

Titan技术白皮书 1.0

目录

目录.....	2
1. 引言.....	5
1.1. 问题领域.....	6
1.2. 论文结构.....	7
2. 闲置边缘资源.....	8
2.1. 特性.....	8
2.1.1. 多样性(异构性).....	8
2.1.2. 稳定性.....	8
2.1.3. 区域性.....	9
2.1.4. 数量庞大(规模).....	9
2.1.5. 统一的复杂性.....	9
2.2. 供应方担忧.....	10
2.2.1. 基础设施安全风险.....	10
2.2.2. 收入不透明.....	10
2.3. 需求方担忧.....	11
2.3.1. DePIN 基础设施的可靠性与技术开销.....	11
2.3.2. DePIN 基础设施的安全性.....	11
2.3.3. 成本与透明度.....	11
2.4. Titan 解决方案.....	12
2.4.1. 解决闲置资源多样性.....	12
2.4.2. 解决闲置资源稳定性与可靠性.....	12
2.4.3. 解决闲置资源区域性.....	13
2.4.4. 解决闲置资源规模.....	13
2.4.5. 解决闲置资源共享和租赁的安全风险.....	13
2.4.6. 解决闲置资源成本和奖励分配问题.....	14
3. Titan 网络组件.....	15
3.1. 架构.....	15
3.2. Titan Agent.....	17
3.2.1. 支持多种设备类型.....	17
3.2.2. 闲置资源标准化.....	17
3.2.3. 资源配额调度.....	18
3.3. Titan DAO.....	19
3.4. Titan Harbour.....	20
3.4.1. 接受 Titan DAO 的监督.....	21
3.4.2. 资源接入与管理.....	21
3.4.3. 资源验证与信誉向量.....	21

3.4.4. 资源调度系统.....	22
3.4.5. 资源收益系统.....	23
3.4.6. 资源提供者数据服务.....	23
3.4.7. Titan 客户端数据服务.....	23
3.4.8. 运维 WEB 平台.....	24
3.5. Titan 区块链.....	25
3.6. Titan 智能合约.....	26
3.6.1. 资源管理智能合约.....	26
3.6.2. 客户任务智能合约.....	26
3.6.3. Harbour 治理智能合约.....	27
3.7. Titan 接口.....	28
3.7.1. 资源提供者接口.....	28
3.7.2. 资源消费者接口.....	28
4. 协议协调.....	30
4.1. Titan 协议.....	30
4.1.1. 闲置资源接入协议.....	30
4.1.1.1. Agent 登录.....	31
4.1.1.2. Agent 与 Harbour 的配置/数据同步.....	31
4.1.1.3. Agent 状态转换与状态同步.....	31
4.1.1.4. 保持活跃(Keep-Alive).....	31
4.1.1.5. 资源平面格式.....	31
4.1.1.6. 事件订阅.....	31
4.1.1.7. 业务调度.....	32
4.1.2. 资源提供者服务协议.....	32
4.1.2.1. 访问授权.....	32
4.1.2.2. 绑定或解绑闲置资源.....	32
4.1.2.3. 配置闲置资源.....	33
4.1.2.4. 查看闲置资源信息.....	33
4.1.3. 客户服务协议.....	33
4.1.3.1. 访问令牌.....	33
4.1.3.2. 查询 Harbour 闲置资源概览.....	33
4.1.3.3. 查询自有资源信息.....	33
4.1.3.4. 订购虚拟机资源.....	34
4.1.3.5. 订购容器资源.....	34
4.1.3.6. 查询客户任务合约.....	34
4.2. 资源验证系统.....	35
4.2.1. Titan 验证网络.....	35
4.2.2. AI 生成的服务负载.....	35
4.2.3. Titan 验证系统的有效性.....	36

4.2.4. Titan 的验证流程和协议.....	36
4.2.4.1. Titan 验证系统应用案例 A.....	36
4.2.4.2. Titan 验证系统应用案例 B.....	38
4.3. 资源调度系统.....	39
4.3.1. 调度策略的公平性.....	39
4.3.2. 调度策略的特殊性.....	40
5. Titan 网络用例.....	41
5.1. 智能边缘 CDN.....	41
5.2. 边缘计算.....	41
5.3. IP 代理.....	42
5.4. 网络爬虫应用.....	42
5.5. 各种 Web2 和 Web3 项目的资源基础.....	43
5.6. 区域性测试与主动监控.....	43
5.7. 边缘 GPU 计算和 AI 代理.....	43
5.8. 数据主权存储.....	44
5.9. 内容和计算本地化.....	44
6. 未来发展.....	46
6.1. 更高的公开透明度.....	46
6.2. 更好的 Titan 区块链核心.....	46
6.3. 防止资源被滥用于恶意活动.....	47
6.4. Titan 客户端数据安全.....	47
6.5. 聚合更多类型的闲置资源.....	48
6.6. 支持丰富的交易模式.....	48
6.7. 落地更多类型的客户业务.....	48
7. 结论.....	50

1. 引言

全球数字经济正在加速发展。随着各行各业采用人工智能(AI)、区块链、物联网(IoT)和实时应用等技术,计算、存储和带宽已成为必不可少的基础设施,就像工业时代的电力一样。这些资源不仅推动创新,也构成了可扩展数字服务的基础。

尽管技术不断进步,但数字基础设施的需求与资源实际利用方式之间的不匹配日益加剧。一份 2023 年全球数据中心能效报告发现,传统数据中心的平均服务器利用率仅徘徊在 12%-18% 之间。与此同时,超过 50% 的个人设备在一天中的大部分时间里处于闲置状态。世界各地充满了未被充分利用的设备——家用电脑、路由器、手机,甚至企业服务器——它们消耗能源却几乎不创造价值,导致了数千亿美元的闲置资源沉没成本。

与此同时,传统的云基础设施面临多重挑战:

- 人工智能(AI)、区块链和边缘计算等新兴技术对弹性、低成本资源的需求呈爆炸式增长。
- 集中式互联网数据中心(IDC)的部署难以充分覆盖边缘计算和实时渲染等低延迟场景。
- 资源浪费和能源消耗问题日益严重。

