BIGBANG.

BIGBANG.

树状区块链系统

技术白皮书 V1.0

摘要

区块链技术作为去中心化的价值传输系统，由匿名人士中本聪首次提出并应用到比特币当中。在比特币系统中，为完成相对复杂的交易类型，中本聪创造性的提出了脚本机制。但当开发者想要通过比特币脚本实现更多的功能时，往往就会受到诸多的限制。为此，Vitalik Buterin 提出的 Ethereum 通过引入图灵完备的智能合约和 EVM 使得基于区块链技术的应用开发成为可能，并被业界称赞为继比特币之后的“区块链 2.0”。但无论是比特币还是以太坊，都面临着由于用户与交易增长过快所带来的拓展性及交易延迟的问题。究其根源，在于当前区块链系统中单链的结构，使得诸多优秀项目在这些问题面前都缺乏足够的灵活性，区块链在物联网这一天生适用的领域的发展也举步维艰。

为解决这些问题，并更好的将区块链与物联网技术相结合，经过不断地探索论证，我们提出了 BigBang Core 树型区块链。BigBang Core 呈“主链 + 多应用支链”的树状结构，通过支链的无限拓展实现单链结构无法解决的交易拓展性和高并发性问题。同时 BigBang Core 作为物联网的基础设施，将建立多实体的设备互信及异构环境下的数据互通。为未来物联网更复杂的商业模式打造稳定可靠的技术基础。

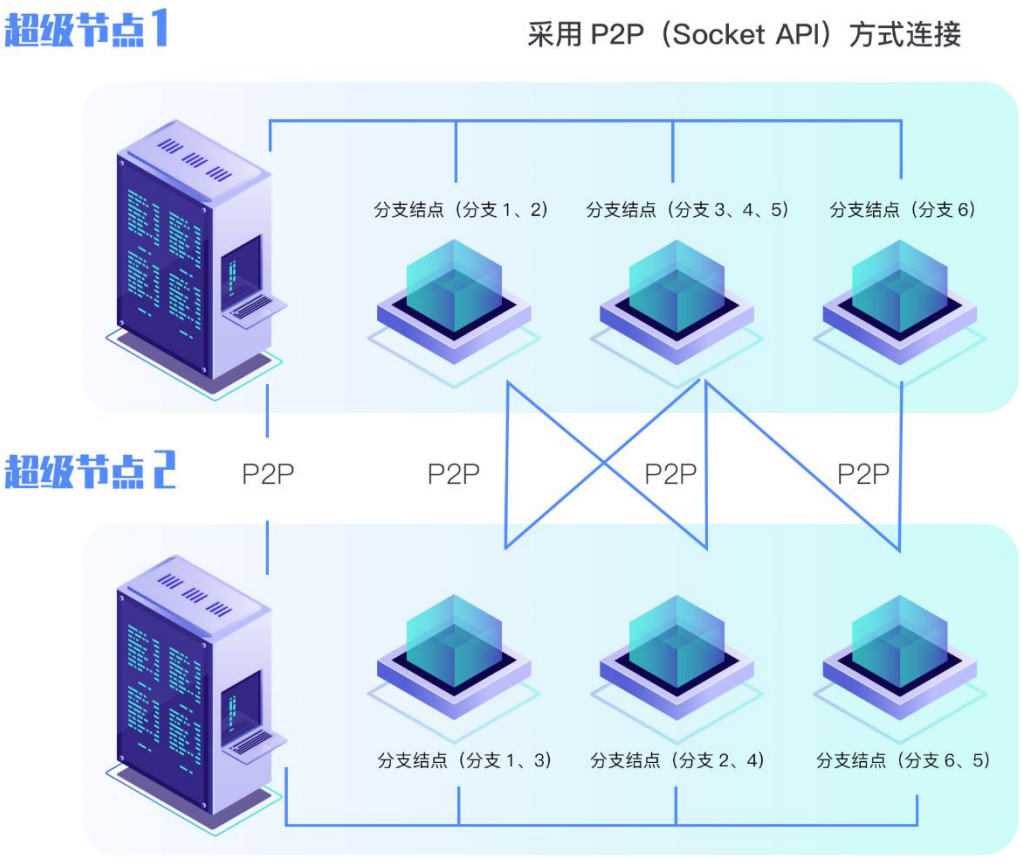
本文重点将针对 BigBang Core 的技术架构及关键技术原理进行详细介绍。

01. 系统描述

1.1 基础介绍

BigBang Core 是构建于 P2P 网络的区块系统，同目前流行的 P2P 数字货币系统类似，以去中心化方式维护透明账本，实现用户数字资产自主安全管理和高效流动。BigBang Core 系统针对 IoT（Internet of Things，物联网）数据业务需求设计，利用区块技术为 IoT 数据业务提供去中心化安全管理平台，实现 IoT 系统所需高并发低延迟等性能要求。

BigBang Core 通过安全共识组织用户交易（transaction），按时间顺序形成数据区块。同 Bitcoin 等单链系统不同，BigBang Core 采用树结构来存储排列区块，可以根据业务类型和数据负载进行分叉形成多个分支。分支之间区块相互独立，新增区块只与自身分支数据相关。在多重分支的情况下，根据业务数据流量，可以分布到多个分支区块中，由此产生的可扩展性和高并发性正是 IoT 系统所需的基本性能。BigBang Core 的多重分支结构由唯一安全主链和众多应用支链构成，安全主链用于支撑全网共识机制，应用支链用于实际业务。在应用支链可以提供最低 2 秒的低延迟交易确认，用户可以指定交易紧迫性，支付相应交易手续费，以此实现低延迟业务。



1.2 共识机制简述

众所周知，在“不可能三角”的各种研讨中，去中心的结果往往意味着低效的 TPS，而物联网的海量数据就成为共识构建中一块无法搬走的巨石，那么在区块链+物联网 IoT 的领域里，究竟什么才会是适合的共识呢，让我们先从共识算法的演进说起。

Proof of X 是目前公链领域内应用较多的一类共识。其中 PoW 最早被应用，但存在资源浪费、算力集中、缺少终局性以及性能低下等。

PoS 是目前有力竞争者，可避免资源浪费、弱化了中心矿池需求、降低 51%攻击可能性，但也同时存在确定记账节点数量困难、存在非预期的中心化问题、Nothing at Stake 等问题。

为了解决以上弊端，当前也诞生了许多混合类共识，希望既融合两者的优势，又能规避某些弊端，包括 PoW+PoS、DPoS+BFT 等。所以混合共识机制可能会是公链后期发展的一个出路。

1.2.1 PoW 共识算法

PoW（Proof of Work）即工作量证明，根据矿工的工作量对数字货币进行分配，矿机的性能越高，数量越多，工作量越大，得到的数字货币就会越多。

BTC 是采用 PoW 方案最典型的原型。它通过挖掘过程包括解决一个数学问题，矿工通过这种技术手段完成了 PoW，就获得了记账权。因为它需要计算力的资源，成功的矿工会得到 BTC 作为奖励。为了控制货币基础，挖矿被设置成了更加复杂的模式。因为每个矿工解决问题的可能性依赖于他的算力，挖矿的难度由系统中所有算力的总和来决定。

对于 PoW 机制的加密货币，矿工是通过竞争解决数学问题来确认和固定转账。第一个解决问题的矿工得到奖励。该问题的复杂是刻意制造的，用来控制货币基础。

这个处理过程被一些人认为是天才之举，很好的解决了拜占庭将军问题。但是被另外一些人批评没有效率因为白白损失了资源。同时，单一的 PoW 机制也面临着 51%算力攻击等安全性问题。

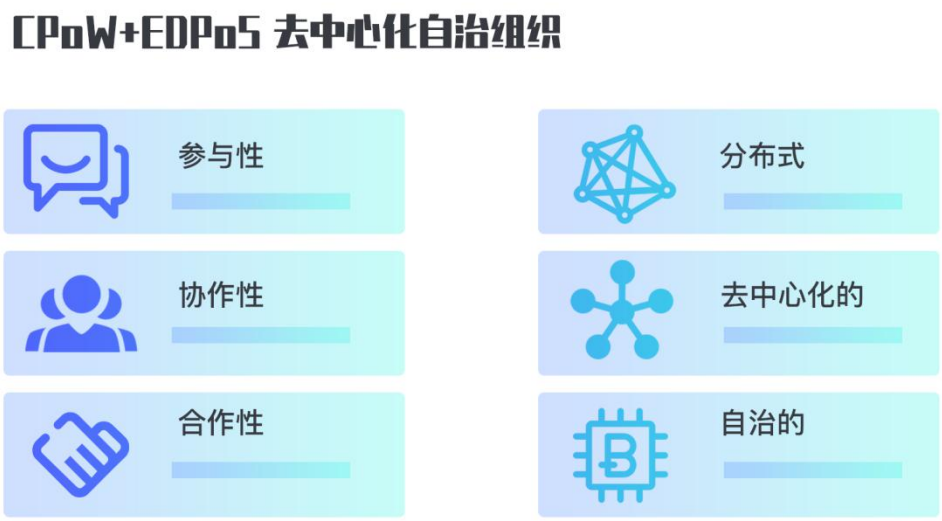
随着 BTC 的发展与区块链的行业发展，PoW 机制的缺点也暴露了出来。持币者无法参与任何决策，话语权集中在矿工的手中，这与去中心化的理念背道而驰，决策权集中在少数矿工手中。

