# Harmony: Open Consensus for 10B People

**Harmony** builds an *open marketplace* at Google-scale for the decentralized economy. This project aims to provide a consensus protocol over the **open Internet** at 10 million transactions per second with 100-millisecond latency and at most 0.1% fee.

Harmony's goal is to be over 1,000 times *faster and cheaper* than Bitcoin and Ethereum. We are rebuilding the decentralized economy with 10x innovations in all components: **transport network** (Google's UDP, Bloom tables, 5G mobile), **consensus protocol** (Byzantine committees, acyclic graphs, monopolist fees), and **system tooling** (unikernels, multi-core in Rust, zero-copy streaming).

We believe *communication* is the key to the future of humans and machines to create in *harmony*. As the gateway for **microtransactions** and online business, our fees must be no more than 0.1% of the transaction value to support new marketplaces of metered content or fractional work.

As the infrastructure for the **world's data firehose**, our bandwidth must scale to 10M tx/sec to support data from supply chain IoT devices or energy grids. Yet, all of the above must settle agreements within 100 milliseconds to support **instant reactions** for autonomous robots or on-chain quotes in exchanges.

We employ technical innovations that are *already proven* in research and implementation. For example, **Google's UDP** currently powers 35% of its traffic (or 7% of the Internet) with 50% latency improvement, **OmniLedger Byzantine protocol** benchmarks to 13,000 tx/sec and 1.5 sec latency with 1,800 hosts, while **unikernels in Rust** archives 10M concurrent connections on a standard 96-core machine on Amazon Cloud.

The rest of this document describes our technical architecture and company roadmap.

Sign up for Harmony! See our talk (simple-rules.com/talk) or our GitHub for the compiler prototype. Our extended team (part time) consists of **four Ph.D.s**, 3 Ex-Google, 2 Ex-Apple, graduates from Berkeley, CMU, Waterloo, Penn and Harvard.

About: *Stephen Tse coded in OCaml for 15+ years and graduated with a doctoral degree from the University of Pennsylvania on* ***security protocols and compiler verification****. He was a researcher at Microsoft Research, a senior infrastructure engineer at Google, and a principal engineer on search ranking at Apple. He founded the mobile search Spotsetter*, a startup ***Apple later acquired****. He may be reached at* s@simple-rules.com.

# Harmony: 为百亿人打造的开放式去中心化共识协议

让我们建立一个Google数据级别的开放市场平台，成为去中心化经济系统的顶梁柱。**Harmony**的目标是，在**开放的互联网**上，提供一个去中心化的共识协议，做到每秒处理千万条交易和数据，保证100毫秒左右的延迟，同时让每笔交易的手续费低于0.1%。

Harmony会做到比世界上最领先的区块链网络还要快1000倍，便宜1000倍，比如和Bitcoin和Ethereum比起来。我们会通过10倍的技术创新，重构去中心化的经济体系下的所有层面：传输层（Google的特制UDP协议， 布隆表格，以及5G移动网络），共识协议层（拜占庭委员会机制，无环图，垄断费），以及系统工具（Rust下Unikernel单内核并行计算，和零拷贝数据流)。

我们相信在未来，*通信与交互* 是人与机器融洽相处的关键。和谐与一致，即是Harmony的内在含义 。要成为一个支持**微交易**的电子商务网关平台，我们收取的交易费必须低于0.1%，才足够支持（大数据时代）市场下簇生的新兴交易模式，比如对信息内容的量化，以及工作内容的微分。作为基础设施，我们必须像是**水电网络**一样，为**全世界**的数据提供稳定可靠的通道。正因为如此，我们的带宽必须能够扩展到至少每秒千万条交易的级别。这样，我们才能够支持那些由供应链、物联网（IoT）、能源网生成的数据。在这基础上，我们还要保障每条交易从握手达成，到被整个网络认证，必须在100毫秒内完成，这样，我们才可以支持那些需要**实时处理能力**的应用，比如全自动机器人，交易所报价。

在技术方面，我们只会采纳那些已通过考验的革命性技术。这些技术都已经历过长期大范围的研究、开发、和测试。举例来说，Google的特制UDP协议，现在已经处理了Google本身超过35%的流量。这相当于整个互联网7%的流量。从实际数据上看，这个协议至少降低了用户50%的网络延迟。OmniLedger上的拜占庭协议，可以处理超过每秒13,000条交易，在1800个主机下运转，只有1.5秒的网络延迟，Rust下单内核技术，则可以在只用一个亚马逊云端96核的主机的情况下，并行同时处理一千万个网络连接。

*Stephen Tse 谢镇滔 (微信 “tsestephen”) 自高中年代起，便一直着迷于编译器和通信协议方面的研究。他曾反编译过ICQ 和X11 的通讯协议，并已用OCaml 语言编程长达十五余年。他博士毕业于宾夕法尼亚大学，专注于研究**安全通讯协议**，以及**编译器校验**方面的技术 。*

*Stephen 曾在微软研究院总部任职研究员，在Google总部任职高级软件工程师，负责基础架构方面的项目，并在苹果公司总部任职主任工程师，主导搜索排序方面的工作。他曾创立 一个专注于移动搜索的公司Spotsetter，**并被苹果公司收购**。Stephen 还是一个前Google 员工的硅谷创业者每周私人聚会的创办人和组织者（TGI-$ — 大口喝酒，大谈机器学习和区块链)。*

# Architecture and Innovations

> *“ Bottlenecks were neither the protocol nor the infrastructure. The bottlenecks were in the* ***implementation of the protocol****. ”*
> Bitcoin Unlimited at Stanford

We argue that all aspects of a consensus protocol are critical for scaling *beyond 1,000x*. Our approach is to **productionize** research innovations to the scale that serves billions of people and devices.

Currently most decentralized applications are limited to *infrequent transfers* of currencies or tokens. Harmony’s high performance will make many *real-world* decentralized applications practical. Later in this document, we will describe many use cases unique to Harmony, including *location oracles, decentralized maps,* and AI *data marketplaces*.

Let’s start with our technical differentiation. We seek and master innovations *already proven in practice*. For Harmony, we focus on the following key components: transport network, consensus protocol, and system tooling.

Our team has extensive experience building systems at the **largest scale** in the top tech companies. We will also show a prototype of our own contract language and compiler in the later sections.