此外,对基础设施的集中控制意味着用户必须在灵活性、隐私和本地化方面做出妥协。这些妥协在今天变得越来越不可接受,而这一切都被强加给公众,同时数字基础设施的垄断阻碍了中小型开发者获取经济实惠的计算资源,从而限制了他们的创新潜力。

幸运的是,一个平行的趋势正在兴起:去中心化。在 Web3 的推动下,个人和组织正在重新获得对数据、身份和基础设施的控制权。数据主权,即拥有和管理个人数字资产的权利,正在成为一项基本期望。集中式云系统无法满足这些新兴需求。人们需要的基础设施不仅要低成本、高性能,还要透明、抗审查和社区所有。

在这种环境下,一个巨大的机遇应运而生:将闲置的计算资源聚合成一个新的、去中心化的数字基础设施层。全球闲置供应虽庞大,却分散、未被利用。当前所缺失的,正是一个能够实现以下功能的协调层:

- 标准化这些异构设备
- 保障和验证其可用性
- 公平地激励参与
- 向终端用户提供可靠的服务

Titan 网络是一个去中心化的协议和基础设施网络,将闲置设备转化为可用的计算、存储和带宽。它结合区块链、智能合约和容器化运行时环境,Titan 使任何人都能以透明、安全和高效的方式贡献或消费数字基础设施。其开源和模块化的设计,使得新一代的开发者、企业和社区能够访问到去中心化、成本效益高且贴近终端消费者的计算基础设施。

1.1. 问题领域

现代计算面临着闲置供应与增长需求之间的悖论。全球数十亿个人设备(个人电脑、智能手机、物联网设备、家用服务器)在一天中的大部分时间里处于闲置状态,这代表了巨大的计算潜力浪费。与此同时,对分布式计算和低延迟服务的需求不断上升,但集中式云提供商却以高成本和有限的地理覆盖范围占据主导地位。这种不平衡导致:

- 基础设施未被充分利用:大量计算设备在消耗能源却没有执行有用的工作。如果加以利用,这些闲置的总容量可以提供巨大的计算能力。
- 中心化之痛:依赖少数云提供商的企业和用户面临高成本、远距离用户的延迟问题、数据隐私担忧和供应商锁定。许多日常工作负载并不需要超高性能的超级计算机,而是需要分布式、中等计算能力且对终端用户低延迟的资源。

新兴的去中心化物理基础设施网络(DePIN)概念为弥合这一差距提供了一条途径。通过将这些闲置资源联网并加以激励,可以创建一个由社区驱动、成本效益高且可扩展的分布式云。Titan 网络正是瞄准了这一机遇,通过提供技术和经济框架,大规模地解锁闲置的边缘资源。

全球分布的大量闲置资源,如果得到充分利用,不仅可以防止资源浪费,还可以为资源所有者创造收入。更重要的是,它们可以补充传统的集中式资源服务,缓解其压力并解决其不足之处。

1.2. 论文结构

本文的组织结构旨在逐步建立对 Titan 网络的理解, 从其解决的问题, 到其架构、协议设计和实际应用。每一章都深化了对系统特定维度的探索:

- 第 1 章: 引言 - 概述了数字基础设施的更广泛背景, 指出了当前模式中的关键低效问题, 并介绍了 Titan 作为大规模解锁闲置边缘资源的解决方案。
- 第 2 章: 闲置边缘资源 - 分析了使用闲置基础设施的性质和挑战, 然后分解了设备特性(异构性、稳定性、本地性、规模)、供需双方的担忧, 以及 Titan 如何系统地解决每一个问题。
- 第 3 章: **Titan** 网络组件 - 描述了 Titan 网络的核心构建模块, 包括 Titan Agent、Titan Harbour、Titan DAO、智能合约和接口。本章接着定义了资源如何接入、验证、调度和货币化。
- 第 4 章: 协议协调 - 详细说明了 Titan 的协议如何管理资源访问、服务交互、验证和调度。包括 Agent、Harbour 和客户之间的通信流程的完整分解, 以及 Titan 通过强大的验证层实现网络安全的方法。
- 第 5 章: **Titan** 网络用例 - 展示了 Titan 如何支持现实世界的应用, 如边缘 CDN、计算、IP 代理、去中心化存储、AI 代理, 并突出了 Web2 和 Web3 的集成机会。
- 第 6 章: 未来发展 - 讨论了计划中的系统增强功能, 包括透明度、区块链升级、安全措施, 以及对额外工作负载和交易模式的更广泛支持。
- 第 7 章: 结论 - 总结了 Titan 通过实现去中心化、可验证和社区所有的计算, 在重新平衡数字基础设施市场中的作用。

每一章都旨在独立成篇, 同时共同构成对 Titan 技术和经济框架的完整理解。

2. 闲置边缘资源

Titan 网络建立在一个简单而强大的理念之上：世界已经拥有比其使用量更多的基础设施——只是它们处于闲置、碎片化和无协调的状态。这些闲置的边缘资源，虽然单个有限，但集体代表了一个巨大且未被充分利用的基础设施层。本章探讨了这些资源的特性，供应方贡献者和需求方用户所面临的担忧，以及 Titan 如何设计来弥合这一差距——通过去中心化的协调，将闲置容量转化为可验证、可信赖和富有成效的基础设施。

2.1. 特性

闲置边缘资源与集中式基础设施有显著不同。它们在硬件能力、网络稳定性、地理位置和可用性方面各不相同。理解这些特性对于设计一个能够有效接入、管理和利用这些资源的系统至关重要。

2.1.1. 多样性(异构性)

现实中存在各种各样的闲置资源。从位置角度看，有家庭、IDC、办公环境等无数场景下的闲置资源。考虑到家庭内部基于设备的闲置资源，常常存在各种闲置设备：智能手机、平板电脑、笔记本、台式机和其他智能设备。

