V.E.L.A.S.™

虚拟扩展学习自治系统（VLX）

基于 AI 的 DPoS 区块链

非募资或营销目的 作者：Alexander Alexandrovych Alexandrov

官方网站：Velas.com

特别鸣谢： Alex Lightman、Jason Butcher、Mondas、Pr0d1gy、Konstantin、Timur、 Orhun、Sina、Trevor、Albert、Jag、Farkhad 和 Ruslan

摘要

Velas 是一个会自主学习和优化的区块链平台，用于安全、可交互操作、可扩展性极好的交易和智能合约。Velas 使用基于 AI 的 DPoS（AIDPoS）共识来确保区块链的高容量交易处理安全，同时不减弱去中心化特性、稳定性和安全性。通过使用基于 AI 的 DPoS 共识算法，消除了人性腐败问题，从而形成了一个容错系统，防止了现有大多数公链项目的主要问题，如 51% 的攻击和无抵押问题。

2019 年 7 月 1 日

免责声明和警告声明：*此 Velas 技术白皮书仅供参考。除历史事实陈述外，本文件中的所有陈述均为前瞻性陈述。这些声明只是预测，反映了作者当前对未来事件的信念和期望，并且是基于假设的，随时可能受到风险、不确定性和变化的影响。我们的经营环境变化迅速，新的风险不时出现。鉴于这些风险和不确定性，请注意不要依赖这些前瞻性声明。实际结果、业绩或事件可能与这些前瞻性声明中包含的结果、业绩或事件存在重大差异。本文所载的作者所作的任何前瞻性声明仅在其作出之日起生效，而作者没有义务，并且明确否认任何义务，更新或更改本文所载的前瞻性声明，无论是由于新的通知事件、后续事件或其他。作者不保证本白皮书的准确性或得出的结论，本白皮书按原样提供。作者未作出并明确否认所有明示、暗示、法定或其他任何形式的陈述和保证，包括但不限于：（i）对适销性、特定用途适用性、适合性、用途、所有权或非侵权性的保证；（ii）本白皮书的内容不存在任何错误；（iii）此类内容不会侵犯第三方权利。对于因使用、引用或依赖本白皮书或本白皮书所含任何内容而导致的任何类型的损害，作者不承担任何责任，即使已告知此类损害的可能性。在任何情况下，对于使用、提及或依赖本白皮书的任何直接或间接、后果性、补偿性、偶然性、实际性、惩戒性、惩罚性或特殊性的任何损害、损失、责任、成本或任何形式的费用，作者均不向任何人或实体承担任何责任，包括但不限于任何业务损失、收入、利润、数据、使用、商誉或其他无形损失。本文件不构成出售邀约、购买邀约或 Velas 代币（VLX）或任何其他产品或服务的建议，无论本文件中是否提及此类代币、产品或服务（如有的话）。此外，本文件中的任何内容均不旨在提供法律、税务或收购建议，本文件中的任何内容均不应被解释为接收、持有、购买或出售 VLX 或参与与 VLX 有关的任何交易（包括任何奖励交易）的建议。作者并不表示本文件中讨论的 VLX、产品或服务（如有的话）适用于任何特定人员。您仅负责确定涉及 VLX 的任何交易是否适合您。*

介绍

Velas 区块链平台采用 AI 优化的神经网络来增强其共识算法。Velas 的目的是解决和修正大多数现有区块链所面临的问题和挑战。

神经网络用于作为计算节点运行和区块形成时间的奖励。矩阵计算服务器（神经网络权重）位于网络成员的节点上，用于接收奖励。与比特币矿工类似，节点需要有大量的算力来计算矩阵。例如，安装功能强大的专用显卡（GPU）。

为了训练神经网络，在训练前采用遗传算法，利用误差反向传播的方法，求出目标函数的最小值。

网络训练算法

遗传算法：

1. 创建一个样本：一个权重随机（基因）的矩阵
2. 竞争：获得目标函数的最小值
3. 选择：按误差排列样本，误差最小的获得胜利。
4. 复制：矩阵元素或基因的交换，从前两个最成功的以 50/50 交换。
5. 重复该循环，直到达到 70% 的概率。

每个节点从区块链数据形成自己的数据集（训练样本）。在这些数据上，对遗传算法进行了分层训练。每下一层都在学习前一层：关于自动编码器原理的训练。

在预备训练后，采用反向传播法将权重降到最小。

网络必须在下一个区块周期之前进行训练。

选择最好的网络。

- 在形成新的区块循环之前，节点形成一个测试数据集。
- 在形成测试样例后，检查矩阵。
- 误差最小的矩阵获胜，进入下一个周期。
- 获胜矩阵的节点从网络接收 Velas（VLX）

销售和购买最有效样本的市场将不断进化，这将打开区块链网络维护算法的发展，并奖励基于 Velas 的其他系统。

在基于量子技术的超高速计算系统时代，在短时间内可以实现接近理想状态的

神经网络，在商用量子计算机中对其进行训练，而随着时间的推移只需增加一些新功能。

通过解决区块链网络维护任务，各 AIDPoS 节点实际上为后续开发神经网络贡献了智能计算。这一切都是在不需要所选节点的运行者理解任何编程语言的情况下，就可以发生的。

网络版本推出计划

阶段 1（预 Alpha ）：

创建区块链系统结构、代币、4 个节点上的交易。节点将由网络组织者在 Pre-Alpha 阶段运行。钱包中的智能合约将允许所有 CPS 币（CoinPayments Coin）用户被 1:1 转换为 VLX（Velas）。创建代币或自定义数字资产。

