CIPC

WHITE PAPER

CIPC-Cosmic Interstellar Public Chain

白皮书

大规模商用高性能区块链平台

V 1.0 SEPT,2022

序

现在，区块链已经发展到上万亿美金的市值，但是受制于性能、可扩展性，急需下一代突破性的平台，Cosmic Interstellar Public Chain目标就是以现代密码学、通讯技术、分布式计算为基础，能够承载去中心化金融、社交、电商、搜索、存储商业应用，提供一个大规模商用高性能的区块链生态平台。该介绍性白皮书最初由Cosmic Interstellar Public Chain的创始人jeffrey 于2018年发布。值得注意的是，Cosmic Interstellar Public Chain 与许多社区驱动的开源软件项目一样，自最初成立以来就在发展。虽然已有数年之久，但我们之所以继续维护该文件，是因为它继续充当有用的参考和对Cosmic Interstellar Public Chain 及其愿景的准确表示。要了解 Cosmic Interstellar Public Chain的最新发展以及如何对协议进行更改，我们建议您使用本白皮书。

1. 下一代大规模商用高性能区块链生态平台

中本聪(Satoshi Nakamoto)在2009年开发的比特币经常被誉为货币和货币的激进发展，是数字资产的第一个示例，它同时没有支持或内在价值，也没有集中的发行者或控制者。但是，比特币实验的另一个(可能更重要的)一部分是作为分布式共识工具的基础区块链技术，人们的注意力正在迅速转移到比特币的另一个方面。通常被提及的应用，包括使用区块链的链上数字资产来代表定制货币和金融工具("颜色币")，相关物理设备("智能资产")的所有权，非同质资产例如域名("域名币")，以及更复杂的应用程序，涉及通过执行任意规则。

(智能合约)甚至基于区块链的去中心化自治组织(DAO)的一段代码直接控制数字资产。Cosmic Interstellar Public Chain打算提供的是一种带有内置完整成熟的图灵完备编程语言的区块链，可用于创建可用于编码任意状态转换函数的"合同"，从而允许用户创建上述任何系统，以及我们尚未想到的许多其他功能，只需通过几行代码编写逻辑即可。支持下一代大规模商用需要区块链平台目标是，支撑上百亿设备并发连接，秒级确认，数十万每秒事务处理能力的高性能区块链生态能力。

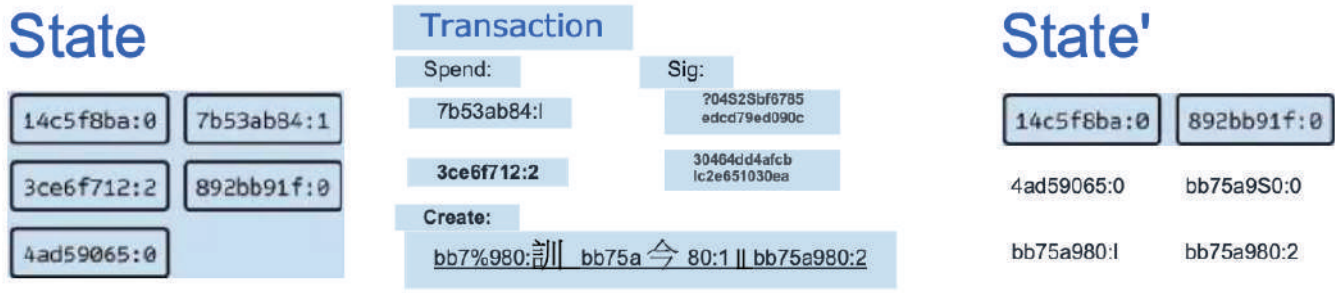
2. 比特币和现有概念简介

2.1 历史

去中心化数字货币的概念以及财产登记等替代应用已经存在了数十年。1980 年代和1990 年代的匿名电子现金协议主要依赖于一种称为Chaumian盲法的加密原语，为货币提供了高度的隐私性，但是由于依赖于中央中介机构，这些协议在很大程度上未能获得关注。1998年，戴炜的b-money成为第一个提出通过解决计算难题以及去中心化共识来创造金钱的想法的建议，但是该建议在如何实际实施去中心化共识方面没有任何细节。2005年，Hal Finney引入了可重复使用的工作量证明的概念，该系统将b-money的想法与Adam Back 的计算困难的Hashcash难题一起使用，创建了一种加密货币的概念，但由于依赖于可信计算作为后端。2009年，中本聪首次在实践中实施了一种去中心化货币,将通过公钥密码管理所有权的既定原语与用于跟踪谁拥有硬币的共识算法（称为"工作量证明"）相结合。

工作量证明背后的机制是该领域的一项突破，因为它同时解决了两个问题。首先，它提供了一种简单而适度有效的共识算法，允许网络中的节点共同商定一组关于比特币分类账状态的规范更新。其次，它提供了一种机制，允许自由进入共识过程，解决了决定谁可以影响共识的政治问题，同时防止了 sybil 攻击。它通过替换正式的参与障碍来实现此目的，例如要求将其注册为特定列表上的唯一实体，并带有经济障碍-共识投票过程中单个节点的权重与计算能力成正比节点带来的。从那时起，提出了一种替代方法，称为权益证明，将节点的权重计算为与其货币持有量成正比，而不与计算资源成正比；关于这两种方法的相对优点的讨论超出了本文的范围，但是应该注意，这两种方法都可以用作加密货币的骨干。

2.2 比特币作为状态转换系统



从技术角度来看，可以将诸如比特币之类的加密货币的分类账视为状态转换系统，其中存在一个"状态"。

由所有现有比特币的所有权状态和"状态转换函数"组成，该状态转换函数接受状态和交易并输出新的状态，作为结果。例如，在标准银行系统中，状态为资产负债表，一笔交易是将$x从A移到B的请求，并且状态转换函数将A的账户中的值减少$X，将B的账户中的值增加$X。如果A的账户中第一个账户的余额少于$x位置，状态转换函数返回错误，因此可以正式定义：

APPLY(S,TX) -> S' or ERROR

APPLY({ Alice: $50, Bob: $50 },"send $20 from Alice to Bob") = { Alice: $30, Bob: $70 }

在上面定义的银行系统中：但：

APPLY({ Alice: $50, Bob: $50 },"send $70 from Alice to Bob") = ERROR 比特币中的“状态”是指所有已开采但尚未花费的硬币（从技术上来说，是“未 花费的交易输出"或UTXO）的集合，每个UTXO都有一个面额和一个所有者（由 20 个字节的地址定义，本质上是一种加密的公共密钥。1）。事务包含一个或多个输入，每个输入包含对现有 UTXO 的引用和由与所有者地址相关联的私钥产生的加密签名，以及一个或多个输出，每个输出包含要添加到的新UTXO。

状态转换函数APPLY（S, TX）->S'可以大致定义如下：

1. 对于TX中的每个输入：
如果引用的UTXO不在S中，则返回错误。
如果提供的签名与UTXO的所有者不匹配，则返回错误。
2. 如果所有输入UTXO的面额之和小于所有输出UTXO的面额之和，则返回错误。

3. 返回S'，删除所有输入UTXO,并添加所有输出UTXO。
第一步的前半部分防止交易发送者花费不存在的硬币,第一步的后半部分防止交易发送者花费其他人的硬币,而第二步则强制保护价值。为了将其用于支付,协议如下。假设爱丽丝想向鲍勃发送11.7 BTC。首先，Alice将寻找一组她拥有的可用UTXO,这些UTXO总计至少为11.7 BTC。实际上，爱丽丝将无法获得准确的11.7 BTC。说她能得到的最小数是6+4+ 2=12。然后，她使用这三个输入和两个输出创建一个事务。第一个输出将是11.7 BTC，以Bob的地址作为其所有者，第二个输出将是剩余的0.3BTC "更改"，所有者是Alice自己。

2.3 挖矿

如果我们可以访问值得信赖的集中式服务，那么该系统的实施将非常简单；只需使用集中式服务器的硬盘驱动器来跟踪状态，就可以完全按照描述的方式对其进行编码。但是，对

于比特币，我们正在尝试构建去中心化的货币系统，因此我们需要将状态转换系统与共识系统结合起来，以确保每个人都同意交易顺序。比特币的去中心化共识过程要求网络中的节点不断尝试产生称为"区块"的交易包。该网络旨在每十分钟产生大约一个块，每个块包含一个时间戳，一个随机数， 对前一个块的引用(即其哈希值)以及自上一个以来发生的所有事务的列表。随着时间的流逝，这将创建一个持续不断增长的“区块链”，该区块链会 不断更新以代表比特币分类账的最新状态。此范例中表示的检查块是否有效的算法如下： 检查该块引用的前一个块是否存在并且有效。检查该块的时间戳是否大于前一个块的时间戳。距离未来不到2小时检查块上的工作证明是否有效。令 S [0]为上一个块末尾的状态。假设TX是具有 n 个交易的区块交易列表。对于0... n-1中的所有i，设置S[i + 1]=APPLY(S[i],TX[i])如果有任何应用程序返回错误，请退出并返回false。返回true，并在该块末尾注册S[n]作为状态。

从本质上讲，该块中的每个事务必须提供一个有效的状态转换，该状态从执行该事务之前的规范状态到某个新状态。请注意，状态不会以任何方式编码在块中。它纯粹是验证节点要记住的一种抽象，只能通过从创始状态开始并按顺序在每个块中应用每个事务，才能（安全地）为任何块计算该抽象。此外，请注意，矿工将交易包含在区块中的顺序很重要；如果一个区块中有两个事务A和B，使得B花费了A创建的UTXO,那么如果A在B之前，则该区块有效。