这些家用设备不仅数量众多，而且拥有丰富的资源和能力。例如，许多智能手机具备：

- 1GB 到高达 16GB 的内存。
- 16GB 到高达 1TB+ 的存储空间。
- 接近笔记本电脑 CPU/GPU 的 CPU/GPU 计算能力。

如果这些资源能够得到充分利用，可以产生巨大的价值。

这些闲置资源可能在多个维度上进一步存在差异：

- 操作系统类型：Windows、Linux、macOS 等，内部还有各种版本。
- CPU 架构：x86、ARM、MIPS 架构等。
- 内存类型和数量：DDR5/4/3 类型，数量如 4GB/8GB/16GB 等。
- 存储类型和数量：HDD、SSD，数量如 256GB/512GB/1TB 等。
- 连接和网络类型：静态/动态 IP、NAT1/2/3/4、光纤/5G/卫星等。

2.1.2. 稳定性

闲置资源的一个关键特性是它们在以下方面相对缺乏稳定性：

- 消费级与商业级设备的可靠性差异。
- 电源供应稳定性(例如，缺乏不间断电源(UPS))。
- 宽带和互联网稳定性(例如，多个家庭共享宽带设施)。

- 由于设备使用而导致的资源可用性(例如, 由于用户设备使用而导致可用资源减少)。

2.1.3. 区域性

从物理距离的角度来看, 多区域分布的闲置资源可以有效地服务于同一区域内的需求, 因为距离近。然而, 由于跨区域通信可能产生不可接受的网络延迟, 整合跨多个区域的闲置资源进行利用是具有挑战性的。

2.1.4. 数量庞大(规模)

闲置资源的主要来源是家庭设备。即使在单个区域内, 如一个城市, 闲置资源的规模也是巨大的。从资源角度看, 与闲置资源通信需要大量的网络带宽, 调度这些资源提供服务也需要强大的计算能力。

2.1.5. 统一的复杂性

虽然闲置边缘资源的多样性、规模和地理分布带来了巨大的潜力, 但它们也引入了现实世界的复杂性。这些设备通常被认为是不稳定且难以跨区域进行一致管理的。对于这些资源的所有者和潜在消费者来说, 这种复杂性引发了关于安全性、可靠性和经济透明度的重要担忧。下一节将从供应方的角度探讨这些挑战, 重点关注是什么阻碍了设备所有者在今天贡献他们的闲置容量。

2.2. 供应方担忧

从闲置资源所有者的角度来看，让资源闲置确实令人惋惜。然而，如果使用这些闲置资源需要承担风险，他们宁愿根本不使用，以避免潜在的损害。

2.2.1. 基础设施安全风险

允许外部使用闲置资源涉及到运行外部程序。如果这些程序是恶意的，可能会出现数据安全问题。这种安全风险不仅危及闲置资源本身，还危及其整个局域网(LAN)。如果闲置资源位于家庭中，家庭局域网上的所有设备都可能面临风险；如果位于 IDC 中，该 IDC 内的所有设备都可能受到威胁。

2.2.2. 收入不透明

资源所有者常常面临其闲置资源所产生收入不透明的问题。许多先前的项目已经聚合了闲置资源来提供服务，产生收入并将一部分分配给资源所有者。固有的问题在于这个过程的黑箱性质；资源所有者无法了解实际收益和自己的份额，可能只收到真实收入的一小部分。

这种黑箱模式等同于欺骗，因为它损害了资源所有者和资源使用者(客户)的利益。因此，资源所有者不愿意支持这种黑箱模式，并对将自己的闲置资源贡献给此类项目犹豫不决。

2.3. 需求方担忧

2.3.1. DePIN 基础设施的可靠性与技术开销

在分布式闲置节点上部署、管理和更新软件的难易程度是一个关键问题。客户需要像传统云解决方案那样的直观界面，如仪表盘，用于在众多边缘资源上进行监控、部署、版本控制和故障排除。没有高效的工具和自动化，运营开销可能会超过低成本基础设施带来的好处。

2.3.2. DePIN 基础设施的安全性

在由社区拥有的设备组成的去中心化网络上部署专有代码或内容，引发了对数据完整性、机密性和执行信任的严重担忧。没有足够的安全性或明确的问责制，客户可能会对采用 DePIN 解决方案犹豫不决。需要确保隔离的访问点、可验证的执行和强大的安全层，以给用户信心，确保他们的应用程序和数据在具有可编程信心的全球分布式基础设施中保持安全。

2.3.3. 成本与透明度

客户，特别是大型企业，寻求以最低成本获得大量的计算和网络资源。虽然闲置资源提供了可负担性，但不透明或黑箱的聚合模型常常抬高价格或不当分配费用，给客户带来很差的附加值。需求方需要一个透明、开放的平台，其中定价是可预测的，中介被最小化，成本反映了实际的资源质量和使用情况。

2.4. Titan 解决方案

针对闲置资源的特性以及供需双方的担忧, Titan 从资源提供者和使用者的角度出发, 为每个关键点提出了相应的解决方案。

2.4.1. 解决闲置资源多样性

为了管理闲置资源的多样性, Titan 协议被设计为在多个维度上抽象异构的闲置资源, 最大限度地减少不同类型之间的差异。

Titan 协议标准化了闲置资源的顶层抽象, 主要基于容器(K8S 容器、Titan 轻量级容器)和虚拟机(VM)建立资源平面。各种闲置资源通过这些 Titan 协议定义的资源平面向 Titan 客户端提供服务。

同时, Titan 协议定义了闲置资源加入 Titan 生态系统的接入和管理协议。因此, 任何闲置资源只需遵守 Titan 协议规范即可加入生态系统并接受 Titan 的调度。在第 3.2 节中, 详细解释了 Titan Agent, 它实现了 Titan 协议定义的资源平面和接入规范。随着 Titan 生态系统的发展, 社区将发布支持更广泛闲置资源类型的 Titan Agent, 从而实现更广泛的参与。