阶段 2（Alpha）：

创建一个稳定的系统，在 10 个节点上部署 Velas，从网络组织者 4 个服务器测试 AI 。引入支持多币种的钱包容器，支持公共和私有发送功能。

阶段 3（Beta）：

添加 AI 填充测试节点，与服务器端 AI 竞争。进一步扩展支持多币种的钱包容器系统，支持所有主流加密数字货币，支持发送、接收和智能合约等功能。

阶段 4（候选发布）：

将 AI 集成到现有 advisor 节点中，其数量由 AI 逻辑设置。

第 5 阶段（发布）：

启动系统的全部功能。用户可以下载神经网络工具包，并使用可视化工具为他们的项目进行优化，并根据贡献获得奖励。代币节点将使用经过训练的神经元网络在大多数环境下进行预备训练，并允许轻松设置和维护代币节点。

Velas 平台概述

术语和定义

- VelasCycle——有限时间段和区块数。每个 VelasCycle 由一个 SimpleBlock+ 一个 CycleBlock 组成；
- CycleBlock – 一个区块，其包含允许在当前 VelasCycle 中出块节点列表；
- SimpleBlock – 包括交易列表 – 不与 CycleBlock 混淆；
- NodeID – 每个节点都包含一个密钥和公钥。公钥是节点标识符，密钥用于生成 BlockSign；
- BlockSign – 区块创建者标识区块编号的签名；
- TxQuery – 查询或“特殊交易”。它必须广播到网络，以显示其在下一个 VelasCycle 内生产区块的意图。此交易将包括 NodeID。生成 TxQuery 需要 100,000 个 Velas 代币。

在 VelasCycle 结束时，每个节点必须通过以下方式定义下一个算法：

1. 从上一个 VelasCycle 收集所有 TxQueries；
2. 按 NodeIDs 对 TxQueries 列表按词典顺序排序；
3. 为下一个 VelasCycle 生成一个潜在节点列表；
4. 收集上一个 VelasCycle 的所有 BlockSign；
5. 用 BlockSign 做一棵默克尔树，这将产生一个 VelasSeed。VelasSeeds 用于同步 Velas 网络中所有节点之间的随机函数。算法具有确定性；
6. 利用一个 VelasCycle 的时间周期和区块时间，计算下一个 VelasCycle 的区块数。例如，VelasCycle – 20 小时，区块时间 – 1 秒；20 小时 *60 分钟 /1 秒 =72000 个区块每 VelasCycle；
7. VelasSeed 用于随机化函数；
8. 最后一步包括在下一个 VelasCycle 中调用随机函数，调用次数和下一 VelasCycle 中的区块一样，以此来同步所有节点；

出块节点的选择标准最初将完全基于抵押的 Vela 数量。因此，一个节点抵押越多代币，它越有可能被选为出块节点，并收到 Velas（VLX）回报。

例如，一个拥有 2,000,000 VLX 的节点被选中的概率是拥有 1,000,000 VLX 的节点的 2 倍。

请注意，大多数抵押 Velas 的节点需要抵押累积 51% 的代币。这是在 Velas 区块链上达成共识所需的最低限度。

VelasCycle 被允许跳过 SimpleBlocks。至少 51% 的区块需要在 VelasCycle 中进行验证。

达成共识的方法

AIDPoS 能够成功地解决现有备选方案的许多缺点和局限性。当网络开始 1:1 转换 CoinPayments Coin（CPS）到 Velas（VLX）时，2,000,000,000 枚代币被预挖。

1. 形成区块链时，所有代币在服务组织者之间共享。
2. 当形成新区块并结束循环区块时，将释放代币。
3. 发行的代币被奖励给出块者。

交易验证是通过在附属网络上“抵押”（持有）代币来执行的。Velas（VLX）的抵押建立了一个可信的验证人网络，该网络将处理和形成一个到链的交易区块。从本质上讲，正是所抵押的 Velas（VLX）数量使人们对区块链的当前状态达成共识。

参加者将获得易于使用的钱包软件来抵押代币。通过收取一段时间的网络费用来补偿验证人的抵押。抵押的越多，分配到的 Velas（VLX）越多。对于那些拥有专用 GPU 的参与者，将提供额外的软件供选择，以提升神经网络，并因其对 AI 的贡献和训练而获得奖励。因此，PoS 方法可以激励大量的长期投资，并结合正确的验证行为和回报，提供可靠的、规模可观的代币流，同时尽可能减少开销或对高级编程语言知识的要求。

Velas 人工直觉 DPoS 算法

人工直觉 DPoS（AIDPoS）用于确保 Velas 区块链安全。AIDPoS 试图提供一种可以替代常用的共识机制，比如比特币传统的 PoW，Peercoin 和 NXT 的 PoS 系统。

Velas AI

人工直觉是一系列用于识别一组数据中关系和模式的算法。网络可以调整输入，从而在不必重新设计输出标准的情况下产生最佳的可能结果。

Velas 系统的选定技术参数包括：

- 每秒交易数：>30,000；
- 每秒区块数：1 秒 -2 分钟，取决于 AI 算法的计算结果；

区块时间取决于网络负载（TPS）。如果网络每秒有许多交易，则区块时间将很短。如果网络没有交易，则区块时间将很长。在生成空块的情况下，它只包含没有主体的区块头。

AI 算法将从以下历史数据中得出：

- VelasNodes 数量
- 每个 VelasCycle 的交易数

VelasCycle 的特殊交易

神经算法将优化以下参数：

- Velas 节点网络
- 区块大小
- 区块时间
- 增加 TPS

优化后的参数将是抵押者的交易佣金总额。

Velas 奖励

奖励将提供给节点 / 区块生成者，用于主动正确地参与系统，并取决于获得的分数。算法更改是动态进行的，考虑到以下参数：

- 每 VelasCycle 时间
- 每区块时间
- 每区块包含的交易

节点分级评分算法。

使用重要性证明算法。之后用于优化的关键参数包括：

- 一个节点的交易数，并考虑其质量。假的交易会导致减分，而真实交易会得到加分奖励。
- 账户余额。额外的分数根据抵押的代币分配。Advisor 节点（区块生产商）在早期阶段预估至少需要 1,000,000 个 VLX。
- 在线时间。根据区块生成节点的总正常运行持续时间分配额外的分数。
- 区块生成事件。每个区块生成节点接收每个生成的区块分数。