## 1. Transport Network

As discussed in *Advances in block propagation* by Greg Maxwell, 12% of network bandwidth is consumed by synchronizing blocks among nodes. More critically, 2.5 round trips are needed to confirm transactions. Maxwell’s study concludes that *relay engines* and *template deltas* are important research directions to improve the propagation performance, which Harmony follows closely.

We base our network stack on **Google’s UDP**, called QUIC, which accounts for 88% of Chrome traffic (35% Google’s traffic or 7% of Internet traffic). QUIC improves *multiplexing* over sessions and achieves *zero round-trip* latency. The multiplexing saturates the network when synchronizing multiple resources, hence drastically improving bandwidth. It also eliminates many session handshakes during connection setup, which are slow and computationally expensive. Furthermore, broadcasting without round trips makes data synchronization robust to network changes and unavailability.

The research paper *The QUIC Transport Protocol: Design and Internet-Scale Deployment* [talk] covers many technical details and deployment insights for QUIC implementations. Another independent study *Taking a Long Look at QUIC: An Approach for Rigorous Evaluation* depicts Google's UDP as the future of the transport layer.

An important technique we use for transaction propagations is *Bloom tables* in set reconciliation and coding. Like the template deltas for Bitcoin mentioned above, Bloom tables enable communication between shards in constant size O(1), rather than proportional to the total transaction size O(n). Graphene employs a similar approach, described in *What's the Difference? Efficient Set Reconciliation without Prior Context*, to huge improvement.

Lastly, we design with 5G mobile network in mind, namely 1ms latency, 1M connections/km², and 10Gbps throughput, as described by Huawei 5G vision. This upcoming transport network supports 100B connections all over the world; their initial tests are already deployed in Japan and Europe.

## 2. System Tooling

One principle guiding our work is to actively seek *novel architecture* and to use *optimal languages*.

Bitcoin Unlimited recently argues the same point of an integrated approach and the quality of implementation. Its talk *Measuring maximum sustained transaction throughput* at Stanford emphasizes proper parallelization and eliminating locks through Bitcoin's system design.

The technical blog post *Terabyte blocks for Bitcoin Cash* also argues the importance of an optimal backend architecture. It claims the economic feasibility of 7M tx/sec at 10-min latency. More concretely, *A roadmap for scaling Bitcoin Cash* discusses many proposals for systems optimization: memory map, parallelization, and indexing unspent outputs.

We take an extreme view on eliminating the bottlenecks in our protocol implementation. Our inspiration comes from *Mosaic: Processing a trillion-edge graph on a single machine* (talk), which achieves 59x speedup on the most challenging distributed benchmarks, including PageRank and community detections. Its novel innovation is to use *Hilbert-ordered tiling scheme* for locality, managing to process graphs with 1 billion nodes and 1 trillion nodes on a single host. Mosaic runs its benchmarks on standard machines with 244 cores and 850K IOPS, which are readily available on Amazon cloud at a low cost.

The most severe overheads of parallel processing are *locks* and *garbage collections*. Typically, the more processing cores in the machine, the more contention for the same resources, which are marked mutually exclusive using locks. Similarly, the more memory in the machine, the longer it takes to scan for free space among all data structures. We implement lock-free, multi-core algorithms in Rust with allocator-free regional memory management. In particular, we follow *Destination-Passing Style for Efficient Memory Management* to achieve stack and region-based O(1) allocators for pause-free processing.

The last barrier is the system kernel itself. Typical network programs spend more than a third of the time in the kernel and in switching contexts. With *unikernels*, program executions happen in Ring 0 to eliminate context switches to kernels. *Unikernels: The Rise of the Virtual Library Operating System* describes the MirageOS and Jitsu that boot in just 20 milliseconds.

One of our approaches is to saturate the network capacity of the underlying system. *Reliable Messaging to Millions of Users with Migratory Data* provides a solution to the C10M problem, namely 10 million concurrent connections on a single host. We will explore the latest techniques there for blockchain, including *zero-copy streaming* and *succinct indexing* with a memory-only database.

## 3. Consensus Protocols

Scalable decentralized protocols are an active area of research. *Consensus in the Age of Blockchains*, written as a systemization of knowledge, is the best introduction to the latest results. We research extensively into the *Most Cited Byzantine Consensus Publications*, which cover many developments for distributed consensus and state machine replication.

One line of research stands out: *OmniLedger: A Secure, Scale-Out, Decentralized Ledger via Sharding* [talk] by Bryan Ford’s lab at EPFL. OmniLedger forms committees for sharded state replication, efficiently handling faults and attacks in the open network. OmniLedger is the basis of our protocol; we will discuss its details in the next section.

One of the main predecessors of OmniLedger is *Algorand: Scaling byzantine agreements*, which uses Verifiable Random Functions and a stateless agreement protocol. Algorand benchmarks to 500k users and 50s latency, achieving 125x of Bitcoin throughput.

Another key, proven technique is using acyclic graphs for inclusive mining to avoid confirmation forks or orphan blocks. *Bitcoin-NG: A Scalable Blockchain Protocol* [blog], building on top of its own early work *SPECTRE: Serialization of Proof-of-work Events: Confirming Transactions via Recursive Elections,* is now a common primitive for separating leader elections from transaction confirmations.

Lastly, our protocol employs a new approach called *monopolist fees*, which uses game-theoretical incentives to disclose true fees without waste. Our model is based on *Rethinking Bitcoin's Fee Market* [blog], which discusses alternative methods for determining simple fees to improve miners' revenues.

# Protocols and Optimizations

After extensive research, we conclude that OmniLedger is the most scalable permissionless protocol. Most importantly, its publication *OmniLedger: A Secure, Scale-Out, Decentralized Ledger via Sharding* is peer reviewed in the top research conference *IEEE Symposium on Security and Privacy*. OmniLedger is tested with 1,800 hosts (25 committees, each consisting of 72 nodes) and 13,000 tx/sec.