1.2.2 DPoS 共识算法

DPoS 是基于 PoW 及 PoS 的基础上，出现的一种新型的保障数字货币网络安全的共识算法。它既能解决 PoW 在挖矿过程中产生的大量能源过耗的问题，也能避免 PoS 权益分配下可能产生的“信任天平”偏颇的问题。那么，DPoS 就能顺理成章成为共识机制 3.0 脱颖而出的代表性共识机制。DPoS 它能够让用户广泛参与到挖矿中来 ，指的是让每一个持币者都可以进行投票，由此产生一定数量的代表 ,或者理解为一定数量的节点或矿池，他们彼此之间的权利是完全相等的。持币者可以随时通过投票更换这些代表，以维系链上系统的“长久纯洁性”。

1.2.3 EDPoS+CPoW

为防止单共识节点崩塌而导致整个生态的运行停止，以及防止因 DPoS 节点集体罢工而导致区块网络整体瘫痪，CPoW（可持续的工作量证明机制）和更加“去中心化”的 EDPoS（可拓展的委托权益证明机制）应运而生。BigBang Core 的安全共识机制为 EDPoS（可拓展的委托权益证明机制）+CPoW（可持续的工作量证明机制），节点收益为出块奖励息加上块内交易总交易费。用户可以用 Token 为 EDPoS 节点进行投票，投票为 EDPoS 节点增加出块概率。当 EDPoS 节点成功产生新区块，对应投票用户也按投票额度分享出块奖励。节点需要筹集超过 Token 供应总数 2% 投票才能成为 EDPoS 节点。



CPoW 作为 EDPoS 共识的补充，每轮 EDPoS 协商过程有一定概率将首要出块权交给 CPoW 共识。参与 EDPoS 过程的 Token 越少，说明 EDPoS 共识的安全性和可靠性越低，这种情况下，通过 CPoW 共识获取出块权概率越高，混合 CPoW 机制增强系统安全性和可靠性。

BigBang Core 的树状结构中，除了安全主链外，其余支链地位对等且相互独立。

EDPoS 节点群通过安全计算共同建立出块序列，同时产生真随机数信标。应用支链出块系列分配由安全主链的随机数信标计算产生。

根据拜占庭容错原理，恶意节点少于 1/3 整个安全计算过程就不会被干扰；合理选择协商算法和参数，可以实现非 51% 攻击情况下，安全计算过程就不会被控制。在 BigBang Core 系统中，共识机制可以达到有较高的一致性，系统性分叉非常罕见，在恶意节点所持 Token 少于参与 EDPoS 总数 50% 情况下，3 个确认可保证主链历史数据不可回滚。

1.3 网络描述

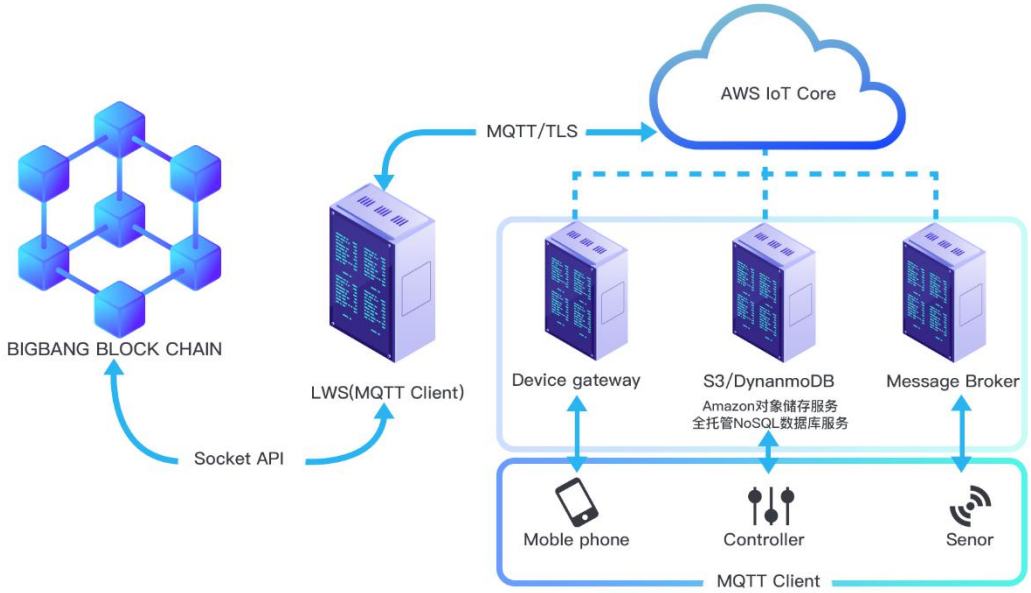
BigBang Core 网络由运行 BigBang Core 软件的节点构成 P2P 网络。BigBang Core 的整体网络架构可分为三层：节点网络层、终端服务层、IoT 终端层。

节点网络层由运行 BigBang Core 核心节点程序的节点构成，节点之间同步校验区块和交易数据，并进行共识组织区块数据。

终端服务网形成分布式终端后台，为 IoT(Internet of Things) 终端提供接入服务。

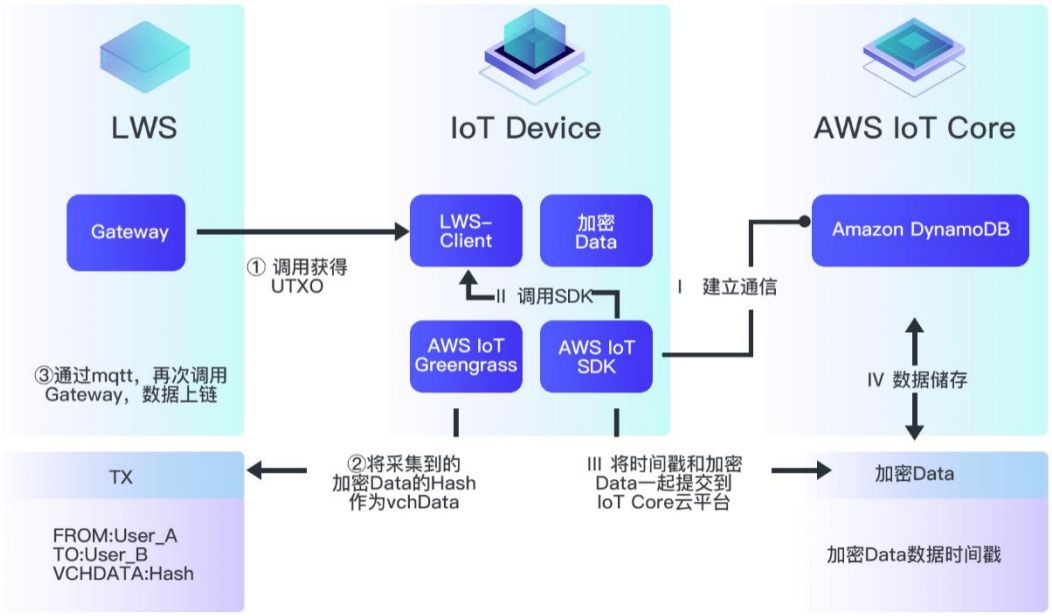
为了支撑庞大的 IoT(Internet of Things) 业务，节点网络与终端服务网共同组成 BigBang Core 服务平台。

IoT 终端层包括智能传感器、控制器和移动终端，内嵌轻客户端程序，本地保存私钥完成交易构建和校验。



1.4 系统软件组成

为更好支持 IoT 复杂环境下的多种应用场景，同时保证区块链服务的可靠运行和普通用户的使用需求。BigBang Core 系统软件的设计总体包括五部分：核心钱包程序、轻钱包后台服务系统、移动端轻钱包程序、嵌入式系统轻钱包 SDK 和在线区块浏览器。



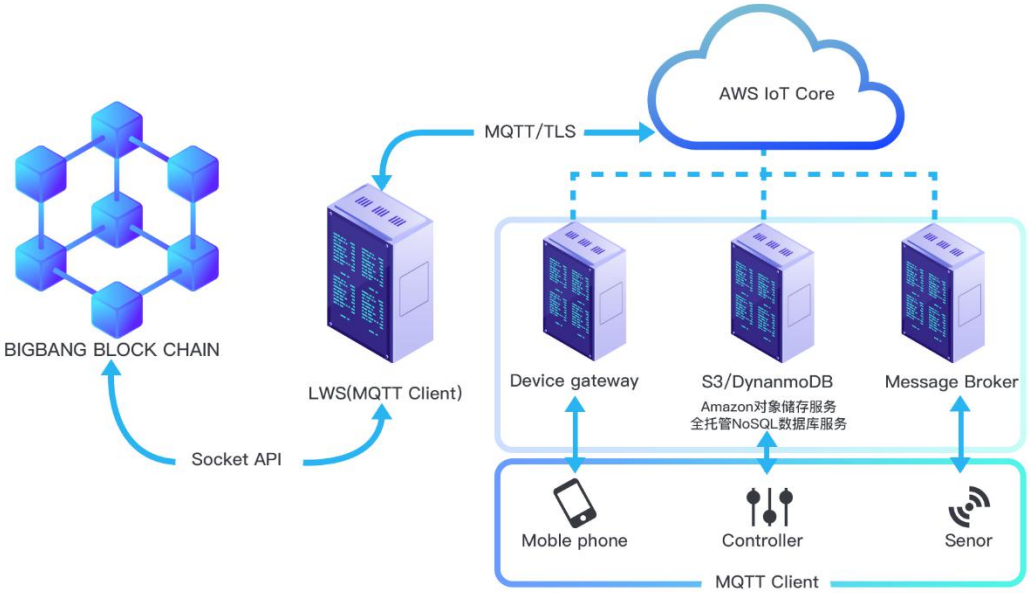
1.4.1 核心钱包程序

核心钱包程序用于主干网络节点和普通用户，对运行环境和硬件有一定要求，可以完整使用区块系统所有功能模块。

1.4.2 轻钱包后台服务系统

LWS 是 light wallet service 的缩写，是架设在 BigBang Core 公有区块链主干网络和终端数据采集传感器设备之间的一座桥梁。通过它，BigBang