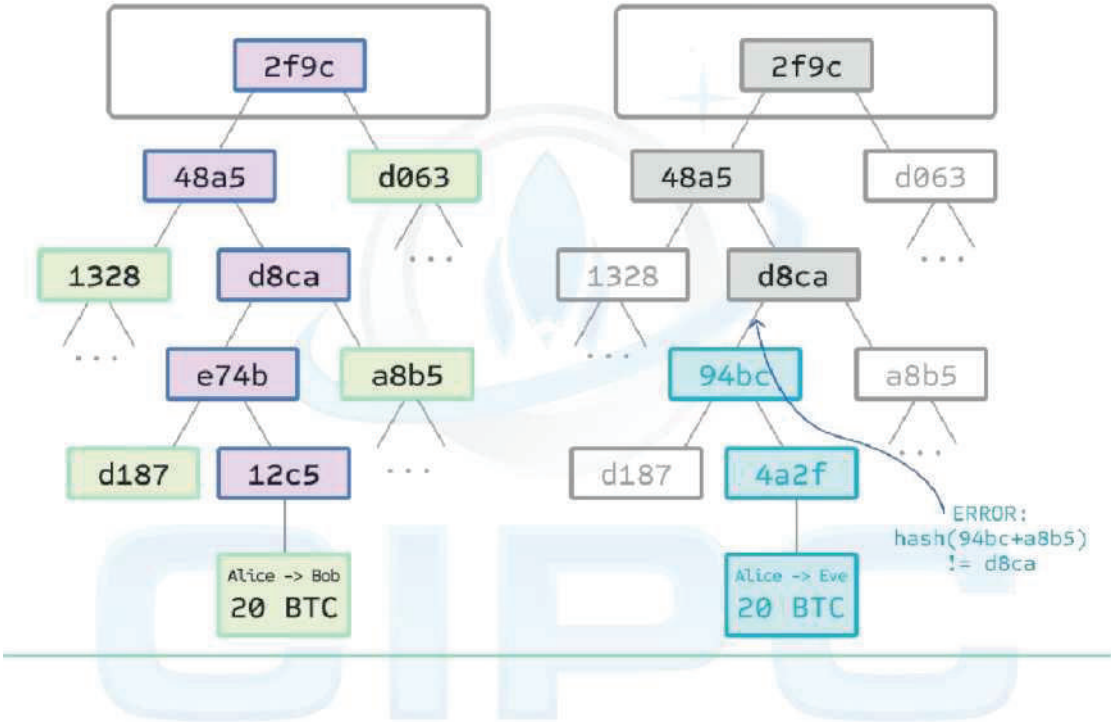
上面存在的一个有效条件在其他系统中找不到的清单是"工作量证明"的要求。精确的条件是，每个块的双SHA256哈希（被视为256位数字）必须小于动态调整的目标，在撰写本文时，该目标约为2187。这样做的目的是使在计算上 "困难"地阻止创建，从而防止sybil攻击者以他们的喜好来重新构建整个区块链。因为SHA256被设计为完全不可预测的伪随

机函数，所以创建有效块的唯一方法是反复试验，反复增加随机数并查看新哈希是否匹配。
在当前目标～2187之前，网络必须平均尝试 269 次才能找到有效的块。通常，网络每隔 2016个块对目标进行一次重新校准，以便平均而言，网络中的某个节点每隔十分钟会产生一个新块。为了补偿矿工的计算工作量，每个区块的矿工都有权包括一笔无偿提供12.5 BTC的交易。另外，如果任何交易的输入总面额大于输出的总面额，则差额也作为“交易费”分配给矿工。顺便说一句，这也是发布BTC的唯一机制，起源状态根本不包含任何硬币。

为了更好地了解挖掘的目的，让我们检查一下恶意攻击者发生的情况。由于已知比特币的基础加密是安全的，因此攻击者将针对比特币系统不受加密直接保护的部分：交易顺序。攻击者的策略很简单：向商家发送100 BTC，以换取某些产品（最好是快速交付的数字商品）等待产品交付产生另一笔交易，向自己发送相同的100 BTC 尝试说服网络，他对自己的交易是第一笔交易。一旦执行了步 骤（1），几分钟后，一些矿工将交易包含在一个区块中，例如区块编号270。大约一小时后，该区块之后的链中将再添加五个区块，每个区块间接指向交易并因此"确认"交易。此时，商家将接受最终确定的付款并交付产品；由于我们假设这是一种数字商品，因此交付是即时的。现在，攻击者创建了另一个向自己发送100 BTC的交易。如果攻击者只是简单地将其释放到野外，则不会处理该交易；矿工将尝试运行APPLY（S，TX），并注意到TX消耗了不再处于该状态的UTXO。因此，取而代之的是，攻击者创建了区块链的"叉子"，首先是挖掘块270的另一个版本，该版本指向与父代相同的块269,但使用新的交易来代替旧的交易。由于块数据不同，因此需要重做工作量证明。此外,攻击者的块270的新版本具有不同的哈希，因此原始块271到275不会"指向"它；因此，原始块271至275不会指向该块。因此，原始链和攻击者的新链是完全分开的。

规则是，用叉子最长的区块链才是事实，因此合法的矿工将在275 链上工作，而仅攻击者在270链上工作。为了使攻击者能够使自己的区块链最长，他需要比网络其余部分的总和拥有更多的计算能力才能赶上（因此“51％攻击”）。

2.4 默克尔



左:仅在Merkle树中显示少量节点即可证明分支的有效性。
右:尝试更改Merkle树的任何部分最终都会导致链上某些地方出现不一致。
比特币的一个重要的可伸缩性功能是，该块存储在多层数据结构中。块的"哈希"实际上仅是块头的哈希，大约 200 字节的数据，其中包含时间戳，随机数，前一个块哈希以及称为 Merkle 树的存储所有事务的数据结构的根哈希在块中。默克尔树(Merkle tree)是一种二叉树，它由一组节点组成，在树的底部包含大量底层节点，这些底层节点包含基础数据，还有一组中间节点，其中每个节点是其两个子节点的哈希，最后是一个根节点，该根节点也由其两个子节点的哈希组成，代表树的“顶部”。Merkle树的目的是允许零散地传递块中的数据: 节点只能从一个源下载块的标头，而从另一源下载与它们相关的树的一小

部分，并且仍然可以放心所有数据都是正确的。之所以可行，是因为散列向上传播：如果恶意用户试图交换假交易进入Merkle　树的底部，此更改将导致上方节点的更改，然后更改上方节点的更改，最终更改树的根并因此更改块的哈希，从而导致协议注册它是一个完全不同的模块（几乎可以肯定带有无效的工作证明）。

默克尔树协议可以说对于长期可持续性至关重要。比特币网络中的一个"完整节点"，用于存储和处理每个块的全部，截至2021年4月在比特币网络中占据约95GB的磁盘空间，并且每月增长超过10GB。当前，这对于某些台式机而不是电话是可行的，并且在以后的将来，只有企业和业余爱好者才能参加。称为"简化付款验证"（SPV）的协议允许存在另一类节点，称为"轻型节点"，这些节点下载块头，验证块头上的工作量证明，然后仅下载"分支"与之相关的交易相关联。这使轻型节点能够以强大的安全性确定任何比特币交易的状态及其当前余额，同时仅下载整个区块链的一小部分。

2.5替代区块链应用

采取基础的区块链概念并将其应用于其他概念的想法也有很长的历史。1998年，Nick Szabo提出了拥有所有者授权的安全产权的概念，该文档描述了“复制数据库技术的新进展”将如何允许基于区块链的系统存储谁拥有土地的注册管理机构，精心设计的框架，包括房屋维护，逆权管有和格鲁吉亚土地税等概念。但是，不幸的是，当时没有有效的复制数据库系统可用，因此该协议从未在实践中实施。但是，在 2009 年之后，一旦比特币的去中心化共识得以制定，许多替代应用便迅速出现。Namecoin-创建于2010年，Namecoin最好地描述为一个分散的名称注册数据库。在 Tor，Bitcoin 和BitMessage 等去中心化协议中，需要某种方式来识别账户，以便其他人可以与它们进行交互，但是在所有现有解决方案中，唯一可用的标识符是伪随机哈希，例如 1LW79wp5ZBqaHW1jL5TCiBCrhQYtHagUWy。理想情况下，人们希望能够拥有一个名称为“ george”的账户。但是，问题在于，如果一个人可以创建一个名为“george”的账户，那么其他人也可以使用相同的过程为自己注册
“george”并模拟他们。唯一的解决方案是从第一个到第一个文件的范例，其中第一个注册器成功，第二个注册器失败，这个问题非常适合比特币共识协议。Namecoin是使用这种想法的最古老、最成功的名称注册系统的实现。彩色硬币-彩色硬币的目的是用作允许人们创建自己的数字货币的协议-或者，在重要的琐碎情况下，在比特币区块链上以数字为单位的数字令牌。在有色硬币协议中，通过公开为特定的比特币UTXO分配颜色来"发行"新货币，并且该协议以递归方式将其他UTXO的颜色定义为与创建它们的交易所用输入的颜色相同。(某些特殊规则适用于混合色输入)。这允许用户维护仅包含特定颜色的UTXO的钱包，并像常规比特币一样将它们发送到周围，在区块链中回溯以确定他们收 到的任何UTXO的颜色。

Metacoins-Metacoin 背后的想法是拥有一个生活在比特币之上的协议，该协议使用比特币交易来存储Metacoin交易，但具有不同的状态转换功能APPLY'。由于metacoin协议无法阻止无效的metacoin交易出现在比特币区块链中，因此增加了一条规则，即如果APPLY' (S,TX)返回错误，则该协议默认为APPLY'(S,TX)=SO这提供了一个创建任意加密货币协议的简单机制，可能具有无法在比特币本身内部实现的高级功能，但开发成本非常低，因为挖矿和联网的复杂性已由比特币协议处理。
因此，通常，有两种方法来建立共识协议：建立一个独立的网络，以及在比特币之上建立一个协议。前一种方法虽然在Namecoin之类的应用程序中相当成功，但难以实施；每个单独的实现都需要引导一个独立的blockchain，以及构建和测试所有必要的状态转换和 联网代码。此外，我们预测去中心化共识技术的应用程序集将遵循幂律分布，其中绝大多数应用程序太小而不能保证拥有自己的区块链，并且我们注意到存在大量的去中心化应用程序类别，尤其是去中心化自治系统组织，需要彼此交互。

另一方面，基于比特币的方法具有一个缺点，即它不继承比特币的简化付款验证功能。SPV适用于比特币，因为它可以使用区块链深度作为有效性的代理。在某个时候，一旦交易的祖先走得足够远，可以肯定地说他们是国家的合法组成部分。另一方面，基于区块链的元协议不能强制区块链不包括在其自身协议范围内无效的交易。因此，一个完全安全的 SPV 元协议实现将需要一直反向扫描到比特币区块链的开始，以确定某些交易是否有效。当前，所有基于比特币的元协议的“轻型”实现都依赖于受信任的服务器来提供数据，这可以说是次优的结果，尤其是当加密货币的主要目的之一是消除对信任的需求时。

2.6 脚本编写

即使没有任何扩展，比特币协议实际上也确实促进了“智能合约”概念的弱版本。比特币中的UTXO不仅可以由公钥拥有，还可以由以简单的基于堆栈的编程语言表达的更复杂的脚本拥有。在这种范例中，一笔交易花费了UTXO 必须提供满足脚本的数据。实际上，甚至基本的公共密钥所有权机制都是通过脚本实现的：脚本以椭圆曲线签名作为输入，根据交易和拥有UTXO 的地址对其进行验证，如果验证成功，则返回1，否则返回0。对于各种其他用例，还存在其他更复杂的脚本。例如，可以构造一个脚本，该脚本需要给定的三个私钥中的两个私钥进行验证(“multisig")，一种适用于公司账户，安全储蓄账户和某些商户托管情况的设置。脚本也可以用来支付解决计算问题的悬赏，甚至可以构建一个脚本，上面写着"如果您可以提供SPV证明向您发送了此面额的狗狗币交易，则该比特币UTXO是您的"，本质上允许去中心化交叉加密货币交换。

