP DEVS NEED TO CARE

- If you're building a dapp on top of the Ethereum Stack, you need to care about the layers of the stack
- Get involved in the stack, from standards to code contributions, to make sure your dapp will succeed on top

---

对于 DApp 开发者来说，Boris 建议他们应该参与到协议标准和代码的贡献之中，因此才能确保自己的 DApp 在此基础上获得成功。

fission

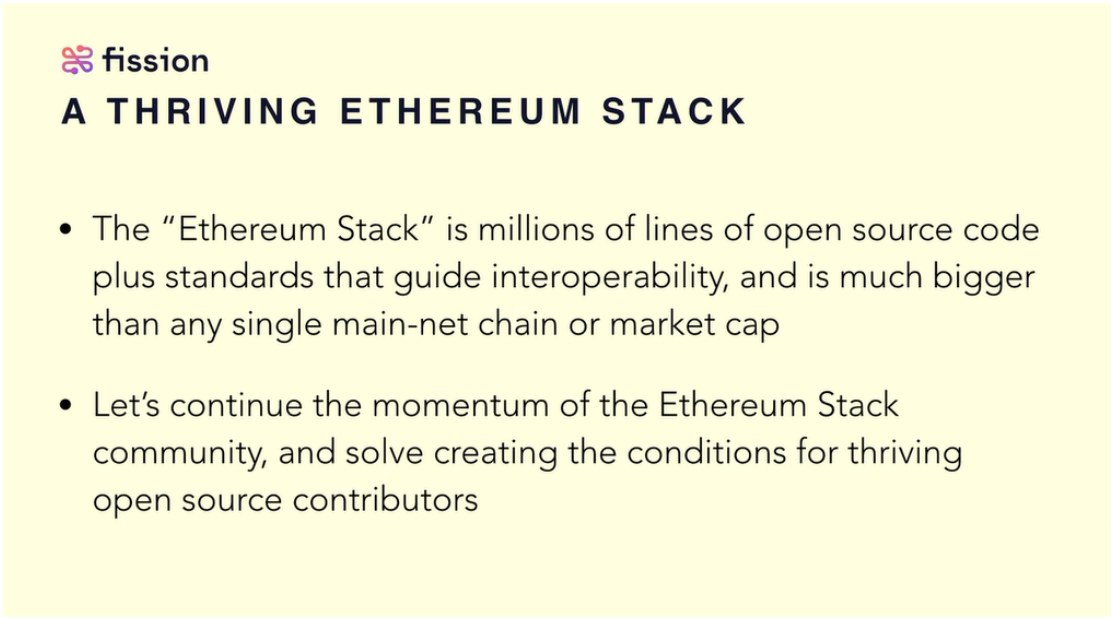
A NEW DEAL

- Open source is a key enabler and foundation for Blockchains & Web3
- We are practicing **commons based peer production** while building & operating global public blockchains
- Figuring out a New Deal for Open Source is crucial if we expect to keep our systems running for decades

开源是区块链和 Web3 的重要基础和发展动力。

在全球公链的建设和运行中，我们要不断实践基于公共资源的对等生产方式。

要保持区块链系统的长足发展，就必须要找到开源的“新政”。



以太坊堆栈由数百万计的开源代码和协议标准组成，并且由此促进其互操作性，它比任何单一区块链的市值都要大得多。

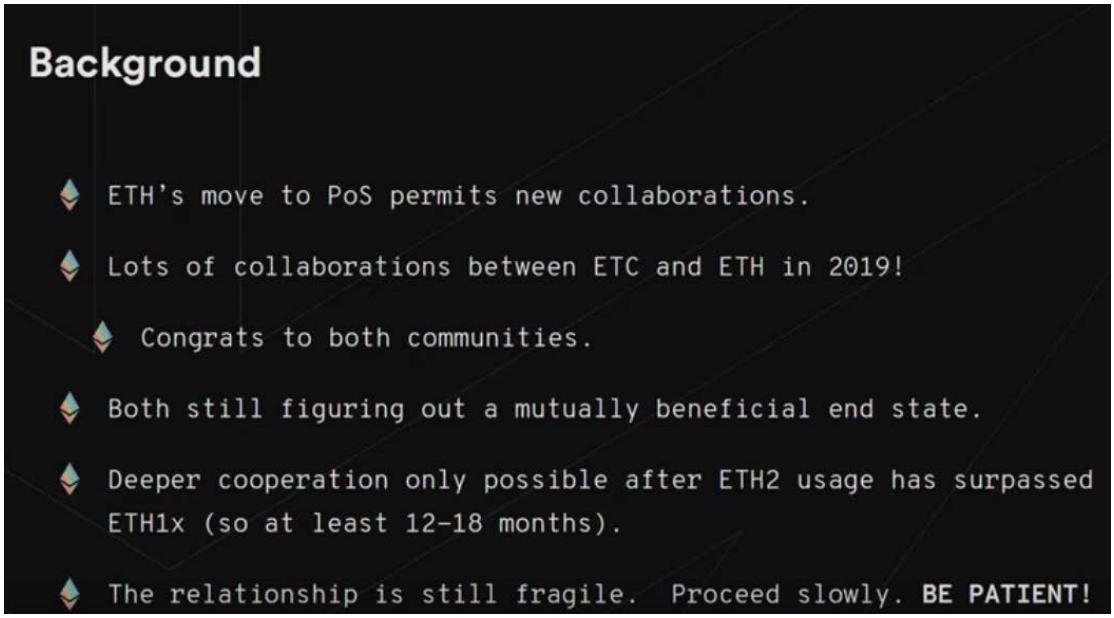
Fission 要做的不是基于开源的“云”供应商，而是“星宿”供应商，建立一个可以在笔记本电脑上运行的去中心化 Firebase，使用最新的 Web3 技术，为开发者提供可以自己掌控数据的工具。

ETC 和 ETH 之间的“正和博弈”

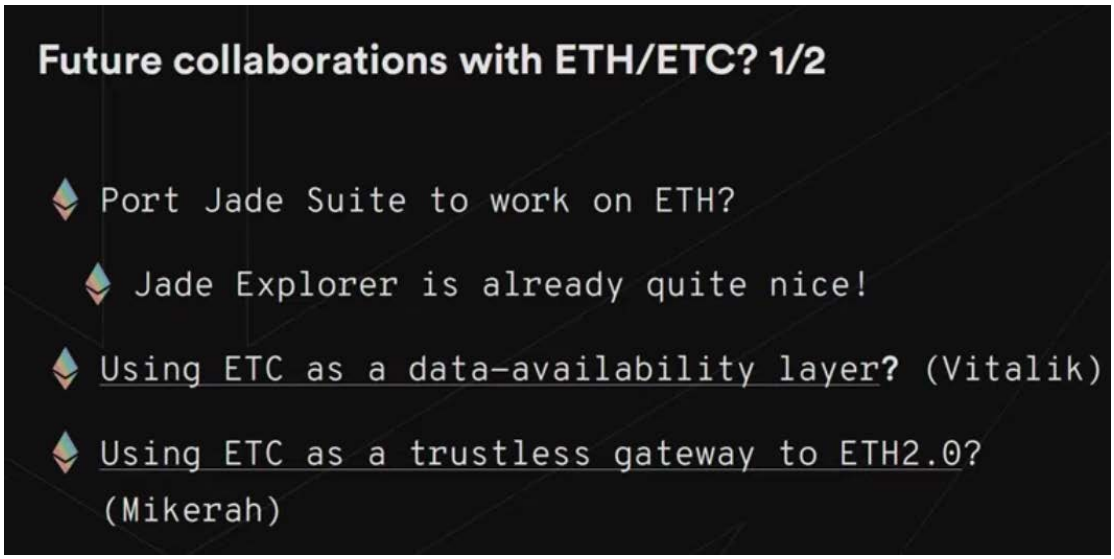
Positive Sum Games Between Ethereum and Ethereum Classic

演讲嘉宾：Virgil Griffith

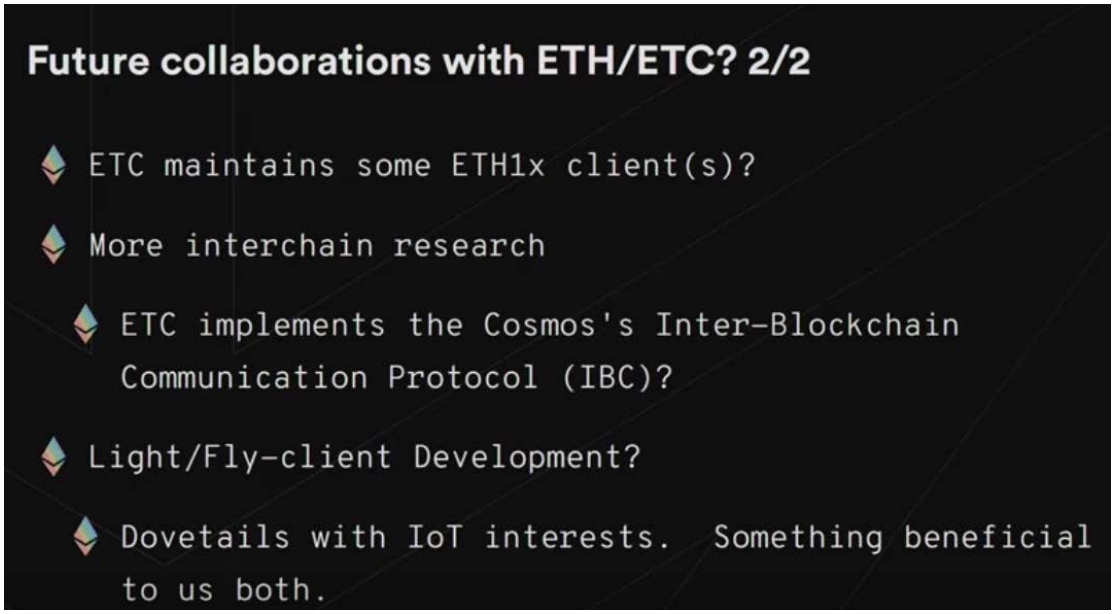
来自以太坊基金会的 Virgil Griffith 以其风趣幽默的风格，给我们讲解 ETC 和 ETH 之间的“正和博弈”。



这一年，ETC 与 ETH 进行了多方面的协作，实现互利共赢。目前双方联系仍不够紧密，期待一年或一年半后与 ETH 2.0 的深度合作！

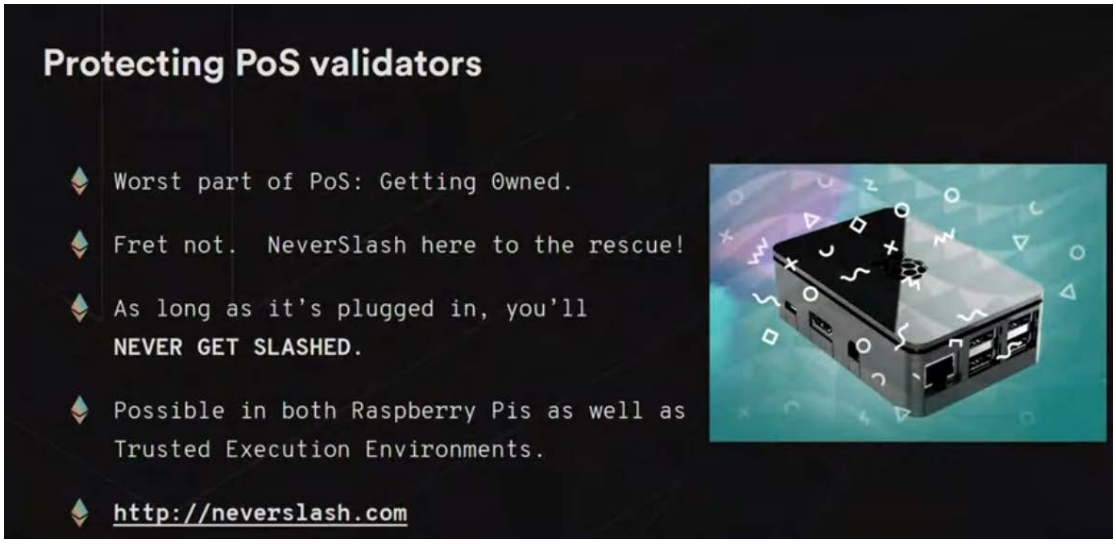


关于未来 ETC 与 ETH 的协作，有无限可能性，如未来 Jade Suite 或许可以在以太坊上运行，ETC 或许会成为数据可用性层，或许 ETC 能帮助 ETH 2.0 奠定去信任基础。