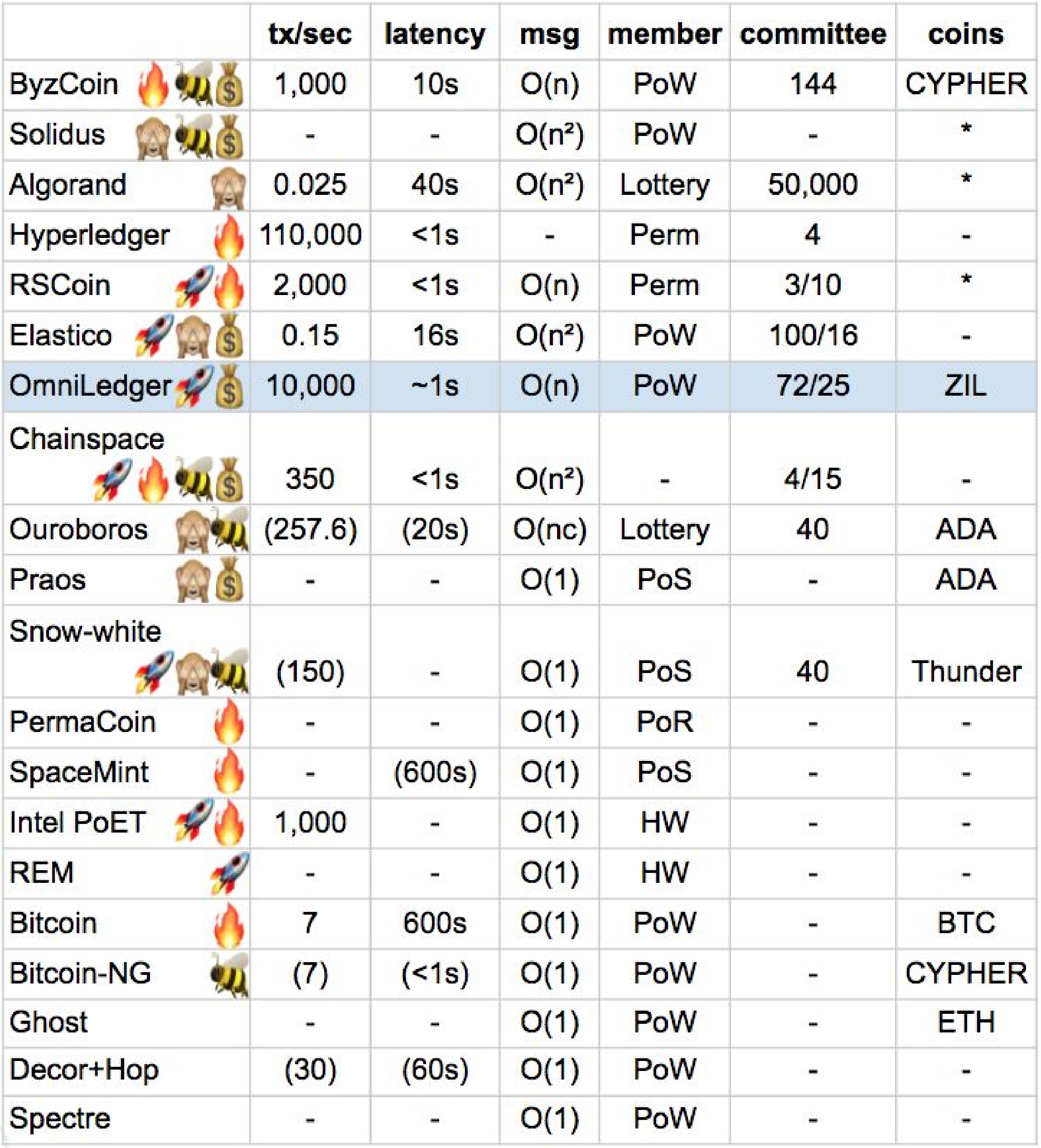
The rest of this section describes other protocols targeting scaling and high performance. Many share similar techniques; therefore, we present a critical framework to understand the principles and optimizations for comparison. Lastly, we formulate our adversary model and address some common attacks.

## 1. High-Performance Protocols

The table on the following page, based on SoK: Consensus in the Age of Blockchains, compares protocols that are peer reviewed or openly validated. The table lists specific transactions per second and measured latency (the second and third columns). Note that the *most promising protocols* are tested with large numbers of nodes and sharding committees (the sixth column). We would emphasize that the **message complexity** (the fourth column) is often overlooked but critical in scaling for production.

To highlight the value and the importance of blockchain scaling, here are the projects closest to Harmony, and their valuations as of April 26, 2018:

1. EOS, valued $14B.
2. Zilliqa, valued $700M.
3. Hashgraph, raising $300M.
4. Thunder, valued $100M.
5. Dfinity, raised $100M.
6. Kadena, raised $12M.
7. Algorand, raised $4M.



| | tx/sec | latency | msg | member | committee | coins |
|---|---|---|---|---|---|---|
| ByzCoin 🔥🐝💰 | 1,000 | 10s | $O(n)$ | PoW | 144 | CYPHER |
| Solidus 🙈🐝💰 | - | - | $O(n^2)$ | PoW | - | * |
| Algorand 🙈 | 0.025 | 40s | $O(n^2)$ | Lottery | 50,000 | * |
| Hyperledger 🔥 | 110,000 | <1s | - | Perm | 4 | - |
| RSCoin 🚀🔥 | 2,000 | <1s | $O(n)$ | Perm | 3/10 | * |
| Elastico 🚀🙈💰 | 0.15 | 16s | $O(n^2)$ | PoW | 100/16 | - |
| OmniLedger 🚀💰 | 10,000 | ~1s | $O(n)$ | PoW | 72/25 | ZIL |
| Chainspace 🚀🔥🐝💰 | 350 | <1s | $O(n^2)$ | - | 4/15 | - |
| Ouroboros 🙈🐝 | (257.6) | (20s) | $O(nc)$ | Lottery | 40 | ADA |
| Praos 🙈💰 | - | - | $O(1)$ | PoS | - | ADA |
| Snow-white 🚀🙈🐝 | (150) | - | $O(1)$ | PoS | 40 | Thunder |
| PermaCoin 🔥 | - | - | $O(1)$ | PoR | - | - |
| SpaceMint 🔥 | - | (600s) | $O(1)$ | PoS | - | - |
| Intel PoET 🚀🔥 | 1,000 | - | $O(1)$ | HW | - | - |
| REM 🚀 | - | - | $O(1)$ | HW | - | - |
| Bitcoin 🔥 | 7 | 600s | $O(1)$ | PoW | - | BTC |
| Bitcoin-NG 🐝 | (7) | (<1s) | $O(1)$ | PoW | - | CYPHER |
| Ghost | - | - | $O(1)$ | PoW | - | ETH |
| Decor+Hop | (30) | (60s) | $O(1)$ | PoW | - | - |
| Spectre | - | - | $O(1)$ | PoW | - | - |

Consensus protocols in open research from SoK Consensus.

| | |
|---|---|
| 🚀 scalable to 100K nodes | 🐝 vulnerable to DoS |
| 🔥 source code available | 💰 incentive to join committee |
| 🙈 vulnerable to tx censorship | |

## 2. Principles for Scaling

Most protocols share similar techniques but different terminologies for scaling. It is critical to specify the *metrics* and the *complexity* of different implementations. Here we present a common framework in OmniLedger's terminology.