2.4.2. 解决闲置资源稳定性与可靠性

为了应对闲置资源稳定性不足的问题, Titan 生态系统采用了多种策略。

首先, Titan 为每个闲置资源分配一个稳定性分数和评级。闲置资源加入生态系统后, Titan 会持续监控多个维度的指标并对资源质量进行评分。根据该分数, 闲置资源被分为几个稳定性级别之一。

其次, Titan 生态系统允许资源所有者对其闲置资源进行抵押。如果出现稳定性问题, 可能会扣除一定比例的抵押品。本质上, 资源所有者通过抵押来保证其资源的稳定性。

在 Titan 生态系统中, Titan Harbour 负责资源调度。其资源调度系统在资源分配期间充分考虑了闲置资源的稳定性水平以及它们是否有稳定性保证(即抵押)等因素。

例如, 对于需要 99% 可靠性的无状态 Titan 客户端工作负载, Titan Harbour 可以选择三个可靠性均为 90% 的闲置资源。从统计学上讲, 这种组合可以满足客户端的可靠性要求。对于有状态的 Titan 客户端工作负载, 由于难以平滑迁移, 因此需要高度可靠的资源, Titan Harbour 可以选择有保证(已抵押)的闲置资源来满足客户端的需求。

随着 Titan 生态系统的不断发展和成熟, 将引入和实施更多解决闲置资源可靠性缺陷的方法。例如, 可以利用机器学习和人工智能, 根据历史稳定性数据预测未来一段时间内闲置资源的稳定性。

2.4.3. 解决闲置资源区域性

为了处理闲置资源的区域性, Titan 生态系统将区域属性视为主属性。每个闲置资源根据其位置连接到负责其所在区域的 Titan Harbour。每个区域至少有一个 Titan Harbour 负责管理该区域内的闲置资源。Titan 生态系统在资源分配期间避免跨 Harbour 的资源调度。

此外, 闲置资源拥有一个 IDC-UUID 属性, 用于标识资源所在的特定 IDC 设施。共享相同 IDC-UUID 值的闲置资源被 Titan Harbour 分组为同一 IDC 内的资源, 这意味着它们可以通过高速局域网互连。

目前, Titan 生态系统内的区域在地理上定义为国家或大洲级别。随着生态系统的发展, 加入的闲置资源增多, 对更细粒度的需求增加, 区域的物理范围可能会被细化, 甚至可能缩小到城市级别。

2.4.4. 解决闲置资源规模

首先, Titan 生态系统根据物理区域部署 Titan Harbour, 每个 Harbour 只管理其指定区域内的闲置资源。

其次, 每个 Titan Harbour 是一个由多个服务程序组成的集群系统。每个服务程序都支持多实例分布式部署。这不仅减少了单个实例的负载, 还降低了因单点故障导致服务中断的风险。

此外, 与闲置资源相关的关键核心数据, 如收益, 通过智能合约锚定到 Titan 区链上。这确保了即使在 Titan Harbour 发生故障的最坏情况下, 资源所有者的收益也受到保护。

2.4.5. 解决闲置资源共享和租赁的安全风险

关于闲置资源的安全性, Titan 生态系统通过彻底的隔离来实现。Titan 客户端工作负载在一个沙箱环境中运行, 确保它们无法突破沙箱与外部环境交互。这保证了闲置资源本身的安全以及其所在局域网(LAN)的安全。

目前, Titan 生态系统实现并支持以下四种类型的沙箱。随着生态系统的发展, 将支持更多沙箱类型:

- **K8S 容器**: 什么是容器? 容器是标准化的、轻量级的、安全的程序执行环境。因为它们与主机的操作系统共享内核, 所以它们表现出极快的启动时间和低资源消耗, 与虚拟机相比非常轻量。同时, 容器提供了一个与主机操作系统隔离的环境。容器内的 Titan 客户端业务程序无法非法访问容器外的设施。K8S 容器是指由 K8S 容器编排系统管理的容器。Titan 客户端可以通过 K8S 或多租户容器编排系统高效地利用 Titan 生态系统的闲置资源。
- **虚拟机(VM)**: 虚拟机在主机上创建一个虚拟化的机器实例, 运行一个独立的操作系统。虚拟机与主机隔离, 防止虚拟机内的 Titan 客户端业务程序非法访问其外部的设

施。虚拟机的优点是能够运行与主机完全不同的操作系统，这由 Titan 客户端的要求决定。缺点是，正是由于需要运行一个独立的操作系统，虚拟机比容器需要更多的资源和更慢的启动时间。Titan 生态系统主要在 Linux 平台上利用 KVM 虚拟化，在 Windows 和 macOS 平台上利用 VirtualBox，并且也支持 VMWare 虚拟机。

- **JS/WASM 容器**: 此容器与上述 K8S 容器不同。它不能运行典型的容器镜像。相反，它是一个基于 Chromium 的 V8 JavaScript 引擎构建的 JavaScript/WASM 运行时环境，能够执行用 JavaScript 和 WASM 等语言编写的程序。Titan 生态系统基于 Worker-D(一个基于 Chromium V8 的开源运行时环境)构建了这个 JS/WASM 运行时环境，确保与主机的隔离。使用 JavaScript/WASM 开发的 Titan 客户端业务程序在 Titan 的 JS/WASM 容器内运行，因此无法以未经授权的方式访问主机操作系统上的设施。JS/WASM 容器的优点是极其轻量、资源消耗最少、启动迅速。然而，缺点是要求 Titan 客户端业务专门用 JavaScript 或 WASM 编写程序，这对许多客户端来说是一个巨大的开销。
- **TERMUX 沙箱**: 这种类型的沙箱主要存在于 Titan 生态系统内的闲置 Android 手机中。它建立在著名的 Android 社区终端模拟器 TERMUX 提供的库之上。其目的是允许 Android 手机在隔离环境中运行控制台程序。有了 TERMUX 沙箱，Titan 客户端工作负载可以作为 Android 控制台程序在 Titan 生态系统内的闲置 Android 手机上运行。客户端的程序无法访问 TERMUX 沙箱之外的手机设施。