如果某个节点由于任何原因（例如，由于网络问题，处理能力不足或运行时间不足）未生成区块，则会扣除该节点上一点分数。当形成一个区块时，也会生成验证人列表。根据收到的分数，验证人被添加到列表中。

上面指定的数据通过神经网络，然后我们接收 y' =f（x）目标函数的数据。每一个 y' 都会经过 softmax 层。在此之后，我们得到最大质量偏差的百分比。

区块生成的奖励取决于在神经网络中收到的贡献的百分比。

奖励代币

假设区块的总奖励为 100%，参与者将按照他们在神经网络中收到的分数所占百分比对总奖励进行分配。

人工直觉（AI）将基于线性回归模型。该模型采用随机方法训练。AI 基于两个模型。利用多维线性回归模型计算类的概率分布密度，贝叶斯分类器利用后验最大概率估计确定正确的决策。

由于在超复杂的环境中应用了大量的输入数据流，因此采用了基于遗传算法的 AI ，因为它能够比采用误差反向传播方法的标准神经网络更有效地进行多次处理。

遗传算法用于使用进化方法解决优化问题，即从各种最合适的解决方案中进行选择。它们不同于传统的优化方法，具有以下特性：

1. 他们处理问题参数的编码形式，而不是它们的值。
2. 基于一定的人数寻找解决方案。
3. 使用的是目标函数，而不是其导数。
4. 算法是随机的。

将遗传算法用于神经网络训练，作为反向传播误差法的替代方法。训练的目的是使成本函数最小化。此外，使用遗传算法可以避免局部极小值中的代价函数。

需要强调的是，误差的反向传播算法通常比遗传算法执行得更快，因为后者要扫描所有可能的结果。然而，梯度法并不总能得到预期的结果，这取决于起始点的选择。此外，误差反向传播方法的一个基本缺陷是局部最优中的“干扰”。这就是为什么遗传 AI 是一种更具创新性和前景的神经网络学习方法。

算法实现细节

确定节点等级的人工直觉算法和确定一个阶段中区块数的算法。

确定区块数量的 AI 算法：每个区块由既定的交易数组成。这个数字取决于区块链的使用强度。如果在区块形成期间，未确认的交易仍然存在，那么有必要通过减少区块之间的时间来增加这个阶段中的区块数量。

当形成一个新的区块周期时，有必要通过使用增长函数的行列式确定每个时期的标准差，来检查前面阶段中未确认的交易数量。接下来，我们通过计算偏差来计算块中应该有多少个交易。然后，我们计算出所需的区块数和出块时间。

确定节点等级的 AI 算法：节点区块生成者在确认出块时得到分数。如果一个节点由于各种原因(缺乏计算能力或网络问题)未形成区块，则从该节点中减去分数。当形成一个区块周期时，也会形成一个 advisor 列表。Advisor 将收到的分数添加到列表中。

人工直觉决定了投票名单中的评分。输入参数为：

a) 节点交易数

b) 帐户余额

c) 网络时间

d) 形成区块的分数

训练示例包括在前一个周期中出块时的错误、延迟和未确认交易。

AI 的结果应该是候选节点列表中的一个等级。这四个参数是输入数据，我们将根据输入数据对流量进行分类。

DNA 是神经网络 (矩阵) 的层，将在分离的节点上计算。可以添加更多的输入参数：

1. 在形成区块之后剩余的未确认交易数。
2. 一个参与交易检查的节点 (针对 51% 的攻击和双倍消耗的攻击的保护)
3. 当一个节点在一个 cycle-block 内时，真实的投票和交易检查速度。

AI 在 Velas 上的应用

在 Velas 平台上使用 AI 的目的是降低共识的成本。

Velas 平台顶部的 AI 框架：

1. 激励网络（节点）参与者在网络中的可靠性、活跃性，最大化相关分数 / 奖励。
2. 阻止关于错误交易的虚假消息，从而提高消息的质量和网络对攻击的抵抗力。
3. 形成每个阶段的计时，从而加速 TPS，减少一般的计算网络工作量。换言之，它是关于在高工作负载期间出块的动态时间，同时将出块任务分配给具有较高计算能力的节点。
4. 正确、最优地分配奖励。

防 51% 攻击：

我们使用 DPoS 算法。这个阶段持续 24 小时。

创建一个阶段时，将创建一个循环块，其中选择具有更多权重的节点。选择到循环块的节点将离开其桩号。

新块必须由 80% 的循环区块节点签名。因此，为了侵入系统，入侵者需要进入一个 80% 以上参与率的循环区块，并创建占整个系统 80% 以上的假节点。在这种情况下，入侵者会拥有整个系统，他们抢劫自己是没有意义的。