Core 核心钱包的区块和交易数据及时地更新和缓存在 LWS 自有的高速内存数据库及本地数据库中。



根据这些数据，它会计算出不同终端设备持有密钥所对应的公钥地址的最新 UTXO 集合，并通过与 AWS 的 IoT Core 的 mqtt 连接，将这些信息发布（publish）到亚马逊云端设施上，由其 message broker 向对应的订阅（subscribe）了这些信息的终端设备转发。相应地，终端设备会根据这些与自己相关的 UTXO 列表，在获取了监控监测采集的数据后打包这些数据到交易中，通过 mqtt 发布到亚马逊 IoT Core。

经由后者的 message broker 向订阅了这些设备的发送交易主题的 LWS 推送，LWS 会校验这些交易，如果验证成功，则会通过 Socket API 向 BigBang Core 核心钱包转发这部分交易，后者收到之后通过 P2P 网络接口向 BigBang Core 全网广播这些交易，出块节点收集这些交易，最终完成其打包区块上链的操作。

LWS 使用 AWS 提供的基于长连接、双向的消息 pub/sub 消息代理解除与巨量连接的 device 端数据交互的耦合关系，解决了设备的高并发性和高扩展性。对于区块及交易数据的存储查询及 UTXO 数据的更新，LWS 使用 AWS 的 Amazon DynamoDB 服务存储它们 KV 键值对数据。

考虑到 BigBang Core 公链网络多支链上高并发 TPS 产生的海量交易数据及打包区块数据，以及海量的 UTXO 数据，利用 AWS 的 ms 级响应延迟的数据存储服务 Amazon DynamoDB, 可以为每个业务分支链创建一个区块数据库和交易数据库，加速数据的检索能力。

LWS 同步主干网络下行的区块链数据的同时，配合高吞吐量、弹性扩展的 Amazon Kinesis 服务，使用 Amazon S3 高度扩展（Scalability）、高持久性（Durability）和高可用（Availability）的分布式数据存储服务缓存巨量的区块文件到亚马逊云端，完成区块实时数据收集和处理，可以为本地物理地址近邻的其它 LWS 使用，甚至向世界范围的 LWS 提供检索服务，另一方面，LWS 在与核心钱包失步或数据错误时，可以使用 S3 中的数据快速恢复。此外，LWS 使用 AWS 的规则引擎 rules engine 将消息转换并路由到 AWS 服务，后端使用 Kinesis 服务分流数据到不同的 AWS 服务，或者接驳 Lambda 服务分流数据。在区域网络传输不均衡的环境中，也可以使用 AWS 的 CloudFront 服务提供 CDN 类似的功能。

使用 PB 级的 Amazon Redshift 关系型数据仓库，可以存储结构化区块链数据，便于 BigBang Core 区块链 web 浏览器、智能设备钱包 app、BigBang Core 区块链开发测试人员调试跟踪程序运行时的数据视图。

LWS 使用高并发语言 golang 开发，程序采用 goroutine 及 channel 设施保证了数量庞大的 device 端同时发送的发送交易到核心钱包主干网络的请求能够及时有效地处理，从而实现了海量交易的高速上链。

1.4.3 移动端轻钱包程序

移动端轻钱包程序可以使终端节点在不运行完整钱包程序的情况下，也能够对交易进行校验。主要用于 IOS 和 Android 移动终端，在网络带宽和硬件性能都有较大限制的情况下，为用户提供安全钱包服务。

1.4.4 嵌入式系统轻钱包

嵌入式系统轻钱包 SDK 为 IoT 智能硬件提供轻钱包 API，可以通过终端服务器接入 BigBang Core 网络，不需要在本地进行繁重的区块同步和区块数据存储，专注与业务相关的交易数据构造和鉴权。

1.4.5 在线区块链浏览器

在线区块浏览器配合钱包节点实时展现区块系统状态，查询历史区块交易数据。

1.5 系统特点

· 树状结构

与 Bitcoin 等传统单链系统不同，BigBang Core 使用树状结构来组织存储区块数据。

· 高拓展

面对物联网海量的数据交易，通过依托于无限拓展的树状结构来实现整个系统的横向拓展。

· 高并发

应用支链每秒的交易笔数(TPS, Transaction Per Second)能够达到5200笔，并且一笔交易最快 2S 即可完成链上确认，能够满足物联网下数据交易频繁且低延迟的场景特点。

· 数据快速接入

通过良好的系统设计，为流式数据和持久化数据均提供稳定接口和后端服务使得数据快速上链。

· 数据公证

BigBang Core 链上存储数据标识及流转记录，数据标识一旦上链就不可更改，可为需求方和提供方提供数据的一致性证明和公证信息。

· 物联网价值流转平台

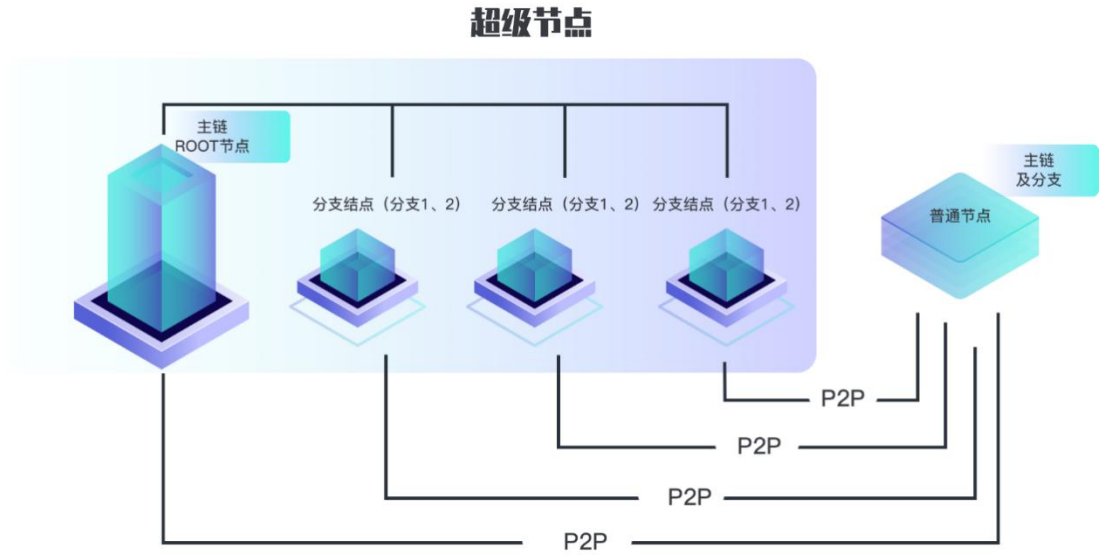
作为物联网技术底层的基础设施，为物联网上的数据传输与价值流转提供稳定可靠的技术平台。

02. 树状区块结构

在如今常见的区块链项目中，所有交易信息均存储在单链区块当中，使得整个系统面对不断增长的交易规模时缺乏足够的灵活性。在 BigBang Core 中，主链数据与应用数据进行了分割处理，以“安全主链 + 多重应用支链”的树状区块结构来存储系统区块数据。

安全主链主要存储交易与安全共识相关数据；应用方则通过从任意链条进行分叉，生成支链（分叉链），专门组织和存储与应用业务相关的数据。并且随着交易规模的扩大，支链可以继续建立子级支链。通过这种类似垂直分割的方式，杜绝了传统单链结构中所有交易填充在主链区块的弊端，实现了整体系统的横向拓展。

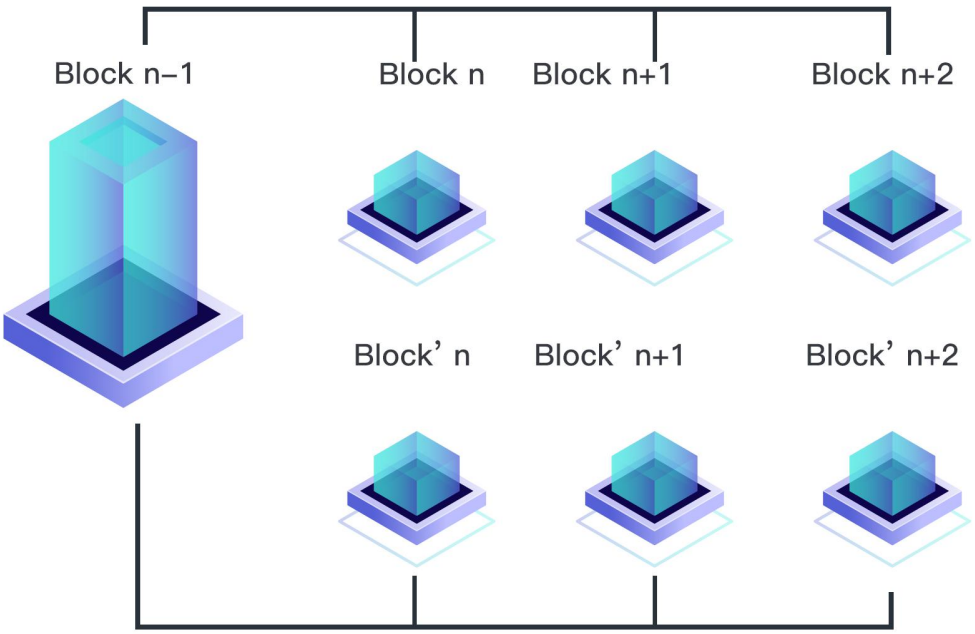
BigBang Core 上的支链数量越多，系统可承载的 TPS（Transaction Per Second）也就更高，在应用支链足够多的情况下，BigBang Core 整体可实现千万甚至亿级的 TPS 承载。