1. Representative sharding
   O(1)-size multi-signatures for 10k nodes vs 16-node PBFT. Crypto sortition via randomness from multi-party computation and commit-then-reveal step.

2. Gradual transition
   Sybil-resistant identities to maintain liveness when swapping shards. A sliding window from a fixed permutation to ensure ⅔ honest majority.

3. Atomic shard-commit
   Each shard uses O(log n) multicast tree-based BFT to unanimously accept cross-shard transactions with O(1)-size coordination.

4. Parallelizing blocks
   Acyclic graphs to capture transaction dependencies transitively. Divide each shard into groups to replace faulty nodes with a view-change.

5. Pruning checkpoints
   State blocks for storage and bootstrapping against Byzantine DoS. Multi-hop, collectively signed back pointers, 100x space savings.

6. Optimistic confirms
   Trust but verify low-value transactions with shard deposits. Guarantee finality in ~1s with penalty linear to loss and detection in minutes.

## 3. Model for Attacks

*A Survey on Security and Privacy Issues of Bitcoin* discusses the common models and many vectors of attacks for decentralized protocols.

One critical problem for sharding is the *high churn* for OmniLedger and Chainspace. Churn happens when network nodes temporarily go offline or unresponsive. We follow The Honey Badger of BFT Protocols; its randomized consensus helps mitigate the churn problem that guarantees liveness without making any network timing assumptions.

Another common problem is *single-shard DoS attacks*. One defense is private leader election and fast view change. OmniLedger optimizes ByzCoin by reusing randomized seeds to elect group leaders. In addition, OmniLedger's state block and *Vault: Fast Bootstrapping for Cryptocurrencies*'s Merkle checkpoints help sync when switching shards.

# Locations and AI

This section highlights some features and applications **unique to Harmony**. Our team has extensive experience with geospatial data and machine learning.

## 1. Location Oracles

It is a challenge to integrate smart  contracts with oracles that serve as authenticated data feeds. Nodes must be able to independently verify an oracle's consistency. We study *Crux: Locality-Preserving Distributed Systems* for optimizing routing and for exposing network topology. The former is useful for sharding that respects physical distances. The latter can take the GPS signals of mobile or IoT devices as proof of location in our applications.

With tight integration of locations, Harmony will be well-suited to support applications for smart cities. For example, autonomous vehicles can fetch verified location data in upcoming trips. Or, imagine thousands of swarm robots, self-organizing around a common mission in an unknown terrain.

## 2. Decentralized Maps

Maps for geocoding and points of interest can be a showcase for decentralized applications in the real world. A good starting point for building decentralized maps on Harmony can be augmented reality games with incentives like Pokémon.

The competitive advantage of decentralized maps is the long-tail, community-specific content. For example, a school can mobilize all of its staff to map out its buildings and playgrounds in a day; any student or organization can then build games and events on top of the location data without coordination.

## 3. AI Data Marketplace

Harmony will also serve as a high-volume data marketplace and optimize its machine learning performance. We follow *Blockchain-based Machine Learning Marketplaces* to build a new decentralized economy based on data.

The key aspects are privacy-preserving applications and transparent data usage. The effectiveness of multiparty computation and homomorphic encryption relies heavily on high performance platform such as Harmony.

# Contracts and Beyond

Harmony also explores the design space and the scaling of smart contracts. We have designed a new programming language, **Min** (see min-lang.com)**,** and built a prototype compiler to demonstrate its ease and the security.

This section describes the background of formal verifications and language-based security. These techniques are the state of the art against vulnerabilities in the decentralized economy.

## 1. Formal Verification vs Hacks

Open and connected systems create opportunities for new applications as well as attacks and hacks. Beyond the web for information and the social network for identity, security for assets and contracts is paramount to building a decentralized economy.

Kevin Hartnett of Quanta Magazine draws the same conclusion in "Computer Scientists Close In On Perfect, Hack-Proof Code"[1] published in Wired:

> *"Key parts of Little Bird's computer system were **unhackable** with existing technology, its code as trustworthy as a mathematical proof... That results made **all of Darpa** stand up and say, oh my goodness, we can actually use this technology in systems we care about."*
>
> *"Previously, when computers were isolated in homes and offices, programming bugs were merely inconvenient. Now those same small coding errors open massive **security vulnerabilities** on **networked machines** that allow anyone with the know-how free rein inside a computer system."*

Yet, Bitcoin and Ethereum are learning these lessons anew. In "A survey of attacks on Ethereum smart contracts"[2], Atzei et al. analyze the $60M DAO attack and other security vulnerabilities:

[1] https://www.wired.com/2016/09/computer-scientists-close-perfect-hack-proof-code by Kevin Hartnett
[2] https://eprint.iacr.org/2016/1007.pdf by Nicola Atzei, Massimo Bartoletti, Tiziana Cimoli

- Interrupting money withdrawal before the balance is updated, draining the entire account's asset;
- Leaky abstractions and arbitrary limits of gas usages and functions in virtual machines, causing untestable behaviors in corner cases;
- Executing a contract at a dynamic address is not checked with its static specification, causing the under-specified behavior of an exception or a fallback.