Velas 节点选择

Velas 节点选择的目的是计算神经网络的矩阵。GPU“机载”是必要条件。网络管理器设置多个节点。矩阵计算的输入从区块链接收。

该算法是公共的，因此每个网络成员都可以：

1. 获取输入；
2. 完成矩阵；
3. 进行控制计算。

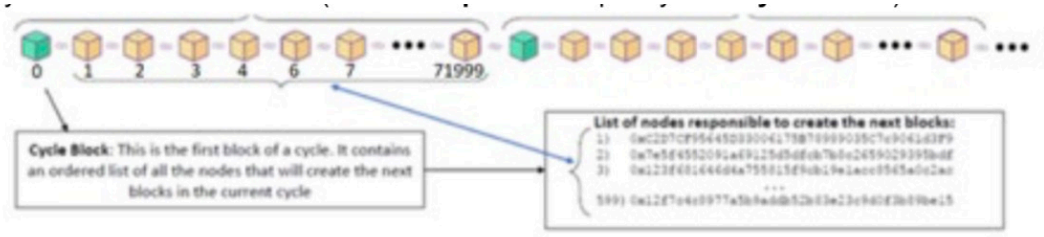
根据神经网络的计算结果，一个节点形成一个具有时序的循环区块，并将其发送给所有网络参与者。

一个节点存储一个循环块的节点参与者的抵押代币。通过计算神经网络，一个节点在一个阶段结束后分配奖励。

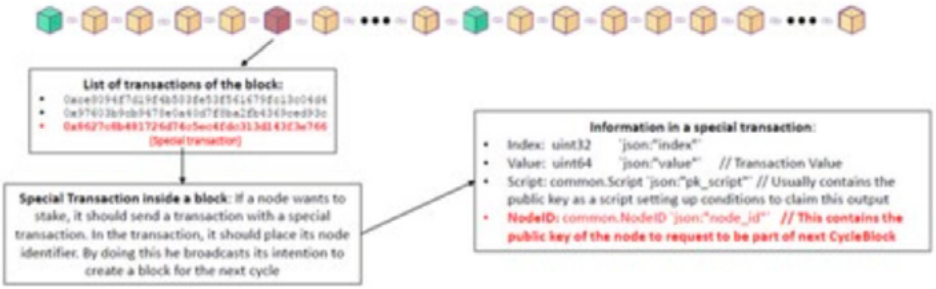
如果检测到攻击，节点将分解一个循环区块并出新块。

循环区块结构

循环时间 = **72001 区块**（72000 **SimpleBlocks** 每循环 +**Cycle-block**）



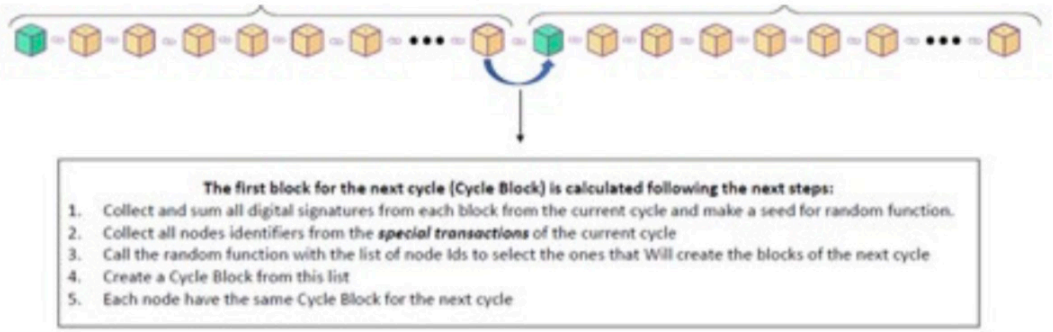
进入网络的节点



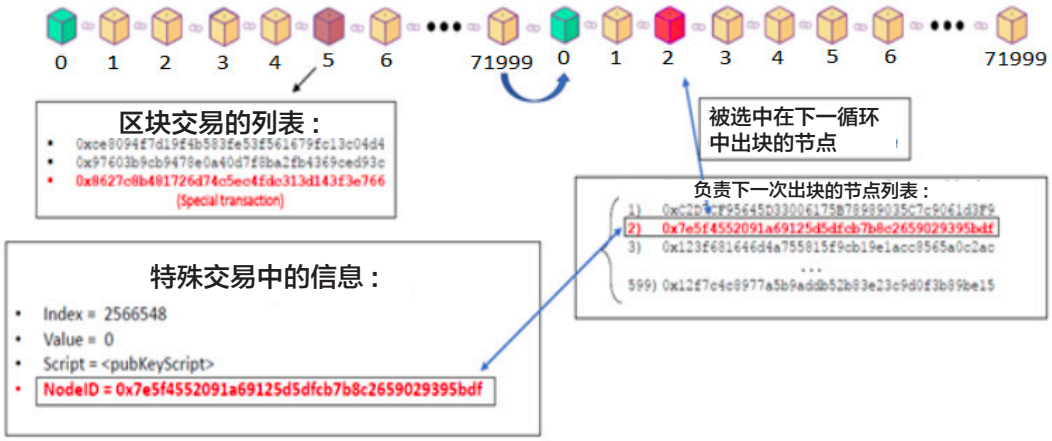
所有试图抵押的节点必须向网络发送一个特殊交易，以便在下面的区块周期中将自己记录为出块者。

区块循环结束

选择节点以出块



Velas 平台上的交易



交易模型的描述

Velas 平台上的 Velas 代币的转让是通过将新代币的公钥重新发布给后续所有者来执行的，同时保留上一笔交易的哈希值。交易的验证是通过链的验证来完成的。

为了防止诸如“双花”这样的难题，可以通过网络验证链的真实性来解决。这是通过引入“Epoch”协议实现的，其提供了更高的安全级别，允许在特定的时间内将区块添加到区块链。这些区块是公开的，可以在区块浏览器上查看和检查到。每个区块包含前一个区块的哈希和一个时间戳。每个包含的时间戳增强了整个链的有效性。