但是，用比特币实现的脚本语言有几个重要的局限性：
缺乏图灵完整性-也就是说，尽管比特币脚本语言支持很大一部分计算，但它几乎不支持所有功能，缺少的主要类别是循环。这样做是为了避免在交易验证期间出现无限循环。从理论上讲，这对于脚本程序员来说是一个无法克服的障碍，因为任何循环都可以通过简单地使用if语句重复底层代码多次来模拟，但是这确实导致脚本在空间上非常低效。例如，实施替代的椭圆曲线签名算法可能会需要256个重复的乘法回合，所有这些回合都单独包含在代码中。

价值盲目-UTXO脚本无法对可提取的金额提供细粒度的控制。例如，甲骨文合同的一个强大用例是对冲合同，其中A和B投入了价值$1000的BTC,脚本在30天后向A发送了价值$1000的BTC,其余的向B发送了。甲骨文确定1BTC的美元价值，但即使这样，与现在可用的完全集中化解决方案相比，这在信任和基础结构要求方面还是一项巨大的改进。但是，因为UTXO是全有或全无，所以实现这一目标的唯一方法是非常低效的破解，即拥有许多不同面额的UTXO（例如每30k中的每k有一个2k的UTXO）并由O选择哪个UTXO发送给A,发送给B缺乏状态-UTXO可以花费也可以不花费；多阶段合同或脚本没有机会保留超出此范围的任何其他内部状态。这使得很难制定多阶段期权合约，分散交易报价或两阶段加密承诺协议（安全计算赏金所必需）。这也意味着UTXO仅可用于建立简单的一次性合同，而不能用于构建诸如分散组织之类的更复杂的"有状态"合同，并且使元协议难以实施。二进制状态与价值盲目相结合还意味着另一个重要的应用提款限制是不可能的。

区块链盲目性-UTXO对诸如随机数，时间戳和先前区块哈希之类的区块链数据视而不见。通过剥夺脚本语言潜在的宝贵随机性，这严重限制了其他几个类别的应用。

因此，我们看到了三种在加密货币之上构建高级应用程序的方法：构建新的区块链，在比特币之上使用脚本，以及在比特币之上构建元协议。构建新的区块链可以无限自由地构建功能集，但要付出开发时间，自举工作和安全性的代价。使用脚本易于实现和标准化，但功能却非常有限，而元协议虽然简单，却存在可伸缩性方面的缺陷。借助Cosmic Interstellar Public Chain，我们打算建立一个替代框架，在简化开发以及更强大的轻客户端特性方面提供更大的收益，同时 允许应用程序共享经济环境和区块链安全性。

3. COSMIC INTERSTELLAR PUBLIC CHAIN

Cosmic Interstellar Public Chain 的目的是创建一个构建去中心化应用程序的替代协议，提供一组不同的权衡取舍，我们认为这对一大类去中心化应用程序将非常有用，尤其是在快速开发时间，小型和小型安全性的情况下。很少使用的应用程序以及不同应用程序进行非常有效的交互的能力非常重要。Cosmic Interstellar Public Chain通过构建本质上最终的抽象基础层来做到这一点： 带有内置图灵完备编程语言的区块链,允许任何人编写智能合约和去中心化应用程序，他们可以在其中创建自己的所有权，交易格式和状态转换功能。可以用两行代码来编写Namecoin的基本版本，而可以在不到20个的时间内建立其他协议（例如货币和信誉系统）。也可以在平台之上构建智能合约，即包含价值并只有在满足特定条件时才能解锁的密码"盒子"，其功能远比比特币脚本提供的功能强大，这是因为图灵完备性的附加功能、价值意识、区块链意识和状态。

哲学
3.1 Cosmic Interstellar Public Chain 背后的设计旨在遵循 以下原则：

1. 简便性：Cosmic Interstellar Public Chain 协议应该尽可能简单，即使以一些数据存储或时间效率低廉为代价。理想情况下，普通程序员应该能够遵循并实现整个规范。以充分实现加密货币带来的前所未有的民主化潜力，并进一步推动Cosmic Interstellar Public Chain 作为对所有人开放的协议的愿景。除非该优化提供了非常可观的好处，否则不应该包括任何增加复杂性的优化。

2. 普遍性：Cosmic Interstellar Public Chain 设计哲学的基本部分是，Cosmic Interstellar Public Chain不具有“功能”。相反，Cosmic Interstellar Public Chain 提供了一种内部的图灵完备的脚本语言，程序员可以使用该语言来构建任何可以数学定义的智能合约或交易类型。想发明自己的金融衍生品吗?使用Cosmic Interstellar Public Chain，您可以。想自己做货币吗?将其设置为Cosmic Interstellar Public Chain合约。是否想建立一个完整的Daemon或Skynet?您可能需要几千个互锁的合约，并一定要慷慨地提供它们，但是，没有什么能阻止您随时随地使用Cosmic Interstellar Public Chain。

3. 模块化：Cosmic Interstellar Public Chain协议的各个部分应设计成尽可能模块化和可分离。在开发过程中，我们的目标是创建一个程序，如果要在一个位置进行小的协议修改，则应用程序堆栈将继续运行而无需任何进一步的修改。应该将Ethash (请参阅黄皮书附录)，改良的Patricia树(Yellow Paper, Wiki)和RLP (YP, Wiki)等创新作为独立的功能完整的库来实现。这样一来，即使 Cosmic Interstellar Public Chain不需要某些功能,这些功能也仍然可以在其他协议中使用。应该最大程度地进行 Cosmic Interstellar PublicChain 开发,以使整个加密货币生态系统受益,而不仅仅是自身。

4. 敏捷性：Cosmic Interstellar Public Chain 协议的细节不是一成不变的。尽管我们会明智地对高层结构进行修改，例如使用分片路线图抽象执行，并且仅将数据可用性纳入共识。在开发过程的后期进行的计算测试可能会导致我们发现某些修改，例如协议架构或 Cosmic Interstellar Public Chain 虚拟机（EVM），将大大提高可扩展性或安全性。

5. 不歧视和不审查：协议不应尝试积极限制或阻止特定类别的使用。协议中的所有调节机制都应设计为直接调节危害，而不是试图反对特定的不良应用。程序员甚至可以在Cosmic Interstellar Public Chain之上运行无限循环脚本，只要他们愿意继续支付每个计算步骤的交易费即可。

3.2 Cosmic Interstellar Public Chain 账户

在Cosmic Interstellar Public Chain中，状态由称为“账户”的对象组成，每个账户都有一个20字节的地址，状态转换是账户之间价值和信息的直接转移。Cosmic Interstellar Public Chain 账户包含四个字段：
随机数，用于确保每笔交易只能被处理一次的计数器该账户当前的币余额账户的合同代码（如果有）账户的存储空间（默认为空）"CIPC”是Cosmic Interstellar Public Chain 的主要内部加密gas,用于支付交易费用。通常，有两种类型的账户：受私钥控制的外部拥有的账户和受其合同代码控制的合同账户。外部拥有的账户没有代码,并且可以通过创建和签名交易从外部拥有的账户发送消息；在合同账户中,合同账户每次收到消息都会激活其代码,从而使其能够读取和写入内部存储并发送其他消息或依次创建合同。
请注意,Cosmic Interstellar Public Chain中的“合约”不应被视为应“履行”或“遵守”的东西；相反,它们更像 Cosmic Interstellar Public Chain 执行环境中的“自治代理”,当被消息或事务“戳”时总是执行一段特定的代码,并直接控制自己的币余额和自己的密钥/值存储区,以跟踪持久变量。

3.3 交易

Cosmic Interstellar Public Chain中使用的术语“交易”是指已签名的数据包，用于存储要从外部拥有的账户发送的消息。交易包含：
邮件的收件人；识别发件人的签名
从发送方转移到接收方的币数量可选数据字段STARTGAS值，表示允许事务执行执行的最大计算步骤数GASPRICE值，表示发件人每计算步骤要支付的费用。

前三个是任何加密货币中预期的标准字段。数据字段默认情况下不起作用，但是虚拟机具有一个操作码，合同可以使用该操作码来访问数据。作为示例用例，如果合同充当区块链上的域注册服务,则它可能希望将传递给它的数据解释为包含两个“字段”,第一个字段是要注册的域,第二个字段是要注册的域。字段是要向其注册的IP 地址。合同将从消息数据中读取这些值,并将其适当地存储。

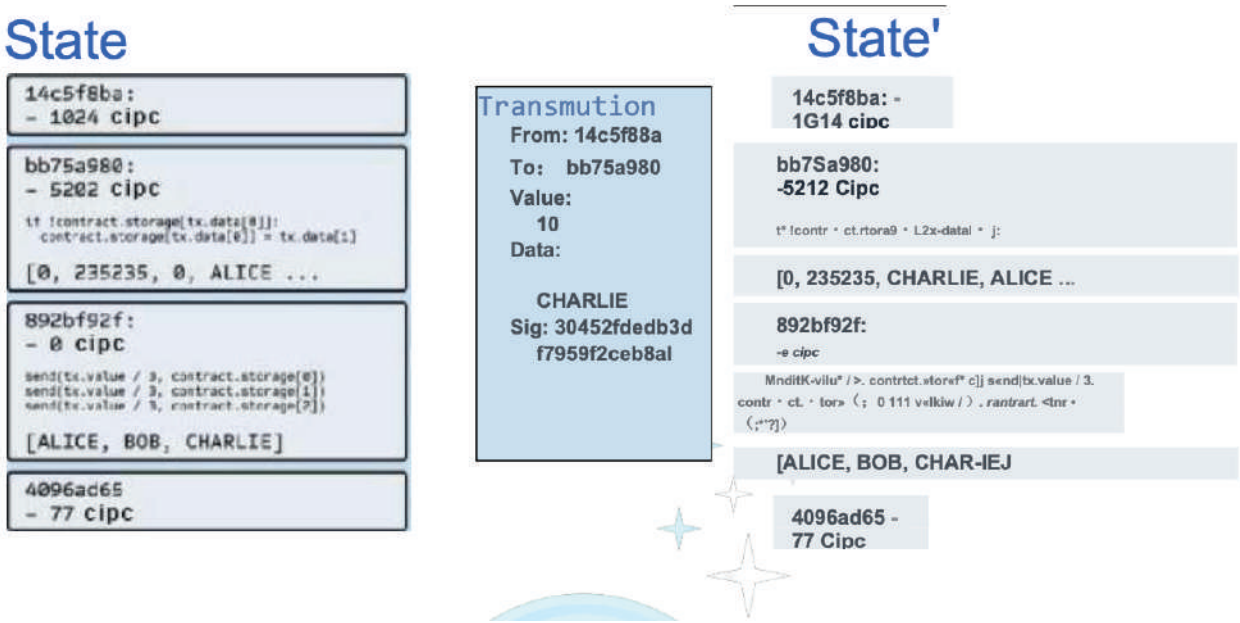
STARTGAS 和 GASPRICE 字段对于 Cosmic Interstellar Public Chain 的反拒绝服务模型至关重要。为了防止代码中的意外或敌对的无限循环或其他计算浪费,要求每个事务对它可以使用多少个代码执行计算步骤设置一个限制。计算的基本单位是“gas"。通常，一个计算步骤花费1gas,但是某些操作会消耗更多的gas,因为它们在计算上更加昂贵，或者增加了必须存储为状态一部分的数据量。交易数据中的每个字节还需要支付5gas 的费用。收费系统的目的是要求攻击者按比例为他们消耗的每种资源（包括计算，带宽和存储）支付费用；因此，任何导致网络消耗大量这些资源中的任何一笔的交易，其燃气费必须大致 与增量成比例。