2.1 分支标识

BigBang Core 系统的区块按时间顺序连接在一起，以多分支形成树状结构。在 BigBang Core 中，安全主链和应用支链统称为“分支”。每一条分支均会有一个独特发的分支标识来进行标记。 安全主链以创世区块 hash 作为支链 ID，应用支链以分叉点后第一个区块的 hash 作为支链 ID。

在分叉之前，父链和支链拥有完全一致的链结构和交易；分叉点之后则相互独立，互不干扰。出现在分叉点前的同一笔 Token 在分叉点之后可以在父链及支链中创建不同的交易发送到不同的地址；分叉点之前的区块数据在父链及支链中也可以通用。 用户在创建交易的时候需要指定一个锚定区块，标定在此区块之后的所有 支链有效。在图 2.1 中，如果锚定区块设定为 Block n−1，创建的交易会被包含到两个支链中；如果设定为 Block n，则该交易只在父链有效，支链中可以创建新交易将 Token 发送其它地址。



2.2 安全主链

安全主链为 BigBang Core 树状结构中的主链，所有的支链均为其“后代”，其被用于支撑全区块系统的安全和共识，在 P2P 网络中主链的同步广播消息转发优先级高于应用支链。安全主链除了记录主链 Token 转移，还保留 EDPoS 节点协商关键过程数据。安全主链的区块之间不能插入子块，只能按照既定出块间隔增长。由于会有相当部分容量记录共识协商过程数据（以 23 个 EDPoS 节点为例，协商数据会占据每区块 115KB 左右的容量），故安全主链的交易容量低于应用支链。 安全主链以区块系统创世区块为起点，通过 EDPoS+CPoW 共识顺序产生区块。安全主链被用于支撑全区块系统的安全和共识，所有应用分支节点都需要同步和校验主链区块头信息。新节点接入网络后，首先完成主链同步，才开始进行对应应用分支同步。

2.2.1 主链特殊交易

在安全主链中，鉴于功能特殊性，有三类与共识机制相关交易是安全主链独有的：EDPoS 节点投票交易；EDPoS 节点登记交易；CPoW 出块奖励交易。

1.EDPoS 节点投票交易

EDPoS 节点产生一个 Delegate 模板地址，首次需要自己发送 Token 到该地址，完成 Delegate 地址链上发布；用户使用与 EDPoS 节点相同的参数创建 Delegate 地址，并将 Token 寄存于该 Delegate 地址，完成 Token 投票。EDPoS 节点可以使用 Delegate 地址的投票作为权重参与 EDPoS 协商过程。 用户将币寄存 Delegate 地址进行投票时，所有权依然属于用户，并且可以随时取出，但是一旦取出，相应节点的投票数量也随之相减。

2. EDPoS 节点登记交易

EDPoS 节点在每轮协商需要筹集足够 Token 投票，并以此创建登记交易提前在链上进行登记和发布自己初始协商参数，只有协商轮次开始前完成登记的节点（超过总票数的 2%）才允许进入协商过程以及获取出块权。

3.CPoW 出块奖励交易

CPoW 共识缺省情况下只用于主链共识出块，对应出块奖励通过这类交易提供给参与者。该类交易的作用类似 Bitcoin 中 coinbase 交易。

2.2.2 主链参数

token 年膨胀率	1%
区块产生间隔	60 秒
区块最大字节数	2MB
初始出块奖励	15
最大 TPS	156
EDPOS 节点最大数量	50
CPOW 难度调整间隔	相邻 C P O W 区块

2.3 应用支链

在 BigBang Core 中，应用方通过在父链发送一种特殊类型的交易 ------ 分叉交易，用于创建应用支链。应用支链的区块产生间隔需要和安全主链一致，其它主要参数可以在创建分支初始化过程中由创建者配置，可配置参数包括 Token 总量和分布、出块奖励和增发方式等。

新创建支链的第一个区块（分支起始块）被保存在分叉交易中。支链的 Token 分布可由创建者定义，有三种方式：

1. 创建独立分支，分支起始块重新设置 Token 总数和分配方式；

2. 完整继承分叉点 Token 分布；

3. 继承分叉点 Token 分布，并在此基础上进行增发，增发部分的分布方式在分支起始块中定义。

自分叉点之后，支链 Token 和父链是完全隔离的。

2.3.2 抵押机制

为了防止恶意人员通过高频分叉对父链带来的资源损耗，每一次建立支链都需要使用父链 Token 进行抵押，分叉交易中用于抵押的 Token 被发送到一个特殊地址进行冻结。抵押 Token 根据父链区块高度和父链区块起始高度差值分阶段解冻，创建者使用自己私钥进行签名后可以将解冻部分 Token 转移到其它地址。创建支链所需抵押 Token 随区块高度和起始高度差值递减，每隔 525600 区块完成一次减半。其中抵押 Token 的基数 N 由父链的初始 Token 供应总量决定。

2.3.3 应用支链出块

应用支链的安全性依赖于安全主链的共识机制。在 BigBang Core 中，应用支链的出块方式主要有三种：

1）EDPoS 节点出块。该种方式可以不设立出块节点，由 EDPoS 节点在获得主链出块权同时为分支产生新区块，该种方式对应公开业务模型；

2）自设立节点出块。应用方也可以在应用支链自设立出块节点，借用安全主链产生的随机信标设立本分支的出块机制。该种方式对应封闭业务模型。

3）**自定义共识机制。**除以上两种出块方式外，应用方也可以自定义共识机制，用于应用支链区块和子块的生成。

应用支链和安全主链相比，除产生正常的支链区块之外，在正常每分钟出块间隔中，还可以产生子块，用于低延迟交易上链。子块间隔不低于 2 秒，且不可以产生空块。产生子块的节点由安全主链同高度区块块独立随机信标决定，子块没有额外出块奖励，但可以获取高额交易费收益。

考虑应用支链中的子块，每个应用支链的交易容量可以提高 30 倍，TPS 可达到 5200。当数据业务需要更高交易容量和并发性时，可以对针对当前应用支链创建多个支链，以此实现高量级 TPS。

2.3.4 支链空块

在 BigBang Core 中，应用支链高度维持和安全主链的区块高度同步，用于支撑应用支链间的跨支链交易以及方便支链区块引用主链同一高度的共识结果，减少便利查询的次数。故为了维持高度一致，应用支链可能产生空块。且在空块和下一个支链区块之间，不能产生子块。

应用支链选择主链的 EDPoS 节点出块时，主链 EDPoS 节点会将应用支链 Token 价值作为优先级参考，并自主选择支持的应用支链。价值比较低的应用支链可能会被部分主链 EDPoS 节点排除在出块列表以外。此时支链为了维持主链的高度同步，会自动填充一个空块。

2.3.5 支链参数

应用支链承载实际业务的交易数据，子块和支链区块及主链区块的大小一致。通过缩小子块的出块间隔从而对支链 TPS 进行扩容，应用支链的基础参数如下：

支链区块产生间隔	60 秒
子块产生间隔	2 秒
支链区块最大字节数	2MB
子块最大字节数	2MB
支链区块生成奖励	缺省 15，后续用户可自定义
子块生成奖励	无
支链最大 TPS	5200

03. 用户密钥与地址

BigBang Core 的地址有两类：公钥地址以及模板地址，分别对应特定公钥和模板。地址长度固定为 33 字节，在交互性界面中，采用编码后的地址作为输入 / 输出参数。

```
pubkey address:
encoded address = '1' + BASE32Encode(pubkey + CRC24q(pubkey))
template address:
encoded address = '2' + BASE32Encode(template ID + CRC24q(template ID))
```

其中 BASE32Encode 采用 Crockford 方案字符集，但不进行该方案中 symbols check 过程。

3.1 密钥和公钥地址

BigBang Core 系统采用 curve25519 作为基本安全算法，用户私钥和公钥均为 32 字节，私钥签名为 64 字节。curve25519 安全性和 P256 相同，同安全性算法中是目前效率最高的非对称安全算法。以类型前缀 + 公钥作为钱包公钥地址。

```
secret key:
9b9d0d52a251cf5933bb65e864dfff41ca26d650baf112324f4ce234dfcfe7aa
public key:
59a317119823caacb0acaed78f029aa2fcfbb9417c298892e660b5fc174cb3b9
encoded address:
1q6smr5zwpnged4m855y43efvzjh9m0mftyqasc5cs8hsg48qmdcgf9g0
```

为了保证用户私钥安全，在本地存储采用 chacha20+poly1305 算法加密，需要用户输入密码才可以使用私钥进行签名操作。

3.2 模板地址

模板地址由类型前缀 + 模板 ID 构成。模板 ID 由 2 字节模板类型 + 参数 Hash 低位 30 字节构成。例如一个 3-5 多重签名模板 ：

```
public keys:
  1: fcd74aa82a1eb098830a2fcc877735a60152b441c16b2212157c4215db074e88
  2: f1a1ced60a7ecdf83735a3380765f2ef77221f367da05bd901e885b9d799aec5
  3: c2885254a2acefaeb05bd94b0e73e483bded994b02ebd0bc6b3523c2dde558dd
  4: e2de897ad0935bbfd6cca48da2ee285c87ae784285df35513180143ec55c8450
  5: b1f1ce918f30b46aa3d2648810f6153410e44122c042998699323b982664a16f
tempate ID:
000244c03d536e6175912b3040aa876388b197c21ae55c283f182403ab610852
encoded address:
2a8463ar34gc3ya2wwmdc55xhh1hrfaj060ns2xb1ds9kvg240803k041
```