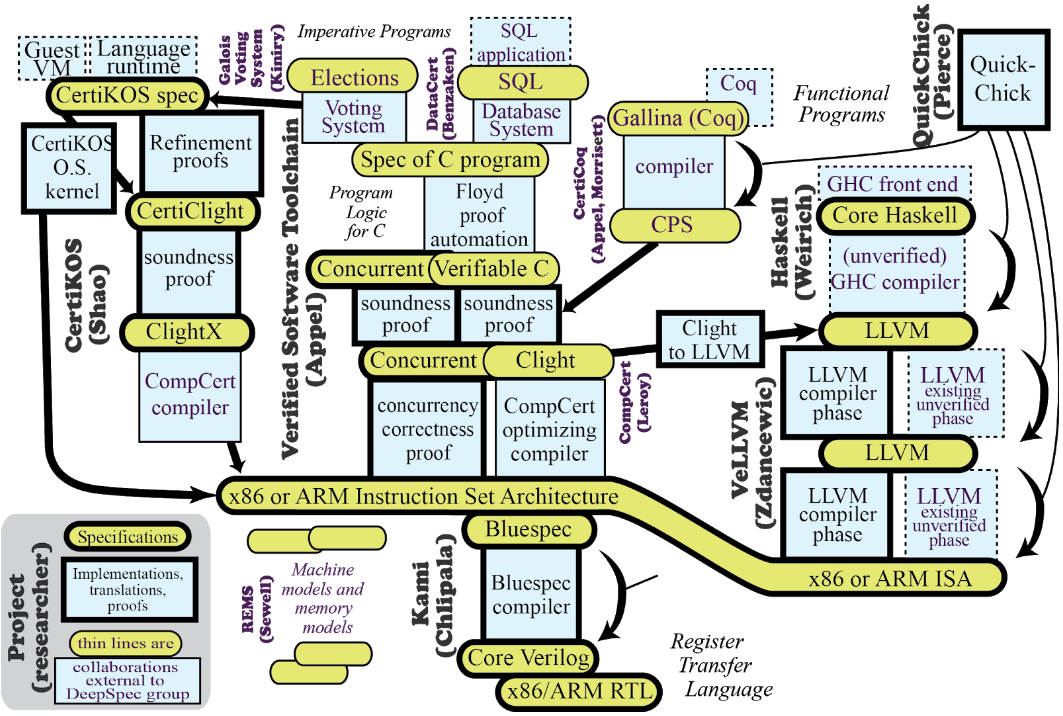
More recently, in "An In-Depth Look at the Parity Multisig Bug"[3], Initiative for Cryptocurrency and Contract (IC3) at the Cornell University analyzes the $30M Parity Multi-Sig Wallet Attack.

## 2. Language-based Security

Reactive methods for security such as testing, auditing, and monitoring are costly and incomplete. They fail to reason or prove beyond doubt that assets are secure or contracts are consistent with top-level goals.

Language-based security[4] is a mathematically based technique to verify that programs behave exactly as they intend at a rich specification. Below is the architecture diagram of "The Science of Deep Specification"[5], one of the maximally funded "Expeditions in Computing" of the National Science Foundation.

[3] http://hackingdistributed.com/2017/07/22/deep-dive-parity-bug by Breidenbach, Daian, Juels, Sirer
[4] http://repository.cmu.edu/cgi/viewcontent.cgi?article=1921&context=compsci A Language-Based Approach to Security, by Fred Schneider, Greg Morrisett, Robert Harper at CMU
[5] https://deepspec.org/page/Research Expeditions in Computing, funded by NSF



Architecture diagram of "The Science of Deep Specification"

We are applying this technique of language-based security to decentralized applications to guard against incidental mistakes or malicious attacks, including[6] transaction-ordering dependence, mishandled exceptions, timestamp dependence, and reentrancy vulnerability.

**Min**, a Harmony subproject, is a new language for programming software with security guarantees, leading to unhackable systems. In short, we devise static types with concise syntax as security specification, analyze decentralized protocols in formal models, and generate optimized code across networks.

## 3. Fairness and Efficiency

> *"Scalability is probably problem number one... There's a graveyard of systems that* ***claim to solve*** *the scalability problem but don't. It's a very significant and hard challenge."* Vitalik Buterin[7] at DevCon3

[6] http://www.comp.nus.edu.sg/~loiluu/papers/oyente.pdf Making Smart Contracts Smarter
[7] A Modest Proposal (sharding to solve the trilemma of scalability + decentralization + security)

Our focus is *on-chain scaling* to support 1,000x more transactions and applications[8][9][10]. Sharding contracts is more challenging than sharding states.

We follow *Chainspace: A Sharded Smart Contracts Platform* to build a distributed commit protocol with audit for a Turing-complete platform. Our approach is to:

1. Define a useful subset of limited scripting
2. Expose transaction dependencies during sharding,
3. Annotate computational contracts with stateless verifiers.

We plan to integrate the following aspects of a consensus protocol but their details are beyond the scope of this document.

- Scripts and contracts: Covenants[11], Ethereum[12]
- Fairness and efficiency: anti-pooling[13], proof of useful work[14], proof of stake[15]
- Security and privacy: multi-signatures[16], attack models[17] [18], verification[19] [20]
- Off-chain and edge clients: Lightning[21], IoT[22] [23]

[8] 10M tx * log(10B consensus) * 60*60*24 sec * 512 UDP bytes = 9PB/day, or 17 mins per global vote

[9] SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies

[10] On Scaling Decentralized Blockchains (max 27 tx/sec and 12s latency despite blocksize or intervals)
On side planes: *"Their capacity, ability to find routes, achieved throughput, latency, and privacy guarantees depend fundamentally on emergent properties of the payment network graph, such as the value capacity of peer-to-peer channels, the discoverability of routes, the online status of nodes involved."*

[11] Bitcoin Covenants (vaults to deter key thefts, poisons to penalize double spendings)

[12] https://github.com/ethereum/wiki/wiki/Sharding-FAQ (data availability vs cryptographic accumulators)

[13] Nonoutsourceable Scratch-Off Puzzles to Discourage Bitcoin Mining Coalitions (proof of retrievability)

[14] REM: Resource-Efficient Mining for Blockchains (vs trusted elapsed time on Intel's SGX)

[15] Ouroboros: A Provably Secure Proof-of-Stake Blockchain Protocol (vs grinding vulnerability)

[16] Securing Bitcoin wallets via a new DSA/ECDSA threshold signature scheme

[17] Bitcoin's Security Model Revisited (cost effective double spending attacks on lightweight clients)

[18] Bitcoin Confirmation (6 confirms to defend against 10% hashrate, 60 confirms against 40%)

[19] Computer Scientists Close in on Perfect, Hack-Proof Code (NSF $10M funding and DeepSpec)

[20] Penn Computer Scientists Join NSF "DeepSpec" Expedition to Eliminate Software Bugs UPenn PL club

[21] A fast and scalable payment network with bitcoin duplex micropayment channels

[22] Blockchains and Smart Contracts for the Internet of Things (marketplace of services between devices)

[23] Blockchains in Mobile Networks (smart city, iot contracts, content marketplace, edge negotiations)

# Team and Collaborators

## 1. Passion and Team

Stephen Tse 谢镇滔 has been obsessed with protocols and compilers since high school. He reverse-engineered ICQ and X11 protocols, coded in OCaml for 15+ years, and graduated with a doctoral degree from the University of Pennsylvania on **security protocols and compiler verification** .