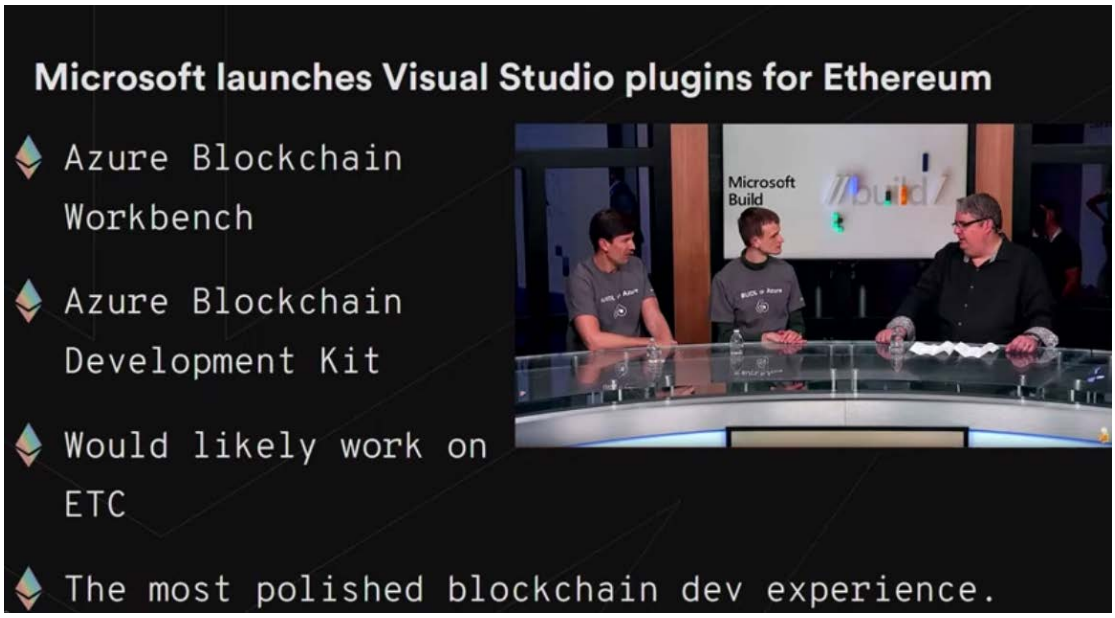


链间研究、轻客户端或 Fly-client 的发展等等都可以作为未来互相协作的方向。

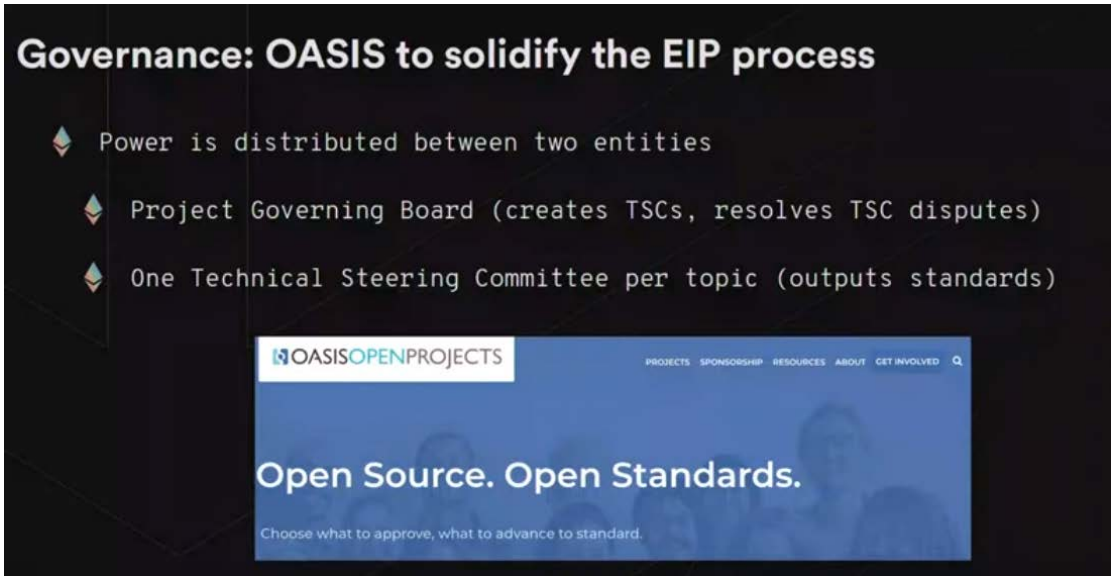
目前，ETH 与 ETC 已通过回教法的审核与批准，此举有利于建立与中东国家的联系。



NeverSlash 项目帮助 PoS 验证节点免于质押加密货币，适用于树莓派(Rspberry Pis)与其他受信任执行环境。



Microsoft 的开发者团队发布了适用于以太坊的 Visual Studio 插件。

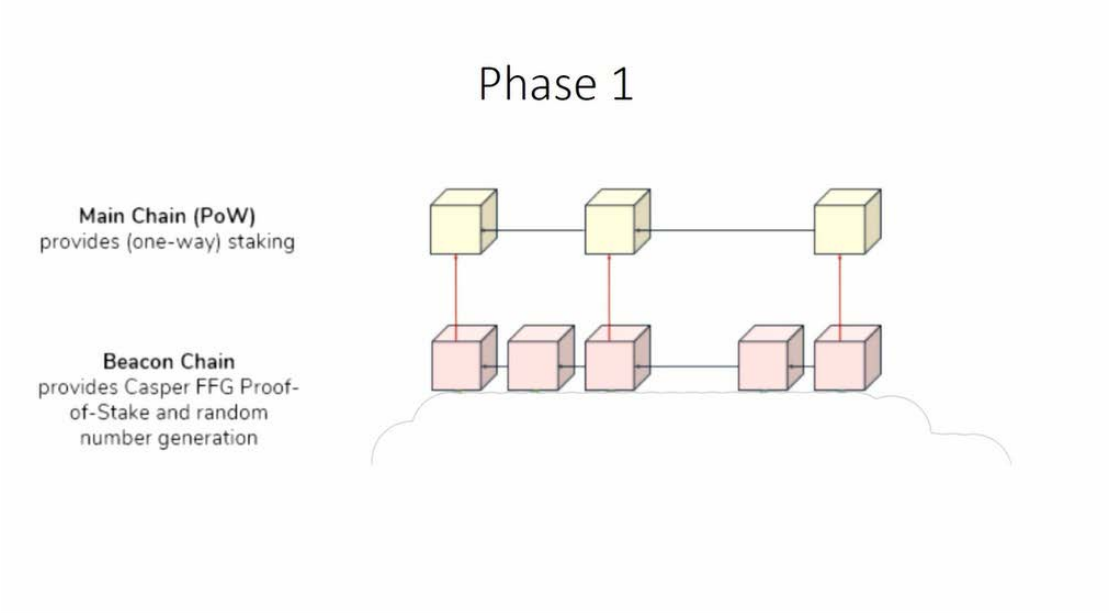


OASIS 为各种项目的开发者提供标准咨询服务等。

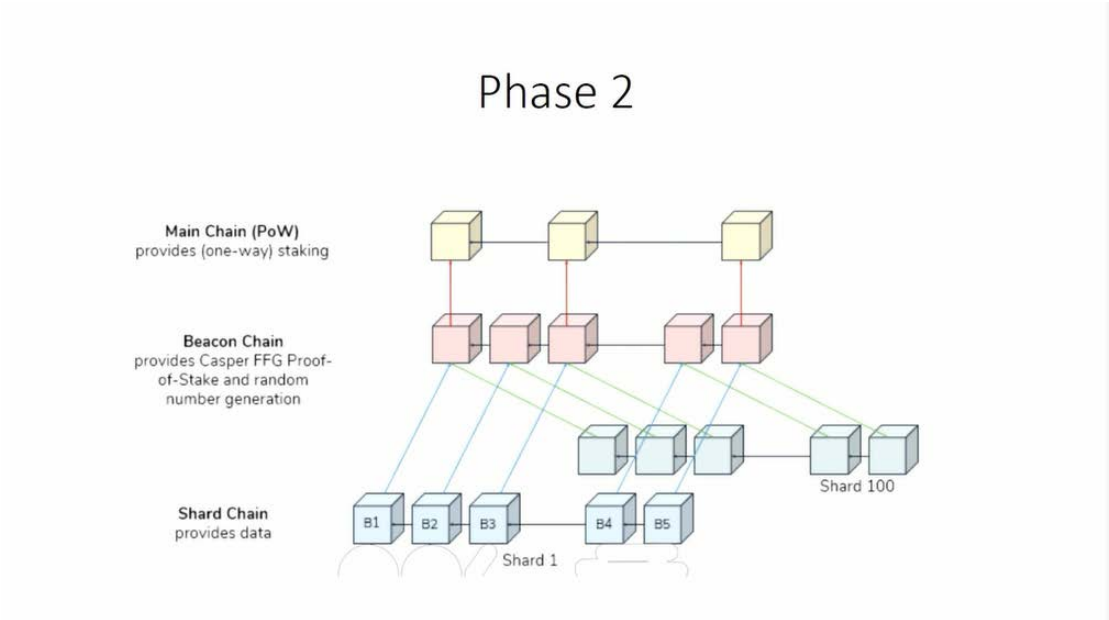
针对 ETC 2.0 的保守提议

A Conservative Proposal for Ethereum Classic 2.0

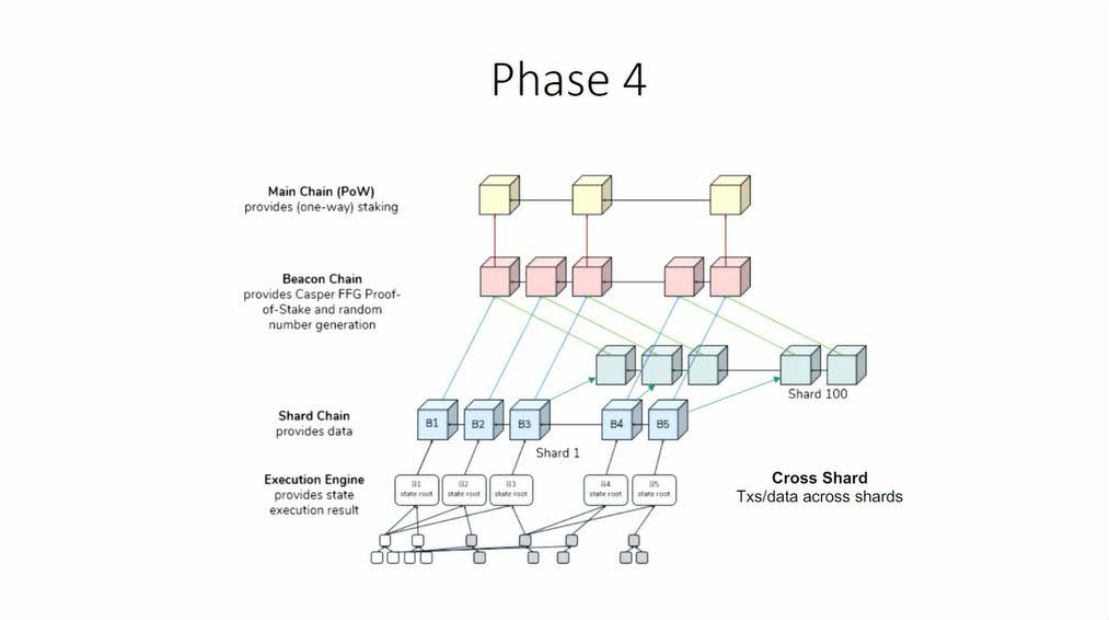
演讲嘉宾：Sunny Aggarwal （peaceBridge 项目参与者）



以太坊 2.0 路线图，阶段 1 就是从 PoW（工作量证明）链转向 PoS（权益证明）信标链。

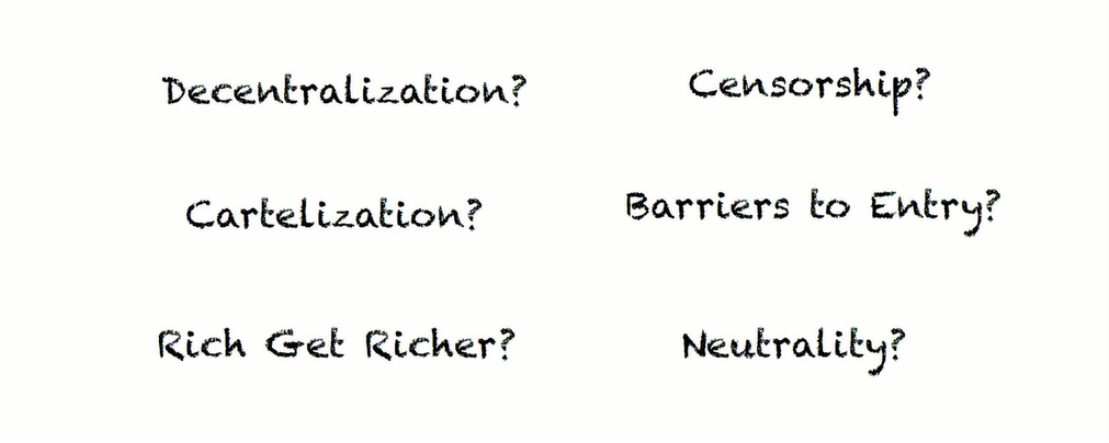


阶段 2 加入分片链（Sharding）



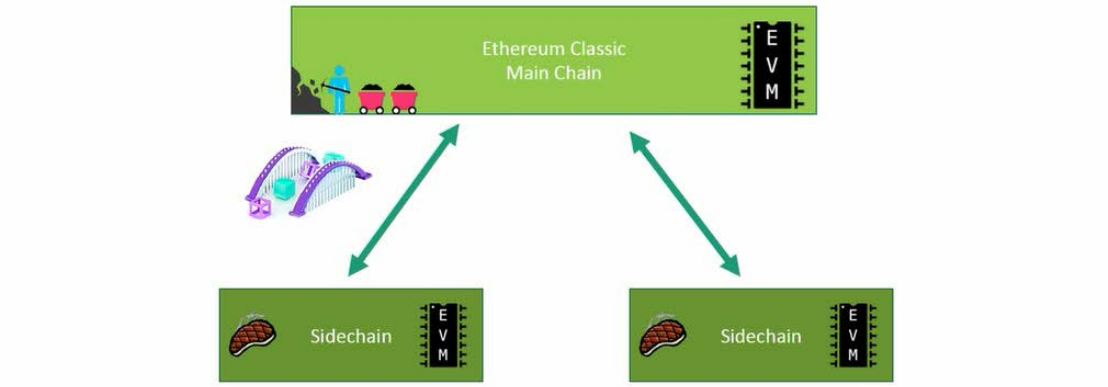
阶段 3 可能要几年后才能实现，到那时可以在分片中加入执行引擎，eWASM 虚拟机。阶段 4 则支持跨链交易和数据交换。