每秒交易数

举例：

- 区块时间 -2 秒 (1 秒 -2 分钟);
- 每个区块的交易量 - 60,000 笔 ;
- 每秒交易数 - 60,000 / 2 = 30,000 TPS;

交易过程

在这个部分中，我们将总结创建和处理 Velas 交易的关键细节

1) 每个交易都有以下参数 :

- 交易哈希
- 区块链交易类型

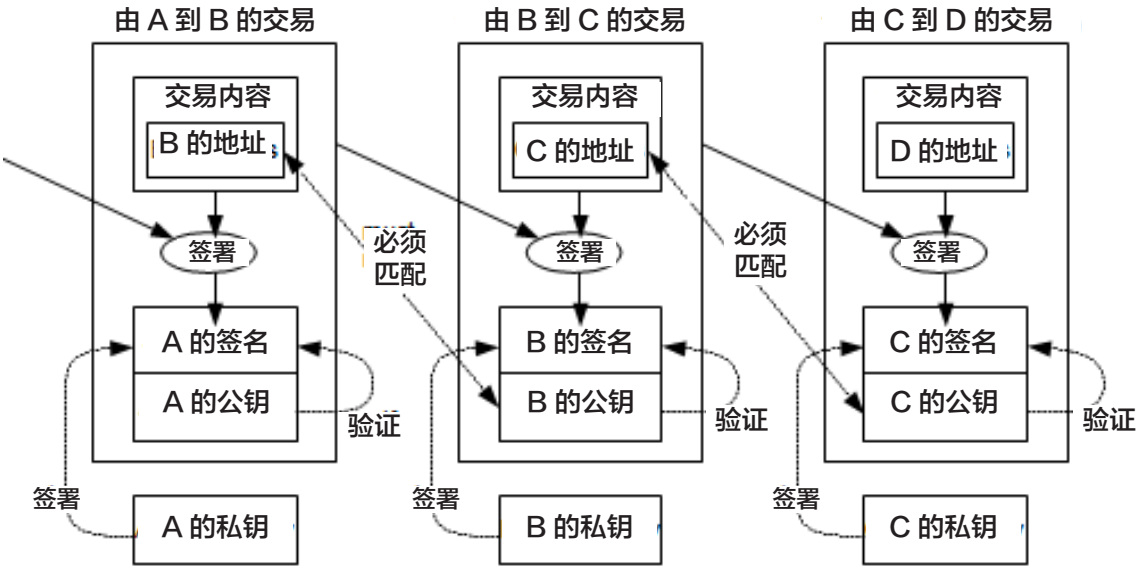
- 交易被区块确认时的区块数或时间戳
- 使用的交易输入数量
- 使用的交易输入列表
- 创建的交易输出数量
- 创建的交易输出列表

2) 所有交易输入的值在处理之前都要经过验证。必须审查所需参数的规格。例如，佣金不能小于或等于零。如果有问题的交易没有得到确认，则该交易将不被处理。

3) 必须启动和处理交易验证程序。这项核查应确保下列事件按计划进行：

- 新交易创建完成
- 生成新代币新交易标识符
- 获得新代币所有者签名
- 执行信息中指示网络节点处理交易的数据的加密
- 成功生成并记录传输到网络所有节点的数据

交易结构



交易确认

所有的Velas交易均被视为是未经确认的，直到它们包含在一个有效的区块中。

近创建的区块由出块节点分发到网络。由于新的区块被添加到现有的块链中，所以每个额外的区块都会增加一个交易确认验证。已发送至网络中但未包含在区块中的交易将无法得到确认。交易的优先级基于其相关费用的多少。

交易成本

当代币的组合、分割或重新发行添加到区块中时，与区块相关的所有交易费用都将分布在网络中的节点中。委员会成员从各区块内的所有交易中按其由网络选出的次序获得 Velas 代币的奖励。

如果区块中所有交易的大小不超过 1 MB，那么最小的 Velas (VLX) 将足以支付与处理相关的所有费用。当未确认交易的数量超过可置于区块中的数量时，节点将选择佣金最高的交易。