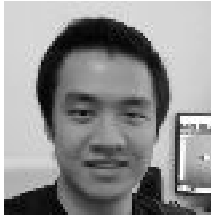
Stephen was a researcher at Microsoft Research, a senior infrastructure engineer at Google, and a principal engineer on search ranking at Apple. He founded the mobile search Spotsetter with institutional venture capital; **Apple later acquired** the startup.

Nicolas Burtey founded a VR video startup in 2012 that **grew to 40 people and raised $10m**. Orah served the needs of thousands of professional content creators in 70 countries by selling GPU-driven live stitching software and 360° cameras.

Nicolas has a bachelor degree in mathematics and computer science, and a master degree in computational photography. His master thesis at Ecole Nationale Supérieure Louis Lumière is “*The representation of space and time in panoramic photography.”*

Alok Kothari has worked on deep learning models for natural language understanding at Apple Siri. He has conducted research in word sense disambiguation, machine translation, and social media retrieval. Alok has published at top conference venues including SIGIR, ICWSM and EMNLP. His research paper won the best dataset award at ICWSM 2013.

Alok published a best seller book “Game Changers,” chronicling successful graduates from his alma mater IIT Kharagpur in India. He obtained his master degree in **artificial intelligence (language technologies)** at the Carnegie Mellon University.



Rongjian Lan was a search infrastructure engineer for Play Store at Google. He has published 10+ academic papers on **spatio-temporal querying** and map-based visualization. Rongjian has researched on decentralized protocols extensively since early 2017.

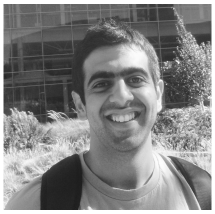
Rongjian is the co-chair of ABC Blockchain Foundation with 100+ engineers from Google, Facebook, Linkedin as members. He was a doctoral candidate of computer science at University of Maryland College Park, and obtained his bachelor degree from the University of Science and Technology Beijing.

Minh Doan worked at Google for 5 years on Assistant, Play and Plus. He was a champion at **USA Computing Olympiad Open** and many other Informatics Olympiads. Minh holds the patent at Google for “*Publisher Click-Ring Fraud Detector.*”

Minh was a doctoral candidate in algorithms and distributed systems at University of California, Irvine. He has a Master’s degree in Computer Science and Applied Math at Moscow State University. Minh published the paper "*An effective ant-based algorithm for the degree-constrained minimum spanning tree problem*” in IEEE Congress on Evolutionary Computation.

Nick White graduated with a BS and MS in **Electrical Engineering from Stanford** University where he focused on *Signal Processing, Optimization and Control*. Upon graduation, Nick taught and researched artificial intelligence and applied mathematics with Prof Bernard Widrow at Stanford.

Nick served as the AI Specialist for Hong Kong-based AI incubator Zeroth.AI where he coached over 20 teams from 5 continents across diverse industries including finance, agriculture, robotics and blockchain. Nick is also a *certified yoga instructor* and an avid surfer.



Sahil Dewan founded his ed-tech startup, FuturEd, where he launched a mobile platform for alumni engagement and fundraising, tapping more than 100 educational institutions. He also served as the country president of AIESEC, a global youth leadership organisation present across 125+ countries.

Sahil is a graduate of **Harvard Business School**, where he also served as the President of the Blockchain & Cryptocurrency Club. He has been involved in the blockchain space while working for Draper Dragon Fund and advising a few blockchain projects.

Kunal Patel is responsible for the security design and architecture of Samsung Pay and various Samsung Knox projects. He is a member of the B2B core security team, which provides internal reviews, partnership evaluations, *program analysis research*, and incident response for numerous Samsung initiatives. Over the years, Kunal has worked with researchers and executives of startups in anti-malware, cryptography, program verification, machine intelligence, and authentication.

Kunal obtained a bachelor degree in computer science from North Carolina State University. His specialties include **systems security, applied cryptography, protocols, and platforms**. Kunal is intensely interested in *understanding intelligence* in all its forms.

Eileen Cao graduated with BA in financial management. She worked at Finance Service Center for Global 500 companies for 6 years, Burtey's startup Orah, and e-commerce gaint JD.com.

Eileen graduated from University of Texas at Austin with a master degree in technology commercialization. Eileen is a member at *American Poolplayers Association* and has won a regional championship in Austin Texas.

Hakwan Lau is a full professor at University of California, Los Angeles. He specializes in neuroscience and machine learning. Hakwan studied at Oxford on the prestigious **Rhodes Scholarship**, was an associate professor at Columbia University, and published 90+ papers in peer-reviewed journals.

Hakwan's latest paper at Science, "*What is consciousness, and could machines have it?*", was well cited among the technical researchers. With Harmony, he is exploring the connection between probabilistic consensus protocols and brain communication. Hakwan is also studying privacy-preserving modeling of mental patience's data on blockchain.

(Academic Consult) Georgios Fainekos is a tenured professor at Arizona State University. He specializes in formal methods of cyber-physical systems, **swarm robotics**, and machine learning for real-time planning. Georgios did his doctoral studies at the General Robotics, Automation, Sensing & Perception Labs (GRASP Labs) of the University of Pennsylvania.

## 2. Expertise and Collaborators

Zi Wang worked at Google from 2006-2015 on Chrome, Google [X], Android and Nexus. He was the first **Global Creative Director** for the Google hardware division and co-founded a Google research lab with a $20M budget.

Zi founded Quantum Bakery, a startup partnering with Google, Corning and Toyota to develop consumer products with ambient intelligence. He holds a bachelor's degree in computer science and a master's degree in economics.

Trausti Kristjansson has worked at the industry's most respected research labs (Microsoft Research, IBM Research, Google Research), founded full-stack startups and led PhD engineers at the top of their field as an engineering Director.

Trausti's expertise includes machine learning for speech and massive compute over billion consumers' data. His research paper in 2004 "Interactive Information Extraction with Constrained Conditional Random Fields" won the award at AAAI, the top academic conference in artificial intelligence. Trausti received his **doctoral degree on machine learning** from the University of Waterloo.