3.3 带参数模板

时下流行的区块链系统可以提供运行于不同 VM（Virtual Machine，虚拟机）之上的脚本或智能合约，可以对区块系统基本账本进行强大灵活的功能扩展。但是截至目前看来，区块系统中的 VM 模块还处于起步阶段，除了存在内在安全漏洞等问题外，运行效率和使用费率也在一定程度上限制了智能合约适用范围。BigBang Core 系统不提供脚本和智能合约系统，而是采用带参数过程模板实现常用的脚本和智能合约功能。采用对应模板地址为用户提供功能调用。BigBang Core 系统提供以下模板：

类型标识	名称	参数	描述
0x0001	带权重多重签名模板	签名所需权重，公钥和对应权重列表	可分别配置参与公钥签名权重，分配不同权限，公钥数不大于 16
0x0002	简单多重签名模板	签名所需权重，公钥和对应权重列表	参与公钥签名等权重，公钥数不大于 16
0x0003	创建支链模板	签名所需数量，公钥列表	创建者可从模板地址取回解冻 token
0x0004	冷钱包挖矿模板	支链创世纪块 hash，创建者解冻抵押 token 地址	采用出块签名公钥和花费地址分离机制，支持安全冷钱包挖矿
0x0005	EDPOS 代理节点模板	区块签名公钥，节点所有者地址	采用出块签名公钥和花费地址分离机制，支持安全冷钱包 EDPOS 过程
0x0006	跨分支交易模板	交易双方地址	用于进行分支交易进行身份识别和条件判断
0x0007	电商交易模板	采购方的公钥和出售方公钥	用于支持数据交易中的订单交易

04. 区块与交易

4.1 区块

BigBang Core 区块的数据结构设计如下：

Elem	Type	备注
nVersion	uint16	版本号
nType	uint16	区块类型，用于区分创世纪块、主链区块、支链区块和支链子块
nTimeStamp	uint32	时间戳
hashPrev	uint256	前一区块 hash
hashMerkle	uint256	vtx 所包含 transaction 构建的 Merkle Tree Root
vchProof	vector<uint8>	用于校验共识合法性数据
txMint	CTransaction	出块奖励交易
vtx	vector < CTransaction >	当前区块包含交易列表
vchSig	vector<uint8>	区块签名

说明：

· 目前区块版本为 0x0001 。

· 时间戳采用 UTC 以秒为单位。

· vchProof 包括了合法性证明系列化数据，在安全主链中，包括 EDPoS 节点广播的计算结果（包括各节点签名），CPoW 区块中还包括工作量证明参数；在应用支链中，包含同高度主链区块 hash 和共识计算结果。

· txMint 不进行签名，签名字段为空。

· 区块签名 vchSig 使用 txMint 输出地址进行签名，签名数据段包含除 vchSig 以外所有字段。

4.2 交易

BigBang Core 采用 UTXO 模型记录交易，包括以下数据：

Elem	Type	说明
nVersion	uint16	版本号
nType	uint16	类型，区分普通交易、投票交易、挖矿
nLockUntil	uint32	交易冻结至高度为 nLockUntil 区块
hashAnchor	uint256	交易有效起始区块 HASH
vInput	vector<CTxIn>	前序交易输出列表，包含前序交易 ID 和输出点序号
addrTo	CDestination	输出地址
nAmount	int64	输出金额
nTxFee	int64	网络交易费
vchData	vector<uint8>	交易数据
vchSig	vector<uint8>	交易签名，可以包含模板参数

说明：

· 目前交易版本为 0x0001。

· hashAnchor 用于指明当前交易起始有效区块以及对应分支。

· 输入列表中的前序交易要求输出地址相同。

· 交易包括两项输出，一项为表中所列（ addrTo/nAmount ），另外一项是隐含的找零输出，地址同输入地址，金额为（ Total Input – nAmount – nTxFee ）。

· 交易签名用输入列表统一地址，签名数据段包含除 vchSig 以外所有字段。

4.3 BigBang Core 跨分支交易

跨分支交易可以用于实现 BigBang Core 分支之间无信任情形下同步价值交换。实际应用中，往往可以将业务按照业务流程、设备种类、空间地

域等关联因素进行划分，分散到多个分支中。互动频繁的设备通常持有同一分支 Token，在同一分支进行数据交易。但作为一个业务整体，和持有其它分支 Token 设备交互的需求也是客观存在的。这种情形下，跨分支交易就可以实现支链之间的 Token 交换。一方面跨分支交易可以在无信任情形完成，利用技术原理保证了对双方的公平性；另一方面跨分支交易在两个支链之间同步进入区块，保证了高效率和有效性。这为包括去中心化交易所、Token 兑换网关等应用提供了良好的底层技术支撑。

05. 共识机制

如前文所述，BigBang Core 系统采用的共识机制为 EDPoS+CPoW，以 EDPoS 为主导，决定下次获得出块权的节点或者指明下一区块由工作量证明共识产生。在 EDPoS 机制未能有效建立时，例如启动初始阶段，CPoW 成为唯一的出块共识机制。

下面对共识机制进行详细说明。

5.1 EDPoS 节点协商过程

EDPoS 节点以所持 Token 投票数作为出块权重，通过随机数计算产生固定出块节点系列。EDPoS 机制建立后，通过 EDPoS 节点之间协商产生随机数。协商过程每分钟进行一轮，通过加权可验证密钥分享（VSS）和拜占庭容错方式进行公平随机计算。

每轮协商都包括以下几个过程：1. 节点登记；2. 加密分片数据分发；3. 秘密分片公布；4. 数据重构和随机信标计算。

在每轮协商之前，每个 EDPoS 超级节点需要利用 ECC 算法产生一组私钥：{a0,a1,…,at−1} ，以及对应公钥：{A0, A1 ,…, At−1} ，满足 Ai = aiG, （i = 0,1,…,t−1）。t 为重构数据的门限值，根据对有效 EDPoS 超级节点设定，t 最大值为 50。

1. 节点登记

EDPoS 超级节点在本轮协商对标区块 16 个区块之前将登记信息广播上链，包括加密后的多项式系数 {A0,A1,…,At−1}, A0 作为节点协商公钥。

2. 加密数据分发

分发开始于协商对标区块之前 16 区块止于前一区块。分发过程开始时，根据登记节点的顺序和权重分配计算序号，每个节点分配到的计算序号数量计算方法为：

[token vote / (total supply X 2%)]

节点 i 根据其它节点 j 发布的协商公钥创建共同密钥 Kij，将秘密分片 sij 以 Kij 加密后广播全网，对应节点 j 解密后可以用节点 i 的登记信息对 Sij 进行校验。其中 Sij 下式计算：

$$s_{ij} = \sum_{k=0}^{k<t} a_k \times j^k$$

由于节点 i 的加密公钥 {A0,A1,…,At−1} 在登记过程已公布，节点 j 通过下式进行校验：

$$s_{ij} \cdot G = \sum_{k=0}^{k<t} j^k \times A_k$$

若上式成立，说明节点 i 发送了正确秘密分片。

3. 秘密分片公布

当前一区块广播后，每个节点将通过校验的所有秘密分片广播全网，全网节点在收集到解密的节点分片，也可以通过上面公式进行校验，最终剔除恶意节点数据后将有效数据进行计算。

4. 数据重构和随机信标计算

全网节点在收集节点 i 的 t 个秘密分片就可通过拉格朗日方程重构 {a0,a1, …,at−1} ，不能收集到 t 个通过校验秘密分片的节点会被剔除，不能进入下一阶段计算。重复上述计算过程，最终可获取所有有效节点数据。此过程中，所有可靠节点计算结果将一致，通过组合计算，得到全网一致的随机信标。由于用于计算的数据分别由各 EDPoS 节点随机提供，在进行到最后一步计算前，都无法获知其它节点的数据。作弊节点在校验和重构阶段就会被剔除，在不考虑 51% 攻击的情况下，没有节点可以控制最终计算结果，因此可以认为产生的随机信标具备真随机属性。

5.2 出块权分配

在 EDPoS 机制没有有效建立起来的情况下，当前区块由 CPoW 共识产生。当 EDPoS 协商成功完成，就会用随机信标进行掷骰过程，假设 EDPoS 超级节点 i 的 Token 投票为 Vi，总的 EDPoS 投票 Vd = V0+V1+...+Vn，总 Token 供应量为 S。

CPoW 等效投票为 Vwork = S * (1 – Vd / S)^3

节点 i 获得出块权的概率 Pi = Vi / （Vd +Vwork)

CPoW 获得出块权概率 Pwork= Vwork / （Vd +Vwork）

重复上过程，就可以得到确定的出块序列。按照出块序列，对应节点完成当前区块出块，并将解密后的协商最后一步计算过程记录进区块，自证出块合法性。

在经过 EDPoS 协商，确定的出块系列可以被所有节点一致计算验证，除了指定确定节点进行出块，有一定概率指定 CPoW 出块。如上面公式，CPoW 区块被选中的概率和参与 EDPoS 协商总 Token 数量相关：

