GeekChain

极客链

开发设计“一链一应用”的技术架构
构建智能节点完备的下一代区块链应用生态

白皮书

V1.0

摘要

本文介绍了 GeekChain 基金会推动的“一链一应用”的区块链应用主链系统架构思想，全面解读了“生态级孵化器”的实现原理，以及 GeekChain 基金会构建的 GCT 生态的运作方式，并介绍了支持 GCT 生态运行的 GeekChain 的具体技术实现。
本文还披露了 GeekChain 基金会的关键构成信息和运作方式。

官网

www.gct.one

目录

1 区块链技术的现实背景 ……3

2 “一链一应用”系统思想概要……4

3 GeekChain 基金会的使命 ……5

4 GCT 生态搭建与经济循环……5

5 GeekChain 的技术实现 ……8

6 基于双链结构的共识算法……14

7 GeekChain 基金会的设定与运作规则 ……25

8 团队与路线图……27

1.区块链技术的现实背景

回顾区块链的发展历史，我们认为已经经过了两个阶段：

区块链 1.0:
核心标志是比特币的发明和被广泛承认。其证明了区块链技术核心理论的正确性和安全性，并且提供了第一个去中心、点对点交易的加密数字货币产品，这是革命性的。

区块链 2.0:
在比特币被发明以后，经常有人为了一些特定的场景，或为了验证一些新的区块链技术，在比特币的基础之上进行修改，进而发行一个新品类的数字货币。人们也渐渐意识到，比特币所发明的区块链技术，是一种新的无中心的信息存储与处理技术，可以渗透到更加广泛的应用场景。在众多尝试中，Vitalik Buterin 发明的以太坊提供了一个理想的“区块链计算机”模型，开发者可以通过编写智能合约的方式开发各种区块链应用（DApp），这极大地释放了区块链技术的能量。

虽然以太坊提供了完整的智能合约的运行平台，但其区块链核心技术还是从比特币里继承的 PoW 共识算法。随着 ETH 价格的上涨，目前基于以太坊的智能合约开发，有着合约运行效率低、合约执行费用高、复杂智能合约开发调试困难等实际问题。例如，曾经流行一时的区块链游戏“Cryptokitties”，就给整个以太坊网络带来了大规模的拥堵。

以太坊创造的智能合约，虽然极大地扩展了区块链技术的应用领域，但上述的种种问题，最终因高昂的成本和缓慢的性能让其在实际应用依旧停留在数字货币、数字证券（ICO）发行等传统金融领域，而其提出的基于智能合约开发 DApp 的愿望并没真正实现。

区块链 3.0 的标志是商业应用的落地
对于区块链的下一步应该如何发展，全球的技术精英提出了大量的新思路、新架构。但相信大家可以达成一个基本的共识，那就是区块链技术的再一次革新，一定会让海量的普通互联网用户，能在日常使用的产品里看到区块链技术带来的改进，并为之付费。

有不少人认为可以通过构建一个更好的“以太坊”来达到这个目的：使用新的共识算法，改进智能合约的脚本语言并构造更高效的执行虚拟机，设计并行能力更强的合约状态存储系统，试图构建一个在性能上全方面超越以太坊的新公链。社区已经按这个“以太坊 Plus”的思路推进了一段时间了，其中有一些项目已经进入了 beta 期，计划在 2018 年发布正式版。这些项目吸引了社区巨大的关注，并已经通过 ICO 的方式募集到了大量的资金。

2.“一链一应用”系统思想概要

我们不认为上述方案是未来的唯一选择，从目前的理论基础和硬件能力来看，要想在保持区块链的强一致的前提下，实现一个有足够性能、并且有足够多节点的安全网络是非常困难的。这在计算机发展史上也有类似的历史时期：UNIX 在 20 世纪 70 年代被发明以后，是非常完美的多用户多任务操作系统。但以当时的硬件水平，只有大型机才能运行起来，高昂的硬件成本导致其只能在金融业使用。但人们对计算机依旧有旺盛的需求(字处理、制表)，这导致了搭载技术上比 UNIX 简陋的多的单任务操作系统(比如 DOS)，能运行一些关键的 Killer App，价格更低的 IBM-PC 因此流行。 以史为鉴，我们认为基于智能合约的区块链平台虽然理论上很完美，但在现在的硬件和网络水平下是无法实现性能、成本、安全的平衡的。要推动 Killer App 在区块链上落地，需要让整个主链的设计（共识算法、块大小、支持的交易（TX）与保存哪些状态）根据应用的场景进行定制。我们把这个思路称作“**一链一应用**”。

“**一链一应用**”的设计让应用开发者能尽快地行动起来，而不是等待新的公链完成漫长的测试和验证。同时也能在实际的场景中去验证各种不同的区块链的新理论，让区块链技术在实际的应用场景中创造价值，进一步推动行业的发展。我们相信在现在这个时期，“**一链一应用**”的设计能比“以太坊 Plus”更快地帮助 DApp 落地。

3.GeekChain 基金会的使命

整合全球尖端资源，通过开发设计“**一链一应用**”的技术架构，构建智能节点完备的下一代区块链应用生态，以“**生态级孵化器**”为定位推动区块链商业应用的落地。

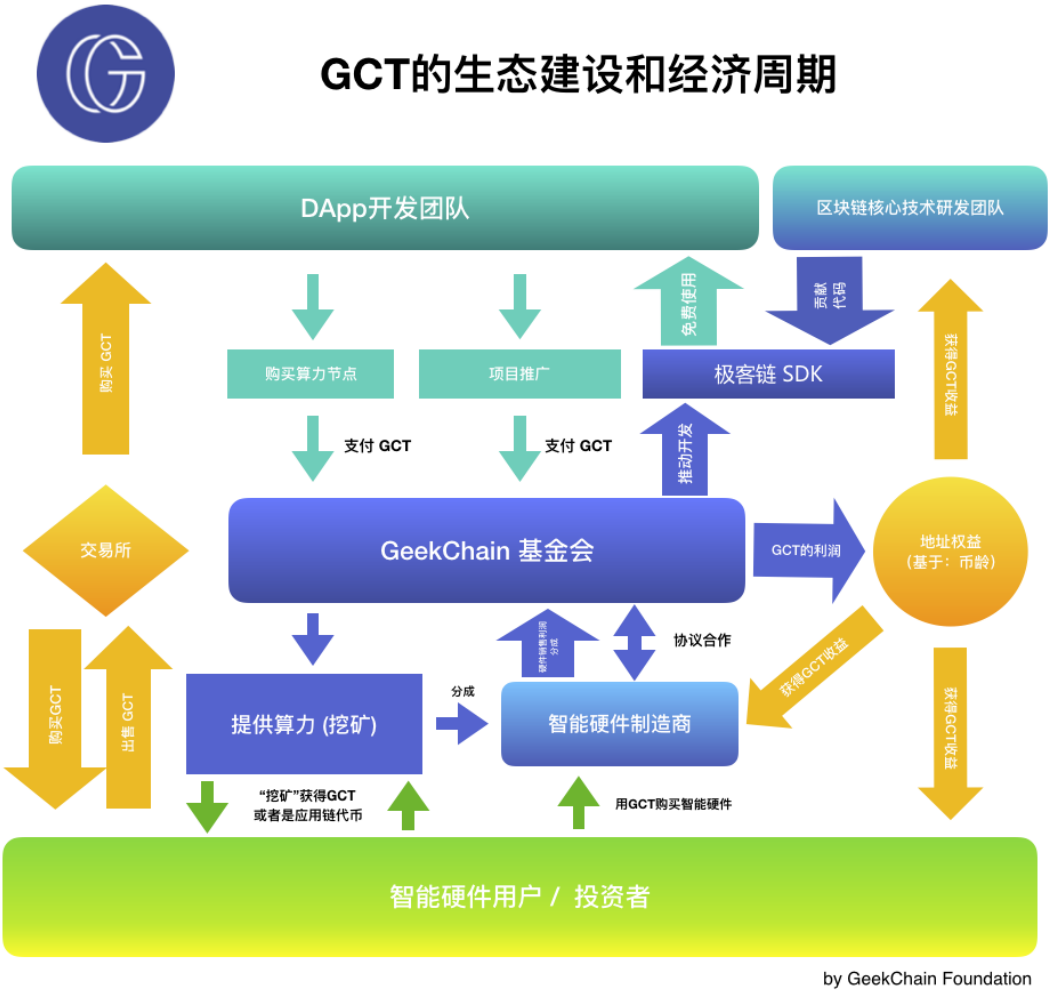
我们计划通过以下方式来达成使命：
推动主链 SDK 的发展，在 SDK 中集成区块链的新技术，降低应用主链的开发门槛。