交易哈希的生成

交易和区块哈希值是使用 Schnorr 签名算法生成的。之后我们将讨论更多关于这种算法的逻辑、优点和工作原理。Velas 将始终支持安全的多重签名交易。

- 版本
- 锁定时间 – 0
- 交易输入
- 哈希 – 交易哈希
- 指数（代币形成时的输出数量）

价值（代币在 CCN 的数量）

- 签名脚本
- 签名
- 交易输出

代币结构

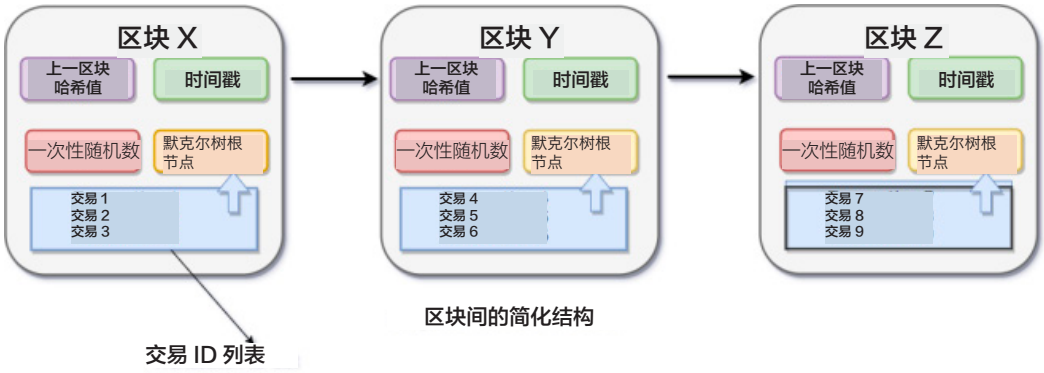
代币是在交易过程中形成的，其结构如下：

- 密钥（交易中产生）
- 交易哈希（从创建时就存在）
- 指数（交易指数）
- 单位（代币数量）

每个区块在加入区块链之前必须得到委员会成员的确认。每个成员验证区块中的所有交易，确认以下数据：交易数量、输入、数字签名和输出。如果验证成功，在将交易合并到区块链之前，将签名发布到网络的其他部分进行最终验证。为了参与其中，节点必须将其目的通知网络，从而获得一枚独特的代币，用于存储在其整个过程中所完成工作的报酬数据。委员会新成员的选拔标准如下：网络节点的活动时间、钱包的大小、参加委员会的频率。

区块头包括以下参数：

- 高度（区块的高度）
- 尺寸（区块大小）
- 版本（区块链版本）
- 之前的区块哈希
- 时间戳（当前时间以秒为单位）
- Bits（命令窗口）
- nonce，一次性随机数（命令窗口）
- 每个区块中的交易数



Velas 区块生成节点（BGN）的详细信息

当前网络节点分为两种类型：

- 区块生成节点（BGN）；
- 所有剩余节点。

连接时，节点向 DNSSeeds 发出请求并接收一个列表 (slice)。BGN 向任何地址发出请求，并接收 BGN 分支的列表 (node_slice)。

节点向列表中的任意节点发出请求，并尝试连接。如果连接成功，则注册节点。如果节点没有空闲插槽，它将给出一个可以向其发出连接请求的从节点列表，否则，它将通过连接到任何可能的从节点来自行注册。

此外，在整个“树”中搜索，这个循环将继续，直到新节点找到一个空闲的连接插槽。

节点同步并发出请求 :sync()

此外，如果节点不在 NAT 后面，它将启动一个服务器来连接剩余的 7 个节点。节点发送一个 ping 码来确定路径长度。节点总是连接到 TCP 服务器。如果节点在没有通知的情况下失去连接，它也会失去这些分数。

网络中有两种类型的消息：

- 节点和主节点间的消息；
- 不通过从节点的消息。

每个附加节点都对前面结构的哈希值进行签名，并将其发送出去，直到 ping 码到达 BGN

这些记录的结构如下：

```
{
  address_node,
  hash,
  sign
}
```

BGN 对哈希进行签名并将其发送回去。此时，节点记录跳转到 BGN。当投票发生时，节点根据跳数和前面讨论的其他参数给 master 打分。如果需要连接一个新节点，该节点将用一个空闲插槽和跃点数 BGN。如果没有空闲插槽，节点只响应“无法连接”，并给出从节点列表。

节点如何控制连接？

基本设置是通过向主控节点发送“notify”命令来完成，主节点必须响应“OK”。

每隔两分钟，节点必须从主节点接收一个新的块。如果节点没有在指定的时间范围内接收到块，则可以开始搜索新的主节点。

如果将区块发送到从节点的尝试失败，并且节点从特定节点接收到错误且连接未关闭，则节点将自行关闭该连接。

如果主节点不可用，但其他节点可用（即成功响应连接尝试），则节点会给这个主节点扣分。

如果节点在 NAT 后面，它每 15 秒发送一次“notify”。代理客户端节点不能是 BGN 或主节点。

交易过程概述

交易自从节点传播到主节点。主节点验证交易，对它们签名并转发。节点离 BGN 越近，需要验证的交易越多。

如果节点不验证交易，则节点不发送交易，并等待算力出现的时刻。

如果交易不正确，节点将关于错误的通知发送到从节点。

交易每秒转发到主节点一次，在发送之间的间隔中，形成一个来自交易的区块。需要减少施加在网络上的负载，因为区块越大，网络上的负载就越少。

完整的区块（由所有 BGN 记录的链块）从主节点向下扩展到从节点。

BGN 的选择

关于到 BGN 的路径长度，每个节点按分数为 master 投票。最大数量为 10，即 1 个跃点；最小数量是 0，即超过 10 个跃点。

自从节点接收评估后，主节点向其 master 打分，并添加从节点的大小。由从节点生成的交易数被添加到节点的大小中。

分数总额

主节点验证从节点给出的数据的正确性。如果检测到错误，则不会考虑分支的分数；

检测到错误的从属节点的分数将被归零。

为了限制在“树”上生成交易时不涉及的节点（矿机）的提升，必须计算交易的数量。

节点签署投票并发送消息。然后，对每对节点的分数进行求和和处理。

接收到最终消息后，BGN 将其发送给其他 BGN。投票后，选出得分最高的节点。形成新 BGN 的列表。BGN 验证选择链中的符号。

投票

所有的 BGN 都构成节点候选列表。如果算法正确，列表对每个人都是一致的。

每个 BGN 验证这个列表，如果它符合它的列表，BGN 就对它进行签名。获胜的列表至少拥有超过 80% 的 BGN 票数。

BGN(区块生成节点)