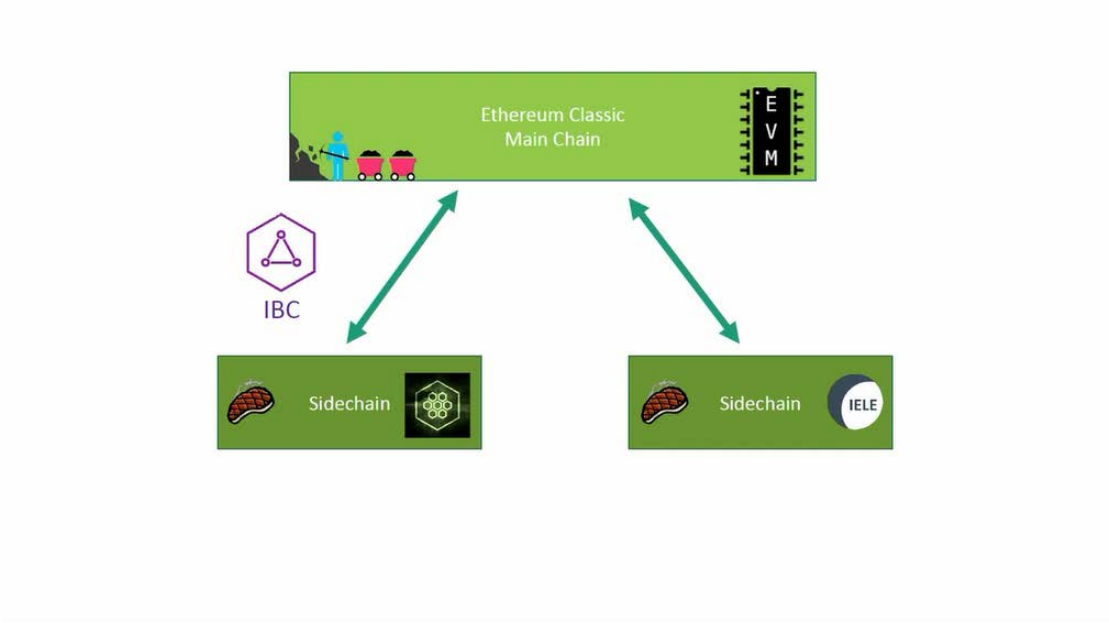
Sunny 对此感到忧虑的一点就是在阶段 1 中，PoW 链上的节点需要“销毁”32 个 ETH 才能参与 PoS 信标链，但无法再从信标链转回 PoW 链。



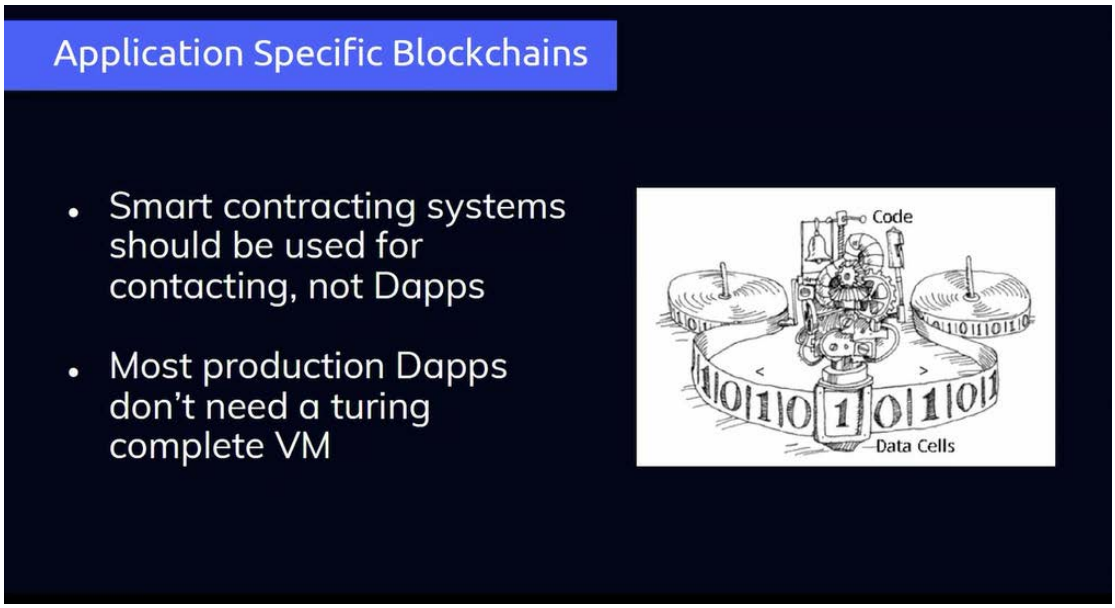
处于实验阶段的 PoS 机制面临的一些问题：去中心化和抗审查性如何？与 PoS 相比准入门槛是否过高？是否能够保持 PoS 的中立性？但最大的问题是 PoS 仍处于高度实验化和理论化的层面，Sunny 对于将此技术应用到价值 300 亿的以太坊网络中感到忧虑，因此他希望更加保守地循序渐进。

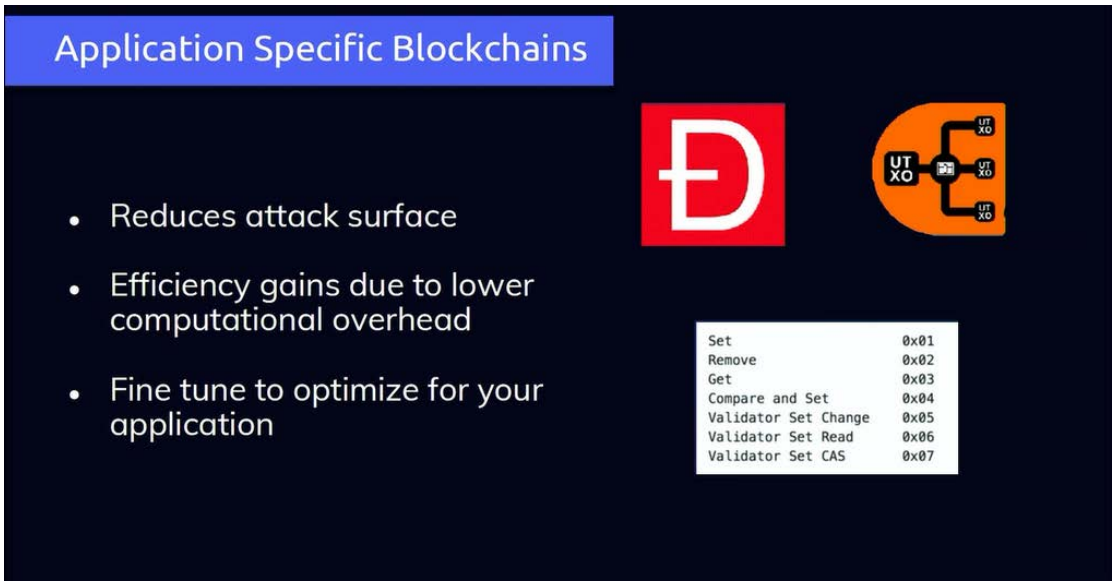
在过去的一年中，以太坊尝试将分片和 Casper 研究整合成一个项目，但其实这两项研究针对的是不同的问题。Boris 认为作为扩容性解决方案的分片技术应该首先获得更多的关注，由 PoW 转向 PoS 则可以循序渐进。





Sunny 为 ETC 2.0 提出的建议是，阶段 1 中使基于 EVM 的 ETC PoW 链与 PoS 侧链协作，节点可以将 ETC 质押在 ETC 主链上，以获得在侧链共识机制中投票的资格，与以太坊不同的是，用户可以使得资金在主链和侧链之间来回流动。如果侧链基于 EVM，那么可以使用 POA 和 TokenBridge 等技术来实现资金的双向流动；但如果侧链和主链的基础设施不同（即侧链可以不基于 EVM），也可以利用 Cosmos、Substrate 等跨链技术来实现链间的数据和通证交换。





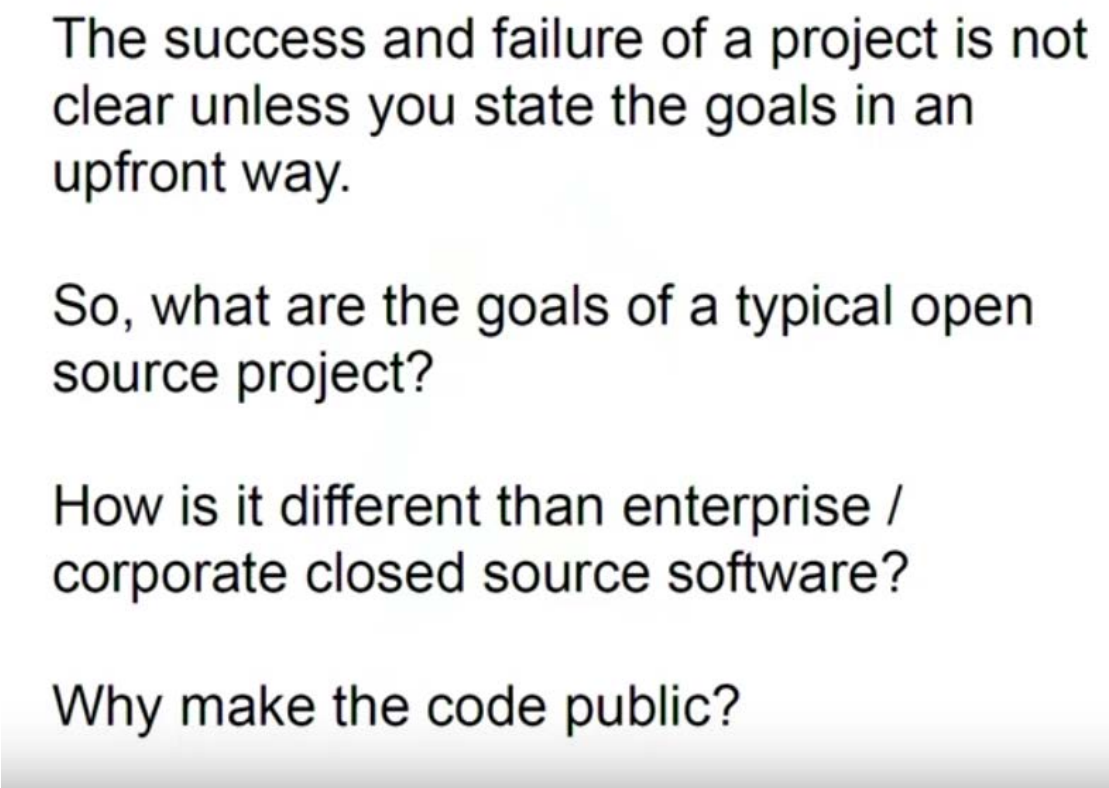
Sunny 倾向于非基于 EVM 的区块链，而是专门针对 DApps 的区块链。理由如下：智能合约系统应该用于合约编写，而非 DApp；大多数 DApp 产品并不需要图灵完备的虚拟机。这样不仅可以减少受攻击面，还可以通过更低的计算限制来更高效率地优化 DApp。另外，Sunny 认为 PoS 完全可以在侧链或者分片链中先行部署，等到时机成熟时再将其部署到主网，间隔有可能达到 5—10 年。

开源软件是否不朽？

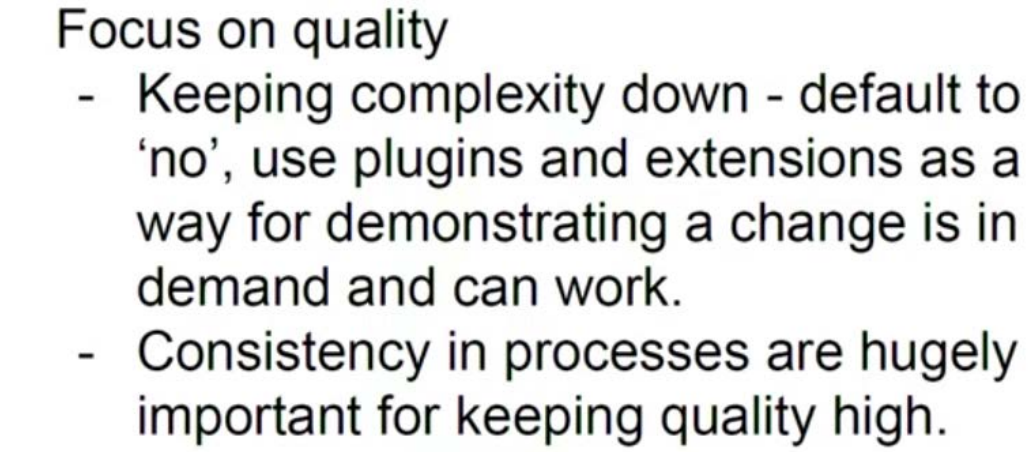
Is Open Source Software Immortal?