1. 为应用主链的开发测试提供公网节点。
2. 为应用主链的冷启动提供足够的公网节点。
3. 通过构建涵盖智能硬件生产、区块链核心技术发展、区块链应用开发、区块链应用社区的生态，推动基于应用主链的区块链应用能够落地，让全球的互联网用户能感受到区块链技术带来的革新。

4.GCT 生态搭建与经济循环

GeekChain 基金会为搭建一个公平、开放的应用主链上线生态，以“**生态级孵化器**”为定位精心设计了一个经济体系。在这个体系中的每一方，都能在利益的驱动下推动生态发展，并共同分享应用生态的收益。我们为这个经济循环创建了 GeekChain Token （GCT）。

- **经济循环设计如下图所示：**



- **经济流程与角色利益:**

➢ **区块链应用开发团队**

根据自己的具体需求，基于 GeekChain 基金会提供的主链 SDK 可以又快又好地实现应用主链。代码完成编写后，应用开发团队可以支付一点点 GCT 来向 GeekChain 购买一些公网节点（可以指定节点的硬件条件和节点所在的物理区域）来进行链的公网测试。测试通过后，开发团队可以继续支付更多的 GCT 从 GeekChain 购买公网节点，组建应用主链的早期网络（一般应用主链应至少有 1000 个以上的节点较为安全）。此外，开发团队还可以通过支付 GCT 向整个生态的终端用户推广自己的项目。当项目的价值被大家认可后，会有很多人认为成为该应用的早期节点更具价值，从而会手动选择主动加入该应用主链。随着该应用的影响力增强，有更多的生态外节点也会加进来。待该区块链

完成了在 GeekChain 上的孵化过程，可以依靠自己的经济循环获得足够的节点，后续也不需要再使用 GCT 购买节点了。

➢ **硬件厂商**

根据自己对市场的判断，针对不同的国家/地区推出各种有不同侧重点的区块链硬件，销售给最终用户（矿工）。当应用主链开发团队向 GeekChain 购买节点时，GeekChain 会根据该应用主链设置的条件，选择最合适的节点。GeekChain 会使用一个公开的方法给 GeekChain 上的节点发放工资（挖矿收入），不同的硬件厂商可以按其和 GeekChain 事先约好的协议获得一定的工资分成。

➢ **普通用户（矿工）**

根据自己的实际情况购买支持 GeekChain 的智能硬件，接入网络后就成为了一个 GeekChain 待用节点开始挖矿。在 GeekChain 项目的早期，可以挖到 GeekChain 发放的最低 GCT 工资（这也是最容易挖到 GCT 的时期）。中期随着有应用主链团队加入生态，经济循环开始运作并开始产生利润。我们会根据各个智能设备保存在 GeekChain 上的工作行为，通过一个公开的算法计算出所有设备每周应该获得的工资收入并发放给矿工。同时，被分配给应用主链的节点还会挖到应用主链提供的 Coin/Token。这个时期矿工通过自己“在应用主链上的劳动”获得主要收入。到了 GeekChain 的繁荣期，有大量的应用主链在 GeekChain 生态上诞生/消失，GCT 的价格也趋于稳定并较难挖出。有经验的矿工可以手工选择想要加入的应用主链，靠选项目的眼光来获得更好的回报。

- **<u>地址资格</u>**

未来不同的主链对于节点的性能要求方向各不相同，有的需要节点有强大的算力，有的需要节点有海量的状态存储能力，有的需要节点能有高速的网络接入速度。但我们相信，共同点之一就是希望节点能够长期且稳定的工作。不同的主链结合自己的特点对稳定和长期的定义可能会各不相同，我们将会不断地收集整理提供多维度的数据来帮助不同的主链项目寻找最适合自己的节点进行部署。符合主链项目稳定性要求的节点绑定的 GCT 地址，拥有该主链项目的节点资格，这个机制我们称作“地址资格”。

在最开始，我们对节点稳定性的判断的主要方法是节点挖取 GCT 的速度。同时为了防止一些常见的作弊手段，我们也会要求能挖取 GCT 的硬件节点所绑定的地址上拥有一定币龄的 GCT 作为担保。
此外，项目方需要通过 GCT 生态推广自己的项目时，也会定义一些目标用户的画像，比如价值投资者(愿意进行中长线投资)。我们也会根据设备绑定的地址的挖矿收益情况，来进行合格用户筛选。

- **地址权益**

单个节点的价格在矿工与应用主链开发者之间是存在价格差异的，而且 GCT 生态的很多业务流程都有手续费的设计。这个利润空间是我们为了整个 GCT 生态的良性发展设计的。利润中的一部分会根据公开的协议分给 GeekChain 的早期的硬件合作伙伴，另一部分则会根据一个公开的地址权益算法，分给所有合格的 GCT 持有者。简单的说，我们整个系统的利润，会按社区持有 GCT 的比例进行分配，让生态中的每一个角色，都有机会从整个生态的成长和繁荣中获利。

- **节点价格**

由于“地址权益”的设计，有一定数量的 GCT 持有者都会倾向于持有 GCT，这也许会导致 GCT 的交易价格随着整个生态的繁荣而持续上涨。而对于开发者来说，如果购买节点的价格上涨过快，不利于应用开发者控制成本并制定合理的计划，最终影响到整体生态的稳步成长和繁荣。为了解决这个问题，我们生态中的节点价格是动态调整的。比如在早期一个节点的价格是 50GCT/每月，到了后期可能就变成 5GCT/每月了。此外，我们还提供了 GCC（GeekChain Coin）来稳定支付手续费。

5.GeekChain 的技术实现

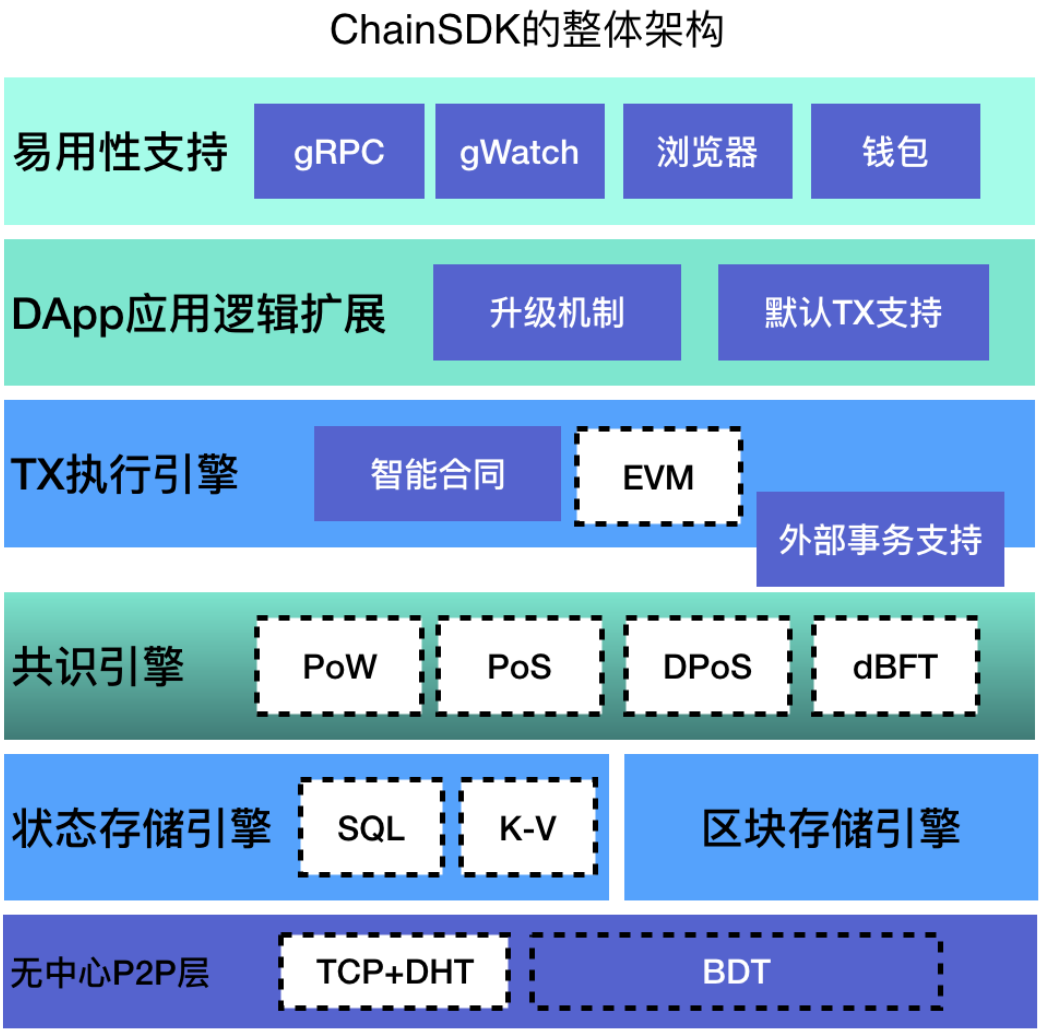
- **ChainSDK 介绍**

ChainSDK 是 GCT 生态中的重要组成部分，基金会鼓励所有的 DApp 开发者用开发自己的定制主链的方法来实现产品目标，ChainSDK 能较大提升这类应