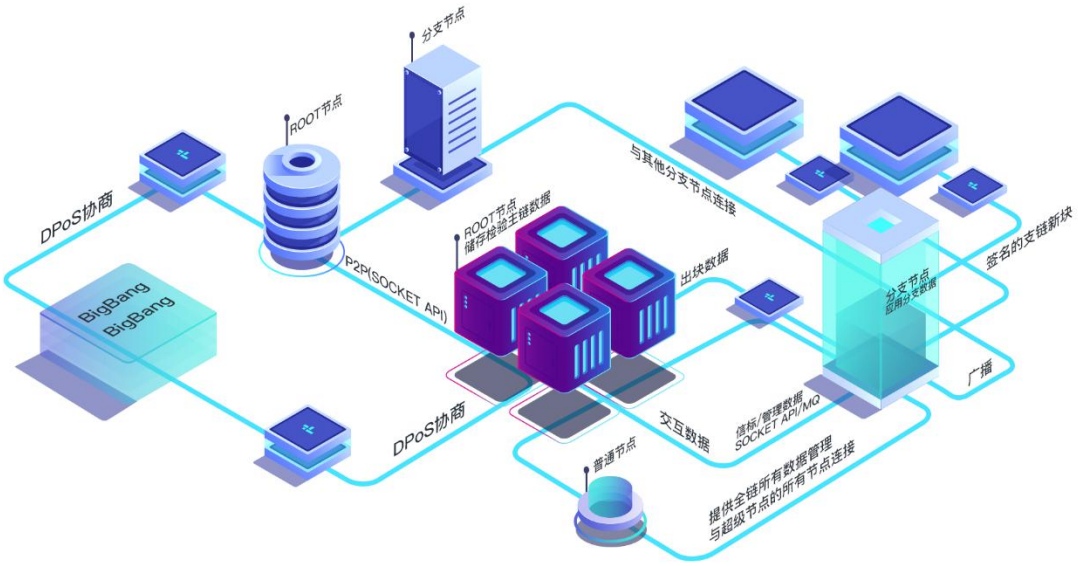
Vd / S	Pwork
0	100%
0.25	62.8%
0.5	20%
0.75	2%
1	0%

在起始阶段，参与 EDPoS 的节点和 Token 比较少，共识机制退化为以 CPoW 为主，当越来越多 Token 参与到 EDPoS 过程，CPoW 出块几率会迅速降低。

06. 网络功能单元

6.1 网络节点

BigBang Core 的网络节点上运行核心节点程序，用于区块数据生成，不同节点之间数据更新同步等功能。其中核心节点程序的构成如下图所示：



底层由一系列基本库和工具类构成，提供包括数据存储、数据库访问、安全算法、应用程序框架、P2P 网络 /HTTP 等底层程序接口。

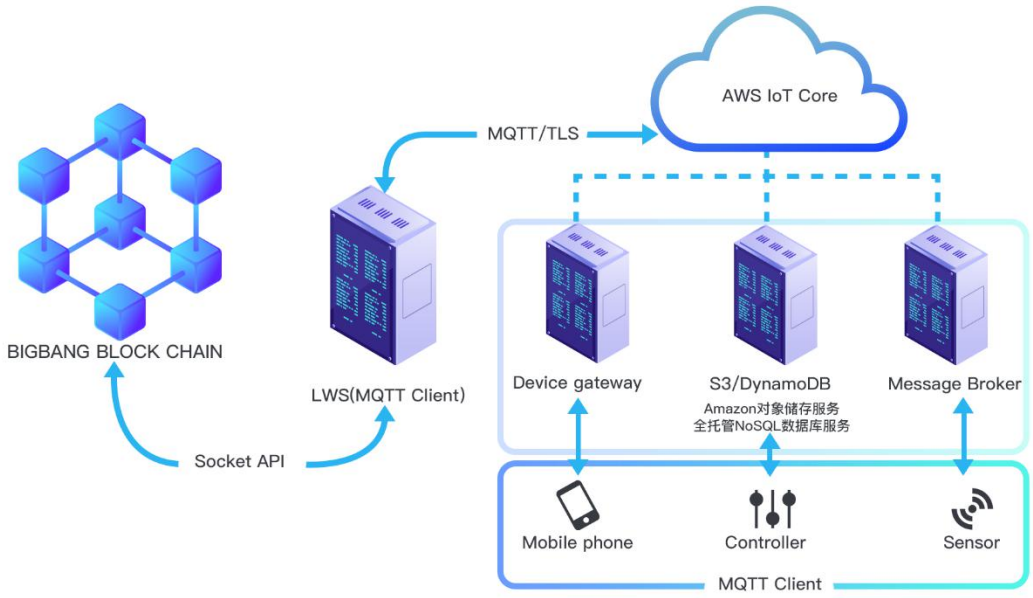
中间层包括区块数据 / 实时交易管理、用户钱包、区块产生构造几部分，分别用于校验管理特定支链区块 / 交易数据、内置用户钱包密钥和用户交易管理、EDPoS/CPoW 共识出块。 上层 P2P 网络层实现节点网络协议，一方面管理调度和其它节点的网络连接以及数据请求，另一方面通过数据分发接口同中间层进行数据同步交换。

核心节点程序通过 JSON-RPC 和 Socket API 两种方式同外部服务进行交互以提供节点控制和功能扩展，JSON-RPC 主要面向 RPC 客户端

程序、钱包图形化界面、钱包节点管理等人机交互应用，Socket API 则为轻钱包服务和分布式节点部署提供高速数据同步通道。这两种外部接口由 API SERVICE 提供核心功能汇聚。

6.2 轻钱包服务与客户端

LWS 是 light wallet service 的缩写，是架设在 BigBang Core 公有区块链主干网络和终端数据采集传感器设备之间的一座桥梁。通过它，BigBang Core 核心钱包的区块和交易数据及时地更新和缓存在 LWS 自有的高速内存数据库及本地数据库中。



根据这些数据，它会计算出不同终端设备持有密钥所对应的公钥地址的最新 UTXO 集合，并通过与 AWS 的 IoT Core 的 mqtt 连接，将这些信息发布（publish）到亚马逊云端设施上，由其 message broker 向对应的订阅（subscribe）了这些信息的终端设备转发。相应地，终端设备会根据这

些与自己相关的 UTXO 列表，在获取了监控监测采集的数据后打包这些数据到交易中，通过 mqtt 发布到亚马逊 IoT Core。

经由后者的 message broker 向订阅了这些设备的发送交易主题的 LWS 推送，LWS 会校验这些交易，如果验证成功，则会通过 Socket API 向 BigBang Core 核心钱包转发这部分交易，后者收到之后通过 P2P 网络接口向 BigBang Core 全网广播这些交易，出块节点收集这些交易，最终完成其打包区块上链的操作。

BigBang Core 客户端程序作为 IoT 设备 Firmware 的一部分，利用设备中央处理器和安全计算协处理器处理包括交易构建 / 解析、HASH、ED25519 签名 / 校验等 BigBang Core 交易相关计算。设备私钥被存放处理芯片安全区域，不可被直接读取。

LWS 和 BigBang Core 客户端之间的通过协议过程实现客户端 UTXO 列表同步和实时更新，LWS 响应 BigBang Core 客户端发送交易请求，通过连接的网络节点将交易广播全网。

IoT 终端发送一笔交易与 LWS 完整的交互流程如下：

Service Req/Reply： BigBang Core 客户端发起服务请求，传递协议版本、钱包地址和所需分支等信息；LWS 在可提供服务的前提下，返回用于构造 APIKey 的数据和所支持分支列表；APIKey 被用于后续消息签名。

Sync Req/Reply： BigBang Core 客户端发起同步请求，传递目前记录 UTXO 列表 HASH；LWS 将 UTXO 列表 HASH 进行对比，在判定客户端失步的情况下，推送对应 UTXO 列表。

Update UTXO： 在 Block/Tx 状态更新（新块产生、交易广播）时，网络节点会通过 Socket API 通知 LWS，LWS 在筛选过滤后将 UTXO 的状态变化推送至对于 BigBang Core 客户端。

SendTx Req/Reply： BigBang Core 客户端根据同步的 UTXO 列表构造交易并进行签名，通过 LWS 将交易广播全网；LWS 返回执行状态。

6.3 分布式超级节点

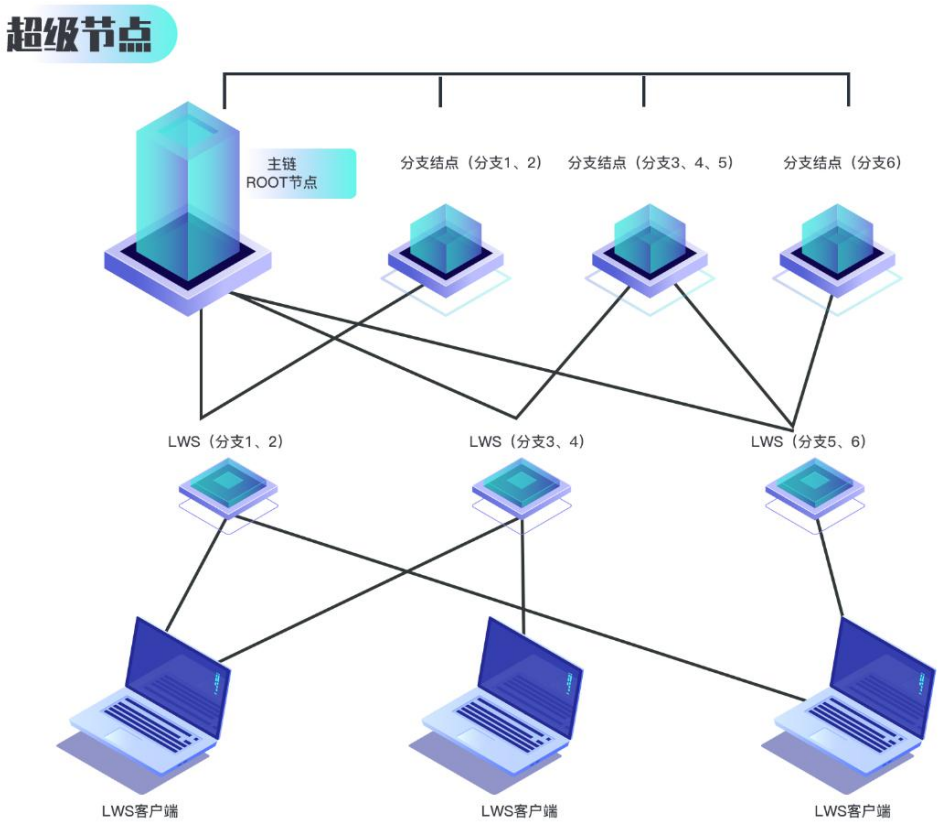
6.3.1 方案主要目标

分布式超级节点的主要目标在于为主链的 EDPoS 节点解决扩展性问题。BigBang Core 作为多支链的区块系统，参与 EDPoS 的节点需要为所有有价值的支链进行同步出块。随着应用支链数量增加，出块节点的负担会日趋加重，单服务器无法满足系统本身可扩展性要求，于是分布式超级节点是解决这一问题的有效方案。