3.4 消息

合同具有向其他合同发送"消息"的能力。消息是永远不会序列化的虚拟对象，仅存在于Cosmic Interstellar Public Chain执行环境中。一条消息包含：消息的发件人（隐式）邮件的收件人随消息转移的乙醚量可选数据字段STARTGAS 值。

从本质上讲，消息类似于交易，只是消息是由合同而非外部参与者产生的。当前执行代码的合同执行CALL操作码时，会生成一条消息，该操作会产生并执行一条消息。像交易一样，一条消息会导致收件人账户运行其代码。因此，合同可以以与外部参与者完全相同的方式与其他合同建立关系。
请注意，交易或合同分配的天然气津贴适用于该交易和所有子执行所消耗的总天 然气。例如，如果外部参与者A用1000瓦斯向B发送交易，并且B在向C发送消息之前消耗了 600瓦斯，而C的内部执行在返回之前消耗了300瓦斯，然后B可以再消耗100 瓦斯才能 耗尽瓦斯。

3.5 Cosmic Interstellar Public Chain 状态转换函数



Cosmic Interstellar Public Chain 状态转换函数 APPLY（S, TX）-> S'可以定义如下：

1. 检查交易是否格式正确(即具有正确数量的值),签名有效以及随机数与发 送者账户中的随机数匹配。如果不是，则返回错误。
2. 将交易费用计算为STARTGAS* GASPRICE，然后从签名中确定发送地址。从发件人的账户余额中减去费用,并增加发件人的现时。如果余额不足,请返回错误。
3. 初始化GAS=STARTGAS,然后每字节释放一定数量的gas以支付交易中的字节。
4. 将交易值从发送者的账户转移到接收者的账户。如果收款账户尚不存在，请创建它。如果接收账户是合同，请运行合同的代码以完成或直到执行结束为止。
5. 如果由于发件人没有足够的钱而导致价值转移失败，或者代码执行用尽了钱，请还原除支付费用外的所有状态更改，然后将费用添加到矿工的账户中。

6. 否则，将所有剩余瓦斯的费用退还给发送方，并将支付的瓦斯使用费发送给矿工。

例如，假设合同的代码为：

```
if !self.storage[calldataload(0)]:
self.storage[calldataload(0)] = calldataload(32)
```

请注意，实际上合同代码是用低级EVM代码编写的；为了清楚起见，此示例使 Serpent (一种高级语言)编写，并且可以编译为EVM代码。假设合同的存储空间开始为空，并发送了一个交易，该交易带有10个币，2000个gas, 0.001个币gasprice和64个字节的数据，字节0-31代表数字2和字节32-63 代表字符串CHARLIE .fn。在这种情况下，状态转换功能的过程如下：

1. 检查交易是否有效且格式正确。
2. 检查交易发件人是否至少拥有 2000*0.001=2 币。如果是这样，则从发送者的账户中减去2个币。
3. 初始化gas=2000;假设事务长度为170个字节，字节费用为5,则减去850，从而剩下1150个gas。
4. 从发送者的账户中再减去10个币，然后将其添加到合同的账户中。
5. 运行代码。在这种情况下，这很简单：它检查是否使用了索引2处的合同存储，注意到它没有使用，因此将索引2处的存储设置为CHARLIE值。假设需要187瓦斯，那么剩余的瓦斯量为1150-187=963。
6. 将963*0.001=0.963币加回到发送者的账户，并返回结果状态。如果在交易的接收端没有合同，则总交易费用将仅等于所提供的GASPRICE乘以交易的长度（以字节为单位），并且与交易一起发送的数据将是无关紧要的。

请注意，在还原方面，消息与事务等效地工作：如果消息执行耗尽，则该消息的执行以及该执行触发的所有其他执行都将还原，但父执行不需要还原。这意味着合同调用另一个合同是"安全的"，好像A用Ggas调用B 一样，那么A的执行保证最多损失Ggas。最后，请注意，有一个创建合同的操作码CREATE。它的执行机制通常类似于CALL,不同之处在于执行的输出确定新创建合同的代码。

3.6 代码执行

CosmicInterstellar Public Chain合约中的代码以一种基于堆栈的底层字节码语言编写，称为"Cosmic Interstellar Public Chain虚拟机代码"或“EVM代码”。该代码由一系列字节组成，其中每个字节代表一个操作。通常，代码执行是一个无限循环，包括重复执行当前程序计数器（从零开始）的操作，然后将程序计数器递增1,直到到达代码末尾或出现错误或STOP或检测到RETURN指令。这些操作可以访问三种类型的空间来存储数据：堆栈，一个后进先出的容器，可以将值压入其中并弹出内存，无限扩展的字节数组合同的长期存储，即键　/值存储。与堆栈和内存不同，堆栈和内存在计算结束后会重置，因此存储可以 长期保留。

该代码还可以访问传入消息的值，发送者和数据以及块头数据，并且该代码还可以返回数据的字节数组作为输出。EVM代码的正式执行模型非常简单。Cosmic Interstellar Public Chain虚拟机运行时，其完整的计算状态可以由元组（block_state, transaction，消息，代码，内存，堆栈，pc, gas）定义，其中block_state是包含所有账户并包括余额和存 储的全局状态。在每轮执行的开始，通过获取代码的第pc个字节来找到当前指令（如果pc> = len（code），则为0），并且每条指令在如何影响代码的方面都有其自己的定义。例如，ADD将两个项目从堆栈中弹出，将它们的和推入，将gas减少1,将pc递增1,SSTORE从堆栈中将顶部的两个项目弹出，然后将第二个项目插入到合约的存储中并指定索引第一项。尽管有很多方法可以通过即时编译来优化Cosmic Interstellar Public Chain 虚拟机的执行，但是 Cosmic Interstellar Public Chain 的基本实现可以用几百行代码来完成。

4. 关键技术

4.1 SPos 共识协议

目前还有相当多的区块链网络，比如最著名的比特币和以太坊（ETH 2.0正在向PoS 2.0 迈进），它们都使用了基于非对称算法的工作量证明系统。这些证明极难生成，但第三方验证很简单。这些网络的安全性是通过全网共识来实现的随着后续块添加到区块链中，链的数量变得非常难以创建。然而，这些计算量大的证明对区块链网络没有其他用途。我们将有用性定义为除了保护分布式账本之外对区块链网络有价值的工作。而其他网络试图将挖矿能力转化为有用的东西，例如以太坊执行称为智能合约，大部分工作没有用或可重用。采矿过程也非常浪费，因为工作的决定因素通常是计算能力，这会消耗大量电力并且需要大量硬件来执行。

SPos的共识机制是，对于每个时间段，系统选择一组称为验证者的实体，并在下一个选举周期中扮演一个关键的重要角色，例如区块生产和最终验证。工作的质量和数量都很高,因此需要运行成本高昂的操作以确保高通信响应能力并建立长期、可靠的声誉。因此,验证者还需要质押作为良好行为的保证。当偏离正常轨道时,质押将根据其严重程度受到惩罚。而不是在他们勤奋遵守规则时获得奖励。任何节点完成节点任务或能够满足 requirements 可以公开成为候选验证者。当然,由于各种原因,验证者只能是有限的,前期会是360个,后期根据全网估计可能会达到数百甚至数千发展情况。我们也鼓励 CIPC 持有者尽可能多地加入到web 生态中,以选举人的身份投票给验证人。如果选举人支持的候选人验证人正式进入验证人节点池，选举人将根据比例获得相应的回报。质押的数量和佣金的比例与验证人不同，没有选举人数量。只要选举人只选择并支持具有良好安全实践的候选人验证人，那么风险低且有一个常数收入来源。这种选举人-验证人的设计有非常强的安全保障。它允许系统选择总质押量大的验证节点，淘汰总质押数相对较少的候选者。

事实上，在任何给定的时刻，我们都想要很大一部分的CIPC被质押。这使得敌对组织很难成为验证节点（因为他们需要建立强大的声誉才能获得所需的支持）并且攻击系统的成本很高（因为任何攻击都会导致大量CIPC削减）。我们的共识机制方案比权益证明机制（POW）高效得多，也比权益证明机制（PoS）快得多：它允许几乎所有持有CIPC的参与者持续参与，从而保持高水平的安全性，同时限制验证节点的数量，因此所有基本的网络操作都是有效的。为了防止验证 者和收集者之间的勾结，验证者是定期随机分配的。区块链技术应用一直是最受关注的话题，而作为承载应用的公链，其性能决定了应用的上限，包括以太坊升 级，致力于提升系统性能。

公链的诞生和智能合约为区块链应用提供了载体，但应用场景也受到公链的限制。区块链大规模应用的障碍之一是可扩展性，即区块链的性能。可扩展性限制了区块链的交易吞吐量，导致拥塞和更高的交易费用，这是以太坊经常面临的问题。扩容是解决网络拥塞的根本解决方案，而多层结构是实现扩容的主要方式之一。通过多层结构，数据可以在在不同的网络分区中并行，以提高处理效率比如以太坊的Layer2扩容方案，将计算过程放到链下，将最终结果回传到链上，增强信息处理。

4.2 Layer2 方案

The Layer 2 extension solution on Optimistic Rollups (ORU), ORU有很多理想的特性。在所有区块链可扩展性技术中, 它具有不可比拟的特点, 具有以下特点:

免信任

与传统的侧链技术不同, ORU是免信任的 (或者更专业地说, 信任最小化)。您可以随时从Rollup中提取资金, 而无需信任ORU上的绝大多数区块生产者是诚实的。

许可

与Plasma不同, ORU是免许可的。任何人都可以成为ORU上的区块生产者, 因为rollup上的所有区块数据都发布在以太坊上, 并且可以从以太坊获得。如何选择下一个领导者是一个具体的实施问题, 而不是一个基本的限制。