用主链的开发效率。因此，ChainSDK 是基金会“**一链一应用**”理念得以落地的关键软件技术。ChainSDK 目前由 GCT 生态中的生态企业“巴克云”负责研发，在完成初步的研发工作后，ChainSDK 将由整个 GeekChain 社区共同维护，并在开放源代码的基础之上，推向整个区块链技术圈，为行业的发展添砖加瓦。

ChainSDK 通过下列关键设计来实现提高应用主链的灵活性和开发效率的目标。首先，通过分层架构允许 DApp 开发团队使用编写配置文件的方法来集成各种区块链技术以控制应用主链的核心技术指标。其次，ChainSDK 使用**智能合同**取代智能合约，让 DApp 开发者可以使用已经熟练掌握的成熟开发语言（JavaScript）来实现自己的业务逻辑。最后，ChainSDK 提供了从开发调试到测试网络到公网布署，进行线上问题跟踪和修改的完整的工具链，帮助开发者快速搭建开发环境。

➢ ChainSDK 的整体架构如下图所示：



ChainSDK 在架构上把一个主链分了七层。每一层都对上一层提供了标准化的逻辑接口来实现内部的透明切换。这个设计可以通过编写配置文件的方法就构造出一个应用主链，这个主链可以自由采用各种关键的区块链技术（比如选择什么共识算法）以及相关参数(比如区块大小)，同时 ChainSDK 开发团队也能透明地在架构中不断添加新的区块链技术，加快行业里新的研究成果的落地速度。

➢ **简单介绍一下这七层：**

最底层是网络层，无中心网络层实现的核心语义是在无中心网络中进行广播和接收广播；

存储层，区块存储层在文件系统中保存区块（包含分叉），状态存储引擎则为应用提供了状态的抽象和存储能力；

共识引擎提供的共识算法是整个系统运转的发动机；

TX 执行引擎实现了“逻辑业务”运行的容器支持，包括原子特性支持、安全性支持、以及外部访问支持；

DApp 应用逻辑层是留给 DApp 开发者扩展的一层，DApp 开发者编写的代码基本上要符合这一层的架构规范；

易用性支持，提供了传统系统友好的 Server To Server 的状态访问和写入接口，以及用户友好的钱包和区块浏览器支持，帮助 DApp 团队开发完整的系统。

ChainSDK 不但设计了一个新的架构来集成现有的区块链技术，有效提高了 DApp 团队的工程能力，还在这个新的架构内进行了一些区块链技术创新。下面是这些技术创新的简单介绍：

- <u>BDT 协议</u>

BDT 协议是由 GeekChain 基金会支持并计划使用的，由巴克云开发的下一代无中心 P2P 协议。设计目标是通过总结团队过去多年的 P2P 协议开发经验，提供更适应中国网络环境和区块链场景的无中心 P2P 技术；同时，实现更好的连通率、更快的连接速度、更高的点到点传输速度，并用更全面的加密结构让网络有更好的匿名性；此外还希望能减少在网络中进行全局广播的带宽消耗，从而提高 DHT 网络的拓扑稳定性和多级广播的速度。

- **智能合同**

智能合同指的是区块链上一个 TX 的执行和校验过程。按这个定义，目前主流公链里的“虚拟机+脚本”的智能合约也是**智能合同**的一种实现。这种允许在公链运行过程中随时可以扩展新合约的设计，虽然非常的灵活，但也带来了大量的安全隐患。此外，为了 GAS 费用的计算，虚拟机还需定制，这无法利用 PL 领域多年来的积累，进而降低了系统的性能。而**智能合同**是直接在区块链的执行层植入 DApp 的逻辑代码，能使用区块链引擎的开发语言的全部能力，并且不需要设计 GAS 机制，从而提高了灵活性和运行性能。
ChainSDK 在**智能合同**上还提供了名字空间的设计，允许在一个主链的运行过程中，创建一个“接口相同，但拥有全新状态”的独立空间。这就用更简单的方法实现了以太坊通过智能合约解决的扩展性需求。比如可以创建一个新的名字空间，只包含转账相关的 TX，这相当于在以太坊上通过 ERC20 合约创建了一个新的 Token。但我们的架构并不允许用户自己实现这些转账操作，从而提高了安全性。

- ChainRuntime

按 GCT 的生态构想，在一个生态设备上会运行多个基于 ChainSDK 开发的应用主链矿工。ChainRuntime 包含了 ChainSDK 的大量实现代码，降低了 Miner 的部署成本和在不同体系架构的硬件之间进行移植的成本。
ChainRuntime 还为这些 Miner 运行提供了一些基础性的通用功能（比如公用设备地址和私钥），从而降低了 Miner 初始化的交互复杂度。最后，ChainRuntime 还提供了沙盒隔离机制，进一步地保障了生态设备的安全，防止恶意 Miner 读取其它 Miner 的敏感信息。

- **跨链操作**

得益于 ChainSDK 的架构设计，事实上让基于 ChainSDK 开发的应用主链是同构的。这些主链的矿工可以在同一个设备上互相识别，进行安全的互相操作，也可以使用相同的网络协议和共同的 TX 格式互相读取/访问。这为实现跨链操作提供了技术基础。

ChainSDK 在 TX 执行引擎层添加了外部事务支持，可以方便 DApp 团队创建包含访问其它区块链网络，以及现有系统的链外事务。通过链外事务的支持，ChainSDK 可以极大地扩展区跨链应用系统的功能边界。

- **混合治理**

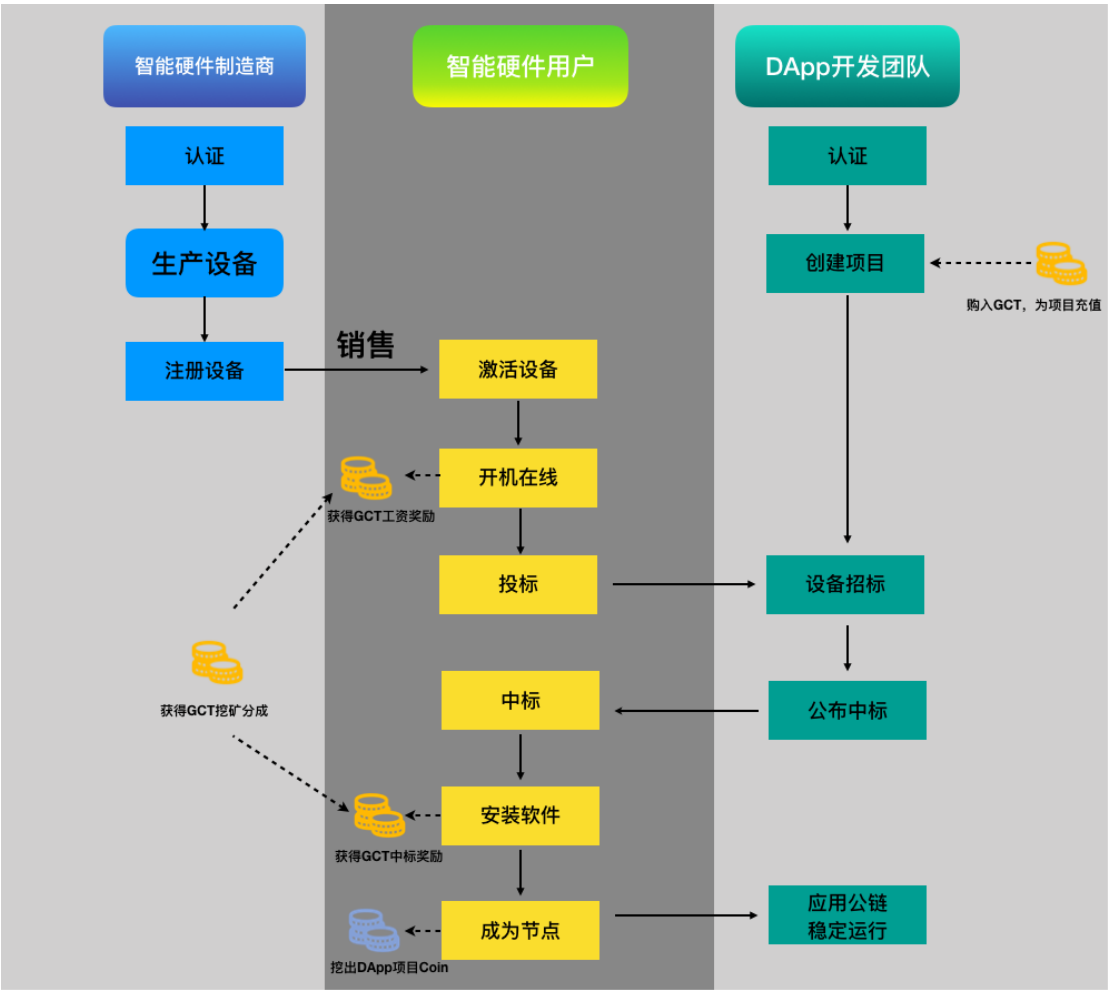
以太坊上智能合约的开发体验某种意义上可以用“一将功成万骨枯”来形容。应用的状态被设定为与合约地址绑定，合约一旦发布就不可以修改，这相当于要求应用开发团队“一次性实现没有 BUG”的系统。虽然这个机制极大地维护了区块链“不可篡改”的特性，但我们认为从软件工程的历史来看这个要求是99.9%的工程团队所无法达到的。ChainSDK 允许主链的开发团队可以在不修改应用状态数据的情况下，进行**智能合同**的更新。