6.3.2 方案介绍

在分布式超级节点方案中，根据业务不同，主要角色分为 ROOT 节点服务器和分支节点服务器。

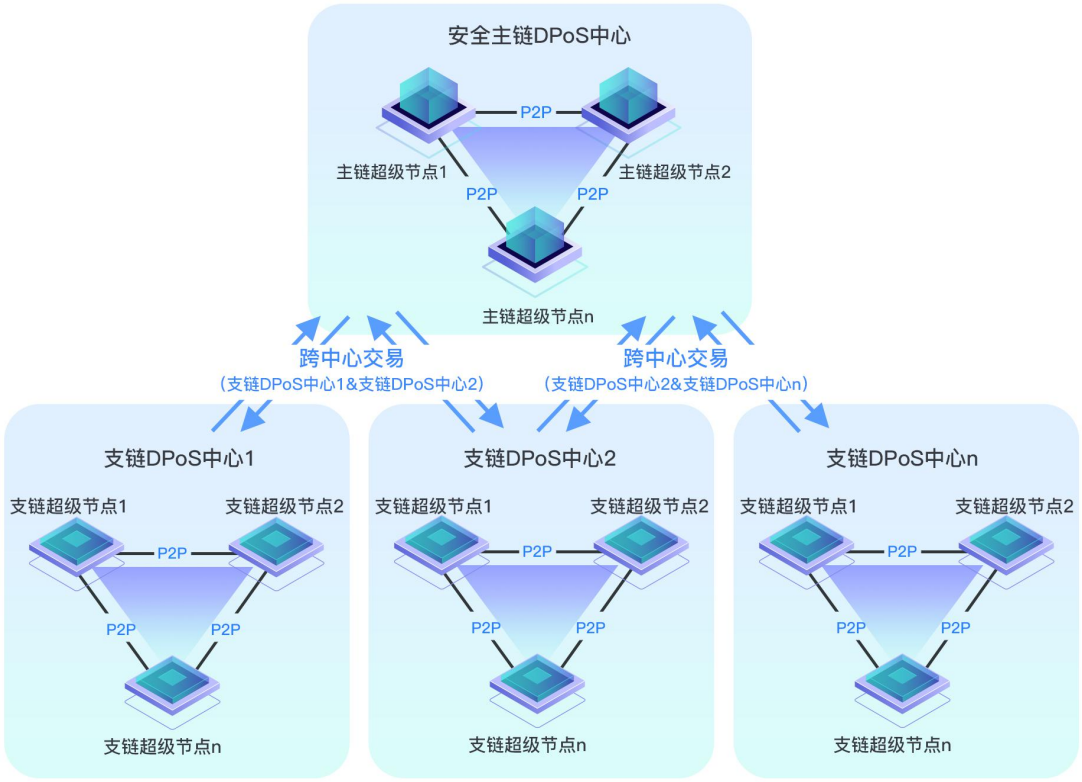
ROOT 节点服务器主要负责共识协商、主链出块、主链数据管理、分支节点管理等工作；ROOT 节点服务器保存有出块签名密钥的相关数据，用于共识协商及主链出块使用；ROOT 节点服务器接入 BigBang Core 网络，一方面进行安全主链上的 EDPoS 协商，另一方面和超级节点内的其它分支节点交互数据；ROOT 节点服务器与其它超级节点的 ROOT 节点服务器通过 P2P(Socket API)连接，或者与普通节点通过 P2P(Socket API)连接。ROOT 节点服务器只存储检验安全主链的数据，应用分支数据由分支节点服务器处理。分支节点服务器和 ROOT 节点服务器通过 Socket API 或 MQ 方式连接，交互安全信标、管理数据等信息。分支节点服务器则专门用于组织应用支链的区块数据，每台分支节点服务器只负责一条或多条分支链数据，并且不能有两台分支节点服务器负责相同的分支链数据。



分支节点服务器通过 Socket API 或 MQ 方式与 ROOT 节点服务器连接，接收处理安全信标和管理数据。分支节点服务器通过 P2P(Socket API)与其它超级节点的对应分支节点连接（ROOT 节点服务器通过与对端 ROOT 节点服务器握手协商时获得对端分支节点的连接地址，并将连接地址分发给分支节点）。当 EDPoS 协商当前 ROOT 节点被选中为出块节点时，ROOT 节点服务器构造主链新块，并把支链出块上下文数据推送给各分支节点服器分支节点服务器构造各自支链新块数据并签名后（分支节点保存有出块签名密钥的相关数据），由分支节点自己广播全网。

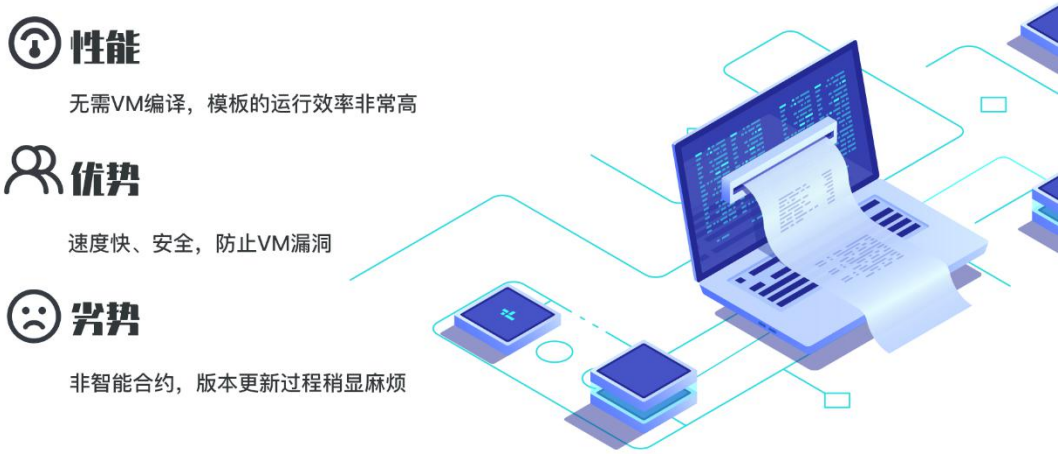
6.3.3 级联方案

当应用支链负载过大，需要进更一步的升级扩展时，分布式超级节点可以通过级联方式构建支链节点服务器集群，用于分流庞大的应用业务负载，如下图所示：



07. BigBang Core 跨链模板和电商模板

BigBang Core 跨链模板和电商模板的实现方式，由于无需 VM 编译，所以相较智能合约和脚本来说，模板的运行效率非常高。同时 BigBang Core 交易运行速度快、安全，没有 VM 一样的可被攻击的漏洞，防止因为 VM 漏洞导致链上代币被盗或归零。但是非智能合约，版本更新过程稍显麻烦，程序发布后，需要同步更新到每个客户端，后期会增加模板自动更新模块。



7.1 跨链交易模板

1） 创建交易模板，包含双方钱包地址和两个参与交易的支链 hash，以及在各自支链上锁定的块高，通过这些数据生成一个交易的模板地址，如果这些数据的任何一个值发生改变，则这个模板地址也会发生改变。该地址可以用于接收任何用户转入的 Token，但是在锁定高度内，要将该模板地址下的 Token 转出，就需要双方的签名数据才能完成。在锁定块高之后，参与交易的双方可以把自己所在的支链上的 Token 转走。

2) 对跨连交易模板地址进行签名，得到签名数据。

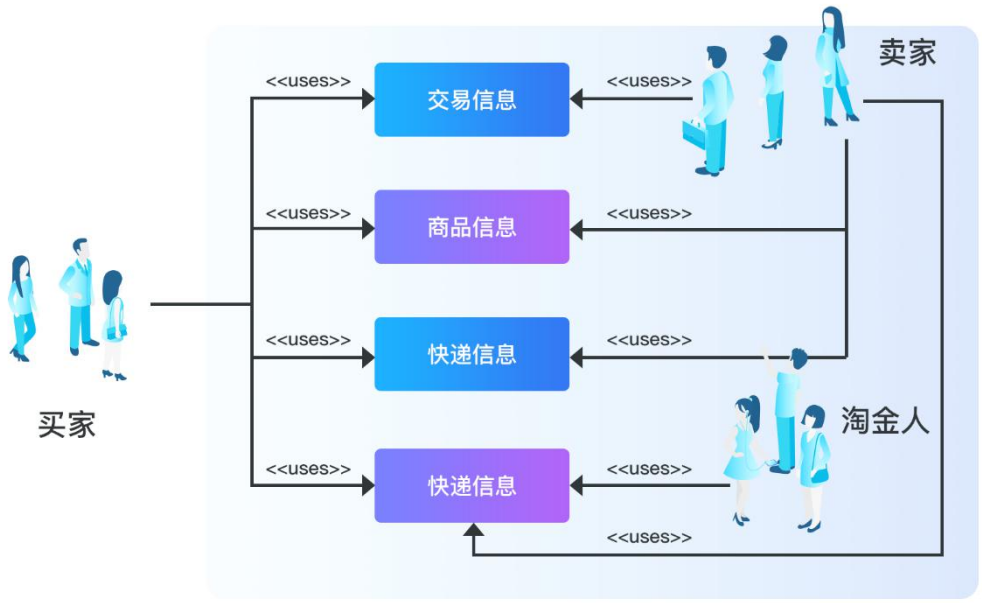
3) 交易双方可以通过对方提供的公钥验证签名数据是否正确，确认该签名数据是否是相应跨链交易模板的签名。要求锁定区块低的用户先将自己的签名结果发送给对方。