不受管理

由于ORU既受信任又无需许可证, 您可以随时提款, 没有人可以阻止您。因此, ORU是免费托管的。

表达性强

与ZK　rollup不同, ORU (从理论和实践上) 表达性强。无论是类似比特币的UTXO支付还是成熟兼容的EVM执行, ORU都能处理。

开放参与

与支付渠道不同，ORU支持智能合约并且对所有人开放，就像Uniswap一样。

资金效率高

与支付渠道不同，ORU不需要用户提前锁定资金。

抗链拥塞

与支付通道和Plasma不同，ORU可以防御链上的拥塞，因为ORU欺诈是在区块层面证明的，而不是像支付通道那样的关闭机制，或者像Plasma这样的退出机制。

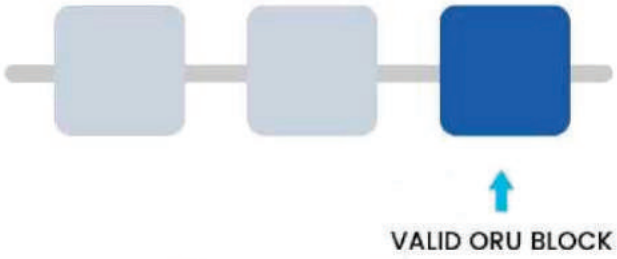
不需要新的密码学

与ZK rollup不同，ORU不需要任何新的密码学。

快速（非即时）确定性

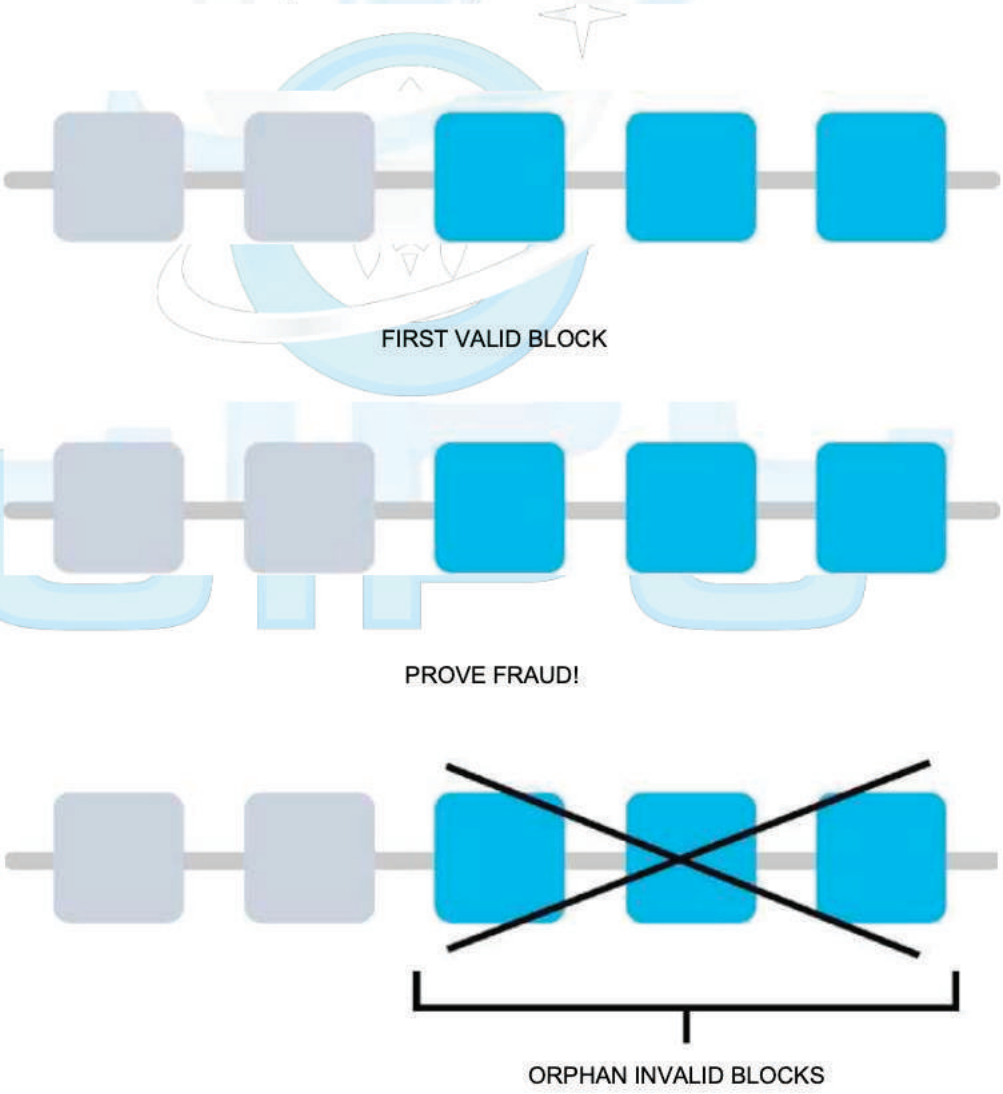
与ZK rollup不同，ORU不需要生成证明，ORU块可以立即发布到以太坊。由于有效的 ORU块无法回滚，因此像以太坊一样的，一旦最终确定，它们就可以使用被发布到以太坊。

CASE 1 : **VALID BLOCK**



CASE 2 : **INVALID BLOCK**

1..One or more invalid rollup blocks are posted to Ethereum



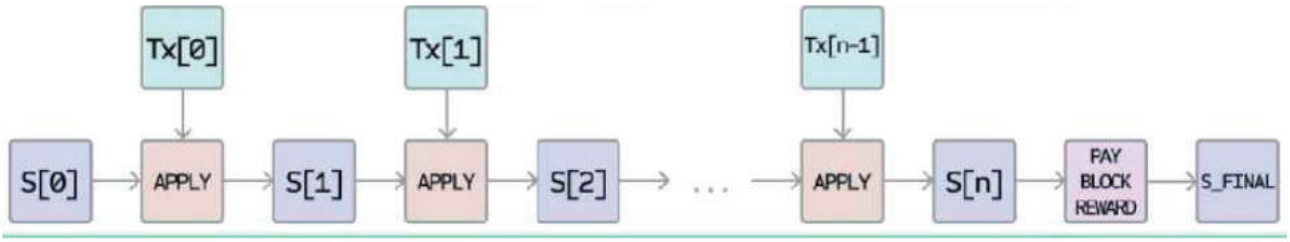
1. 聚合器收集rollup上的交易，将它们打包到rollup块中，并将rollup块与押金（我们将很快解释为什么需要押金）一起发送到以太坊上的智能合约（或另一个类似以太坊的区块链运行大量的智能合约）。这个汇总块不会被翻译或执行——智能合约只记录块哈希并跟踪所有汇总块的哈希。Rollup区块本身并不存储在智能合约中，但每个人都可以在以太坊的历史交易中找到它们。

2. Rollup块包含一个状态根，即rollup 块的状态树的根。如果状态根无效，任何人都可以在挑战期间使用欺诈证明来证明它是无效的。这可能是因为汇总块中的事务无效，或者因为状态根无效。如果一个rollup块被证明是无效的，合约会回滚rollup链，无效块后面的所有rollup块都将成为孤立块。一旦证明欺诈成功，部分押金将支付给验证者，其余部分将被销毁。

3. 如果在挑战期结束前没有提交欺诈证明，合约将敲定rollup区块并允许聚合器提取存款。当用户从Rollup链提款到主链时，需要在Rollup链上发起提款请求，只有在合约中最终确定汇总块后才能提取资金。

4.3 BFT区块链算法

BFT(ByzantineFaultTolerance)解决了原始拜占庭容错算法效率不高的问题，　将算法复杂度由节点数的指数级降低到节点数的平方级，使得拜占庭容错算法在实际系统应用中变得可行。每个见证人出块时依然全网广播，其他见证人收到新区块后，立即对此区块进行验证，并将验证签名完成的区块立即返回出块见证人，不需等待其他见证人自己出块时再确认。从当前的出块见证人看来，他生产了一个区块，并全网广播，然后陆续收到了其他见证人对此区块的确认，在收到2/3见证人确认的瞬间，所以交易时间只需要3秒。



Cosmic Interstellar Public Chain 区块链在很多方面都与比特币区块链相似，

尽管它确实有所不同。Cosmic Interstellar Public Chain与比特币在区块链架构方面的主要区别在于，与比特币（仅包含交易清单的副本）不同，Cosmic Interstellar Public Chain区块包含交易清单和最新状态的副本。除此之外，其他两个值、块号和难度，也存储在块中。Cosmic Interstellar Public Chain 中的基本块验证算法如下：

1. 检查所引用的前一个块是否存在并且有效。
2. 检查该块的时间戳是否大于所引用的先前块的时间戳，并且距未来不超过15分钟。
3. 检查区块编号，难度，交易根，叔块根和gas限制（各种Cosmic Interstellar Public Chain 特定的低级别概念）是否有效。
4. 检查块上的工作证明是否有效。
5. 令S[0]为上一个块末尾的状态。
6. 令TX为区块的交易清单，包含n个交易。对于0... n-1中的所有i，设置S[i+1]=APPLY（S[i]，TX[i]）。如果有任何应用程序返回错误，或者直到该点为止在区块中消耗的总gas超过GASLIMIT，则返回错误。
7. 令S_FINAL为S[n],但加上支付给矿工的块奖励。
8. 检查状态S_FINAL的Merkle树根是否等于块头中提供的最终状态根。如果是，则该块有效；否则，该块无效。

一个常见的问题是合同代码在物理硬件上的“执行位置”。这有一个简单的答案：

执行合同代码的过程是状态转换函数定义的一部分,状态转换函数是块验证算法的一部分，因此，如果将事务添加到块B中，则由该事务产生的代码执行将被执行。由现在和将来的所有节点执行，这些节点下载并验证块B。

5. 应用领域

通常，Cosmic Interstellar Public Chain 之上有三种类型的应用程序。第一类是金融应用程序，它为用户提供了更强大的方式来使用他们的资金来管理和订立合同。其中包括子货币、金融衍生产品、对冲合同、储蓄钱包、遗嘱，甚至最终包括某些类别的全面雇佣合同。第二类是半金融应用，涉及金钱，但所进行的工作也有大量的非货币方面。一个完美的例子是用于解决计算问题的自我执行赏金。最后，有些应用程序（例如在线投票和去中心化治理）根本就不具备财务能力。