这其实赋予主链开发团队更大的权利，可能会在一定程度上削弱了主链的去中心化特性。但我们认为完全去中心化（PoW 在矿池面前去中心化也正被极大挑战）的治理一定是理想且低效的，区块链项目使用混合治理的模型可能是这个历史阶段的必经之路。我们提供给应用团队更大的权利，是因为相信在经济利益的驱动下，应用团队会有更大的动力来推进系统的完善和演进。

由于篇幅的限制，在白皮书里只针对 ChainSDK 的关键技术特性进行介绍。ChainSDK 目前正在公测，可以通过关注 ChainSDK 官网（https://www.chainsdk.io/）来了解更详细的设计和进展情况。

- GeekChain 介绍

GeekChain 是 GCT 生态构想的区块链实现。GeekChain 不但实现了 GCT 生态的各个角色间进行数字货币的交易功能，还基于区块链技术实现了 GCT 生态的关键业务流程，这让 GCT 的生态演进拥有了超越传统公司组织的社区推动力。基金会坚持”Code is law” 的理念，尽可能地让整个生态里的流程在 GeekChain 上运行，减少生态中人为介入的环节。任何人和组织只要认可我们生态的规则，都可以不受地域和时间的限制，平等自由地加入进来。
在 GeekChain 上运行的主要逻辑如下图所示:



- <u>GeekChain 实现</u>

在 GeekChain 主链上线后，所有 GCT 生态下的硬件节点都会成为 GeekChain 的主链节点。不过由于这些节点的主要目的还是给其它的应用主链提供资源，所以 GeekChain 的节点软件主要设计目标如下：

1）这是一条低成本的独立主链，用于记录整个 GCT 生态的经济活动。

2）尽可能少地占用硬件节点的资源。

3）对节点的工作情况进行监控。

4）对节点进行一定的控制，包括安装项目方的节点软件。

5）对节点的安全情况进行管控，防止项目方的代码对设备造成损坏。

GeekChain 基金会坚持贯彻“吃自己的狗食”的技术原则，会主动在核心的生态场景中对大量处于研发和验证阶段的基础技术进行验证，并在生态演进的

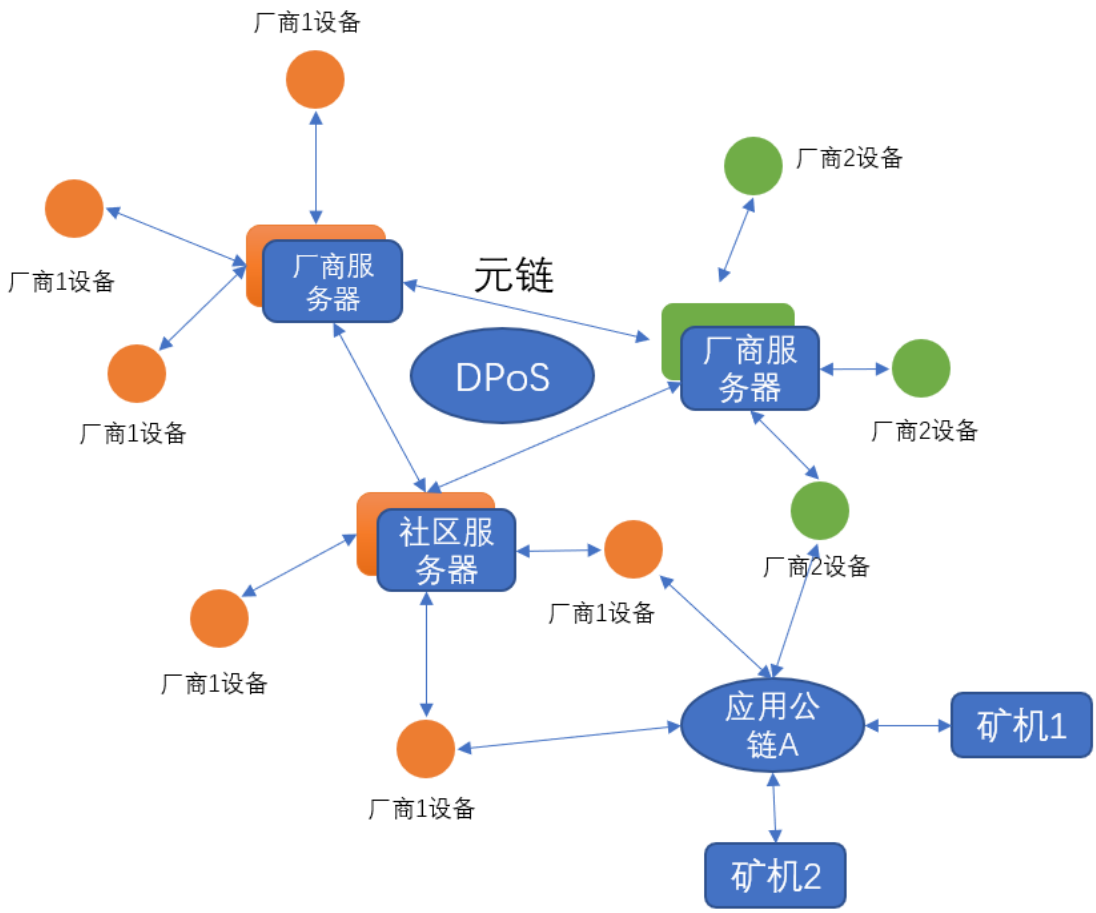
过程中进行持续的改进。GeekChain 基于 GCT 生态中的 ChainSDK 实现，使用基于**双链**的共识算法。

6.基于双链的共识算法

- **双链架构**

根据 GeekChain 的实际应用场景和 GCT 希望适配更多低配置硬件的情况，我们选择了 ChainSDK 提供的一种在 DPoS 共识算法的思想内核的基础上进行改进的新共识算法，我们称做“**双链**”(Double Chain)共识算法。

➢ **整体架构如下图所示：**



- **元链**

由 GCT 生态中的设备厂商、社区领袖、相关生态企业提供的服务器组成。我们通过社区基于 GCT 投票的方法海选出 2n+1 个元节点，并把节点的地址信息

写入元链的创世块中。元链使用 dBFT/DPoS 共识算法进行出块(具体使用哪种共识算法主要看元链上的节点数量，在项目早期我们使用 dBFT，后期可切换到 DPoS)。由于 dBFT 和 DPoS 共识算法已经广为人知，我们这里就不再对这两个共识算法做进一步的介绍了。

