

虚空链白皮书

要点

- 1、这是一个区块链操作系统
- 2、提供帐户，身份验证，数据库，异步通信以及在数以百计的CPU或群集上的程序调度

当前区块链平台存在的问题

- 1、能否支撑百万级用户访问量
- 2、是否免费服务，找到增值服务收费点
- 3、平滑升级避免bug
- 4、低延时的用户体验
- 5、串行能力
- 6、并行能力

委托权益证明机制（DPOS）共识算法

插播一段BTS的DPOS

DPOS是由Dan Larimer发明的。

首先搞清楚两个概念：权益所有人就是股东，授权代表（就是委派见证人）就是董事会成员，创造区块是委派见证人的事情，跟股东没关系。

1、选举机制

在BTS网络中，存在一个24小时不停工作的股东大会。每个股东选举一名授权代表，代表自己行使生产区块的权利（股东可以在自己的钱包的状态指示器看到授权代表的状态）。授权代表需要尽职尽责，否则股东可以取消授权，转而授权其他代表。

2、生产区块

票数最多的前100名代表有生产区块的权利。生产区块的排序是按照之前确定的一张表轮流进行。每名代表分配到一个时间段开始生产。代表没有修改交易详细内容的权利，比如发起人、接收人、或者余额。如果在自己的时间段内，该名代表因为各种原因比如网络无法生产区块或者把交易信息放入区块，那么下一位代表创造出的区块，将会变成两倍大的区块，包括前一位代表遗漏的交易信息，而确认时间将会延长至20秒而不是10秒。出错的代表就会被投不信任票，被从生产状态中清除出去。100名代表的数量是可以进行变化的，这样可以调整网络开销。

3、避免分叉

作为代表为了避免损失挖矿费，一定会力争100%在线的，如果发生上面提到的代表错过

自己时间段的情况，交易确认时间就会变长，如果发现交易者发现10个区块中有5个错过了生产时间段，基本可以确定自己是处于分叉区块链上，应该停止交易，等待分叉回归。

还有一种机制避免分叉，就是相邻区块的生产代表之间是有直接连接的，这种连接是和生成区块的报酬有关系的。

4、去中心化的怀疑

有人怀疑挖矿权集中于100人是否还是去中心化？（v神2017.7.28在深圳的发言就提到VOIDCHAIN不属于去中心化）实际上这100百人是权利平等的，无法将手中权利集中于更少的节点。多名代表可以联合作恶，但是51名代表联合作恶的可能性还是非常小的（比起比特币挖矿权的集中要好很多了）。退一步即使他们能够联合作恶，但是造成的影响也不大，因为很快就会被其他代表识别纠正进行分叉并无视作恶攻击者的区块链。

VOIDCHAIN中的DPOS

基本相同的是，代币拥有者选举代理，不同的是每一轮是前21名代表当选挖矿者，前20名是系统自动排序，第21名是按所得投票数目对应概率选出。所选择的生产者会根据从块时间导出的伪随机数进行混合。以便保证出块者之间的连接尽量平衡。

同一时间，只有一个生产者，频率是每3秒一个区块，就是63秒一轮。

如果某代表错过出块时间，而且24小时内也没有出块，系统认为此代表已失联，会取消挖矿代表权利。

1、交易确认

一般情况下，交易确认完成是需要15个确认，最多就是45秒。假如一个节点观察网络中出现连续2个区块丢失，则认为自己95%可能性在区块链的分叉分支上。如果出现3个连续的丢块以后，则有99%的可能性在一条分叉出来的区块链上。为避免分叉出现，可以生成一个预测模型，将节点丢失的信息，最近的参与率以及其他因素形成策略，快速地警告用户。处理分叉15/21确认是比较简单有效的方法。

2、交易证明（TaPoS）

基于交易的权益证明，要求每个交易都包括最近的区块头的哈希。这个可以防止分叉链上出现大量交易，系统能够感知用户在分叉链上。

账户

账户应该是DAPP的所有者，2-32个字符可作为选择，账户应该预留代币支付管理费，帐户名称还支持命名空间，因此帐户@domain的所有者是唯一可以创建帐户@user.domain的用户。

1、消息处理

每个账户可以发送结构化消息给其他账户，并且可以定义接受处理脚本，也可以进行转发消息。每个账户有自己的数据库，而且只能被自己的程序访问。

2、基于角色的权限管理








权限管理就是检查事物的签名。

VOIDCHAIN软件提供了一个声明式权限管理系统，可以让帐户细粒度和高级别地控制谁在何时能够做什么。

特别的，身份认证和权限管理是和账户的应用程序逻辑分离的，这样能够更方便进行开发。账户还支持多级管理，就是实现附加其他账户私钥。满足现实环境中的组织情况的层次化权限结构。还支持同一账户的权限分离，比如无需提供私钥，让账户实现其他功能，比如社交等。