5.1 代币系统

区块链上的代币系统有很多应用，从代表美元或黄金等资产的子货币到公司股票，代表智能财产的单个代币，安全的不可伪造的优惠券，甚至是与传统价值完全没有联系的代币系统，都可以有作用点。令牌系统非常容易在Cosmic Interstellar Public Chain中实现。要了解的关键点是货币或令牌系统m，从根本上讲是一个具有一个操作的数据库：从A减去X单位并将B给予X单位，条件是（1）A在交易之前至少具有 X 单位，并且（2）交易已由A批准，全部实现令牌系统所需的就是将这种逻辑实现到合同中。

在 Serpent 中实现令牌系统的基本代码如下：

```
def send (to, value) :
if self.storage[msg.sender] >= value:
self.storage[msg.sender] = self.storage[msg.sender] - value self.storage[to]
= self.storage[to] + value
```

这本质上是本文上面进一步描述的“银行系统”状态转换功能的字面实现。首先，需要添加一些额外的代码行来提供分配货币单位的初始步骤，以及其他一些极端情况，并且理想

情况下，将添加一个功能，以使其他合同可以查询地址的余额，但这就是全部。从理论上讲，基于Cosmic Interstellar Public Chain的代币系统可能充当子货币，可能具有链上基于比特币的元货币所缺乏的另一个重要功能：直接以该货币支付交易费用的能力。实施该方法的方式是，该合同将保持币余额，用它可以退还用于向发送方支付费用的币，并且它将通过收取所收取的内部货币单位并将其转售来补充该余额。持续不断的拍卖。因此，用户将需要使用币"激活"他们的账户，但是一旦有币就可以重用，因为合同每次都会将其退还。

5.2 金融衍生品和稳定价值货币

金融衍生产品是“智能合约”的最常见应用，也是在代码中最简单的实现之一。实施金融合同的主要挑战是，大多数金融合同都需要参考外部价格行情；例如，一个非常理想的应用是智能合约，该合约可以对冲 Cosmic Interstellar Public Chain（或另一种加密货币）相对于美元的波动，但是这样做需要合约知道CIPC/USD的价值。最简单的方法是通过由特定参与方（例如纳斯达克）维护的“数据馈送”合同来设计，以便该参与方能够根据需要更新合同，并提供允许其他合同发送合同的接口，消息发送到该合同，然后返回提供价格的响应。

考虑到这一关键因素，对冲合同将如下所示：
1. 等待甲方输入1000个币。
2. 等待乙方输入1000个币。
3. 在存储中记录通过查询数据馈送合约计算得出的1000币的美元价值，说这是$x。

4. 30天后，允许A或B "重新激活"合约，以便将价值$x的币（通过再次查询数据馈送合约以获取新价格来计算）发送给A,其余部分发送给B。
这样的合同在加密商务中将具有巨大的潜力。关于加密货币的主要问题之一是其易变的事实。尽管许多用户和商家可能希望使用安全性和便利性来处理加密资产，但他们可能不希望面对一天损失其资金价值23%的前景。到目前为止，最常见的解决方案是发行人支持的资产。这个想法是，发行人创建一种子货币，他们有权发行和撤销单位，并向其提供（离线）某单位指定基础资产（例如黄金）的任何人提供一种货币单位美元）。发行人然后承诺将向所有向后退回一单位加密资产的人提供一单位基础资产。这种机制允许将所有非加密资产“提升”为加密资产，前提是可以信任发行者。

然而实际上，发行人并不总是值得信赖的，在某些情况下，银行基础设施太弱或太敌对，以致无法提供此类服务。金融衍生品提供了另一种选择。在这里，一个分散的投机者市场，而不是由单个发行人提供资金来支持一项资产，而是押注加密参考资产（例如CIPC）的价格将上涨，而发挥了这一作用。与发行人不同，　投机者没有选择违约的理由，因为对冲合约将其资金托管在托管人手中。请注意，这种方法并未完全分散，因为仍然需要可信的来源来提供价格行情，尽管可以说，即使在降低基础架构要求方面，这仍然是一个巨大的进步（与作为发行人不同，发行价格供稿不需要许可证并可能被归类为言论自由），并减少了欺诈的可能性。

5.3 身份和信誉系统

最早的替代加密货币Namecoin尝试使用B类似itcoin的区块链提供名称注册系统，用户可以在其中将其名称与其他数据一起注册到公共数据库中。引用的主要用例是用于DNS系统,将域名（例如"bitcoin.org"（或在Namecoin 的情况下为“ bitcoin.bit"））映射到IP地址。其他用例包括电子邮件身份验证和可能更高级的信誉系统。这是在Cosmic Interstellar Public Chain 上 提供类似于Namecoin的名称注册系统的基本合同：

```
def register (name, value) :
        if !self.storage[name]:
                self.storage[name] = value
```

合同很简单，它只是 Cosmic Interstellar Public Chain 网络内部的一个数据库，可以添加到该数据库中，但不能对其进行修改或从中删除。任何人都可以注册一个具有一定价值的名称，然后该注册将永久保留。更复杂的名称注册合同还将具有一个“功能条款”，允许其他合同对其进行查询，以及名称“所有者”（即第一注册者）更改数据或转让所有权的机制，甚至可以在其顶部添加信誉和信任网络功能。

5.4 去中心化文件存储

IPFS目标是超过甚至取代HTTP,通过一个文件系统，将网络中所有的设备连接起来，构建一个更快速、更安全、开放分享的互联网。
CIPC团队认为，低成本高速的IPFS存储方案必然是未来去中心化应用网络的必要基础设施。IPFS网络兼具去中心化与成本。在数字化生活中，隐私越来越为人所重视，去中心化存储必将成为未来人们的必需品。iCloud曾经被黑客利用云端系统漏洞，使得50多位好莱坞女星的裸照泄露。这并不是孤例，只要互联网的存储依然是中心化的，这样的事情依然会持续不断地发生。

CIPC内置LibP2P协议模块。LibP2P被用作IPFS的网络层。使用LibP2P的一个节点连接到另一个节点进行通信，能够使用各种不同的传输方式，这些传输方式包括连接延迟，以及通过不同协议进行通信，并根据需求进行协商。

LibP2P模块在IPFS中主要负责数据的传递功能，即路由、网络、交换等。DCE以保护用户数据隐私与安全为己任，将提供永续高效低成本的存储方案给DCE 用户。

该协议的一项重要功能是，尽管似乎可以信任许多随机节点而不是决定忘记文件，但可以通过秘密共享将文件分成许多部分，从而将这种风险降低到接近零。观看合同以查看每件作品仍在某个节点的财产中。如果合同仍在付款，则可以提供加密证明，证明有人在那里仍在存储文件。

5.5 去中心化的自治组织

“去中心化的自治组织”的一般概念是指具有一定成员或股东集合的虚拟实体，这些成员或股东可能拥有多数成员的67%的权利，可以使用该实体的资金并修改其代码。成员将集体决定组织应如何分配其资金，分配DAO资金的方法可能从赏金，薪水到甚至更奇特的机制(例如内部货币来奖励工作)等等。这实质上是复制传统公司或非营利组织的法律陷阱，但仅使用加密区块链技术来执行。到目前为止，有关DAO的讨论主要围绕"去中心化自治公司"(DAO)的"资本主义"模型，该模型具有接收股息的股东和可交易的股票。另一种可能称为"权力下放的自治社区"的选择将使所有成员在决策中享有平等的份额，并要求现有 成员的67%同意添加或删除成员。这样，一个人只能拥有一个成员身份的要求 将需要由该组集体执行。有关如何对DAO进行编码的一般概述如下。最简单的设计只是一段自我修改的代码，如果三分之二的成员都同意更改，则该代码会更改。尽管从理论上讲代码是不可变的，但人们可以轻松解决此问题并事实上通过将代码块包含在单独的合同中，并将要调用的合同的地址存储在可修改的存储中，从而提高了可操作性。在这种DAO合同的简单实现中，将存在三种交易类型，通过交易中提供的数据加以区分：

[0,i,K,V]使用索引i注册提案，以将存储索引K上的地址更改为值V[1,i]对提案i进行投票[2,i]如果已经进行了足够的投票，则最终确定提案i，然后,合同将为每个条款提供条款，它将保留所有开放式存储更改的记录,以及谁投票支持这些更改的列表，它还将具有所有成员的列表。当任何存储更改获得三分之二的成员投票时，完成交易即可执行更改。一个更复杂的框架还将具有对诸如发送交易，添加成员和删除成员之类的功能的内置投票功能，甚至可以提供“流动民主”式的投票委派（即，任何人都可以指派某人为其投票和分配），是传递性的，因此如果A分配B，B分配C,则C决定A的投票）。这种设计将使DAO成为一个分散的社区，从而有机地成长，使人们最终可以将过滤掉谁是成员的任务委托给专家，尽管与"当前系统"不同，专家可以随着时间的推移轻松地出现和消失，随着个别社区成员改变其阵营。

另一种模型是一种分散的公司，其中任何账户都可以拥有零个或更多的股份，并且需要三分之二的股份才能做出决定。一个完整的框架将涉及资产管理功能，提供买卖股票的要约的能力以及接受要约的能力（最好具有合同内的订单匹配机制）。代表团也将以流动民主的方式存在，从而概括了"董事会"的概念。

6. 进一步的应用

6.1 储蓄钱包

假设爱丽丝想确保自己的资金安全,但担心自己会丢失或有人会窃取她的私钥。
她将币与银行鲍勃的合同如下:
爱丽丝一个人每天最多只能提取1%的资金。鲍勃一个人每天最多可以提取1%的资金,但是爱丽丝有能力用她的钥匙进行交易以关闭该功能。爱丽丝和鲍勃在一起可以撤回一切。通常,每天1%的收入足以满足爱丽丝的需求,如果爱丽丝想提款更多,可以联系鲍勃寻求帮助。如果爱丽丝(Alice)的钥匙被黑客窃取,她将奔向鲍勃(Bob)将资金转移到新合同中。如果她丢失了钥匙,鲍勃最终将把钱取出。如果鲍勃被证明是恶意的,那么她可以关闭他的退出能力。

6.2 作物保险

使用天气数据而不是任何价格指数,可以轻松地订立金融衍生品合约。如果爱荷华州的农民购买的衍生品根据爱荷华州的降水量反向支付,那么如果发生干旱,农民将自动收钱,如果有足够的雨水,农民会很高兴,因为他们的收成很好。一般而言,这可以扩展到自然灾害保险。

6.3 分散的数据提要

对于有差异的金融合约,实际上可能可以通过称为SchellingCoin的协议分散数据馈送。SchellingCoin的基本工作原理如下:N个参与者都将给定数据的值(例如CIPC/USD价格)放入系统中,对这些值进行排序,并且介于25%和75%之间的每个人都会获得一个令牌作为奖励。每个人都有提供其他人将提供的答案的动机,并且许多参与者可以实际达成共识的唯一价值是显而易见的默认值:事实。这创建了一个去中心化的协议,该协议理论上可以提供任何数量的值,包括CIPC/USD价格,柏林的温度,甚至是特定硬计算的结果。