演讲嘉宾：Zachary Belford

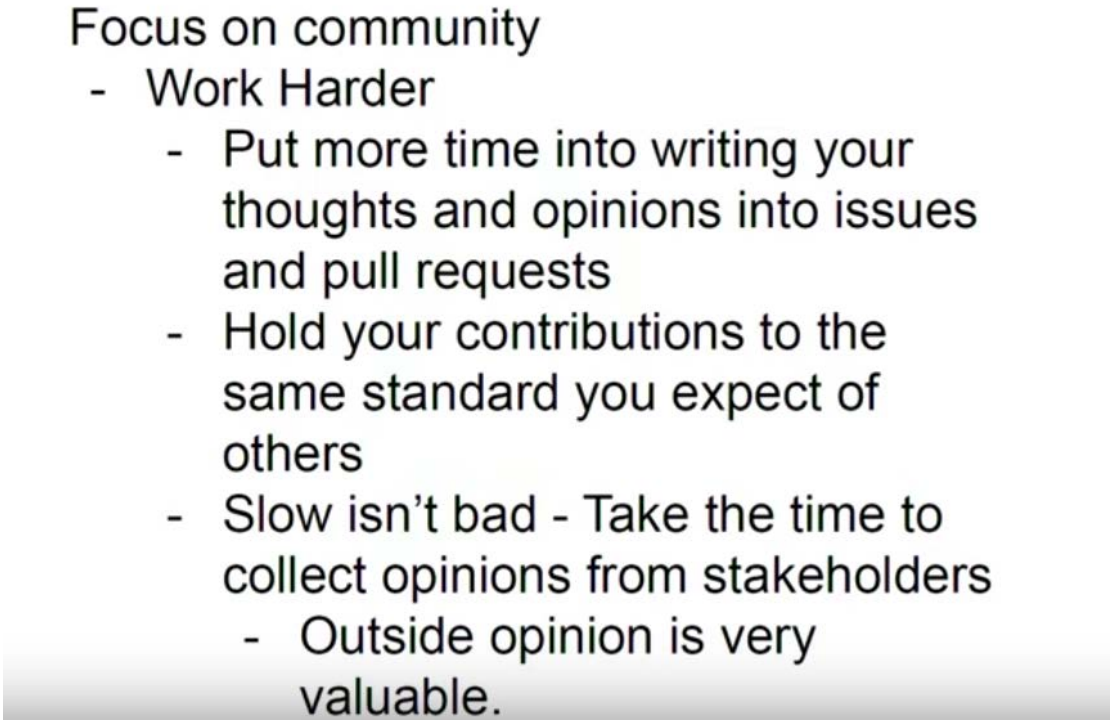
Zachary Belford 是 ETC Labs 开发工具团队的队长之一，他将带我们探讨开源软件是否能创造不朽传说。



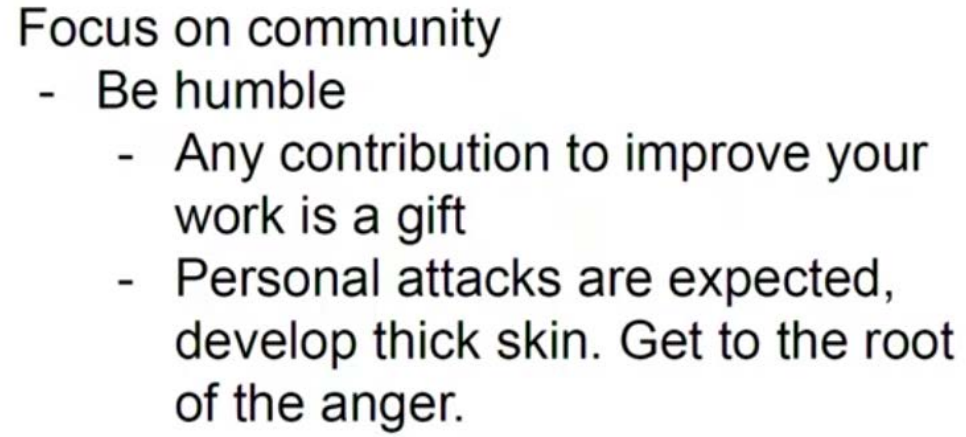
当一个项目中的开发者意见存在分歧，并听取了一方的意见，界定这个项目是否成功，在于该项目是否实现了最初的目标。典型开源项目的首要目标是提高代码的质量，提供更好的开发文档，让人们更易于入门。企业闭源软件的目标则是赚更多利润。



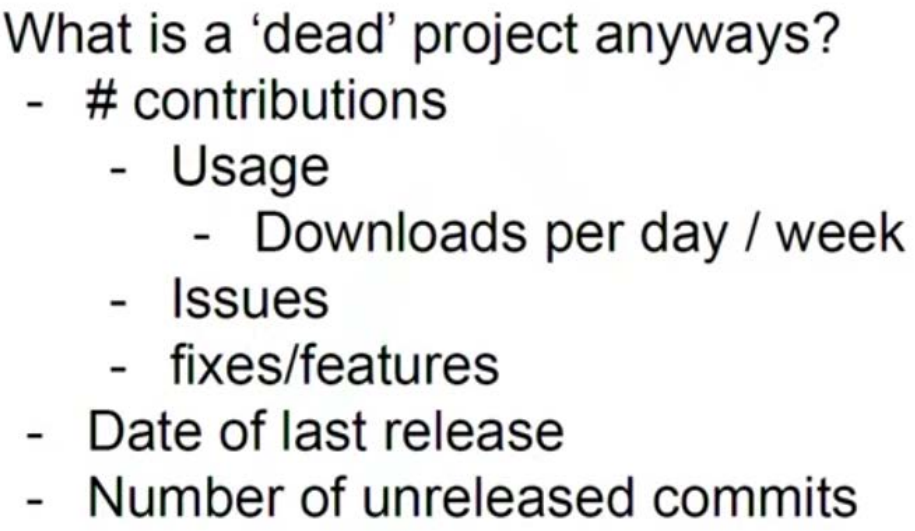
开源软件要重视质量，尽量降低复杂度，代码实现、更新过程中要保存一致性。



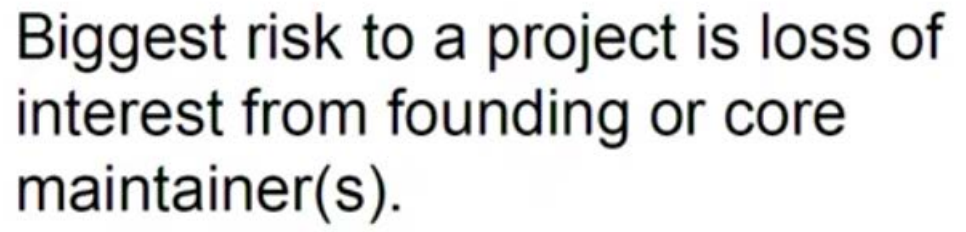
社区建设也应得到重视！开发者应多花时间在阐述自己的想法、意见上，标准对内对外一致，多听取权益持有者的宝贵建议。



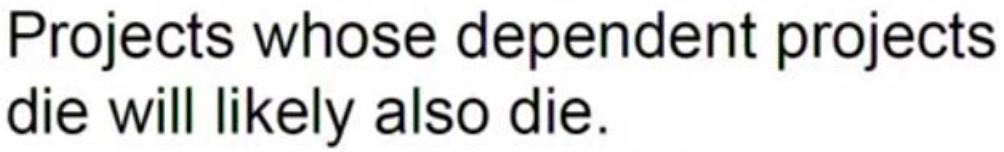
开发者也应秉承谦虚的态度哦！



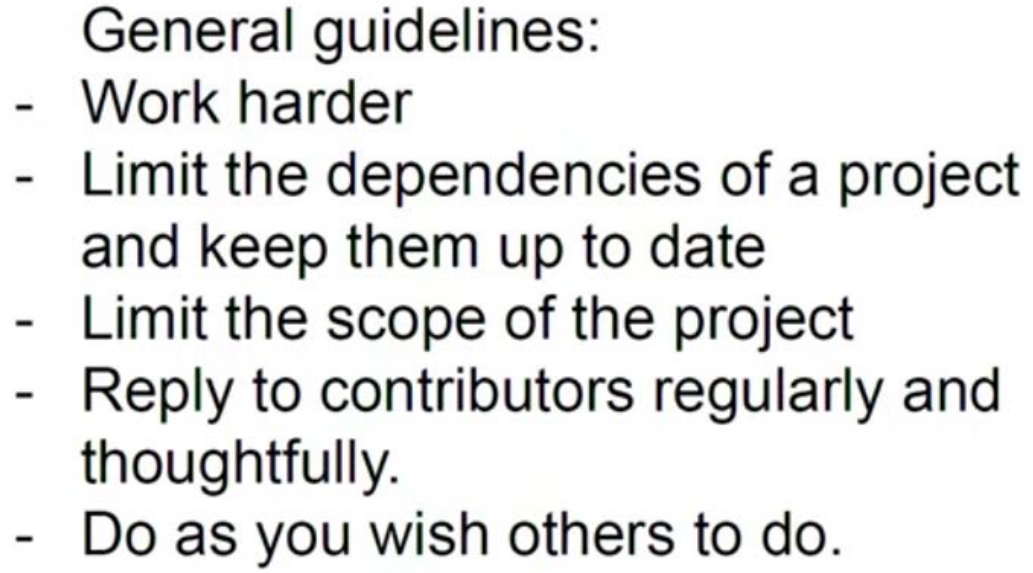
“失败”项目的特征是开发者过分在意各自的贡献、前版本的发布日期、未发布的代码量。



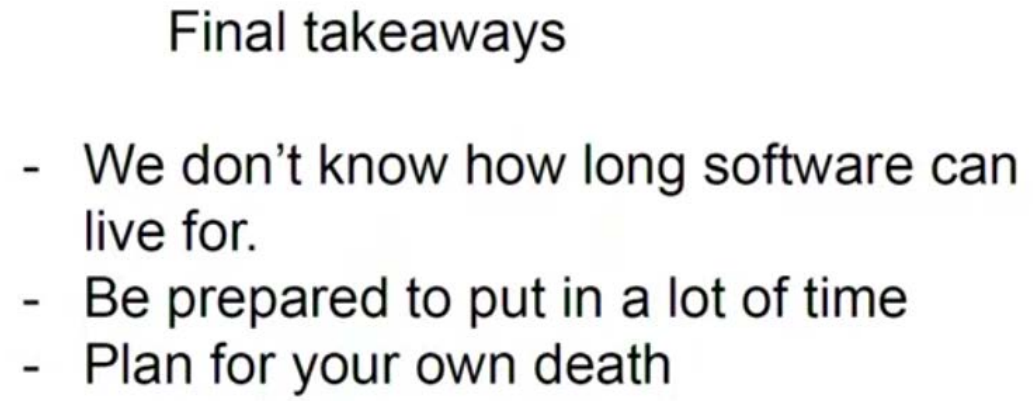
创办团队或核心维护人员失去了对一个项目的热情，那么这个项目也会随之面临巨大的风险。



一个项目的依赖项目宣告失败，那么这个项目也可能面临失败的下场。



要想造就开源软件的不朽，开发者就应更加勤勤恳恳，限制项目的依赖性，保持更新项目，划定项目的业务范围。



总而言之，开源软件能存活多久，我们不得而知，但如果你对感兴趣，那么就应做好充分的

心理准备，准备投入许多时间到开发软件的工作中去。

从技术角度看区块链合并

Merging Blockchains: A Technical Perspective

演讲嘉宾：Wei Tang

作为 Parity Technologies 的核心开发者，Wei Tang 将与我们一起从技术的角度看区块链合并。

51% Attack

Problem: If there's not enough hashrate, a network can be 51% attacked.

如果一个网络没有足够的哈希值，那么它将有可能受到 51%攻击。

Testing

Problem: Testnet is hard to maintain.

- It is nearly always the case that PoW testnets dont have enough miners and peers.
- There's seldom incentives to support them. It's not real money. Plus for dapp's own testing, it's trival to spin up a private testnet.
- The debate of whether to keep the current testnet or spin up a new one is never ending.

测试网难以维护是个问题。

Interoperability

Problem: Separated blockchains almost never works together.

不同的区块链几乎无法互操作。

Blockchain Merging

Merge blockchains so that:

- Share PoW security to better protect against 51% attack, and better support testnets.
- Enable (optional) interoperability.
- Unite communities after split.

然而，合并区块链能解决上述问题：

1. 通过共享 PoW 安全，从而更好地抵御 51%攻击和支持测试网的运作。
2. 实现互操作性。
3. 使分叉后的社区也能再次结合。

Merging Ethereum

Idea: Merge the state root.