命名权限级别

使用账户进行命名权限级别，进行分权管理。每个命名权限级别定义一个权力，这个权力可以是其他帐户的密钥和(/或)命名权限级别组成的多签名检查的阈值。

| Authorities | | |
|---|--|---------|
| Signing | | |
|  | STM7xh66F5ZHyfN9u4rNZmTioBteZhvWdqDwaR2kRS5tBXeCjTr2z | |
| Owner | | |
|  | STM7iTj8quuiqX7aUH2WfYXfAqqTDbpLBFwxoCUzEeKE745mvBY41 | |
| Active | | |
|  | STM5DiwrKfp4ngW7fwCDej1Kd3efogY5GxsMKfkz3xi4AsNGYWU2D | |
| Posting | | |
|  | streemian | 1 33.3% |
|  | esteemapp | 1 33.3% |
|  | STM89kBiwpi5RBCBrgAFWd5p5uayTvv57B6Zb4oBenWx8ChjeLMTf | 1 33.3% |
| Threshold | | 1 33.3% |
| Memo | | |
|  | STM7S4xvQgdvuQ8SFAD8vzFcLskoUdLqzEPbudcXuyoku2iirwzt6v | |

命名权限级别

所有帐户都有一个可以做所有事情的“owner”权限组，和一个除了更改所有者组之外可以执行所有操作的“active”权限组。所有其他权限组均派生自“active”权限组。

命名消息处理程序组：每个帐户将自己的消息处理程序以命名嵌套的方式予以组织。这些命名的消息处理程序组可以通过配置其权限级别被其他帐户引用。

权限映射：系统还允许每个帐户定义任何帐户的命名消息处理程序组与其自己的命名权

限级别之间的映射。

权限评估的只读属性：权限评估过程中对权限是只读的，这就意味着权限评估可以并行，也意味着可以快速验证权限，而不需要考虑启动可能回滚的昂贵的应用程序逻辑。最后，这意味着当挂起的事务继续执行时，事务权限的评估可以继续执行，而无需从头重新执行。

系统中的强制延迟：事物应用之前必须等待一段比较小的时间段。在此期间，这些消息可以被取消，这就是强制延迟。强制延迟时间长短取决于操作的重要程度，越重要时间越长。

VOIDCHAIN软件为用户提供了一种在密钥被盗时恢复其帐户控制的方法。

帐户所有者可以使用在过去30天内活动的任何其批准的帐户恢复合作伙伴的密钥，在其帐户恢复合作伙伴的允许后，重置其帐户上的所有者密钥。在没有帐户所有者的配合情况下，帐户恢复合作伙伴无法重置其帐户的控制权。

对于攻击帐户的黑客而言，由于其已经“控制”该帐户，因此尝试执行恢复过程没有任何收获。此外，如果他们的确进行恢复的过程，那么恢复合作伙伴可能需要身份认证和多因素认证（如电话和电子邮件）。这或者会暴露黑客的身份，或者黑客在恢复过程中毫无所得。

应用程序的确定性并行执行

1、通信延迟优化

延迟时间是一个帐户将消息发送到另一个帐户并收到响应所需的时间。

VOIDCHAIN软件的目标是使两个帐户能够在单个区块内来回交换消息，而不必在每个消息之间等待3秒。为了实现这一点，VOIDCHAIN软件将每个区块分为周期（cycle）。每个周期分为线程（thread），每个线程包含事务列表。每个事务包含一组要传递的消息。该结构可以被可视化为树，其中各层依据其特性被顺序或者并行地进行处理。

在一个周期中生成的交易可以在任何后续周期或区块中传送。区块生成器将不断把周期添加到区块中，直到最长的区块时间间隔达到，或者没有新的可传送交易生成。

2、区块链技术组件是模块化的，也就是每个应用只调用自己使用的模块。

3、自主最优任务安排：每个区块的生产者对交易的复杂度都有一个自己主观的判断。具体到网络层，每一个区块生产者使用自己的算法和度量衡量交易占用的网络资源，有人认为该笔交易消耗大量资源可以选择在自己的块中拒绝该笔交易，也会有人不认可这种观点，会收录该笔交易到自己的块中。因此只要被任意区块收录的交易最终就会被全网认可，不过这种情况下需要花费1分钟才能将交易广播出去。