6.4 智能多重签名托管

比特币允许多重签名交易合同,例如,给定五个密钥中的三个可以花费资金。

Cosmic Interstellar Public Chain允许更多的粒度;例如,五分之四的人可以花掉所有东西,五分之三的人每天可以10%,五分之二的人每天可以花0.5%。此外,Cosmic Interstellar Public Chain多重签名是异步的-双方可以在不同时间在区块链上注册其签名,最后一个签名将自动发送交易。

6.5 云计算

EVM技术还可以用于创建可验证的计算环境，允许用户要求其他人执行计算，然后可选地要求提供证明，以确保在某些随机选择的检查点进行的计算正确完成。这允许创建一个云计算市场,任何用户都可以使用其台式机,笔记本电脑或专业服务参与其中。抽查和保证金可以用来确保系统值得信赖（即，节点无法从中牟取暴利）。尽管这样的系统可能并不适合所有任务；例如,要求高级别的进程间通信的任务很难在庞大的节点云上完成。但是,其他任务则更容易并行化。像SETI @home, folding @home 和遗传算法这样的项目都可以在这样的平台上轻松实现。

6.6 点对点竞猜

可以在Cosmic Interstellar Public Chain区块链上实现许多对等竞猜协议，例如Frank Stajano和Richard Clayton的Cyberdice。最简单的竞猜协议实际上只是下一个区块哈希上的差异契约，并且可以从那里建立更高级的协议，从而以几乎为零的费用创建博彩服务，并且无法作弊。预测市场。如果提供了Oracle或SchellingCoin，则预测市场也很容易实现，并且预测市场与SchellingCoin一起可能被证明是futarchy作为分散组织的治理协议的第一个主流应用。

7. 杂项和关注点

7.1 修改的 GHOST 实现

"贪婪的最重观测子树”(GHOST)协议是Yonatan Sompolinsky和Aviv Zohar 于 2013 年12月首次提出的一项创新。GHOST 背后的动机是,由于过时的高速率,确认时间短的区块链目前遭受安全性降低的困扰-因为如果矿工A挖了一个区块,然后矿工B恰好在矿工A的区块传播到B之前挖了另一个 区块,那么矿工B的区块将最终被浪费,并且不会对网络安全做出贡献。此外,还有一个集中化问题:如果矿工A是具有30%哈希功率的采矿池,而B具有10%哈希功率,则A将有70%的时间产生陈旧区块的风险(因为其他30% 的时间A生成了最后一个块,因此将立即获得挖掘数据),而B则有90%的时间生成过时的块的风险。因此,如果块间隔足够短以致于过时率较高,则又由于其太小,A将实质上更有效。结合这两种效果,快速产生区块的区块链很可能导致一个采矿池具有足够大的网络哈希能力百分比,从而可以对采矿过程进行实际控制。

如Sompolinsky和Zohar 所述, GHOST通过在计算哪条链是“最长”链时包括陈旧的块,解决了网络安全损失的第一个问题。也就是说,不又要计算区块的父代和其他祖先,而且还要计算区块祖先的陈旧后代(Cosmic Interstellar PublicChain 行话中的“叔块”),以计算哪个区块具有最大的总工作量证明它。为了解决集中化偏见的第二个问题,我们超越了Sompolinsky和Zohar所描述的协议,并且还为旧货提供了块奖励:陈旧的块获得其基本奖励的87.5%, 而包括陈旧块的侄子则获得剩余12.5%。但是,交易费用并不授予叔块。

Cosmic Interstellar Public Chain 实现了GHOST的简化版本,仅下降了七个级别。具 体来说,定义如下:

一个块必须指定一个父对象,并且必须指定0个或多个叔块中包含的一个叔块必须具有以下属性:

它必须是B的第k代祖先的直接子代，其中2<=k<=7。它不能是B的祖先—个叔块必须是一个有效的块头，但不必是先前验证过的甚至是有效的块，一个叔块必须与先前区块中包含的所有叔块以及同一区块中包含的所有其他叔块不同（非双重包含）对于B区块中的每个U叔块，B的矿工获得的币库奖励增加3.125%，U的矿工获得标准币库奖励的93.75%。

使用GHOST的这种受限版本具有最多7代的叔块，这是出于两个原因。首先,无限的GHOST会在计算给定块的哪些叔块有效的过程中包含太多的复杂性。其次，Cosmic Interstellar Public Chain 中使用的具有补偿的无限GHOST消除了矿工在主链而不是在公共攻击者链上采矿的动机。

7.2 费用

由于发布到区块链中的每笔交易都在网络上增加了下载和验证交易的成本,因此需要某种监管机制(通常涉及交易费用)来防止滥用。比特币中使用的默认方法是收取纯自愿性费用,依靠矿工充当看门人并设定动态最低费用。这种方法在比特币社区,尤其是在 becau中受到了好评。因为它是“基于市场的”,所以允许矿工和交易发送方之间的供需确定价格。但是,这种推理方式的问题在于交易处理不是市场。尽管将矿工提供给发送者的服务理解为交易处理在直观上很有吸引力,但实际上矿工包括的每笔交易都需要由网络中的每个节点进行处理,因此交易的绝大部分成本处理工作由第三方决定,而不是 由是否决定将其包括在内的矿工承担。因此,极有可能发生常见的悲剧性问题。

但是,事实证明,在基于市场的机制中存在此缺陷时,如果给出特定的不准确简化假设,则会神奇地将其自身抵消。参数如下,假设:

1. 交易导致k项操作,将奖励kR提供给包含该奖励的任何矿工,其中R由发送者设置,并且k和R事先(大致)对矿工可见。
2. 一个操作对任何节点的处理成本为C (即,所有节点的效率相同)
3. 有N个采矿节点,每个节点具有完全相等的处理能力(即总数的1/N)
4. 不存在非挖掘的完整节点。

如果预期的回报大于成本,那么矿工将愿意进行交易。因此,由于矿工有1/N的机会处理下一个区块,因此预期回报为kR/N,而矿工的处理成本仅为kC。因此,矿工将包括 kR/N>kC或R>NC的交易。注意,R是发送方提供的每笔操作费用,因此是发送方从交易中获得的收益的下限,而NC是整个网络一起处理操作的成本。因此,矿工有动机只包括那些总功利超过成本的交易。但是,实际上与这些假设有几个重要的偏差:

1. 矿工的确比其他验证节点支付了更高的交易费用，因为额外的验证时间会延迟区块传播，从而增加区块变陈旧的机会。
2. 确实存在非挖掘的完整节点。
3. 在实践中，采矿权的分配可能最终从根本上是不平等的。
4. 确实存在投机者，政治敌人和疯狂者，其效用功能包括对网络造成损害，并且他们可以巧妙地建立合同，而其成本要比其他验证节点支付的成本低得多。

(1)使矿工倾向于减少交易(2)增加NC；因此，这两个效果至少部分相互抵消。为了解决这些问题，我们只需设置一个浮动上限，没有块可以进行的操作超过BLK_LIMIT_FACTOR乘以长期指数移动平均数。具体来说：

blk.oplimit=floor((blk.parent.oplimit*(EMAFACTOR-1)+-floor(parent.opcount* BLK_LIMIT_FACTOR))/EMA_FACTOR)

BLK_LIMIT_FACTOR和EMA_FACTOR 是常量，暂时将设置为65536和1.5，但在进一步分析后可能会更改。

还有另一个因素无法激励比特币中的大块：大块将花费更长的时间传播，因此变旧的可能性更高。在Cosmic Interstellar Public Chain 中，高耗气量区块的传播也可能需要更长的时间，这是因为它们在物理上更大，并且因为它们需要更长的时间来处理交易状态转换以进行验证。这种延迟抑制因素在比特币中是一个重要的考虑因素，但由于GHOST协议，因此在Cosmic Interstellar Public Chain中考虑较少。因此,依赖于受监管的街区限制可提供更稳定的基准。

7.3 计算与图灵完备性

一个重要的注意事项是，Cosmic Interstellar Public Chain虚拟机是图灵完备的。这意味着EVM代码可以对任何可以想象得到的计算进行编码，包括无限循环。EVM代码允许以两种方式循环。首先，有一条JUMP指令允许程序跳回到代码中的前一个位置，还有一条JUMPI指令进行条件跳转，允许诸如x<27 的语句：x=x*2。其次，合同可以调用其他合同，可能允许遍历递归。这自然 会带来一个问题:恶意用户是否可以通过迫使矿工和整个节点进入无限循环而使其本质上关闭?出现此问题是由于计算机科学中的一个问题（称为"停止问题”：在一般情况下，无法判断给定程序是否会停止。
如状态转换部分所述,我们的解决方案通过要求事务设置允许执行的最大计算步骤数来工作,如果执行花费更长的时间,则可以恢复计算,但仍需支付费用。消息以相同的方式工作。为了显示我们解决方案背后的动力,考虑以下示例：
攻击者创建运行无限循环的合同,然后将激活该循环的交易发送给矿工。矿工将处理交易,运行无限循环,并等待其用尽天然气。即使执行用完了并且中途停止了,交易仍然有效,并且矿工仍然为每个计算步骤向攻击者索要费用。

攻击者创建了一个非常长的无限循环，目的是迫使矿工长时间保持计算，以至于在计算完成时将出现更多的块，并且矿工不可能将交易包括在内索取费用。但是，攻击者将需要为 STARTGAS 提交一个值，以限制执行可以执行的计算步骤的数量，因此，矿工将提前知道该计算将采取过多的步骤。
攻击者看到的合同带有某种形式的代码，例如 send(A, contract.storage[A]);
contract.storage[A]=0,并发送一笔交易量足以执行第一步但不能执行第二步的交易(即进行提款但不让余额减少)。合同作者无需担心会受到此类攻击，因为如果执行在更改的途中停止，则它们将被还原。

财务合同通过获取9个专有数据提要的中位数来工作，以最大程度地降低风险。攻击者接管了其中一个数据馈送，该数据馈送旨在通过DAO　—节中介绍的可变地址调用机制进行修改，并将其转换为运行无限循环，从而试图迫使试图从中索取资金的任何尝试财务合同用完了。但是，金融合同可以在消息上设置限制，以防止出现此问题。

图灵不完全性的另一种选择是图灵不完全性，其中不存在JUMP和JUMPI,并且在任何给定时间都只能在调用堆栈中存在每个合同的一个副本。使用此系统，可能不需要描述的费用系统以及围绕我们解决方案的有效性的不确定性，因为执行合同的成本将受到合同规模的限制。另外，图灵不完整甚至没有那么大的限制。在我们内部构思的所有合同示例中，到目前为止只有一个需要循环，甚至可以通过对一行代码进行 26 次重复来消除该循环。考虑到图灵完成的严重含义和有限的收益，为什么不简单地使用图灵不完整的语言呢?但是，实际上，图灵不完整远非一个巧妙的解决方案。要了解原因，请考虑以下合同：