EIP-2225 的主要理念就是在保留以太坊现有数据结构的前提下，通过合并状态根(state root)来合并以太坊与以太经典这两条基于以太坊区块链。

Merging Ethereum

From one state root to two state roots:

- Hash it again: `hash((stateRoot1, stateRoot2))`
- Binary merkle tree.

那又如何合并状态根呢？可以对两个区块链的状态根进行哈希来合并。

Merging Ethereum

Keep one chain's proof of work, discard the other. The other chain's miner is expected to migrate.

- Merger blockchain
- Mergee blockchain

然后，将会保留一条 PoW 链，而丢弃另一条链。原本在废弃链上挖矿的矿工能够迁移到合并链上是大家所期望的事情。

Merging Ethereum

Use account versioning (EIP-1702):

- Each account has an associated version. When the account is executing, use the VM corresponding to the version.
- Define two versions, one for original VM on the merger blockchain, and the other for original VM on the mergee blockchain.

关于状态执行规则，EIP-1702 提出使用账户版本控制的解决方案。

Extensions

Problem: How do merger chain and mergee chain communicate with each other post-fork?

- UI: Add unique prefix for merger and mergee chain.
- EVM: Additional opcodes similar to CALL for cross communication.

通过添加独特的前缀到合并链与被合并链上，添加操作码，就能实现两条链之间的沟通。

Sedbin

A minimal construct of a master blockchain that allows security sharing for proof of work.

为了解决只能合并两条区块链的弊端，Wei Tang 提出了 Sedbin 项目，Sedbin 是能够为工作量证明机制提供共享安全的主区块链的最小构造。

Sedbin

Master blockchain is a minimal construct that allows storing arbitrary data, with no state execution rules.

- Data should be of fixed length of Hash.
- All of them are expected to be participating blockchain's block pre-hash.
- List encoded in a binary merkle tree.

Sedbin 项目上设有主区块链和参与链。主区块链是可存储任意数据的最小构造，链上不设置状态执行规则。

Sedbin

Participating blockchains require no interaction with the master blockchain for block validation.

- It can freely decide to join or leave a Sedbin master blockchain at any time.
- It does not need modification in networking to join a sedbin.

在任何时候，参与链都能自由加入或离开 Sedbin 主区块链。参与链能独立完成区块验证，不需要主区块链的参与。

Sedbin

For basic integration:

- Participating blockchains are expected to use the same PoW algorithm, the same block time, and the same difficulty adjustment algorithms. (But we can have multiple Sedbin master blockchains!)
- If miners are not mining all participating blockchains, some chains may result in longer block time. (But security is not compromised!)

要想实现基础链合并，参与链的 PoW 算法、区块时间、难度调整算法就要与主区块链保持一致，但有多种 Sedbin 主区块链，所以可以按实际情况与兴趣进行选择。

Sedbin

For advanced integration:

- Participating blockchain can choose to commit to master blockchain once per N blocks, thus result in short block time.
- However, it requires participating blockchain to support multiple seal types, and is more complex.

如果还想达到高级合并的水平，参与链可以按时段提交区块给主区块链，以缩短区块时间，但难度会更大。

Immutability

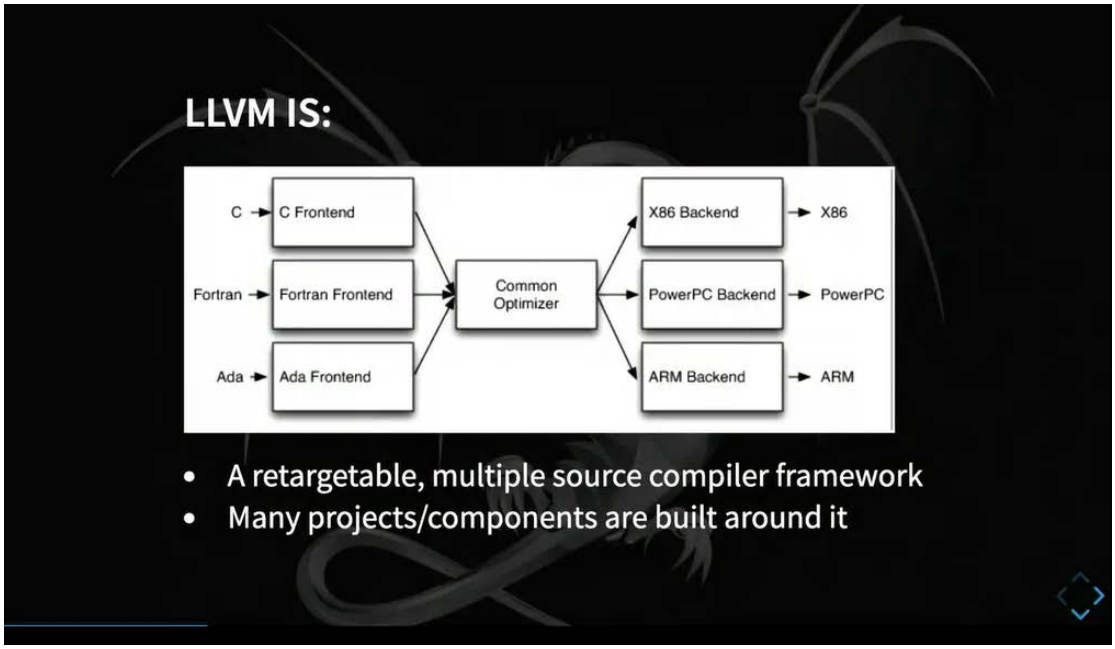
Each chain continues to have their own definitions of immutability, and can carry out hard fork in their own schedules.

两条不同的链合并后，双方对不可篡改性的定义不需达成一致，他们仍能按自己的计划实现各自的硬分叉。

EVM 与 LLVM 编译器

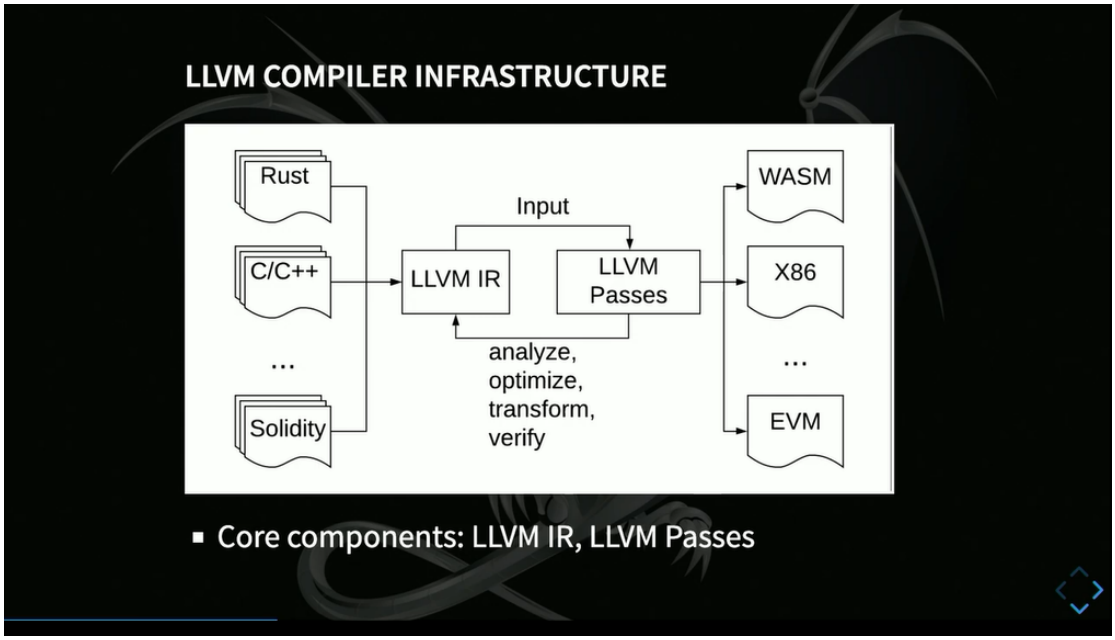
The EVM LLVM Project

演讲嘉宾：Alan Li

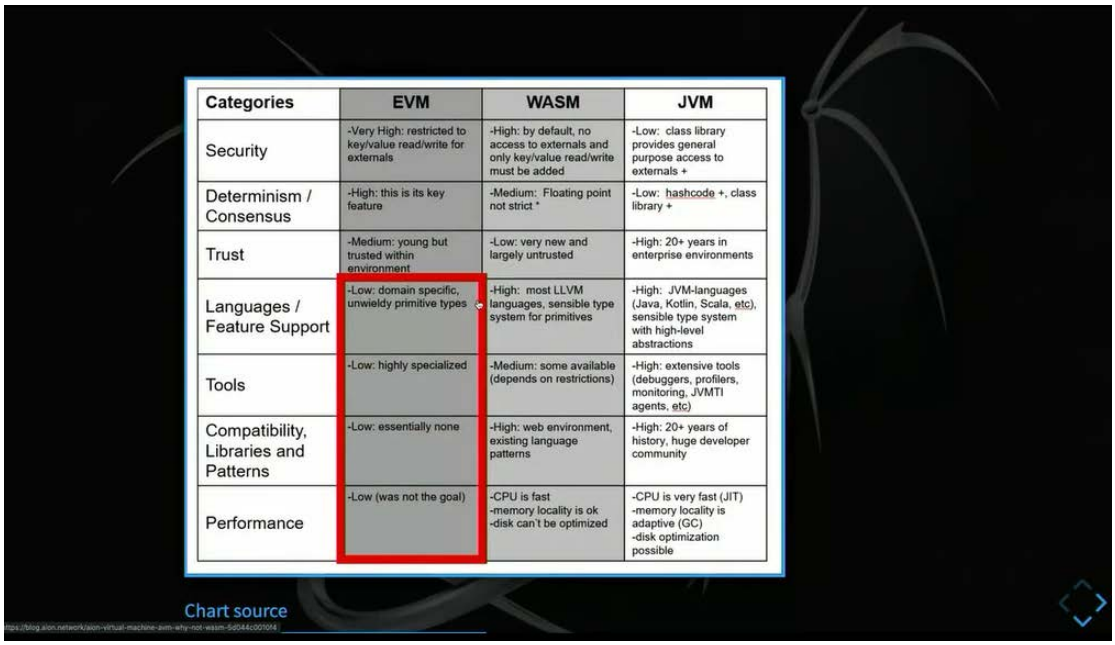


EVM-LLVM 项目：面向 EVM 架构的一种新的基于 LLVM 的编译器后端

LLVM（底层虚拟机）作为一个可重定向的多源编译器框架诞生近 20 年，是最优秀的开源项目之一。



LLVM IR 和 LLVM Passes 是 LLVM 的核心组成部分，前端可以将多种编程语言转换成 LLVM IR 中的低级语言，之后可以使用 LLVM Passes 对其进行分析、优化、转换和验证，最后转换成面向各种架构的代码。



Categories	EVM	WASM	JVM
Security	-Very High: restricted to key/value read/write for externals	-High: by default, no access to externals and only key/value read/write must be added	-Low: class library provides general purpose access to externals +
Determinism / Consensus	-High: this is its key feature	-Medium: Floating point not strict *	-Low: hashcode +, class library +
Trust	-Medium: young but trusted within environment	-Low: very new and largely untrusted	-High: 20+ years in enterprise environments
Languages / Feature Support	-Low: domain specific, unwieldy primitive types	-High: most LLVM languages, sensible type system for primitives	-High: JVM-languages (Java, Kotlin, Scala, etc), sensible type system with high-level abstractions
Tools	-Low: highly specialized	-Medium: some available (depends on restrictions)	-High: extensive tools (debuggers, profilers, monitoring, JVMTI agents, etc)
Compatibility, Libraries and Patterns	-Low: essentially none	-High: web environment, existing language patterns	-High: 20+ years of history, huge developer community
Performance	-Low (was not the goal)	-CPU is fast -memory locality is ok -disk can't be optimized	-CPU is very fast (JIT) -memory locality is adaptive (GC) -disk optimization possible

那么为什么 EVM 需要 LLVM 呢？通过这张对比表格我们可以看到 EVM 在编程语言和性能支持、工具系统、兼容性等方面都不及 WASM 和 JVM，而 LLVM 可以为基于 EVM 的生态提供开发者友好的工具、不过时的开发环境、优化的性能、更多的语言支持以及安全工具等。不过时的开发环境具体指：强大成熟的生态系统支持、支持现有的以及未来的 EVM、更容易将用户转化到 ETC 上、更容易过渡