4) 通过对方提供的签名数据和自己的签名数据，就可以将跨连交易模板中的 Token 转走，此时需要该用户的钱包处于开启状态。

5) 如果你将交易发出后没有在“交易者优先打包高度”内转走 Token，那么第三方会介入打包过程，第三方可以是普通节点或超级节点，可以通过模板中的 From 与 To 地址以及签名数据帮助打包并获得第三方打包奖励。

6）注：跨链交易模板是一次性使用模板，对代码进行进一步的升级，可以修改成为可重复使用的跨连交易模板。

7.2 电商支付交易模板



交易流程如下：

a.卖方销售商品，买方购买商品，去中心化点对点交易，买方拒绝付款的交易 Token 进入资金池。

b.交易费用包括交易金额和安全金额，这两个金额都可以自定义，安全金额用于安全支付场景，用以应对卖方收到 Token 后不发货或买方恶意将 Token 转入资金池的情况发生。

c.交易过程可能会发生的三种情况：交易正常完成、到达块高后买方未确认、买方不满意。

正常完成：

买方满意商品，将确认信息（签名数据）发送给卖方，卖方利用签名信息进行提款操作。

到达锁定块高未确认：

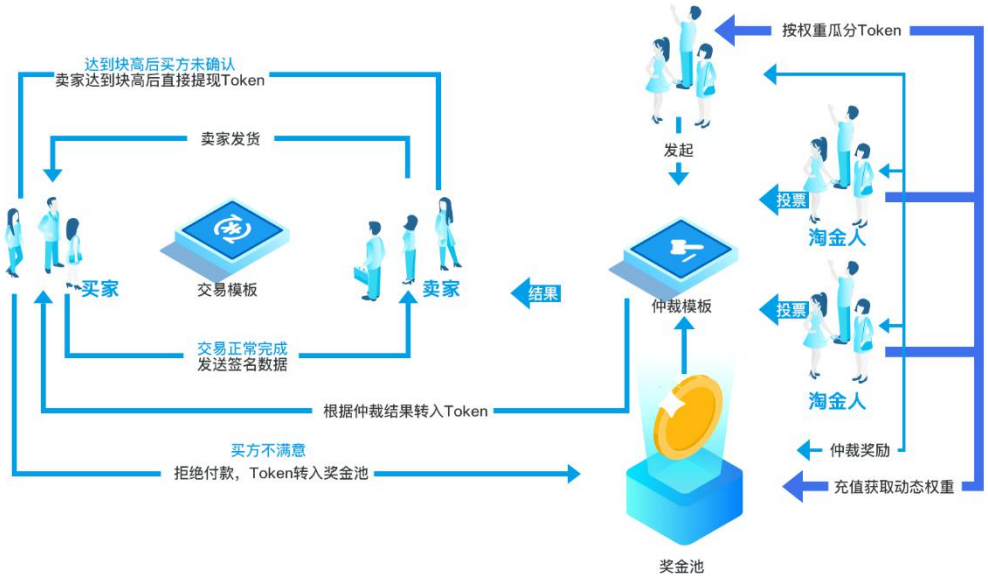
卖方不需要获得买家签名数据，直接提现。

买方有异议：

买方可将 Token 转入资金池，卖方收不到 Token。

d. 如果资金由于节点作恶故意转入资金池，那么可以由第三方“仲裁”方发起仲裁模板，通过节点投票来决定胜方与败方，将 Token 转给胜方的同时，一定比例的 Token 会奖励给投票者与发起者。

e. 一个淘金周期内，淘金人可以向资金池充值，获得动态权重，一个淘金周期结束后可以按权重分享资金池中的 Token。



交易原理如下：

a.买卖双方创建一个交易模板，制定交易规则。

b.去中心化交易，锁定块高内买家只能选择将签名数据发送卖家，或将 Token 转入资金池，锁定块高后卖方可自由提款。

c.为了安全，设置安全金额，买卖双方抵押相同数量的 Token 用于保证交易的大概率正常结束。比如一个商品 S 的价格为 100Big，卖方可设置抵押安全金额为 10Big，在模板创建后卖方需要发出商品并支付 10Big 的抵押金，这时候买方需要支付 110Big 去购买商品 S，交易结束后，买方获得商品 S 和 10Big，卖方获得 110Big。

应用举例：

IPFS 存储市场

卖家：提供 IPFS 存储空间；

买家：购买 IPFS 存储空间；

挑战方：由任何节点在挑战周期内随时挑战（时空证明）；

淘金人：所有往资金池中充入 Token 的参与者。

电子商务市场

卖家：提供商品；

买家：购买商品；

仲裁者：由任何节点在资金允许回撤高度内发起仲裁；

淘金人：所有往资金池中充入 Token 的参与者。

7.3 奖金池模板

在一个分红周期中，淘金人往资金池转入 Token，根据充值的多少获得动态权重，随着电商交易支付模板中产生的违约交易的增多，进入资金池的违约金也越来越多。在一个分红周期结束时，淘金人根据各自的权重，瓜分该资金池中的Token。系统中可以有多个资金池。随时都可以充值Token，随时都有违约金进入，随时都有分红，只是可能处于不同的分红周期。

其原理如下：

a.任何人都可以创建资金池模板，设定分红周期，如果分红周期相同则模板地址相同。

b.一个分红周期分为三个阶段——充值计算权重、违约金进入、分红。

c.淘金人往资金池转入 Token，获得动态权重，动态权重根据某一个分红周期中的充值阶段进入资金池的 Token 的总量进行动态计算。

d.分红周期结束，根据动态权重，瓜分 Token。瓜分 Token 的时候需要提供自己的收益证明（默克尔树证明）和自己抵押的 Token 数量。

e.下一个周期开始，淘金人需要重新充值以便获得动态权重。

08. 后记

本文针对 BigBang Core 的总体技术框架进行了基本阐述，技术的发展日新月异，BigBang Core 也会根据行业整体的技术进步和实际的项目需求不断进行迭代，包括本文档在内的相关技术文档也会随时保持更新。在今后的项目发展过程中，BigBang Core 将先以物联网为出发点，致力于解决当前物联网所面临的一系列问题，夯实 BigBang Core 在物联网行业的产业基础。同时也将向其他领域进行探索和拓展，不断打造 BigBang Core 在诸多领域的应用生态。

09. 术语解释

树状区块链：以主链为根，通过分叉的方式不断生成支链为枝叶的多重树状区块结构。

分支：在 BigBang Core 中，任何一条链均可称之为“分支”，每一条分支均有一个单独的分支标识。

安全主链：BigBang Core 树状结构当中的主链（或者称为“根（root）”），所有的应用支链均为其“后代”。主要用于记录主链 Token 转移及安全共识时协商的数据。

应用支链：BigBang Core 树状结构中的支链（Branch），所有的应用支链最终都可追溯至安全主链。主要用于存储应用方的业务数据，是实现区块链系统拓展的重要组成部分。

分叉 ：在区块链系统中，通过规则改变，使得新旧规则的区块不兼容，从而形成以规则升级前的区块为分叉点，两条独立运行的链。

锚定区块：用于标识新交易自该区块之后都可视为有效，用户在创建交易时进行指定。如果该区块之后有多个应用支链，则该交易对该区块之后的多个支链均有效。

共识机制：在分布式系统当中，为了保持数据在各个不可靠的节点间的一致性，使节点之间达成数据写入提案的过程。

支链区块：应用支链用于记录当前链条公钥地址交易和部分过程模板地址交易的区块。

子块：应用支链独有的区块类型，在两个支链区块之间生成，间隔为 2 秒，专门用于记录当前应用支链下的 IoT 即时业务交易。

拜占庭容错原理：是一种面向拜占庭问题的容错算法，解决的是在网络通信可靠，但节点可能故障情况下节点之间如何达成共识。

VSS：Verifiifiable Secret Sharing，可验证的秘密共享。是一种用于数据安全存储、传输及合法应用的安全协议。由秘密份额的分配算法和秘密的恢复算法两种算法构成。

EDPoS：Extensible Delegated Proof of Stake，可扩展的委任权益证明。是一种类似董事会结构的区块链共识机制。其基本特点为社区每一个持币人进行投票，然后选出特定数量的见证人，由见证人负责交易的确认和区块的生成。

CPoW：Continuity Proof of Work，可持续的工作量证明机制。是一种通过算力竞争的方式夺取区块链记账权的共识机制，最早被中本聪应用在比特币系统当中。

数据节点：运行 BigBang Core 核心节点程序，但是并不负责出块，可通过配置文件配置同步特定支链的区块数据，并通过相应接口为轻客户端提供数据请求服务。