2.4.6. 解决闲置资源成本和奖励分配问题

为确保闲置资源收入的透明性，Titan 生态系统通过智能合约将核心收益数据锚定到区块链上。闲置资源的收益及其分配通过发布在 Titan 区块链上，实现完全公开透明。

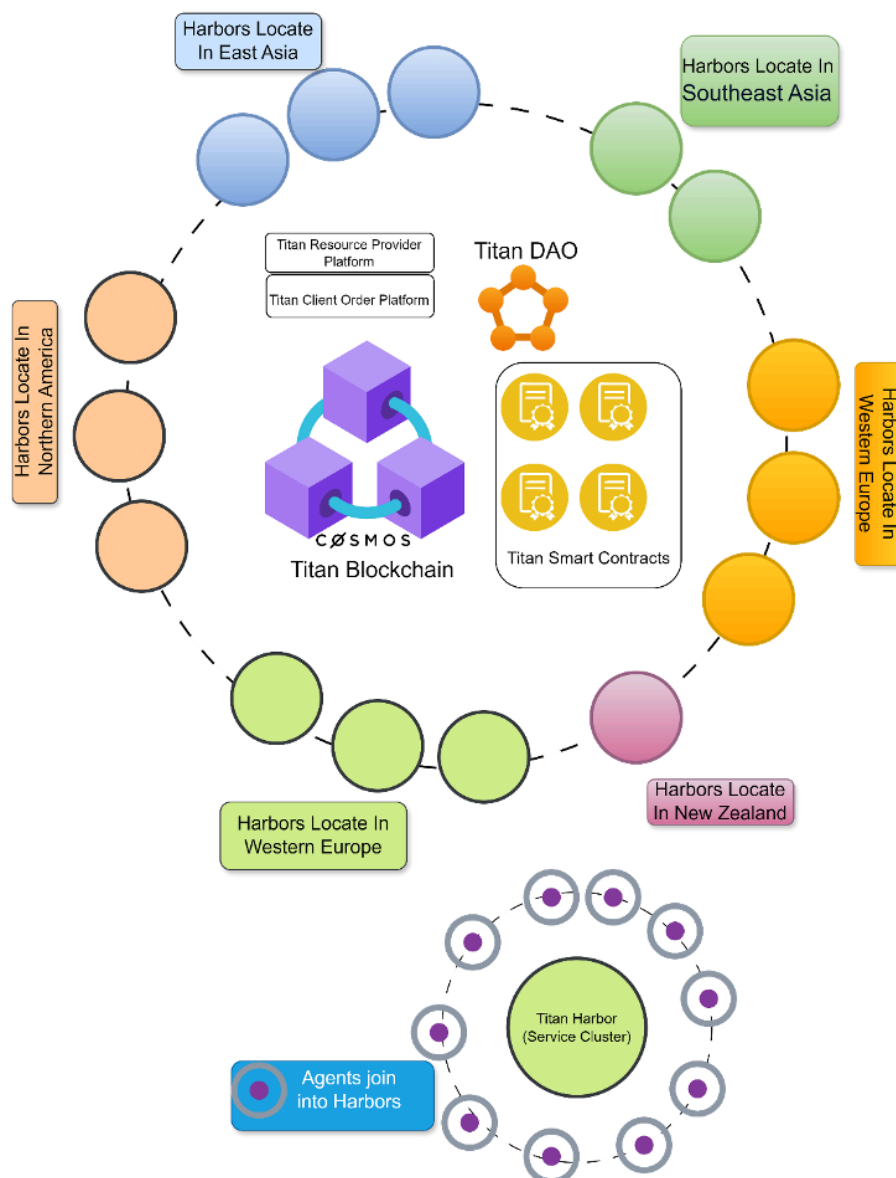
服务计算基础的支付金额和成本也必须公开透明。Titan 生态系统使用 Titan DAO 为每个区域的各种资源类型设置单价、折扣和其他定价信息。这些信息通过智能合约发布到 Titan 区块链上。Titan 客户端的支付正是根据 Titan 区块链上的定价信息和订购的资源数量计算的。

每个 Titan 客户端的订单都作为一个订单智能合约发布到 Titan 区块链上。对于每个闲置资源，详细说明它为谁提供了服务、服务提供的时间以及收到的相应收益的信息也将发布到 Titan 区块链上。有了这些驻留在区块链上的公开透明数据，资源所有者和 Titan 客户端都可以清楚地了解价值的来源和流向。

3. Titan 网络组件

本节对 Titan 的核心子系统进行了高层次的概述。这些子系统是构成 Titan 生态系统的基本组件。它们利用 Titan 协议作为基础通信协议，协同工作，为闲置资源、资源所有者和资源使用者等参与者提供高质量的服务。

3.1. 架构



[图片描述: 该图显示了各种闲置资源通过 Titan Agent 连接到各自区域的 Harbour。Harbour 管理和调度这些资源。客户端通过 Titan 客户服务平台提交资源请求。Harbour 使用管理的闲置资源来满足这些请求。客户端任务作为智能合约锚定在 Titan 区块链上。Titan DAO 治理整个生态系