到下一代引擎。

EVM-LLVM 可以作为：独立的智能合约编译器、分析器、优化器、验证器；SDK 库元件。

目前项目状态

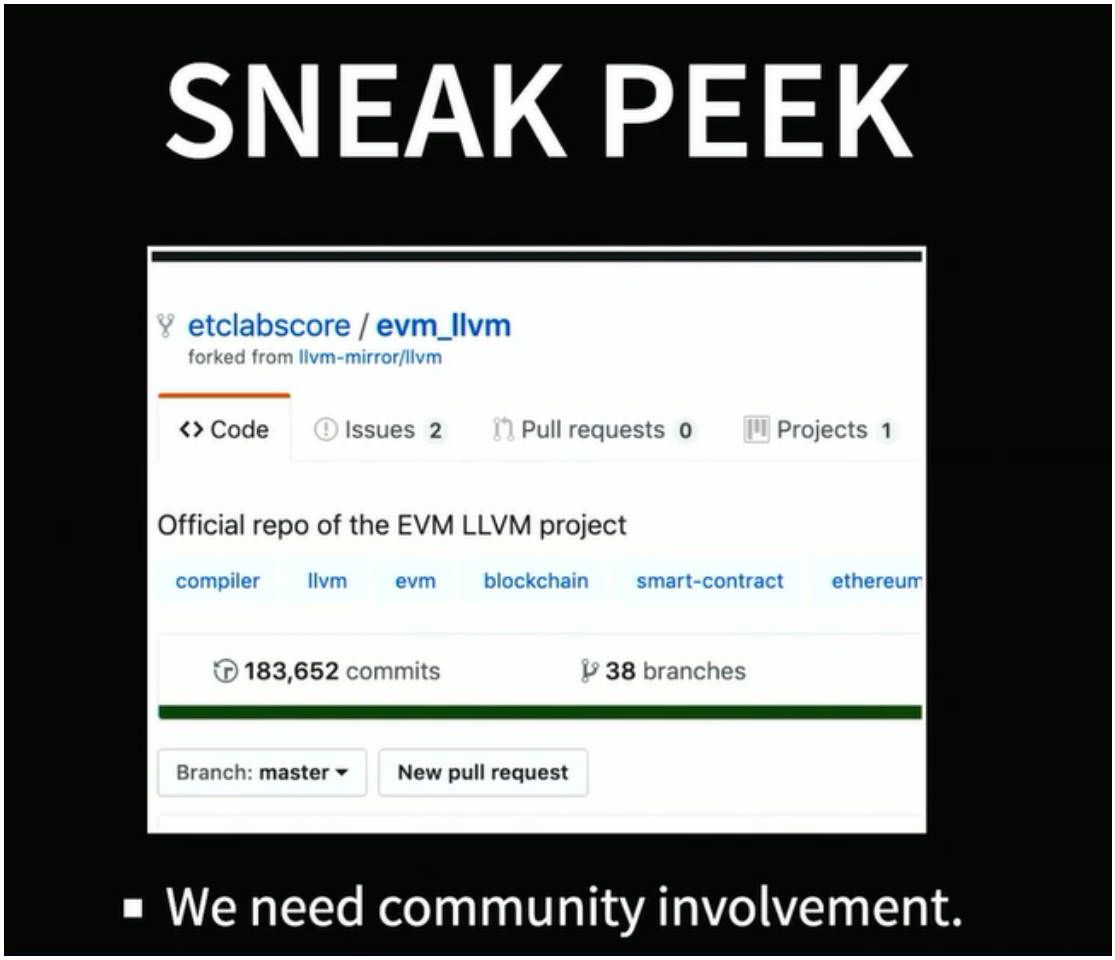
EVM LLVM: ENGINEERING DETAILS (1)

- Based on LLVM 10.0.dev
- 256 bit support
  - 256 bit support exposed to langauge frontends
  - Narrower-width data types are expanded internally
- EVM-specific opcodes exposed as intrinsics
  - Such as: BLOCKHASH, CALLER, GASPRICE, ...
- Stackifier
  - generic optimization passes on register-based opcodes
  - Converts register-based opcodes to stack opcodes

EVM LLVM: ENGINEERING DETAILS (2)

- open source from the beginning
- EVM-specific optimizations (planned)
  - Stack manipulation scheduler
  - Gas consumption optimizations
- Contract ABI generation
  - Currently relies on frontend
- Compatibilities with existing tools

技术细节



大家可以在 etclabscore 的 github 中找到 EVM-LLVM 项目，近期会提供更完善的维基供大家查阅。

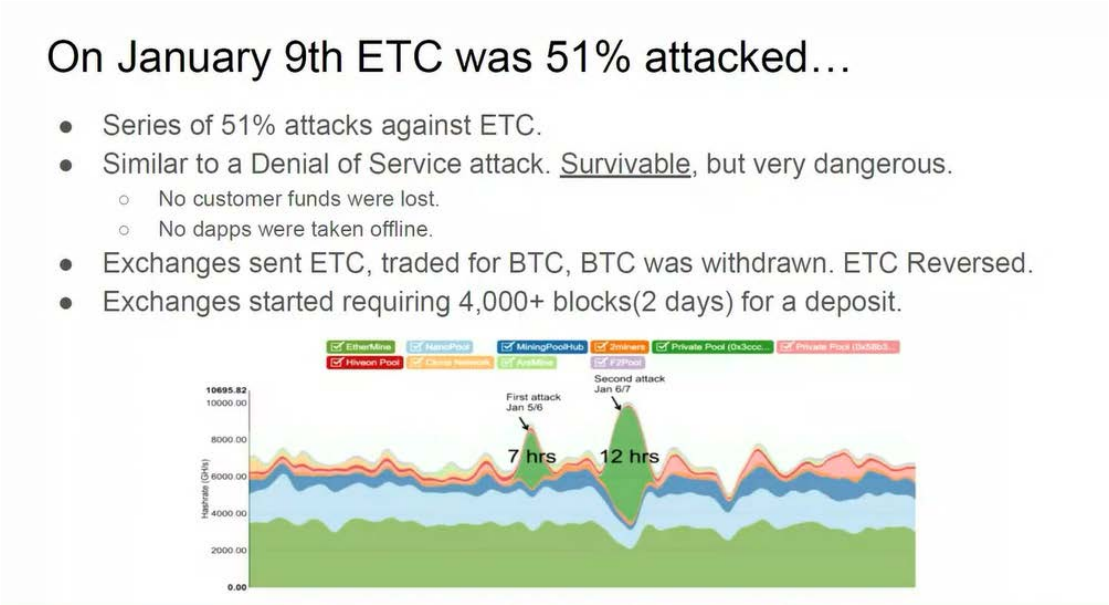
Alan 在现场做了实战演示，有兴趣可以在会议视频中查看。

ECIP 1049： ETC 为何应该使用 SHA3 工作量算法

ECIP-1049: Why Ethereum Classic should adopt SHA3 Proof of Work

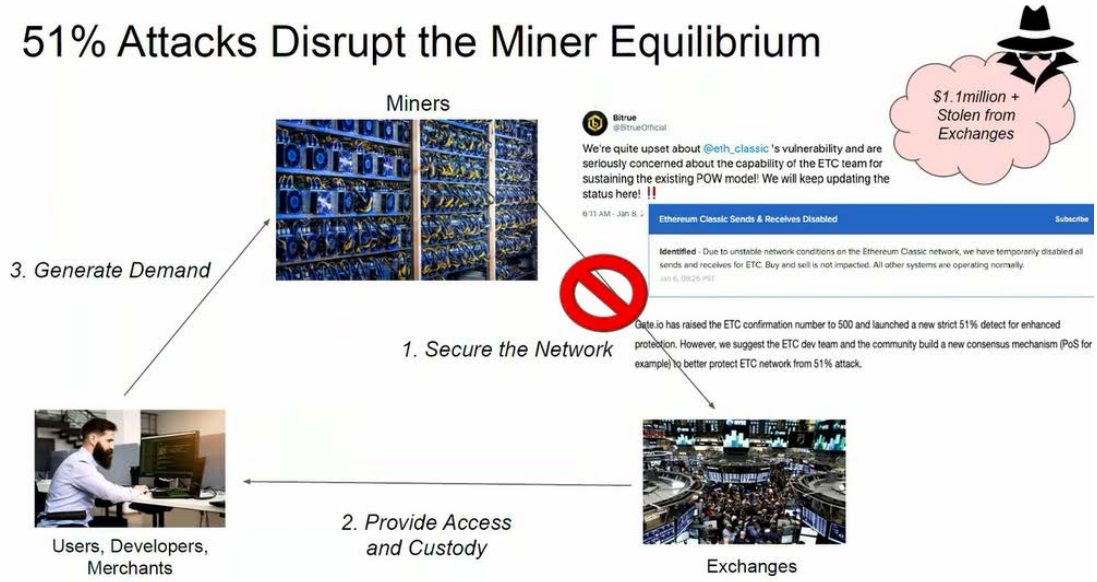
演讲嘉宾： Alex Tsankov

ETC 为什么需要改变共识算法？Alex 的观点是：主观来看，ETC 社区目前规模还不大，有利于社区成员达成共识；并且目前 ETC 有能力做到这种转变，可以由此维持社区共识。



在客观因素方面来看也亟需改变优化共识算法。ETC 在今年初遭受了一系列 51%

攻击，类似于拒绝服务攻击，虽然 ETC 存活了下来，但是仍然具有相当的风险。



PoW（工作量证明）共识机制要运作的三大要素在于：矿工、交易所、交易者/开发者。矿工负责保障网络的安全性并且挖出新的代币，然后发送给交易所，交易所随后可以将代币转换成法币，使得矿工能够覆盖能耗成本。但是当 51%攻击发生时，交易所需要超过 4000 个区块的确认，而这需要耗时 48 小时。这时诚实节点不再能向交易所提供足够的押金以换取法币，也就不再能覆盖自己的成本。这将导致负反馈循环，使得作恶节点能够反复进行双花，交易所也因此需要更多的区块确认。我们需要注意交易所的确需要衡量一个项目的盈利空间，能否安全地转换成法币。

Ethash 的弊端：

1. 抗 ASIC 理论上可行，但在实践中总是失败。正是随着市值的增长，自然滋生出了对 ASIC 的需求

2. 对矿工内存需求很大

3. 一些使用 Ethash 的链遭受到 51%攻击

4. 不支持 Flyclient 和轻客户端

5. 对抗 ASIC 能力的错误预估

6. 缺乏安全监测、标准和文档（在以太坊的 wiki 中只有四段相关描述，这对于基于加密技术的加密货币来说远远不够）

Analysis: Why choose SHA3?

```
#define FOR(i,n) for(i=0;i<n;++i)
#define H(i,r,p,d) int crypto_hash_##i(u8 *h,const u8 *m,u64 n) { Keccak(r*8,m,n,6+25*p,h,d); return 0; }
typedef unsigned char u8;typedef unsigned long long u64;static u64 ROL(u64 a,u8 n){return(a<<n)|(a>>(64-n));}static u64 L64(const u8*x){u64
r=0,i;FOR(i,8)r|=(u64)x[i]<<8*i;return r;}static void F(u64*s){u8 x,y,j,R=1,r,n;u64 t,B[5],Y;FOR(n,24){FOR(x,5){B[x]=0;FOR(y,5)B[x]^=s[x+5*y
];}FOR(x,5){t=B[(x+4)%5]^ROL(B[(x+1)%5],1);FOR(y,5)s[x+5*y]^=t;}t=s[1];y=r=0;x=1;FOR(j,24){r+=j+1;Y=2*x+3*y;x=y;y=Y%5;Y=s[x+5*y];s[x+5*y]=
ROL(t,r%64);t=Y;}FOR(y,5){FOR(x,5)B[x]=s[x+5*y];FOR(x,5)s[x+5*y]=B[x]^(~B[(x+1)%5]&B[(x+2)%5]);}FOR(y,7)if((R=(R<<1)^(113*(R>>7)))&2)*s^=
1ULL<<((1<<y)-1);}}static void Keccak(u8 r,const u8*m,u64 n,u8 p,u8*h,u64 d){u64 s[25],i;u8 t[200];FOR(i,25)s[i]=0;while(n>=r){FOR(i,r/8)s[i
]^=L64(m+8*i);F(s);n-=r;m+=r;}FOR(i,r)t[i]=0;FOR(i,n)t[i]=m[i];t[i]=p;t[r-1]|=128;FOR(i,r/8)s[i]^=L64(t+8*i);F(s);FOR(i,d)h[i]=s[i/8]>>8*(i%
8);}H(shake128,21,1,168)H(shake256,17,1,136)H(sha3224,18,0,28)H(sha3256,17,0,32)H(sha3384,13,0,48)H(sha3512,9,0,64)
```

Complete SHA3 Algorithm

为什么要选择 SHA3？简单、强大、便捷。

SHA3 - The strongest hashing algorithm on Earth.

Safety

- SHA3 is the safest Hashing algorithm around today.
- Winner of NIST Secure Hashing competition (2007-2012).
- Very different, fundamentally from SHA2. Less risk.