C0: call(C1); call(C1);
C1: call(C2); call(C2);
C2: call(C3); call(C3);
C49: call(C50); call(C50);
C50: (run one step of a program and record the change in storage)

现在，将交易发送给Ao因此，在51个交易中，我们有一个合同，占用250个　计算步骤。矿工可以通过在每个合同旁边保持一个值来指定其可以采取的最大计算步骤，并为递归调用其他合同的合同计算该值，从而尝试提前发现此类逻辑炸弹，但这将要求矿工禁止创建的合同其他合同(因为上面所有26个合同的创建和执行都可以轻松地合并为一个合同)。

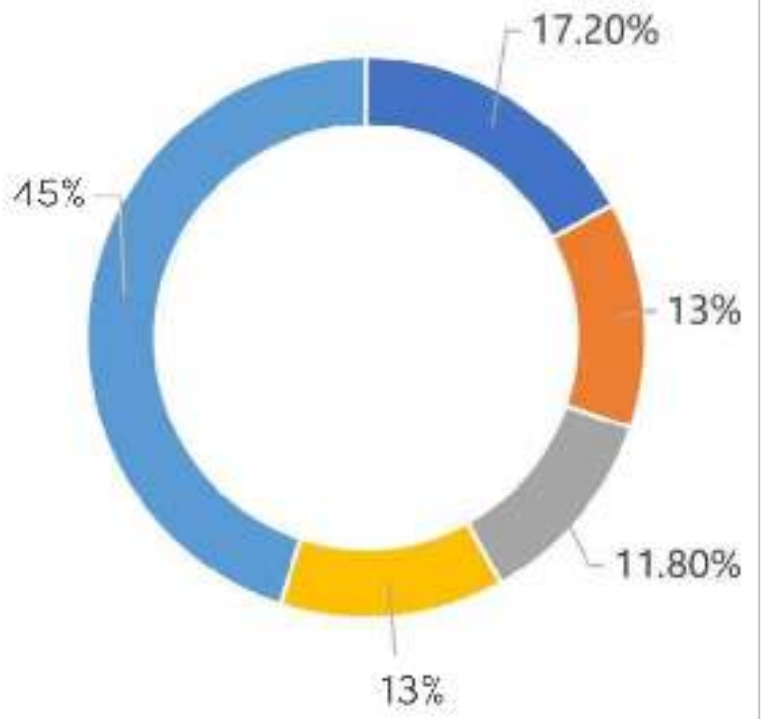
另一个有问题的点是消息的地址字段是一个变量，因此通常甚至可能无法提前知道给定合同将调用哪些其他合同。因此，总而言之，我们得出一个令人惊讶的结论：图灵完备性令人惊讶地易于管理，　而缺乏图灵完备性同样令人惊讶地难以管理，除非存在完全相同的控制措施，但是在那种情况下为什么不仅仅让协议图灵完整?

8. CIPC 的发行机制

Cosmic Interstellar Public Chain网络包括其自己的内置货币CIPC,其双重目的是提供主要的流动性层，以允许在各种类型的数字资产之间进行有效交换，更重要的是，提供了一种支付交易费用的机制。
发行模型如下:

发行代币Cl PC总量97亿枚

- 创始节点17.2%
- 基金会13%
- 种子销售11.8%
- 团队节点13%
- 社区45%

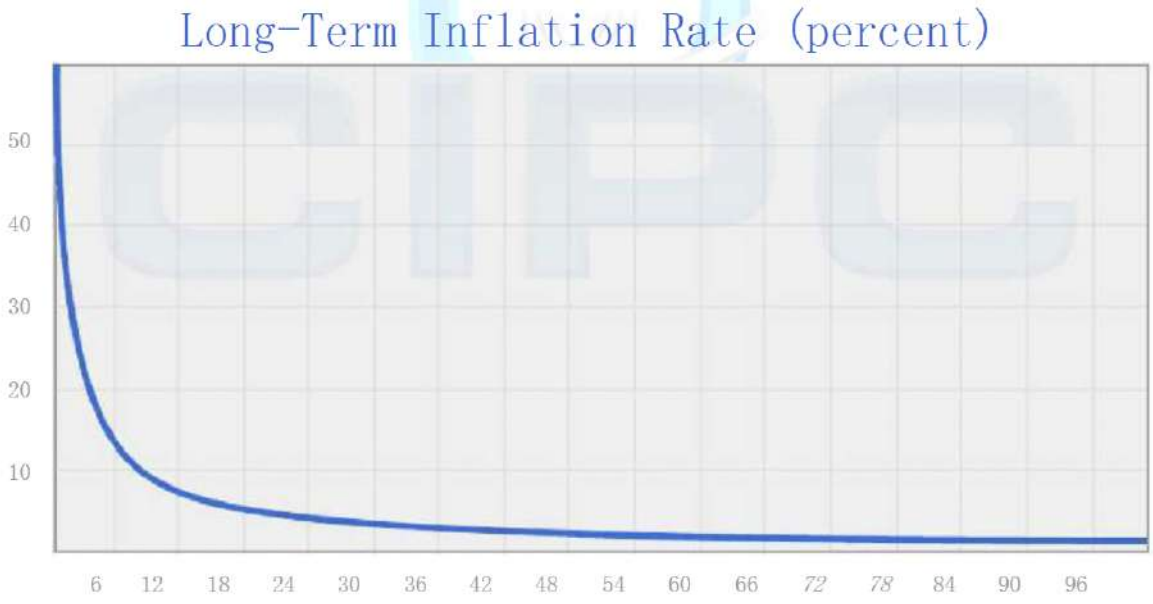


CIPC 币将以协定价格出售，一种旨在资助Cosmic Interstellar Public Chain组织并支付开发费用的机制，该机制已被其他平台（例如COINLIST）成功使用，较早的买家将受益于更大的折扣。出售所得的资金将全部用于向开发人员支付薪水和赏金，并将其投资到 Cosmic Interstellar Public Chain 和加密货币生态系统中的各种营利性和非营利性项目中。

从那时起，每年永久性将总销售量的0.26倍分配给矿工。

CIPC	发射时	1 年后	5 年后
货币单位	1.198 倍	1.458 倍	2.498 倍
买方	83.5%	68.6%	40.0%
售前储备支出	8.26%	6.79%	3.96%
售后二手储备	8.26%	6.79%	3.96%
矿工	0%	17.8%	52.0%

长期供应增长率(百分比)



尽管货币是线性发行的，就像比特币随着时间的推移，供应增长率仍然趋于零。上面模型中的两个主要选择是（1）捐赠基金的存在和规模，以及（2）永久增长的线性供应，而不是像比特性发行减少到0.217倍以提供相同的通货膨胀率，那么Cosmic Interstellar Public Chain 的总量将减少16.5%，因此每个单位的价值将增加 19.8%。

永久线性供应增长模型降低了某些人认为比特币过度集中财富的风险，并为生活在当今和未来时代的人们提供了获取货币单位的公平机会,同时又保留了获取和持有货币的强大动力醚，因为"供应增长率"的百分比仍会随时间趋向于零。我们还得出理论，因为硬币总是由于粗心，死亡等原因而随时间流失，并且可以将硬币损失建模为每年总供应量的百分比，因此流通中的总货币供应量实际上将最终稳定在一个值上，等于年度发行量除以损失率（例如，损失率为1%,一旦供应量达到26倍，则将开采0.26倍，每年损失0.26倍，从而建立平衡）。币那样有上限的供应。捐赠池的理由如下，如果捐赠基金不存在，并且线

9. 发展规划

2022年11月21日，宙际公链参与并成为土耳其全球区块链行业管理委员会的管理职能部门之一，也被土耳其政府指定为国家使用的公链，与土耳其财政国库部、金融犯罪调查委员会、资本市场委员会、中央银行共同合作执行，参与到区块链行业管理，帮助发展加密货币行业，并开展国际加密货币、交易所的监测和管理。

现实世界硬件与区块链和原生代币的紧密结合是一种新颖而有价值的创新，可以应用于其他类型的网络和物理层。区块链的未来不在于谁拥有最大的算力或获得最便宜的算力，而在于区块链中的挖矿证明与提供有价值且可验证的服务相关联。

CIPC已采取或打算采取并启动多项开发流程，包括但不限于：

- 研究将这些想法应用于其他硬件物理层（如智能手机、汽车、家用电器）的适用性
- 探索通过类似设计提供5g60GHz+毫米波连接的潜力
- 研究并实施更多共识证明，以确保CIPC网络在开发过程中保持安全
- 激励系统的博弈论分析，探索kubernetes集群管理容器更高的cap
- 矿工使用的评分算法
- 创建和发布CIPC无线规范
- 制造相对专用的设备模块，以便在CIPC网络启动时可用
- 研究超越基本CIPC原语的智能合约环境的部署
- 可扩展智能合约的强大架构和设计模式
- 区块链互操作性
- 前向纠错技术的持续工作和发展

10. 结论

Cosmic Interstellar Public Chain 协议最初被认为是加密货币的升级版，通过高度通用的编程语言提供高级功能，例如区块链托管、提款限制、金融合约、博彩市场等。Cosmic Interstellar Public Chain协议不会直接“支持”任何应用程序,但是图灵完备的编程语言的存在意味着理论上可以为任何交易类型或应用程序创建任意合约。然而，Cosmic Interstellar Public Chain更有趣的是，Cosmic Interstellar Public Chain协议已经远远超越了货币。在许多其他此类概念中，围绕分散文件存储、分散计算和分散预测市场的协议有可能极大地提高计算行业的效率，并通过添加以下内容来极大地促进其他点对点协议的发展，第一层是经济层，最后，还有大量与钱无关的应用程序。

Cosmic Interstellar Public Chain协议实现的任意状态转换功能的概念为平台提供了独特的潜力。Cosmic Interstellar Public Chain不是旨在用于数据存储，博彩或金融中特定应用的封闭式，单一用途协议，而是以设计为开放式的，我们认为它非常适合作为基础设施，在未来几年中，该协议层可用于大量的金融和非金融协议。

区块链发展遇到性能、可扩展性瓶颈，急需下一代突破性的平台，Cosmic Interstellar Public Chain 以现代密码学、通讯技术、分布式计算为基础，承载去中心化金融、社交、电商、搜索、存储商业应用，构建大规模商用高性能的区块链生态。

Cosmic Interstellar Public Chain 像许多社区驱动的开源软件项目一样，自最初成立以来就已经发展起来。要了解 Cosmic Interstellar Public Chain 的最新发展以及如何对协议进行更改，我们建议您使用本指南。