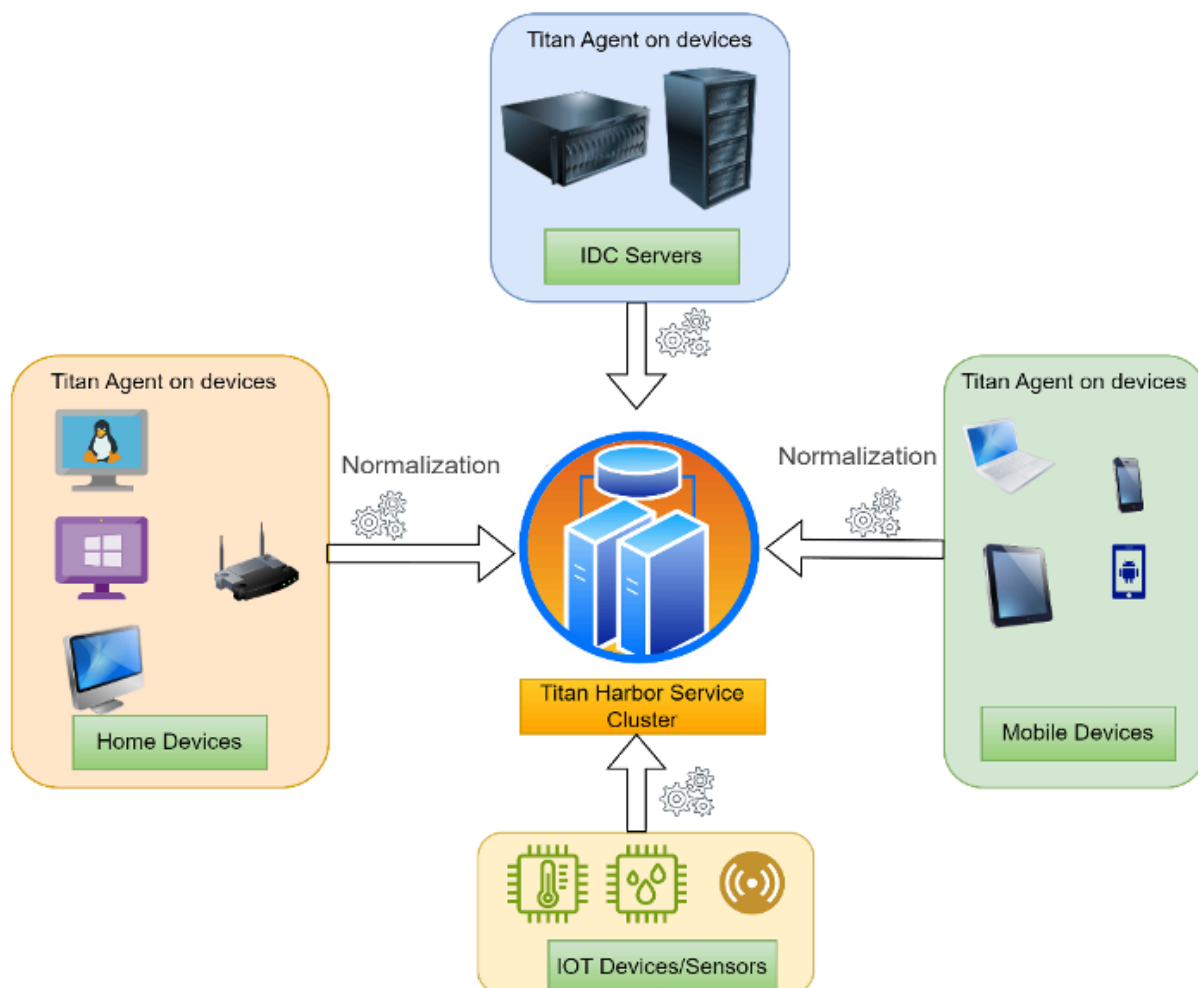
统, 包括 Harbour 和服务平台。]

上图展示了 Titan 的部署架构。

- 各种类型的闲置资源, 经过 Titan Agent 的标准化处理后, 汇集到其各自区域对应的 Harbour。
- Harbour 负责管理和调度这些闲置资源, 向客户端提供资源服务。
- 客户端通过 Titan 客户服务平台, 发布其业务资源需求。一个或多个 Harbour 调度其管理的闲置资源来满足客户的资源需求。
- 每个客户任务都通过可编程智能合约锚定到 Titan 区块链上, 确保整个过程的透明度。智能合约不仅包含详细需求, 还包含与合同完成所需条件相关的数据信息和逻辑代码。
- Titan DAO 作为生态系统治理的核心, 通过模块化的治理框架实现社区共治。Titan DAO 的主要职责包括但不限于:
 - Titan 治理, 例如监督 Titan Harbour 和促进 Titan 生态系统的持续发展。
 - 争议解决, 包括处理与客户服务质量和闲置资源收益相关的争议。
 - 生态系统监督, 确保 Titan 资源的适当使用和 Titan 生态系统的健康发展。

3.2. Titan Agent

Titan Agent 是一套开源软件包，旨在根据 Titan 协议标准化闲置资源并将其连接到 Titan Harbour。只有经过这个过程，闲置资源才能被 Titan 调度和利用。



3.2.1. 支持多种设备类型

可以作为闲置资源的设备类型是多种多样的。Titan Agent 可以在各种硬件（例如，Windows、Linux、macOS 个人电脑、数据中心级基础设施）以及移动设备（例如，安卓手机、平板电脑、物联网设备）上运行。因此，Titan Agent 可以标准化各种闲置资源，并将它们集成到 Titan 闲置资源网络中，让它们为 Titan 生态系统的协同发展做出贡献。

3.2.2. 闲置资源标准化

标准化闲置资源是为了遵守 Titan 协议的约束。由于资源的多样性，需要尽可能地抽象掉

底层的差异，为客户端提供流畅的用户界面，使他们能够方便地利用闲置资源。

从资源提供者的角度来看，他们不仅关心 Titan 生态系统产生的收入，还关心资源安全。标准化闲置资源可以保证安全，因为 Titan 协议强制要求标准化必须以隔离的方式进行，独立于主机。

Titan Agent 根据闲置资源的类型和特性选择合适的标准化技术。例如，对于 Windows PC 类型的闲置资源，Titan Agent 使用 VM 技术进行标准化。对于手机类型的闲置资源，它使用 JS/WASM 容器。对于 Linux PC 或服务器，它利用 Kubernetes (K8S) 容器，以此类推。

3.2.3. 资源配额调度

需要一个资源配额计划，不仅要显示闲置资源的可用配额，还要显示它们可用的时间段。Titan Agent 需要根据资源配额计划动态调整 Titan 的资源使用限制。

闲置资源所有者可以设置 Titan 可用性配额计划。例如，可以根据时间段为家庭带宽资源分配配额。不同的时间段允许 Titan 使用不同的带宽配额：例如，在深夜休息期间，Titan 可能会利用几乎全部带宽，但在家庭成员需要带宽的时候，Titan 必须释放足够的容量，以避免影响他们的体验。