Performance

- Extremely efficient in hardware, magnitudes faster than SHA2.
- Reduces number of chip instructions (And, Xor, Rot, Not) necessary vs SHA2 (And, Xor, Rot, **Add**, Or, Shr)

Compatibility

- Accessible to Smart Contract and used throughout the EVM

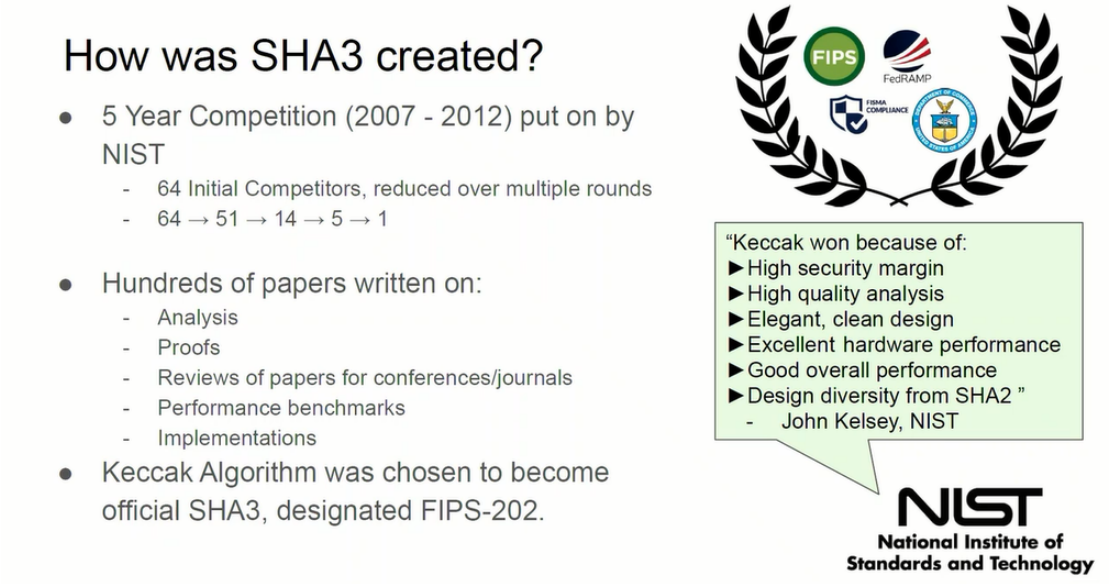
安全性：

1. SHA3 是目前最安全的哈希算法
2. 在 2007-2012 年 NIST 算法竞赛中胜出
3. 与 SHA2 算法全然不同，风险更小

与 SHA2 性能对比：

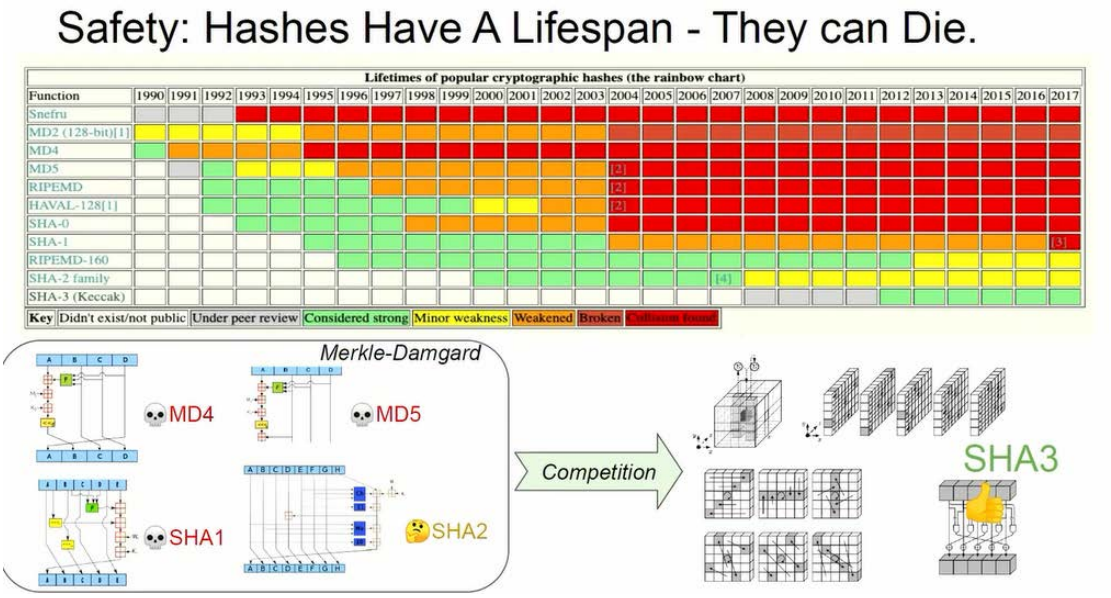
1. 更高效、跟快速
2. 芯片指令更少

兼容性：智能合约和 EVM 可用



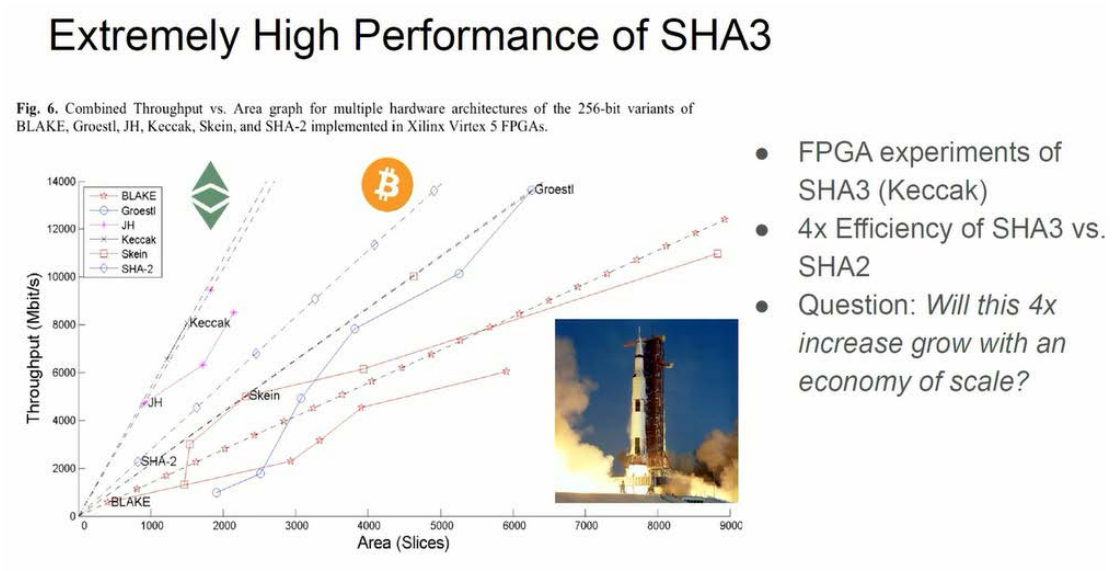
SHA3 是如何创造出来的？

五年的 NIST 竞赛中，全世界的高校为其创作了上百篇论文（分析、证明、评论、性能评测、执行情况），最后得出的结论是 Keccak 算法在安全性、设计、性能等方面都足够优秀。

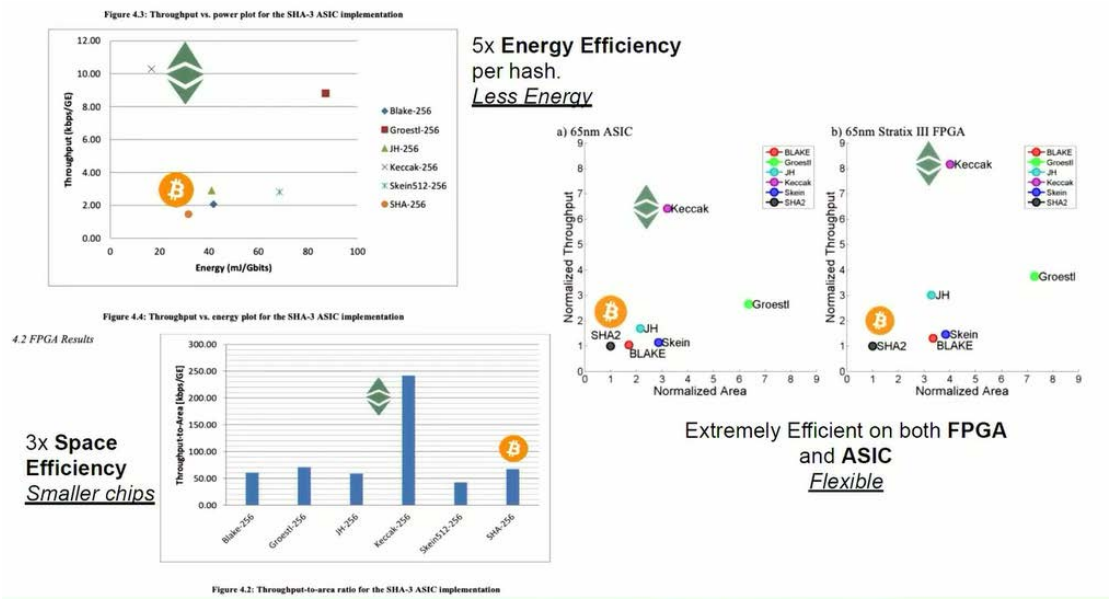


彩虹表：哈希算法的生命周期

Ethash 从未正式成为哈希算法。某个算法被证明能够被破解之后，就会被停止使用。目前尚在使用中的算法包括 RIPEMD-160（性能不高）、SHA2（比特币算法）以及 SHA3（Keccak，暂未被任何 PoW 链使用）。



SHA3 性能比 SHA2 高出 4 倍。



与 SHA2 相比，SHA3 能效率高出 5 倍，空间占用效率高出 3 倍，面对 FPGA 和 ASIC 也更有效率。

SHA3 mining will be very different from SHA2

	SHA2 (Bitcoin)	SHA3 (Ethereum Classic)
Algorithm	Double SHA2-256	Keccak256
Block Time	600 seconds	15 seconds (*latency matters!*)
Supply Cap	17.8m/21m (85%)	112m/210m (53%)
Launch	2009	2015
Use-Case	Store of Value, Transfer of Wealth	Computation, Smart Contracts, IoT payments.
Smallest Unit	0.00000001 (satoshi, 8 decimals)	0.000000000000000001 (wei, 18 decimals)
Nonce Space	32 bit Nonce	64 bit Nonce + 256 bit MixHash (320bit capacity)

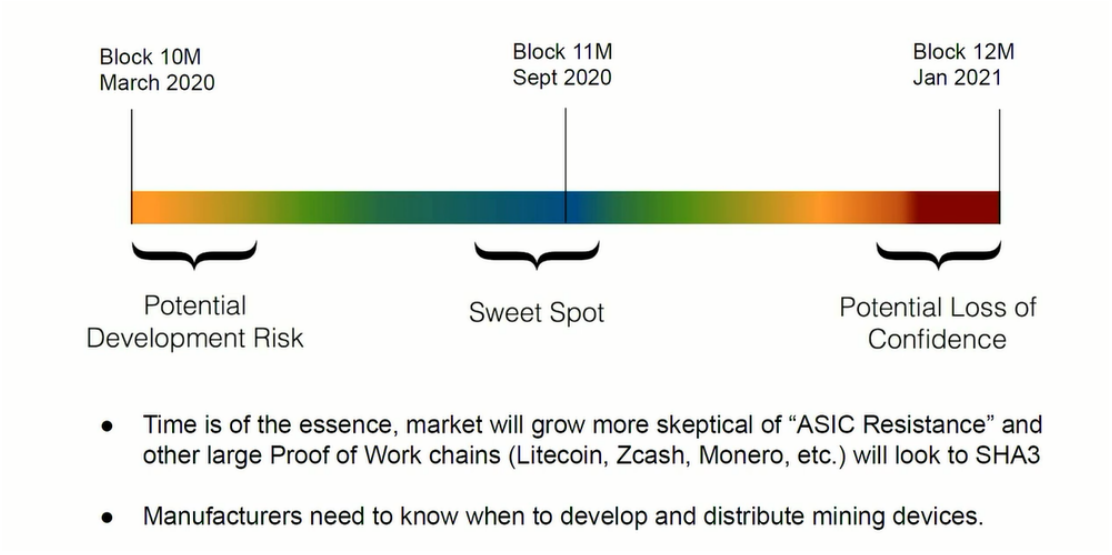
SHA2 和 SHA3 的主要不同点。

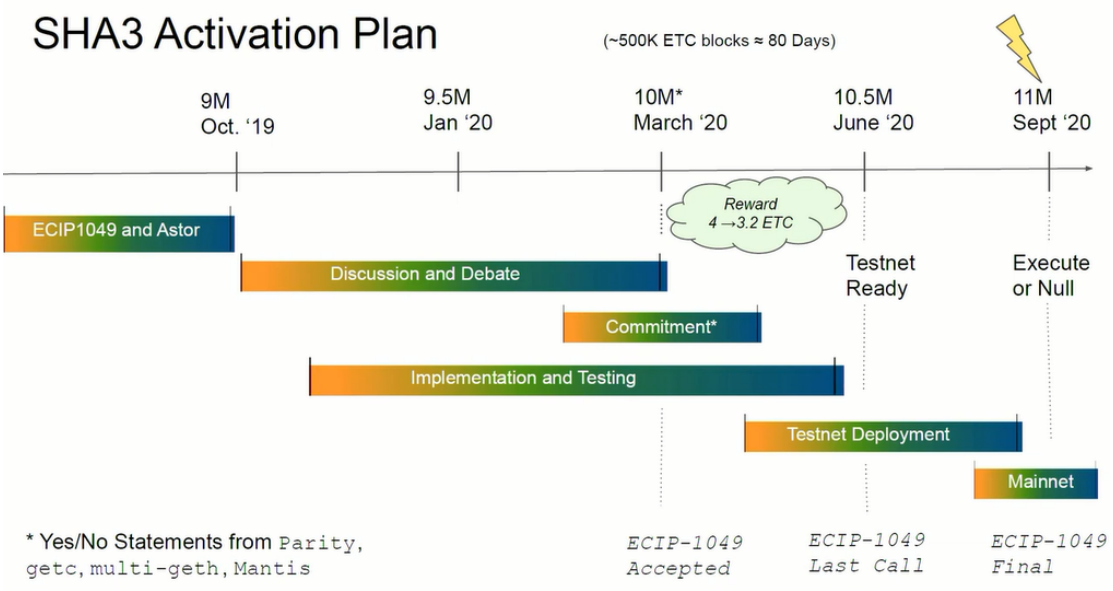
Achievements so Far: Astor SHA3 testnet.