Bruce Huang 黄海旻 was an engineer lead at Microsoft for 7 years, a director at Alibaba Cloud and at Credit Ease. Later, he was the CEO of Madailicai, a top peer lending company in China. Bruce is a certified snowboard instructor. He obtained his master's degree in computer science at the Simon Fraser University.

Isaac Zhang 张得志 started designing civil engineering systems at age 11. He cofounded a smart city and indoor mapping startup, Locision, that raised $20M.

Isaac seeks to be a thinker and tinker, and enjoys **programming in Mathematica** the most. He obtained his doctoral degree on localization at the University of Nottingham.

Aaron Li started programming since the age of 7 and started mining Bitcoin since 2011. He won the prestigious ACM **SIGKDD best paper** award on language topic modeling while completing his master's degree at the Carnegie Mellon University.

Aaron was a researcher at the Google Machine Learning team and a founding engineer at Scaled Inference, an AI startup advised by Prof Michael Jordan. A New Zealand citizen, Aaron was granted the Extraordinary Ability Visa in the US to create fundamental, disruptive technologies for humans.

Navneet Singh led mobile engineering at Google for 6 years. He has also been the Senior Vice President at the **largest payment processor** WorldPay and the Head of Data Science at the mobile payment provider Ezetap.

Navneet graduated from the Massachusetts Institute of Technology with bachelor's and master's degrees in Computer Science. He speaks 6 languages, has extensive experience with distributed machine learning, and has a deep passion for transforming businesses with data-driven innovation.

Michael Young co-founded and led 3 venture-backed companies to acquisitions, including one to a key supplier of Apple. Bringing expertise in computer vision, he has closed deals with Fortune 500 companies, the US Department of Defense, and strategic partners across China and Japan.

An inventor of 20 patents, Michael raised $35M+ and was selected as one of "**America's Best Entrepreneurs**" by Bloomberg. He graduated from the University of Pennsylvania with a bachelor's degree in Physics and was a Ph.D. drop-out from Stanford.

Kushagra Shrivastava runs the Xoogler investment syndicate. He worked at Google from 2006-2014, leading marketing and strategy for Google Play and Android Apps. Graduated from the INSEAD Executive Program for marketing, Kushagra was the chief executive for products at an advertising startup acquired by Pinterest.

Kushagra is the Senior Director for the formerly Yahoo Small Business. He is also a fellow at **Stanford Center for Legal Informatics**, founding its Blockchain Group and actively contributing to its Computable Contracts project.

# Milestones and Roadmap

## 1. Achievements and Milestones

Our prototype compiler is at https://github.com/min-lang/min with the MIT open source license. It compiles Min, the security-verifying programming language, directly to machine code, eliminating the common dependencies of libraries or system tools. Currently, the compiler bootstraps itself in x86-64 instructions and supports development in Mac OS. In an unpublished repo Min also compiles to Java VM without any third-party tools.

The author (Stephen Tse) has been tinkering with the language design and compiler of Min for more than a decade. Our publications on security protocols and language design include:

- Verified interoperable implementations of security protocols, ACM Transactions on Programming Languages and Systems, 2008;
- Dynamic security policies, Doctoral thesis at the University of Pennsylvania, 2007;
- Run-time Principals in Information-flow Type Systems, IEEE Symposium on Security and Privacy, 2004.

While the context of the research above has been web protocols and information security, our experience and results extend to decentralized applications and digital assets. *Our theories* are based on type systems of lambda calculi; they link security specification or policies to dynamic networks with run-time principals. On the other hand, *our tools* are based on ProVerif and OCaml; they verify that the protocol implementations are interoperable with the specification to guarantee information-flow security.

## 2. Roadmap for Launch

- 2018 Q2
  - Validating **OmniLedger protocol** in Go sustains 10k tx/sec and adversarial attacks with 10k nodes
  - Completing legal setup and distribution model for tokens
  - Forming a team of **5 engineers** and raising a $?M seed round

- 2018 Q3
  - Opening public benchmarks for 10k nodes
  - Validating **lock-free and allocator-free algorithms** in Rust scale linearly on a 96-core unikernel on Amazon Cloud
  - Validating **Google's UDP** sustains broadcast at 100ms latency with 10k nodes and saturated 10 Gbps links

  - Raising $?M as the initial private token presale

- 2018 Q4
  - Opening **public mining** of Harmony tokens with testnet and token economics
  - Deploying to **10k devices** for IoT, autonomous robots, supply chain operations
  - Raising $?M as the final private token presale

- 2019
  - Deploying to 100k tx/sec for payments and financial institutions
  - Deploying to **100k nodes with 100k tx/sec** and 1s latency
  - Listing Harmony tokens on crypto exchanges
  - Supporting contracts, anti-pooling, multi-signatures, mobile clients

# Tokens and Sales

The token model of Harmony aims to build a **sustainable platform** with help from decentralized developers and aligned investors. We balance the long term commitment of *development* (40% tokens reserved for developers, 6 years of research expenditures, 4 years of founders' vesting) with the tiered incentives of *investment* (40% bonus, 3-month 40% lockup).

The total hard cap is **$?M USD**, and there are **21 billion** tokens. Investment is denominated in the 1-hour weighted USD price of ETH. All unlocks start from *the listing on an exchange.*

## 1. Token Model and Incentives

- Sales & floating: 20% tokens (40% no lockup, 10% monthly over 6 months; for bonus, 100% release after 1-year lockup)
    - $0M - $?M: 40% bonus, seed round
    - $?M - $?M: 20% bonus, initial private sale
    - $?M - $?M: 0% bonus, final (public or private) sale

- Developer & community: 40% tokens (1-year 28% lockup, 1.5% monthly over 4 years)
    - Open benchmark, mining, airdrops

- Foundation & research: 28% tokens (28% no lockup, 1.0% monthly over 6 years)
    - Reviewed publication, research grants, reserve

- Founder & team: 12% tokens (1-year 28% lockup, 2% monthly over 3 years)

## 2. Planned Use of Proceeds

- 40% for technical development of protocol platform
- 20% for community engagement, developer programs
- 15% for marketing, business development
- 10% for operations, equipment, cloud servers
- 10% for collaboration with academic research
- 5% for compliance, legal, finance

# Questions and Answers

## 1. Frequently Asked Questions

1. *Who are your competitors with published research and open code/benchmark?*

    - Dfinity: extensive publications on crypto, raised $100M+, talk at Stanford
    - Algorand: SOSP '17 paper, Turing award laureate Silvio Micali, raised $4M
    - Zilliqa, maxed at $550M market cap, with verified protocols & verified contracts
    - Thunder, Ocean, IOS, Emotiq, Ellcrys, and æternity cite OmniLedger or ByzCoin

    In contrast:

    - Stellar, maxed at $11B market cap, is a permissioned network (see Ripple story)
    - IOTA, maxed at $14B market cap, had serious vulnerabilities and incompetencies
    - Rchain, maxed at $890M market cap, lacks technical evidences
    - *"Blockchain Consensus Protocols in the Wild"* debunks Tendermint, Symbiont, R3 Corda, Iroha, Kadena, Chain, Quorum, MultiChain, Sawtooth Lake, Ripple, Stellar, and IOTA against Hyperledger Fabric.