3.3. Titan DAO

Titan DAO 是 Titan 生态系统的中央治理中心，肩负着其可持续发展的责任。Titan DAO 监督生态系统内两个核心服务平台(资源提供者服务平台和客户服务平台)的运营和服务质量，并批准其质量改进计划。

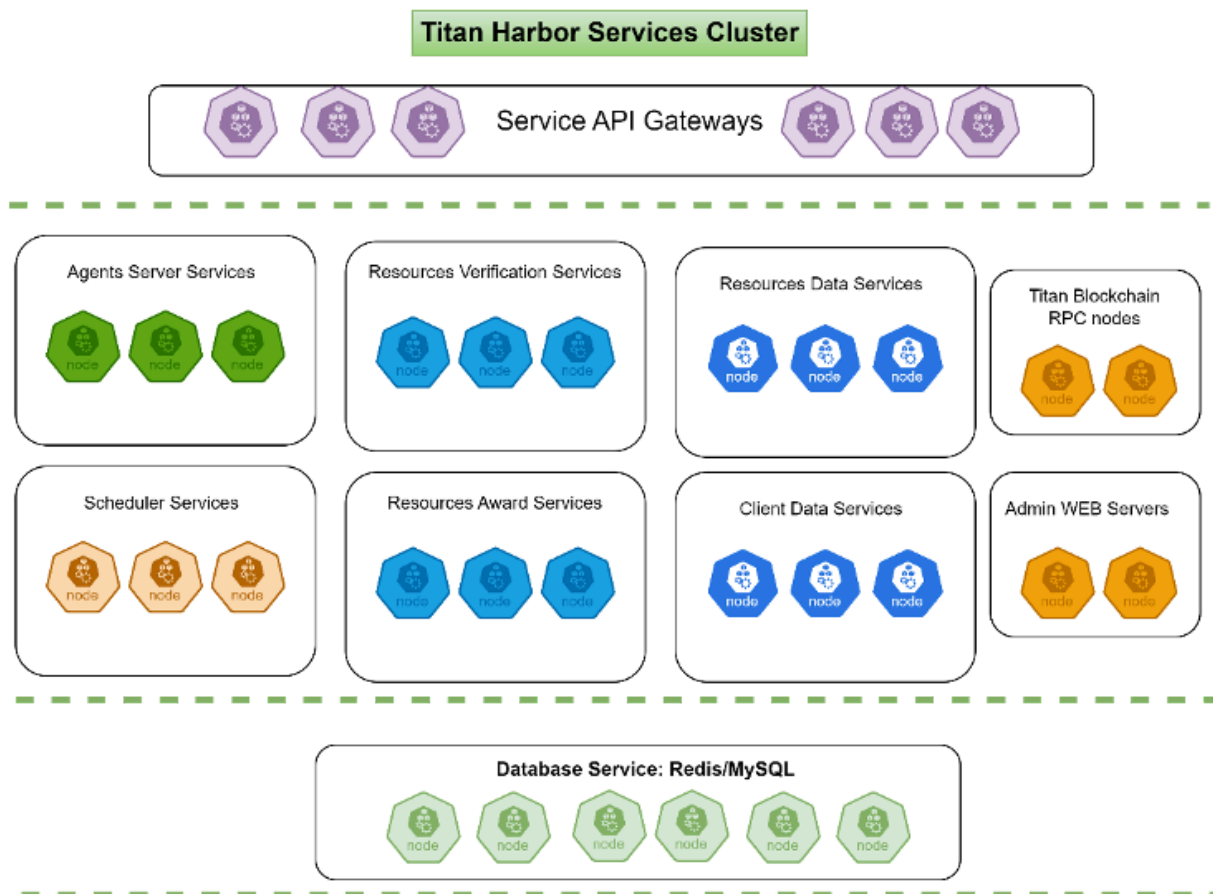
根据 Titan 生态系统不断变化的需求，Titan DAO 批准新的 Titan Harbour 运营计划或撤销现有的计划。Titan DAO 持续监控生态系统内 Titan Harbour 的运营和服务质量，并批准其质量改进计划。

Titan DAO 负责处理 Titan 生态系统内的争议，例如闲置资源提供者提出的争议或来自 Titan 客户端的服务质量争议。

当 Titan 生态系统缺乏足够数量的特定类型闲置资源以满足需求时，Titan DAO 可以向社区发布该资源类型的征集计划，并制定相应的激励方案。

3.4. Titan Harbour

Harbour 充当 Titan 生态系统的中心枢纽。它们北向与客户服务平台对接, 响应客户资源需求;南向与由 Titan Agent 代表的闲置资源对接, 接入并管理其区域内的所有 Titan Agent。



Harbour 是一个由各种服务进程组成的服务集群, 包括但不限于:

- Agent 管理服务 (Agent-Server)
- 资源验证服务 (Agent-Reputation)
- 资源调度服务 (Scheduler)
- 资源收益服务 (Awarer)
- 资源提供者数据服务 (Provider-Data-Server)
- Titan 客户端数据服务 (Customer-Data-Server)
- 运维 WEB 平台

Harbour 服务集群实现了 Titan 协议定义的协议, 允许外部实体通过遵守这些协议来访问和与 Harbour 内的各种服务进行交互。

为了防止单点故障, Harbour 服务集群内的所有服务进程都支持多实例部署。这些服务进程运行在 K8S 容器集群上。

3.4.1. 接受 Titan DAO 的监督

Titan 生态系统内存在多个 Harbour, 每个 Harbour 管理一定数量的闲置资源。这种分布式部署既分散了负载, 又避免了单点故障的风险。

Titan DAO 负责管理 Titan Harbour 的各个方面, 例如决定 Harbour 的区域和数量(例如, 在哪个区域部署多少个 Harbour 集群)。它还监督 Harbour 的服务质量监控和改进, 跟踪诸如对客户资源请求的响应速度和闲置资源调度的质量等指标, 并要求 Harbour 不断改进这些指标。