元链的区块中保留以下 TX:
a. 设备分组 TX
b. 设备的工作汇报 TX
c. 身份认证 TX

身份认证 TX 是元链的持续运作的关键。有 n+1 个元节点签名的身份认证信息会上链，通过这个机制可以投票批准新的元节点加入元链，或投票淘汰不再参与的厂商以及不正常工作的元节点。
元节点工作在 IDC 机房中的 x86 服务器上，这些服务器拥有良好的计算性能与磁盘性能，并通过稳定工作的多路专线接入互联网。这样的硬件环境让元链在常规情况下拥有稳定优秀的出块性能，并可以根据需要采用较大的块大小。

- <u>GeekChain 与元链</u>

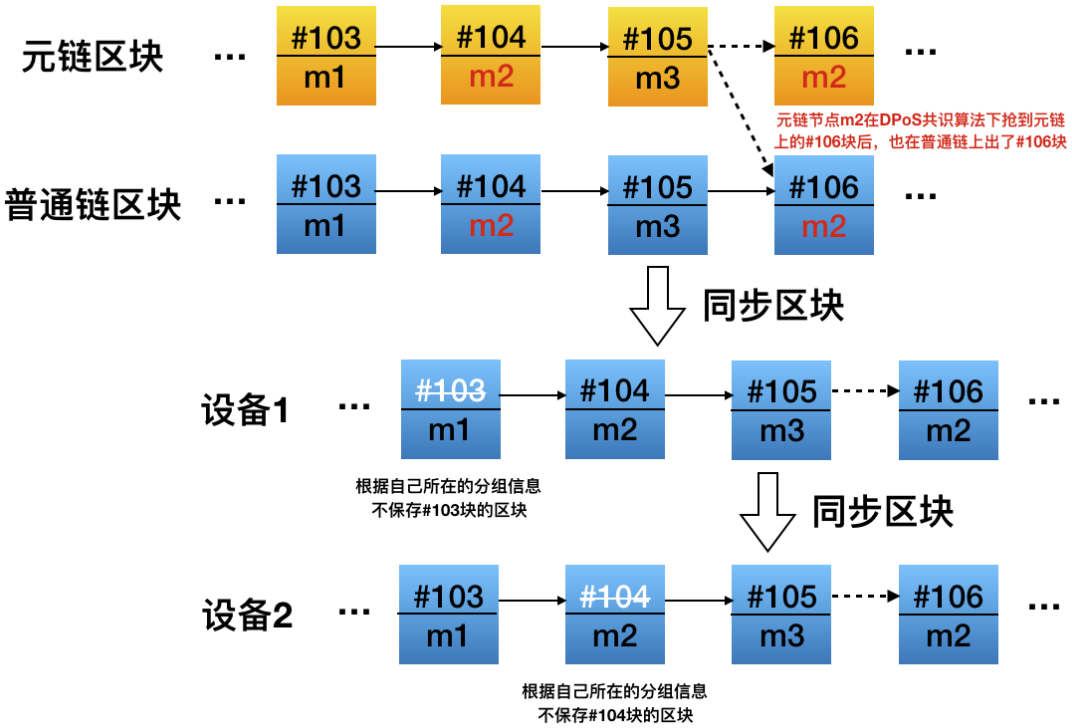
元链上的所有节点也同时属于 GeekChain。在 GeekChain 上，还有大量的不同厂商生产的各种型号的设备节点。普通链上的节点在运行中会不断读取元链上的信息来高效地工作。包括:
1.根据元链的出块信息，确定下一个块由哪个节点出（GeekChain 的块也是由元节点出的）
2.通过读取元链信息，确定当前节点所在的分组，进而确定需要保存的区块数据，完成数据分片。由于 GeekChain 需要记录设备参与项目的工作信息，我们预计 GeekChain 上会有很大的数据量，所以使用存储分片技术是 GeekChain 必然的选择。
3.读取元链的合法厂商信息，确定其它设备上报的数据信息是否合法。
4.往元链上报 GeekChain 设备的工作信息。

通过上述设计可知，GeekChain 上的主要 TX 只包括 GCT 转账信息和 GCT 生态项目的相关逻辑。共识算法逻辑，GCT 生态设备/数据的上报和合法性判读逻辑都上移到了元链，进而提高了 GeekChain 出块稳定性，加快了出块的

速度，降低了一般设备成为 GeekChain 节点的性能要求，并实现了 GeekChain 上的数据分片，降低了一般设备成为 GeekChain 节点的存储容量和 IO 性能要求。

- **共识机制流程**

完成上面的概念和理念介绍后，下面完整的描述一下 GeekChain 的共识运作流程。



1. 元节点通过 dBFT 或 DPoS 共识算法出块。
2. 运行在元节点所在服务器的 GeekChain 节点在元链出块后，跟随出块。
3. 普通节点读取元链上的分组信息，确定自己所在的分组。第一次进入网络的设备还需要在元链上进行节点注册。
4. 设备上的普通节点根据自己的分组信息选择一个元节点保持连接，用来更新区块和投递 TX。这种设计能提高 TX 的确认速度，并减少在普通设备的窄带网络里进行 TX 广播对网络造成的额外带宽消耗。
5. 设备可以根据自己的分组信息来删除不属于自己组的 GeekChain 链区块。
6. 设备把运行日志通过节点工作汇报 TX 投递到元链，用来获得工资收入(GCT)。
7. 设备之间互相发送普通 TX 来调用功能。

8. GeekChain 浏览器默认不展示的元链上的区块信息。
9. 元链定期根据已登记的 GeekChain 设备节点信息进行创建设备分组 TX。

- **挖矿与手续费**

事实上，在 GeekChain 采用的**双链**机制的控制之下，只有元节点才有机会出块，这让传统的基于出块的挖矿激励无法实施。我们使用类似“最低工资”的设计，在 GCT 生态繁荣起来以前，让早期加入 GeekChain 的设备厂商和购买设备的用户，能获得 GCT 收入。(“最低工资”的发放计算与发放方法后面会详细介绍) GeekChain 基金会发行的所有 Token 中，有很大一部分预留给了发放“最低工资”。这个阶段是 GCT 生态的孵化期。随着加入 GeekChain 的节点越来越多，“最低工资”池里剩余的 GCT 总量越来越少，后面一个设备能获得的“最低工资”收入也会越来越少。从结果上来看，随着时间的推移，GCT 越难挖出。

而且按 GeekChain 的生态构想，GCT 的持有者共同组成了 GeekChain 社区，共同推动 GeekChain 的发展的同时，也会分享 GCT 生态运转过程中产生的利润。所以随着 GCT 生态的扩大，GCT 的价格也会随之上涨。而在 GeekChain 上进行 DApp 开发的周期通常都不会太短，我们希望生态各方使用 GCT 进行商业活动时能对成本进行一个相对准确的预估。换句话说，系统里的流程性的手续费，以及一些必要的支出成本，更倾向于按法币定价。
因此，GeekChain 里除了 GCT(GeekChain Token)以外，还有一种内置的货币 GCC(GeekChain Coin)，价值恒定的与 1 美元相等。元链会根据一套 Oracle 机制，在元链上设置 GCT 与 GCC 的兑换比例，并允许用户在自己的地址上进行 GCT 到 GCC 的单向兑换。但与 GCT 不同的是，GCC 并不能转移到别的地址上。也就是说，GCT 转化为 GCC 后，只能用来在 GeekChain 生态中的支付场景里使用。如果用户自己不进行兑换关系的操作，系统会在需要缴纳 GCC 做手续费的 TX 执行时，自动进行兑换。

- **GeekChain 的主要链上业务逻辑(TX)**

GeekChain 在设计上只为了实现 GeekChain 基金会的生态构想。所以在技术上我们并没有通过传统的“虚拟机+智能合约”的架构来完成 GeekChain 基金会的业务需求，而是通过 ChainSDK 倡导的**智能合同**思路在 GeekChain

的 TX 执行层完成了这些业务流程的实现。从这个意思上来说，GeekChain 是一条遵循“**一链一应用**”的 DApp 主链。这个选择再配合之前描述的**双链**的共识算法，极大地降低了运行 GeekChain 全节点的硬件配置，让更多的智能设备有机会成为 GeekChain 全节点。同时，这个设计也让 GeekChain 的整体代码更加简洁（不包括 ChainSDK 代码规模在 3 万行以内），更少的代码量也从概率上减少了逻辑漏洞，使整个系统更加安全。

- **GeekChain 实现的核心智能合同的主要逻辑(注：相关经济参数可能会在主网上线前多次修改)**

➢ **厂商认证**

FIRM_REGISTER (addr_list， desc， deposit)