2. *Why do you target 10M tx/sec?*

    Because 10M tx/sec is **plausibly feasible** and **extremely useful** in 3 years. Terabyte Block Project argues for the economic feasibility of 7M tx/sec, while OmniLedger and Rust propose *linear scaling* over network nodes and machine cores. These innovations together will bring a 1,000x breakthrough to scaling transaction rate.

    Every day, the US equities market totals 9.0B trades (104k tx/s), Facebook handles likes by 800M people (93k tx/s), and WorldPay processes 110M payments (1273 tx/s). These applications are the prime targets for high-volume transactions on *a single protocol*.

3. *How will Harmony change an average user's life?*

    Computers automate tasks, Internet delivers information, smartphones bring mobility at almost no cost. Harmony will bring the next revolution of the **decentralized economy** to the masses, in which enforcing transactions and contracts is essentially free.

    Harmony will enable **disintermediation of trust** where anyone can create businesses without a central authority. For example, all 10B people can vote on a bill in 17 minutes;

or, organization resources can be efficiently re-allocated every second.

4. *Who are the first 1,000 passionate users?*

    ○ Token traders with bid/ask quotes to be openly binding on exchanges.
    ○ **Tips for casual entertainment** or content without creating accounts or channels.
    ○ A marketplace for services and payments of IoT devices.

5. *Bitcoin needs 60 minutes for 6 confirms and allows 7 tx/sec, while Ethereum allows 20 tx/sec. Is Harmony a million times faster?*

    Yes, our goal is 10M tx/sec and 100ms latency. Hence, Harmony allows 1,000,000x more and **36,000x faster** transactions.

6. *If OmniLedger is 13,000 tx/sec, how does Harmony make the leap to 10M tx/sec?*

    Through 10x-100x innovations in all layers of the protocol: networks, algorithms, and implementations.

    OmniLedger achieves 13,000 tx/s with 1,800 hosts in a research benchmark. It assumes single core, 20Mbps and 100ms network. Our further boosts to 10M tx/sec might come from: 100x more nodes (including light clients), 10x network (1Gb and 20ms world round trip with backbone relay), and 10x manycore graph processing.

    Some subtle but key points in OmniLedger are: O(1) signatures and O(log n) commits for PBFT, and optimistic confirms depending on transaction values. The former helps scale along the number of nodes, while the latter helps scale along the number of small transactions.

    Other **disruptive** technologies are Google's UDP for broadcasting blocks without round trips or re-transmissions, as well as multi-core unikernels for lock-free and allocator-free streaming processing in Rust.

    See Section "Architecture and Innovations" for other techniques and reference.

7. *What's Proof of Synchronization and Bandwidth? Did you coin the term?*

    TorCash invents Proof of Bandwidth for Tor relay routing. (See *A TorPath to TorCoin: Proof-of-Bandwidth Altcoins for Compensating Relays*.) Harmony combines the bandwidth contribution and **the work of synchronizing transactions** as the basis of fairness and incentives. We are working out the details at the moment.

## 2. Further reading

- Bitcoin's Academic Pedigree, Communications of the ACM, 2017 Dec
- Bitcoin and Cryptocurrency Technologies [the definitive reference book]
- Decentralization in Bitcoin and Ethereum Networks [blog]
- A survey on security and privacy issues of Bitcoin [blog]
- Programming and proving with distributed protocols [blog]
- Verdi: A Framework for Implementing and Formally Verifying Distributed Systems
- Mechanising Blockchain Consensus
- My VM is Lighter (and Safer) than your Container (8000 VM, 2.3ms boot, 480K size)
- Initiative for cryptocurrency and Contract (IC3 at Cornell, Berkeley, UIUC and Technion)
- Decentralized and Distributed Systems (DEDIS) (Bryan Ford at EPFL, Swiss)
- Top conferences: IEEE S&P, ACM CSS, USENIX Security
- Related conferences: NDSS, NSDI, SOSP
- Specialized conferences: FC, Scaling Bitcoin, Breaking Bitcoin, Bpase

# Appendix: Min Language for Contracts

Min, a subproject of Harmony, is a new programming language for writing smart contracts in an easier and safer way. Our site min-lang.com has the full source code and description. Here’s a glimpse of its features and syntax.

Memory management costs enormous development effort or it dominates runtime cycles. Min's innovative type inference automates ownership annotations in a region-based memory model, so code remains at a high-level abstraction without the complexity of a garbage collector.

```
Term = Tag
 Nil
 Apply fun:Term arg:Term
 If test:Term pos:Term neg:Term
 Binary left:Term op:S right:Term

rewrite : Term? Term =
  Binary (Binary a '&' b) '|' c? If a b c
  Binary a '&' b? If a b Nil
  Binary a '.' b? Apply b a
  Binary a '@' b? Binary b ';' a
```

Min's precise types also guarantee correctness before execution and optimize for machine performance. Scaling to a million page requests or intensive scientific computations won't need delegating to foreign functions. With Min, you won't have to worry about null pointers in critical services, or stealing and tampering of your digital assets in smart contracts.

```
black_scholes
 s : ℝ # stock price
 x : ℝ # strike price
 t : ℝ # expiration time in years
 r : ℝ # risk-free interest rate
 σ : ℝ # volatility
 : ℝ
 = s φ(d_1) - x e^(-r t)φ(d_2) @
  φ = Normal.cdf
  d_0 = log s/x + (r + σ^2/2)t
  d_1 = d_0 / σ√t
  d_2 = d_1 - σ√t
```