令牌（代币）模型和资源使用

资源限制：

保持系统可靠运行的资源，包括以下三方面

1. 带宽和日志存储(磁盘);
2. 计算和计算积压(CPU);
3. 状态存储器(RAM)。

日志存在于全节点中，主要用于重构所有应用程序的状态。

可计债务(The computational debt)是指通过对消息日志重新生成状态的计算消耗。如果可计债务增长太大，就有必要对区块链的当前状态进行拍照，并抛弃区块链的历史状态。如果计算债务增长过快，区块链将使用6个月的时间来重放1年的交易。因此，对可计债务进行精心管理是至关重要的。

存储状态是从应用程序逻辑访问的数据，比如交易信息。

授权能力

系统支持VOIDCHAIN账户所有人将自己闲置或者未完全使用的带宽转让或者转租他人。区块链生产者能够识别这种行为执行其决定为其分配带宽。

VOIDCHAIN系统的主要优点之一是，应用程序可用的带宽完全独立于任何令牌价格。

代币价格的波动与可用带宽没有关系，不过代币价格升高将影响一个区块生成者能够购买的带宽、存储和计算量，反过来可以利用上升的代币价值来提高网络性能。

状态存储成本

应用程序的开发者必须一直持有代币直到带宽和计算资源被委托状态被删除。每个账户都需要一定的存储空间，因此也就必须持有一定数量的代币。

块奖励

每生成一个区块，系统都会奖励出块者一定数量新的代币，这些新代币就是发行机制。出块者获得的代币数量由所有区块生成者所公布的期望报酬的中位数决定。系统可以通过设置来限制获得的代币的上限。

中位数（又称中值，英语：Median），统计学中的专有名词，代表一个样本、种群或概率分布中的一个数值，其可将数值集合划分为相等的上下两部分。对于有限的数集，可以通过把所有观察值高低排序后找出正中间的一个作为中位数。如果观察值有偶数个，通常取最中间的两个数值的平均数作为中位数。（百度百科）

社区福利应用

除了成为挖矿者之外，还可以选择成为3个智能合约（应用程序）的投票支持者。系统为这些智能合约设置一定比例收益（需要减去挖矿奖励），换句话说，这些智能合约根据自己从股东获得的选票数获得代币。如果不能获得信任，就会被新当选的智能合约替换掉。

治理

治理的权利归代币所有者，就是股东。所有对区块链的修改必须得到股东投票同意，挖矿者具有执行和监督权限，比如冻结账户，更新程序，提出对底层协议的变更。如果挖矿者拒不执行投票结果，那么，非挖矿节点（非生产的全节点验证器(交换器等)）将会否定挖矿者的决议。

冻结账户

为了防止智能合约的异常行为或不可预知行为，区块生成者有权选择生成的区块中包括哪些交易，从而使他们有冻结账户的能力。系统通过对一个账户17 / 21活跃区块生成者的投票实现授权这种能力。如果生成者滥用权力，他们可以被淘汰，账户将被解冻。

改变帐户代码

应用程序发生异常而且不可阻止时，经过投票授权的挖矿者（17 / 21被选中的区块生成者的投票）有权直接修改应用程序的代码。

宪法

宪法是VOIDCHAIN中的大法，主要定义当代码无法执行的用户之间的义务，制定了一些公认的准则，确立了法律和司法权的界限。每一笔在网络中广播的交易都必须在其签名信息中包含宪法的哈希值，以明确约束合约签名者。宪法还定义代码之外的人类的意图，当系统出现错误时，这个意图可以分清楚是bug还是系统特性，并且判断社区对此的修复措施是否正确。

升级协议和宪法

对宪法或协议进行变更，需要完成以下步骤：

1. 区块生产者（原文miner/delegate/witness，因此没有译作矿工）提交一个宪法变更动议，并获得17/21以上的赞成票；
2. 区块生产者将17/21以上的赞成票维持连续30天；
3. 要求所有用户都使用新宪法的哈希值确认交易；
4. 区块生产者采用修改源代码的方式反映宪法变更，使用git提交的哈希值将变更提交到区块链上；
5. 区块生产者继续将17/21以上的赞成票维持连续30天；
6. 变更的代码7天后生效，源代码修改通过后，将有1周的时间来对所有节点的进行升级；
7. 所有没有升级为新代码的节点将自动关闭。