节点在期间开始时“被激活”，并打开 BGN 套接字。此节点还向其他 BGN 发送问候语。

BGN 之间的关系类型是“全映射”。节点根据循环区块中指示的调度出块。

自从节点接收到交易后，BGN 将其转发给所有 BGN。

生成区块的节点收集所有未经确认的交易并出块。并被转发到所有 BGN。

BGN 验证交易的正确性并对区块进行签名。如果一个区块被超过 80% 的 BGN 签名，那么它就被认为是可接受的。因此，这个被接受的区块沿着“树”展开。

每个阶段的开始时，下一个 BGN 的投票就启动。与交易一起，BGN 从它们的从节点接收投票，并验证交易数量和分支大小的一致性。如果有不一致，整个分支就会扣分。

验证后，BGN 将投票结果转发给其他 BGN。这样，就形成了节点候选列表。

在“age”的中间，形成了新的 BGN“age”列表和新的循环块。该列表由所有 BGN（80%）签名，并沿“树”向下延伸。

节点

一个节点在一个区块中注册并发送“register”消息，如果这个节点是由它的从节点（与其链接）投票选出的。相关参数列表包括：

```
{
  address
  weight
  hash []addresses slave nodes
  transaction []hashes
}
```

一个新节点被添加到区块链。区块链由六个“age”组成。如果节点在此期间不活动，它也会失去所有的分数和注册资格。

钱包（代币）

钱包的目的是创建交易。它通过自己的节点执行请求。钱包是由特殊应用程序创建的，可以传递给任何其他应用程序。钱包余额是未使用交易输出的数量。

面向用户的应用程序

应用程序被用作钱包（代币）的存储。它必须链接到一个节点。

安全性

针对 51% 算力和双重支付攻击保护：

我们使用 DPoS 算法。期间持续 24 小时。

在创建一个阶段时，将创建一个循环区块，区块中权重更大的节点会被选择。被循环区块选中的节点将留下他们的质押代币。

新的块必须由 80% 的循环区块节点签名。

因此，要入侵系统，入侵者需要进入一个参与率超过 80% 的循环区块，并创建占整个系统 80% 以上的假节点。在这种情况下，入侵者才拥有整个系统，然而他们抢劫自己是没有意义的。

三种类型的节点：

- 主节点（来自 3 个节点，需要 3 个或更多的节点来保证网络在任何时候都具有可用性！）为了实现 AI 的去中心化功能，主节点功能将在第四阶段发布的网络候选版本中转移到 Advisor。
- Advisors—区块生产节点。在发布时，候选 Advisor 将向网络发出信号，表明他们拥有足够的 GPU 算力，并将根据他们的声誉、正常运行时间、计算速度和代币持有情况由网络进行选择。如果被选中，这将允许他们运行神经网络，并从网络中获得更高的回报，帮助网络逻辑的 AI 组件进行训练和去中心化。
- 典型节点

所有 Advisor 在生成新区块时检查交易。如果 80% 的 Advisor 验验证确认该区块，它将进入网络。

如果 20% 的典型节点将一个区块定义为伪区块，主节点将检查这个区块。

投票率是选择循环区块节点的标准之一。

如果 80% 的 Advisor 投票给一个伪区块，20% 的节点没有接受这一区块，这个区块将不会被添加到区块链中。经 Advisor 确认，若区块由主节点区块验证后确认该区块是假的 , 该节点将报告发生攻击并获得奖励。