设备厂商通过发起 TX “FIRM_REGISTER”来成为一个 GCT 生态里的被认证厂商。发起这个交易要求转入一定数额的 GCT 保证金（最小金额由当前元链节点公布）。当该设备厂商由于违规被注销时，这些保证金会被没收，回到 GCT 可挖池。

Desc 里包含一段结构化的信息来介绍厂商。在 GCT 生态的早期，这里还可以包含该厂商与 GeekChain 基金会签订的合作合同。

GeekChain 也可以设置为厂商注册需要 2 次确认，那么只有元节点投票通过后该厂商才能成为一个认证过的厂商。

厂商还需交纳金额为保证金 10%的 GCT 手续费。手续费会进入 GCT 的当期利润池。

➢ **注册设备**

DEVICE_REGISTER(addr_list， desc ，option)

认证厂商可以发起这个 TX 在 GeekChain 上注册一个设备。

desc 里是一个结构化的描述信息(JSON 格式)，用来描述设备的基础能力。

option 里可以指定设备的厂商工资分成比例。

一个厂商允许注册的总设备数量与该厂商缴纳的保证金数额有关。

该操作需要交纳固定 0.1 GCC 的手续费。手续费会进入 GCT 的当期利润池。

➢ **设备节点基础上报**

DEVICE_REPORT(action， info)

已经在 GeekChain 里注册过的设备，发起这个 TX 来上报动作或状态。为了让 GeekChain 的主要容量和性能为 DApp 项目方服务，设备的基础上报的频率会尽量降低，内容会尽量精简。
action: 本次上报的动作。一般有设备启动、设备正常运行、设备关闭、设备连接网络等。
info:动作的附加信息，比如设备正常运行动作包括下列信息： 网络、支持的计算类型、可用内存大小、可用磁盘大小、磁盘性能。
状态上报关系到设备的保底 GCT 工资发放，为了防止作弊，GeekChain 基金会要求厂商必须在其生产的设备上使用反作弊技术。设备作弊除了会被取消资格外，还有可能会罚没一部分厂商的保证金。
该操作投递到元链上，不需要手续费。

➢ **DApp 开发团队注册**

DEV_REGISTER(addr_list， name， desc， deposit)
DApp 团队通过发起 TX “DEV_REGISTER”来成为一个 GCT 生态里的被认证 DApp 团队。发起这个交易要求转入一定数额的 GCT 保证金（最小金额由当前元链节点公布）。当该 DApp 团队由于违规被注销时，这些保证金会被没收回到可挖池。
GeekChain 也可以设置为 DApp 团队注册需要 2 次确认，那么只有元节点投票通过后该 DApp 团队才能成为一个认证过的团队。
DApp 团队还需交纳金额为保证金 1%的 GCT 手续费，手续费会进入 GCT 的当期利润池。

➢ **DApp 项目创建**

PROJECT_CREATE(name， desc， value)
认证 DApp 团队通过发起 TX “PROJECT_CREATE”来创建一个 DApp 项目。创建成功后该 DApp 团队成为项目的 Owner，GeekChain 会返回一个唯一的 DApp 项目地址。
name: 项目的名称，仅仅用于显示。
desc: 项目的详细描述信息，仅仅用于显示。
value: 如项目创建成功，则转入这么多 GCT 作为项目的余额。
该操作固定收取 10 个 GCC 的手续费。

➢ **DApp 项目余额**

使用 DApp 项目的地址，可以在 GeekChain 浏览器上查询到一个 DApp 项目的全部信息。其中最重要的信息就是项目的余额。下面的很多操作，特别是持续的租用设备的操作，都依赖项目地址上的 GCT 余额。
任何人都可以用向项目地址发起普通转账的方式，给项目充值。
该操作的手续费与普通转账一致。

➢ **结束 DApp 项目**

PRJECT_END(addr， end_time)
当前项目的 Owner 通过发起 TX “PROJECT_END”来结束一个项目。如果 end_time 未传入则立刻结束该项目，否则到指定时间才会让项目结束。
项目结束后如还存有余额，会按给项目充值记录的倒序进行顺次返还。
该操作不收手续费。

➢ **转让 DApp 项目**

PROJECT_TRANS(addr， new_owner， value)
当前项目的 Owner 通过发起 TX “PROJECT_TRANS”可以改变项目的 Owner。new_owner 的地址上必须有项目当前余额+value 数量的 GCT 才能交易成功。交易完成后，value 数量的 GCT 会从 new_owner 的地址转入原 owner 的地址。
该操作收取 10 GCC + value*0.1% GCT 的手续费。

➢ **设备招标**

PROJECT_CREATE_TENDER(project_addr， tender_sname，device_min_num， device_max_num， min_task_time，device_price， device_ability， start_time， end_time， deposit ，desc)
由项目的 owner 通过发起 TX “PROJECT_CREATE_TENDER”为项目进行一次设备节点招标。招标内容的核心内容有本次招标的开始时间与结束时间（结束时间可不设置），招标要求参与的最少节点数，节点的最低硬件条件，项目方愿意支付的最高节点价格(单位为 GCC/周)。该 TX 执行成功后，会返回本次设备招标的唯一地址。
招标设定的设备最小能力要求 device_ability 除了设定基本的硬件配置要求外，还可以设定硬件的平均稳定工作时长，硬件地址的 GCT 币龄等条件。

招标还需要项目的 owner 支付一笔保证金，如果本次招标按非标准流程结束，保证金会分给本标的各参与方。GeekChain 上保存了元节点设置的最低投标保证金。
该操作固定收取 10 GCC + 保证金*1% GCT 的手续费。

➢ **设备投标**

DEVCIE_BID(tender_addr， device_price， min_task_time， deposit)
GCT 生态设备通过发起 TX “DEVCIE_BID ”来报名参加一次 DApp 招标活动。报名的时候需要说明自己愿意以什么样的价格参加投标，以及愿意在项目中工作的最短时间。
报名投标还需要设备支付一定的保证金，保证金的最小值为 device_price。
如果设备投标成功后未参与项目，保证金会被扣除分配给相关损失方。
更低的设备价格、更长的工作时间以及更多的保证金，都有助于设备中标。
该操作收取固定 0.1 个 GCC 的手续费。

➢ **公布中标节点**

TENDER_ENABLE_DEVICE(tender_addr， device_addr_list)
项目的 Owner 通过发起 TX “TENDER_ENABLE_DEVICE”来公布中标的设备。device_addr_list 里的设备地址从当前已投标的设备中选出。
只要项目的招标没有结束，项目的 Owner 就可以随时发起这个操作让设备加入。该操作固定收取 1 个 GCC 的手续费。

➢ **项目启动**

PROJECT_START(tender_addr)
当中标的设备数满足了项目方发布的标书需求后，项目的 Owner 就可以通过发起 TX “PROJECT_START”来启动一个项目的标段了。项目的标段启动后，每周项目会从项目的余额账户中自动按中标时约定的 device_price 给正常工作的设备节点发放 GCT 工资了。
该操作固定收取 1 个 GCC 的手续费。

➢ **设备主动退出项目**

DEVICE_EXIT(tender_addr)
设备通过发起 TX “DEVICE_EXIT ”来退出一个项目。如果当时项目的状态处于 STOP 状态，那么本次退出会返还设备投标时缴纳的保证金。

该操作收取固定 0.1 个 GCC 的手续费。

➢ **设备在项目中正常工作**

当设备通过读取 GeekChain 上的公开信息得到自己已经中标的信息后，就需要通过各种方法安装项目方的挖矿软件，尽快开始在项目中正常工作。安装项目方的挖矿软件是一个链下的过程，可以通过设备厂商的固件自动操作，也有可能是设备拥有者自己手动完成，GeekChain 不对这个过程进行强制约定。

确定设备是否在项目中正常工作需要综合三个方面的定期上报信息

a. 设备固件发起的 TX “DEVCIE_PROJECT_REPORT”（投递到元链上），用来汇报从设备固件视角观察到的设备工作状态。该操作使用设备的地址签名。