- Creation of ECIP-1049 Proposal
- Implementation of SHA3 Proof of Work Client based on Parity.
- Creation of Open Source SHA3 CPU Miner - “Toy Miner”
- Astor SHA3 Testnet, up for 200,000+ Blocks
- Analysis of FPGA miners on Xilinx.

目前 ETC 所取得的成果：

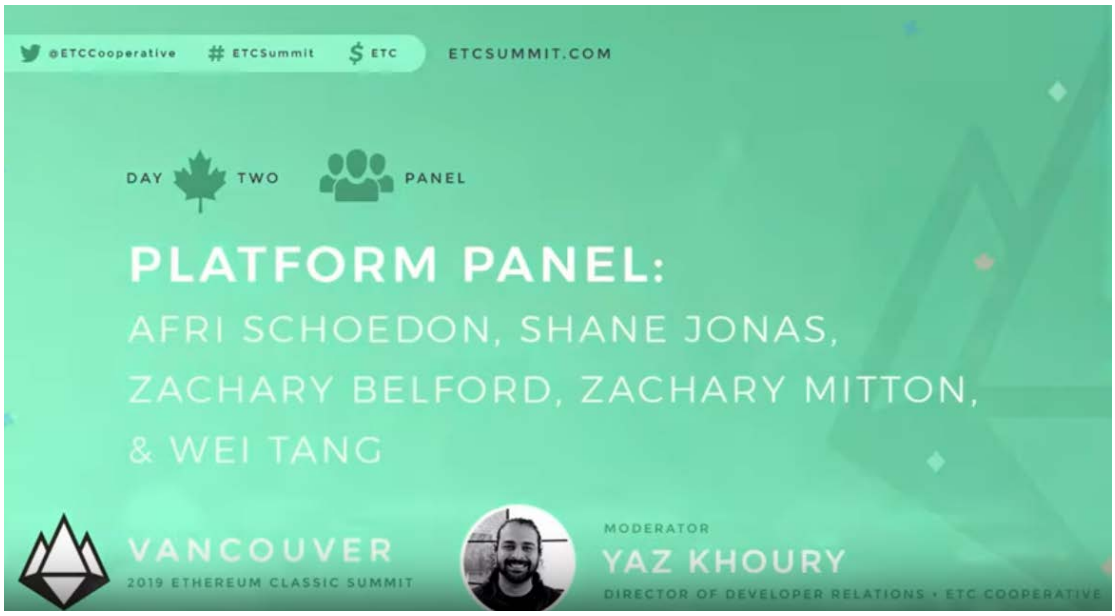
1. 创建了 ECIP 1049
2. 在 Parity 上执行了使用 SHA3 算法的工作量证明客户端
3. 创建了开源的 SHA3 CPU 矿机—Toy Miner
4. SHA3 测试网
5. 在 Xilinx 中分析 FPGA 矿机





SHA3 路线图：最佳部署时机是在区块高度达到一千一百万时（2020 年 9 月）。

欢迎加入关于 SHA3 的讨论！



Yaz：Zac，你提过稳定性更高的客户端，能给我们详细介绍一下吗？

Zac：如果在多客户端的情况下，其中一个客户端运作异常，就会面临被攻击的危险。所以我认为交易所应该使用专用软件(custom software)来监控客户端的运作。

Afri：我相信多客户端的开发能迫使我们提高客户端的质量，进行跨客户端共识测试。但我也赞同我们不需要太多客户端。

Wei Tang：现在我们可以做的事情是把 Dapp 开发和共识与运行执行层分离开来。

Zachary：请问您是研究 EVM 领域的吗？我想问当初为什么要开发有多种执行引擎的虚拟机？

Wei Tang：因为当时没有为以太经典设计的虚拟机，所以就萌生了这一想法。

Zachary：我明白了。

Yaz：现在我们有 Parity、Multi-Geth 等客户端在为 ETC 服务，对此您们有什么看

法吗？

Shane：很高兴 hyperledger 项目能够加入。

Wei Tang：我认为我们应该重视硬分叉协作。

Yaz：作为一个亚特兰蒂斯硬分叉的协调者，您觉得对于此次硬分叉升级，ETC 面临的最大的挑战是什么？

Afri：最大的挑战是社区间人与人之间的沟通。

Yaz：为什么您说我们更应该研究一个重视安全性与测试的网络？

Wei Tang：如果以太坊上的客户端共识机制不同，出现异常时，我们至少需要一天时间来停止网络的运作，需要的时间太长将导致巨大的经济损失，这就是为什么我认为网络安全性很重要的原因。

Yaz：好的。那为什么工具对于开发者建设 ETC 来说很重要？

Shane：就好比我们现有的 JavaScript Web3，同时也支持多种编程语言，所以你不仅仅可以在浏览器中与智能合约互动，还能选择多种语言进行互动。ETC 社区中，大部分 JS Web 库是由 10 个团队一起维护的，我觉得这浪费了很多开发者的时间。

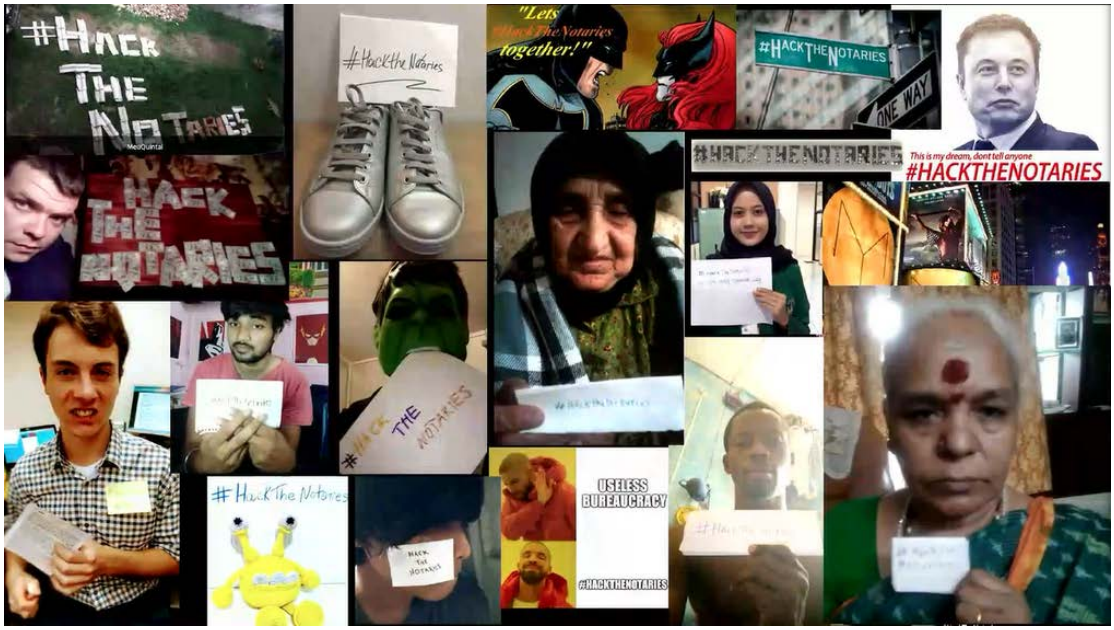
Yaz：Wei，您能给我们更加详细地介绍以太坊地兼容性吗？

Wei Tang：现有的 EIP-1884 提议重新定价操作码，以取得 Gas 费用与资源消耗之间的平衡。

受信任区块链应用如何重新定义政府治理

Redefining Governments Through Trusted Blockchain Applications

演讲嘉宾：Edilson Osorio Junior



Edison 在开头介绍了巴西存在的一个显著社会问题，即人民公证多、公正难的现状。但他们发起了一个#Hack the notaries 的运动，目的是将区块链用于解决公正问题。

OriginalMy 所采用区块链的正是 ETC，将人们的授权、签名和电子化信息都上链，因此 ETC 的不可篡改性显得十分重要。

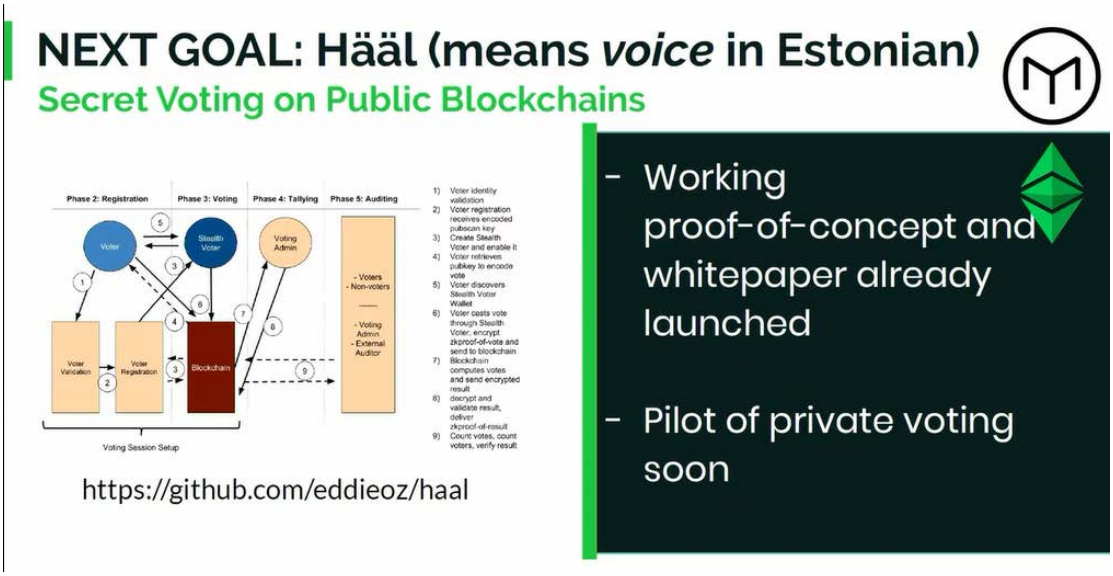
目前他们是一个部分去中心化组织，但之后会向去中心化发展，并且在区块链的可用性方面下了很多功夫，让完全不懂得区块链运作原理的人也能轻松使用。

在使用 ETC 方面，他们的 Dapp 在 ETC 的交易量中总是居于前三。

他们是如何说服政府使用区块链的呢？举个例子，MUDAMOS 是非政府组织 ITS-Rio 创建的公开请愿签名收集平台。巴西有这样一个条例：一旦公开请愿获得 1%的人民赞同，就必须在众议院甚至是国会中进行投票，但是要验证这些签名的有效性却非常耗时耗力。当采用 MUDAMOS 后，提升了公开请愿的效率，巴西政府通过了两条法例。

他们已经在公证方面取得了许多合作，目前巴西人不再需要去公证处证明文件，这对巴西来说意义重大。

他们还开展了同性民事结合运动。同性伴侣只需要在区块链中签署一份合约，就能获得与法定婚姻同等的权利。



Edilson 还提到在他演讲的前一天，巴西进行了第四次区块链电子投票。

之后他们和 ETCLabs 的伙伴们一起撰写了一个新协议，证明在区块链上也能进行无记名投票。

近几年获得的一些奖项，作为真正能够改变社会的受信任区块链，OriginalMy 在今年受到了联合国和欧盟的认可。

从技术角度看区块链合并

Merging Blockchains: A Technical Perspective

演讲嘉宾：Wei Tang

此时此刻，ETC 峰会已接近尾声，接下来由 IOHK 的共同创始人、执行董事 Charles Hoskinson 给我们带来闭幕主题演讲。

ETC 社区是一个多样的去中心化社区，这里拥抱着各种文化，你们或许会喜欢我，也可能不喜欢我，但我欣赏这样的氛围。ETC 一直在发展壮大，第一届 ETC 峰会在香港举行，第二届在韩国，峰会举办得越来越好。追溯到 2015 年，当时人在香港，接到调查任务，我们并不懂什么是权益证明，但我们有一支研究工作量证明的团队。当时我们也很重视隐私这一问题。未来数年 ETC 社区也将同样需要做出一些困难的决定。

什么是隐私、不可篡改性、互操作性引起了社区的激烈讨论。

以前我对比特币很感兴趣，但愈来愈发觉它存在许多方面的缺陷，于是我开始转向研究山寨币。为什么 ETC 能够超越以太坊，甚至比特币？因为 ETC 生态系统有足够的自由，虽然社区人之间可能存在分歧，但沟通过后大家依然是朋友。

这两天大家的演讲都很精彩，可以看出大家都很有想法。当别人叫我们不要去做某件事的时候，我们没有停下脚步，而是造就了今天的 ETC，只要把这种精神一代一代传承下去，不断创新，ETC 就会不断向前发展。