根据VOIDCHAIN操作系统的默认配置，更新区块链来添加新功能这一进程需要2到3个月时间，而修复那些不需要更改宪法的非关键性漏洞需要1到2个月时间。

特殊情况，当出现损害用户利益的有害漏洞或安全漏洞时，区块生产者可以加速宪法变更过程。

脚本&虚拟机

VOIDCHAIN支持任何语言开发的脚本和虚拟机都可以通过api进行集成。

模式定义的消息

账户之间发送的消息都是由共识状态的模式定义的。这种架构允许消息在二进制和JSON格式之间的无缝转换。

模式定义的数据库

数据库状态也是类似上面模式的定义。

将身份验证与应用程序分离

为降低资源消耗，VOIDCHAIN操作系统将身份验证和应用程序进行分离，验证逻辑分为三个阶段：

1. 确认消息在内部是一致的；内部一致就无需访问区块链状态，对外就是只读的。可以并行进行。
2. 确认所有的前置条件都是有效的；同样前置条件也是只读的，也可以并行。

3.修改应用程序状态。这里就是写入操作，串行对每个程序进行处理。

交易进入区块之前是需要身份验证的，但交易一旦被包含在区块链中，就不再需要执行身份验证操作了。

虚拟机独立架构

当前正在评估的虚拟机：

1、Wren

Wren(<http://wren.io> (<https://link.jianshu.com?t=http://wren.io>))是一种小型的、快速的、基于类别的编程语言。之所以选择Wren语言和虚拟机，是因为它的短小精悍、易于文档记录 and 理解的代码库。它还具有非常好的性能，并且可以很容易地嵌入C++应用程序中。

2、Web组件(WASM)

WASM是构建高性能Web应用程序的新兴Web标准，通过少量适配就可以被明确定义和沙箱化。WASM的好处在于业界广泛支持，因此可以用熟悉的语言开发开发智能合约，例如C或C++。

3、以太坊虚拟机(EVM)

这个虚拟机已经被用于大多数现有的智能合约，并且可以在VOIDCHAIN系统区块链上使用。可以想象，在VOIDCHAIN操作系统区块链上，EVM合约可以在内部沙箱中运行，只需要少量适配就可以与其他VOIDCHAIN应用程序交互。

跨链交互

VOIDCHAIN操作系统支持促进区块链间的跨链交互，这是通过简化消息存在证明和消息序列证明来实现的。但是对于开发人员隐藏跨链交互和验证的细节。



VOIDCHAIN的跨链交互

用于轻客户端验证的默克尔证明(LCV)

对于客户端来说，要轻，也就是说不会关心所有的交易，更多关心与自己相关的交易，同样对于交易所而言，将轻量级的默克尔存款证明维护在自己的链中，也就无需依赖全节点矿工，而后者将会节省更多的资源。

因此，LCV的目标之一就是产生相对轻量级的交易存在证明，目标之二是该证明能被其他人通过跟踪一个轻量级数据集进行验证。换句话说，目的就是证明一个特定的交易包括在一个特定的区块中，并且这个区块是被包含在已经验证的特定区块链历史中。

那么来看看比特币怎么做的。

比特币的SPV是所有节点读取区块头数据完整记录，而区块头数据每年增长4MB，假设每秒产生10笔交易，一个有效的证明需要512 bytes，出块时间为10分钟，也就是每个区块中包括6000个哈希值。对于VOIDCHAIN来说出块时间为3秒的区块链来说是最低效的。

VOIDCHAIN操作系统的LCV只需要验证包含某个交易的区块头数据，使用哈希链表架构这种方式，数据集大小可以保持在1024 bytes内，即可证明任何一个交易是否存在。这是基于验证节点保留着前一天的所有区块头数据（2 MB大小），然后证明这些交易只需要200 bytes大小的证明数据。

还可以根据应用场合进一步优化，如果需要根据所有区块头数据，那么也只有420 MB/年这样的数据量。一般情况可以根据使用的区块头数据时间和数量之间找到合适的平衡点。同样的一个区块链可以“懒惰地”只记录过去数据的哈希值作为之前数据的证据。

在链与链之间经过一定密度的相互关联之后。他们将会变得越来越高效。一条链可能会包含另外一条链的全部历史记录，那么就不再需要互相证明。从性能的角度来说，这将极大地减少链间互相证明操作的频率。

跨链通信延迟和完成证明

跨链通信时，矿工必须等待其他区块链不可逆地确认之后才会接受其为有效的输入。但是使用VOIDCHAIN系统软件，依靠DPOS和17/21，确认时间只有45秒。这样就会发生某个链上的矿工还没有完全确认完成，VOIDCHAIN只有等待的情况。VOIDCHAIN操作

系统通过分配一个顺序的标识编号给每一笔到达账户的信息来避免这种情况（证明历史交易数据之间没有缺失），用户可以使用这些标号来证明所有给这个账号的消息已经被处理并且是被按顺序处理的。

(/apps/download?utm_source=nbc)