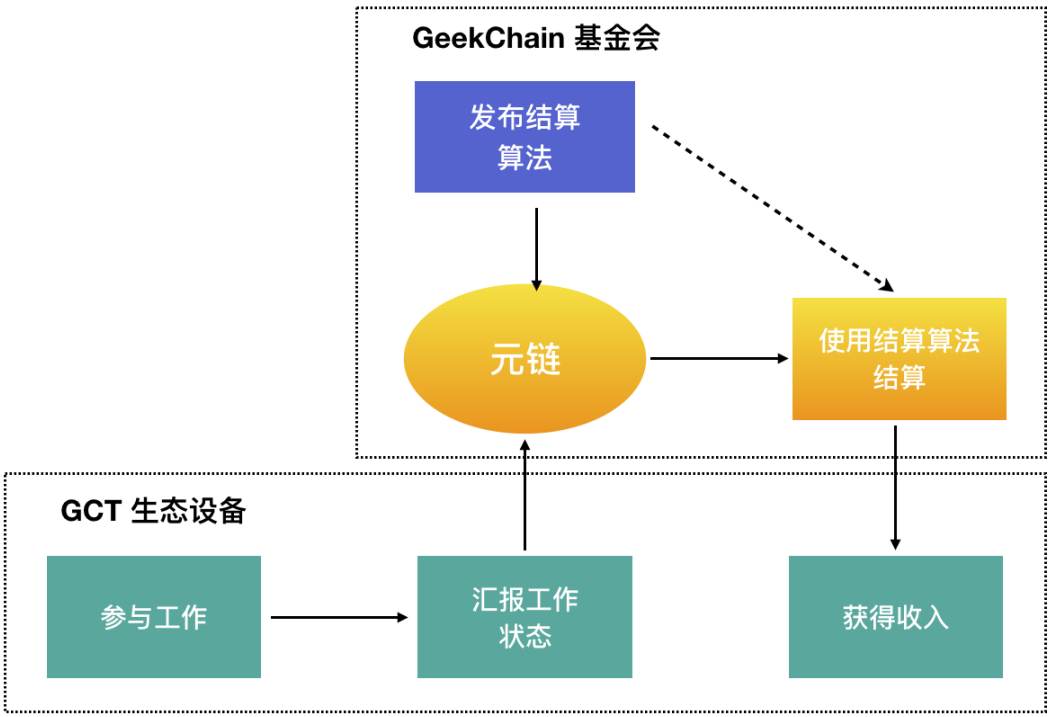
b. 项目方软件发起的 TX “SOFT_PROJECT_REPORT”（投递到元链上），用来汇报从 DApp Miner 视角观察到的设备工作状态。该操作使用设备的地址与 DApp Miner 地址双重签名。

c. 项目方后台发起的 TX “CHAIN_PROJECT_REPORT”（投递到元链上），用来汇报从 DApp Owner 视角观察到的设备工作状态。该操作使用项目方地址签名。

项目方可以和基金会一起，通过分析上述 3 种定期上报的信息，以公开的标准来判定设备是否在项目中正常工作。项目会自动从自己的余额里给正常工作的节点按约定价格发放工资。

- **<u>计算与记账分离</u>**

通过上面的核心流程分析可知，在 GeekChain 上的设备并不通过共识机制来获得挖矿收入。在 GeekChain 上获得 GCT 收入的过程可以抽象为下面流程：



我们把这个过程称作“**计算与记账分离**”。相比通过共识算法获得挖矿收入，其过程无法做到 100%的可靠，有一定的作弊空间。另一方面，这个获得收入的方法也有一定的后滞性，不够及时。但是，让矿工获得收入的方法与共识算法解绑，极大地扩展了区块链在共享计算方面适用的领域（按当下设计，项目方软件甚至可以不是一个区块链矿机软件）。此外，后置的分配算法，也解决了传统公链核心经济参数无法调整的问题。

- **如何解决“计算与记账分离”思路的缺点是 GeekChain 的关键技术创新，总结一下我们使用的技术：**

a. 公开与公平保证。

设备的工作情况通过大量的多角度信息上报保存在链上。再使用公开的算法进行分析，得出节点获得的工资收入。输入公开、算法公开，任何人都可以重复验算，保证了公开与公平。

b. 低成本记录大量日志。

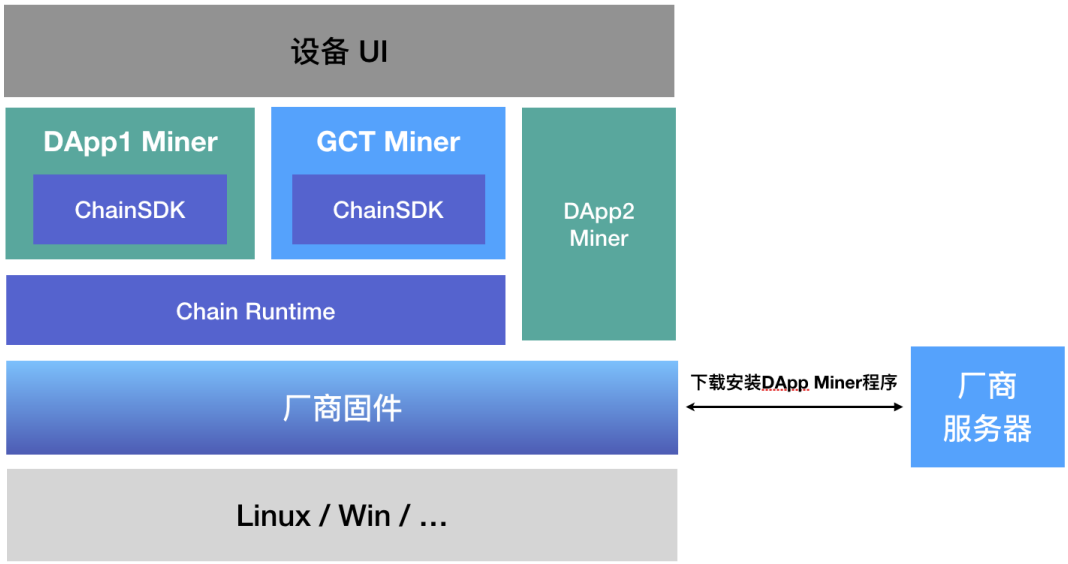
使用**双链**的架构，所有的工作日志都记录在不需要手续费的元链上。元链运行在服务器上有巨大的存储空间，并且可以删除旧的（1 年前）的日志，GeekChain 上只有节点的工资发放记录，不会对 GeekChain 带来容量压力和 TPS 损耗。

c. 防作弊。

由于在元链上保存了大量的多角度原始信息，所以可以不断升级反作弊算法来发现作弊信息。在 GeekChain 的很多环节引入的保证金制度，也大大增加了作弊的成本。在一个可以进行后置分析的系统中进行作弊的风险大、回报小，从而降低了各方作弊的意愿。

- **厂商固件**

一个典型的 GCT 生态设备上的软件技术栈如下图所示：



a. 在生态设备上一定会运行一个 GCT Miner。这个程序可以通过厂商固件的内置自动化逻辑参与 GCT 生态项目的设备招标工作，也可以通过设备的控制 UI，由用户手工参与 GCT 生态项目的招标。

b. 一个生态设备上只要性能许可，可以运行多个 DApp 的 Miner。其中有一部分 DApp 的 Miner 程序，可以不基于 GCT 提供的 ChainSDK 来构建。

c. 为了安全和体验，厂商固件可以有下载、安装、升级已知 DApp Miner 的能力。不过 DApp Miner 程序在设备上的移植工作，需要 DApp 团队与设备厂商协商完成。很明显，使用 ChainSDK 来开发 DApp Miner 程序，能大幅度地降低在不同的 GCT 生态设备之间移植 Miner 程序的成本。

- **其它功能**

得益于 GeekChain 基于的 ChainSDK 的架构支持，GeekChian 天然具有良好的跨链能力。GeekChain 在这种能力上实现了可靠的代币互换，以便用户按一个公认的汇率，用跨链技术在各种主流的 Token 中安全地互相兑换。

7.GeekChain 基金会的设定与运作规则

- GeekChain 基金会的组织架构原则

GeekChain 基金会是一个独立的法律实体，拥有独立的营运方式和管理架构，由顶尖专业人士提供咨询，实施各种措施以应对相关法规的变更和要求。基金会单独运作以开发 GeekChain 主链和相关系统，与各生态企业，投资机构的法律实体分开。从而满足 GCT 持有人的利益，开发稳固可扩展的系统，也便于他们在公平的商业协议下与各生态企业合作。基金会管理代币的发行和分配。

- GeekChain 基金会的管理目标

基金会计划投入资源建立一个公平透明的管理流程，并将生态系统内所有参与者的意见和需求纳入考虑。这种开放的管理模式将能监督参与规则、代币发行、市场推广、利润分红、法律事项，以及内容与合规准则等相关决策。GeekChain 基金会将负责管理和监督 GCT 保存的安全性，以及披露基金会持有的 GCT 的分配和使用情况，还会支持和推进与 GeekChain 区块链相关的研发工作，处理与生态系统成员相关的所有事务，包括合作协议签订与合同的执行跟进。