这是我们解决“nothing-at-stake”问题的办法。

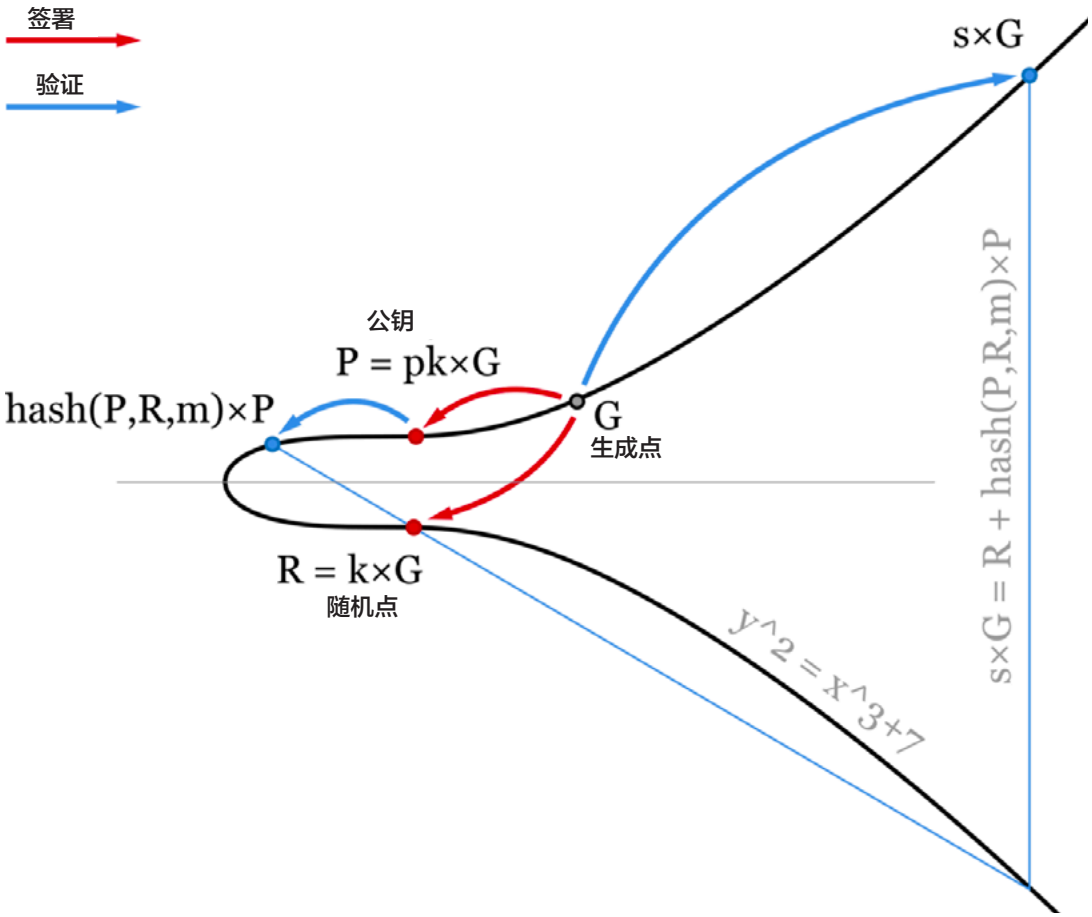
哈希算法

我们使用 Schnorr 签名而不是 ECDSA。Velas 实现将使用 Secp256k1 曲线，因为它是可预测的，因此性能更好。对 Velas 区块使用 Schnorr 签名的最大好处是可以获得可观的性能提升。

由于 Schnorr 签名的线性特性（签名的“总和”等价于“总和”的签名），我们可以对区块中所有交易 / 输入的验证签名进行批处理。

并且，我们只需要存储一个单独的签名，通过简单的计算就可以聚合该区块的所有签名。而这只需要更少的存储成本就可以达到更快的验证。毕竟，节点现在只需要两个相对简单的计算就可以根据区块签名验证所有交易的签名，区别于以往成千上万次计算。

使用 Schnorr 算法而不是 ECDSA 还允许密钥聚合和改进隐私性，这样就再也无法将多重签名交易与常规交易区分开来。



ECDSA 算法的可视化。为便于说明，椭圆曲线是在实数上绘制的。
参考： https://medium.com/cryptoadvance/how-schnorr-signatures-may-improve-bitcoin-91655bcb4744

中间人保护

该系统通过以下方式防止 MITM 攻击：

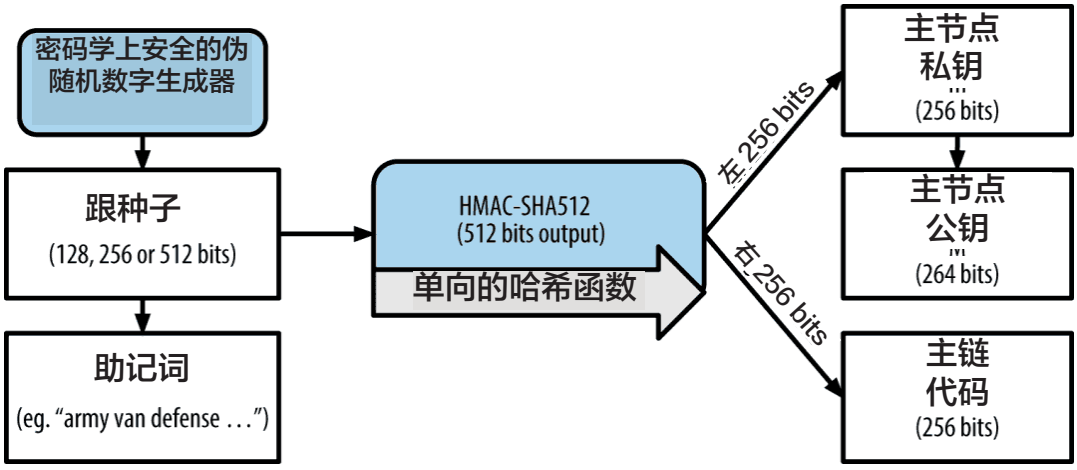
- 每个节点的公钥存储在区块链中。
- 此外，在区块链中还有一个别名—公钥串。
- 当容器从 subscriber A 向 subscriber B 发送时，subscriber A 在区块链中接收 subscriber B 的公钥，并对其容器加密。它以加密的形式发送到 subscriber B 的的支持多币种的钱包。Subscriber B 的使用私钥解密容器。

支持多币种的钱包技术

Velas 区块链可以为所有支持的代币创建公共和私有容器。Velas 将为所支持的各种加密货币，如比特币、以太坊和代币、 柚子币、瑞波币、门罗币等生成私钥。密钥通过用户的私钥种子创建。所有容器都可以通过创建 Velas 钱包时的原始种子或私钥中恢复。这些容器允许创建所有其他代币的链上链下可扩展性方案，并作为 Velas 中所有的智能合约的一个生态系统钱包。用户还可以受益于将当前安全保存在 Velas 钱包中的所有代币存储起来，并为非原生代币创建多签备份。

传输存储私钥的容器。

1. 从节点列表中选择可以作为代理的节点。例如 : 设置了一个标志，并且节点有一个公共地址。
2. 然后，subscriber A 将节点地址发送给 subscriber B。
3. 两个 subscriber 都通过指定的通道连接到代理服务器。在代理服务器上创建映射（Map），图中的代理服务器，Key 是通道名称，Value 是两个连接的结构。
4. 连接完成后，subscriber A 将其公钥发送给 subscriber B。
5. subscriber B 验证 subscriber A 的地址并发送公钥。如果它们一致，拥有包含私钥容器的 subscriber B 将使用 subscriberA 的公钥加密容器并将传输给他 / 她。
6. 接收容器后，subscriber A 使用他 / 她的私钥解密。
7. 关闭连接。



（完）