- GeekChain 代币发行

GeekChain 基金会计划发行 10 亿 GCT，GCT 的发售时间和对象的相关详情详见下面章节。GeekChain 基金会承诺 GCT 永不增发。

我们后续会在 www.gct.one 上持续更新 GCT 的分配情况，欢迎有兴趣的参与者持续关注。

- **GCT 代币发售收益的使用**

初期 GCT 代币发售所获得的资金主要用于以下用途：

1）完成 GeekChain 的主链开发；

2）帮助早期生态企业完成相关软硬件的开发与销售；

3）GeekChain 基金会的日常运营费用；

4）GeekChain 生态的市场推广费用。

- **GCT 的发行与分配**

GCT 发行总量为 10 亿，具体比例见下表：

	占比	锁仓时间	说明
基石	10%	18 个月	GCT 登陆交易所后允许 1 千万进入流通。
私募	21.785%	不适用	GCT 登陆交易所后释放 16.785%，这部分是市场流动性的主要来源。剩下的 5%锁仓 6 个月。
生态企业	10%	18 个月	分配给智能硬件企业(极路由等)。具体份额按合作协议和合同交付情况而定。
	10%	18 个月	分配给平台软件企业 (巴克云等)。具体份额按合作协议和合同交付情况而定。
	10%	N/A	基金会为 DApp 生态企业合作预留
挖矿	28.215%	只能挖出	基金会持有，通过 GeekChain 的设备在线机制挖出。已预分配 1500 万 GCT 给极路由挖矿。
社区推广	10%	不适用	基金会持有，用于社区推广。已预支 5 千万 GCT 用于早期推广。

如上表所示，GeekChain 基金会在 GCT 分配上按“433”的理念进行分配。即 40%分配给社区，所有人都可以通过参与 GCT 挖矿或参加 GCT 社区活动的方法来获得这部分的代币。30%分配给早期投资人，这是 GCT 基金会早期运营资金的来源。30%分配给早期的生态企业，支持他们完成 GCT 生态构建的启动工作。基金会通过”433”的比例架构，平衡好投资者，社区参与者，生态企业之间的利益，共同推进 GCT 生态的繁荣与发展。

GeekChain 基金会的早期投资者都认同 GCT 生态的长期建设理念，主动加入 18 个月的代币锁仓计划。18 个月代币锁仓要求就加入计划的地址上持有的 GCT 在 8 个月内不允许转出。从持有的第 8 个月以后，每月可以解锁总量的 7.5%。在 18 个月后，地址上还保留的 25% GCT 会自动进入下一轮的锁仓计划。截止到白皮书公布后 1 个季度，市场上流通的 GCT 不超过 2.5 亿个。

➢ **下表是流通 GCT 的地址分布说明：**

0xf108857ddef66824c1c689b8c03bd4ec9d044b42	15，000，000	极路由挖矿预支
0xc5db52998df44f885dc0c2e30576ea8f2009ee57	10，000，000	基石（解冻部分）
0x3e97f00c7b4ed9ecfd237a33eea5b81ce015d174	50，000，000	社区推广
0xddea10837773a0efa3ffadc1cf388c475a003baf	25，000，000	私募赠送
0xddea10837773a0efa3ffadc1cf388c475a003baf	50，000，000	私募
0xddea10837773a0efa3ffadc1cf388c475a003baf	142，857，142	私募

- **代币互换**

在 GeekChain 主网上线以前，我们在以太坊上使用 ERC20 Token 的方式来发售 GCT，以完成 GeekChain 基金会的早期需要。GCT 的 ERC20 合约地址为 0x8d6df35bd383033ad6e4a050f8eabf3677b540f6。待 GeekChain 主网上线后，我们会在以太坊上上线兑换的智能合约，方便用户自助完成代币互换。

8.团队与路线图

- **基金会**

➢ **基金会主席：Adam Loveall**

Adam 曾为多家美国跨国公司负责解决方案架构，软件架构和基于云计算（AWS 和 Bluemix）大型企业系统的设计。在主持运营 GeekChain 基金会之

前，他曾担任顶级的信息安全工程师，并在密码学、PKI、加密和密钥管理工具方面拥有丰富的经验。

➢ **基金会全球战略官：David Zhang**

David 拥有美国马里兰大学派克分校，计算机硕士学位。曾在 Planned Systems International 担任高级软体工程师和首席系统架构师，负责与多家全球领先的金融机构合作开发基于区块链体系的数位身份识别软体。对金融行业的区块链应用有着深刻的理解。曾经在 CompSciResources， LLC 和 Primescape Solutions， Inc.和担任产品工程师和高级软体工程师，负责公司面美国证券交易委员会 ITPLUS 系统的开发。

- <u>生态企业</u>

➢ **巴克云**

巴克云是 GCT 生态“**一链一应用**”系统架构实现的重要推动者，目前正主导社区开发 ChainSDK、GeekChain 主链和底层 P2P 协议等关键生态的软件系统。巴克云名字来源于巴克球(富勒烯)，象征着团队对未来互联网“优美、无中心、坚不可摧，充满可能性”的构想。公司创始人刘智聪是原迅雷首席工程师，是迅雷多个核心产品的架构师，曾经主持迅雷的主要基础核心系统（包括 P2P、云存储、机器学习）的研发工作。

➢ **心游科技**

心游科技是一家立足于大型精品游戏开发及整合运营能力，提供面向下一代游戏产品和周边增值服务，并拥有快速把握产品及用户能力的新生代运营研发一体化游戏公司。

心游科技的创始人湛振阳，曾出任北京金山软件有限公司移动软件事业部副总经理、金山数娱亚丁工作室副总经理、北京金山软件有限公司西山居工作室总经理、北京金山软件有限公司副总裁/金山游戏 COO，在金山时代发展曾被称为金山五虎将。

公司其他核心团队成员均来自于金山、Gameloft 等国内外知名的游戏开发商，大多数核心成员具有 6-10 年游戏研发经验以及 30 万人同时在线的大型网游产品运营经验，是目前游戏行业内少数的同时拥有丰富研发和发行经验的游戏团队。

- **投资人**

➢ **硬币资本 INBlockchain**

中国区块链领域规模最大，最具影响力的投资集团，是全球最大的比特币持有者之一，也是有意义的区块链应用的早期投资者。

- **顾问**

➢ **陈广涛 GT.CHEN**

去中心化系统和区块链专家，陈广涛先生在 2009 年创办了多贝网和多贝云。今天的多贝已经是全球最大的去中心化流媒体网络，每天为全球数百万人提供去中心化的通信服务。陈广涛先生拥有 9 年的去中心化系统和 5 年的区块链架构设计经验，参与过多个区块链项目的架构设计工作。

- **发展路线图**

2018 年 1 月 12 日 GCT 登陆 BigOne 交易所，开启 GCT/BTC 首个交易对
2018 年 2 月初 GeekChain 基金会与巴克云正式签约
2018 年 4 月初 GeekChain 基金会与“极路由”正式签约
2018 年 4 月初 BDT 协议内测
2018 年 6 月底 白皮书正式版公布
2018 年 7 月 26 日 “极路由”退出 GeekChain 基金会
2018 年 7 月 28 日 公布首家 DApp 生态企业：心游科技
2018 年 7 月底 BDT 协议公测+开源
2018 年 9 月底 ChainSDK 面向合作伙伴开放开源，开始内部测试
2018 年 11 月 15 日 链银 2.0 钱包测试版发布
2018 年 11 月 26 日 ChainSDK 内测发布
2018 年 11 月底 GeekChain 内测，~~推出首批搭载 GeekChain Miner 硬件，~~暂时不推出新硬件，用好现在的极 X
2018 年 12 月 4 日 多条应用主链开始内测
2019 年 1 月 16 日 基于 ChainSDK 实现的公链 INT 正式上线
2019 年 1 月 31 日 BDT 协议万人公测结束， ChainSDK 公测开启

2019 年 4 月 20 日 GeekChain 测试链小范围公网测试开始

2019 年 5 月 1 日 链银 1.0 停止服务

2019 年 7 月底 GeekChain 测试链大范围公测启动

2019 年 10 月底 GeekChain 主网计划上线

首次发布于 2018 年 6 月

修改于 2019 年 5 月

GeekChain Foundation Limited