3.4.2. 资源接入与管理

Titan Harbour 接受来自 Titan Agent 的连接请求。Agent 根据 Titan 协议的约束向 Titan Harbour 注册其资源属性。

Titan Harbour 为每个 Agent 分配相应的数据存储空间, 以记录其资源属性, 例如设备类型(家用 PC 或 IDC 服务器)、操作系统(Windows 或 Linux)、CPU 架构(ARM 或 x86)等。

当 Agent 在线时, Harbour 将其资源标记为可用, 并根据不同维度将其添加到相应的资源池中。当 Agent 下线时, 其资源被 Harbour 标记为不可用, 并从相关资源池中移除。

Harbour 管理每个 Agent 的当前在线/离线状态。同时, Agent 定期生成关于自身资源状态的报告并发送给 Harbour, 使 Harbour 能够及时更新其数据库以反映资源可用性的变化。

Harbour 知道哪些客户任务分配给了哪个 Agent 以及这些任务的运行状态。通过这些数据, Harbour 能够实时了解其整体资源状况、包括当前正在使用的资源、可用的闲置资源等等。

Harbour 根据多个维度构建各种资源池, 并将闲置资源分类到这些池中, 以便快速进行资源调度。例如, 根据客户任务的部署类型维度, 闲置资源可以被分类到 K8S Pod 资源池、Linux VM 资源池等。

3.4.3. 资源验证与信誉向量

资源的真实性和有效性对 Titan 生态系统至关重要。例如, 如果一个资源提供者承诺提供 100 Mbps 的上行带宽, 但实际只提供 1 Mbps, 将这个资源分配给客户会对其业务运营产生负面影响, 并降低客户体验。

为了最大程度地保证资源的真实性和有效性, Titan 需要实现一个多维度、多层次的验证系

统。需要多维度和多层次是因为 Titan 生态系统的资源是多样的(例如, 计算、带宽、存储等), 需要不同的验证方法。因此, 必须在多个层面上考虑当前的验证方法以最小化资源作弊。

此外, 利用 Titan 生态系统内大量的家庭和移动设备, Titan 开发了一个可用性验证服务(AVS)。客户可以使用 Titan 的 AVS 定期验证他们的服务, 确保其运营的有效性。

Titan 的验证系统随机对闲置资源发起验证, 并将结果通过智能合约锚定到 Titan 区块链上, 以确保公开透明。

验证结果最终影响闲置资源的信誉向量。信誉向量是一个多维度的指标集, 每个指标表示资源在某一特定维度上的可靠性(例如, 在线指标、CPU 能力指标、内存可用性指标、带宽可用性指标等)。

当 Titan Harbour 为客户任务调度闲置资源时, 它会优先选择信誉更好的资源。因此, 信誉更高的闲置资源有更大的机会为客户服务并获得更多收入。

3.4.4. 资源调度系统

Titan 客户端通常通过 Titan 的客户服务平台建立服务需求。在创建需求时, 客户端可以主动选择一个或多个 Harbour 为其服务提供闲置资源。如果客户端没有主动选择, 客户服务平台会根据多个维度的匹配标准(例如, 客户端任务的目标运营区域)自动分配一个或多个合适的 Titan Harbour。

收到来自客户服务平台的资源需求后, Titan Harbour 的闲置资源调度系统开始为客户任务进行资源分配。此分配过程涉及一系列策略和算法的组合, 这些策略和算法会随着时间的推移而演变, 在 Titan DAO 的监督下不断改进。

目前, Titan Harbour 的资源调度策略考虑以下几个方面:

- 根据客户任务选择的运行时环境类型进行选择。例如, 容器环境需要选择支持容器运行时的闲置资源, 而虚拟机(VM)环境需要选择支持虚拟机的资源。闲置资源支持的运行时环境是在其由 Titan Agent 进行标准化时根据资源性质确定的。
- 优先选择信誉向量更好的闲置资源。例如, 信誉向量中较高的在线指标表示稳定性更高, 而较低的值则表明资源在服务期间下线的可能性更高, 迫使调度系统重新调度以再次满足客户的需求。
- 资源分配需要充分考虑冗余, 因为闲置资源的可靠性无法与商业级资源相媲美。例如, 如果客户任务需要 1 GB 的带宽, Titan Harbour 可能会在调度时额外分配 200 MB(20% 的冗余)。
- 在客户任务运行期间持续跟踪资源变化, 在需要重新调度时迅速响应, 以尽量减少对客户业务的干扰。

- 多个 Titan Harbour 的调度系统之间可能进行协作，以实现更好的整体资源调度。

此外，Titan Harbour 的资源调度系统必须支持计划任务，即在指定的未来时间为客户服务执行资源分配。这要求系统评估 Harbour 管理的闲置资源在该特定未来时间点的预计可用性，从而防止在时间到达时执行计划分配时可能发生的失败。

总之，Titan Harbour 的资源调度子系统不断努力寻找更优的资源分配解决方案。未来，Titan 计划引入机器学习 (ML) 和人工智能 (AI)。这些进步将利用对 Titan Harbour 数据库内各种数据的分析来预测闲置资源的可用性，并提出更优的资源分配策略。

3.4.5. 资源收益系统

资源收益系统负责计算 Harbour 管理的闲置资源的收益，并定期通过智能合约将此收益数据锚定到 Titan 区块链上。

闲置资源收益包括 Titan DAO 设定的激励奖励、为 Titan 客户端任务贡献资源所获得的收益、质押奖励等。

资源收益系统还负责计算闲置资源产生的罚款，例如未能通过 Harbour 验证系统的抽查，或未能为 Titan 客户端任务提供承诺的服务质量。

3.4.6. 资源提供者数据服务

资源提供者数据服务的主要职责是实现 Titan 协议中定义的数据服务协议，为资源提供者提供数据访问和修改服务。

闲置资源通过 Agent 连接到 Titan Harbour。Harbour 通过其资源提供者数据服务，为资源提供者提供管理其拥有的闲置资源的接口，包括但不限于：

- 管理资源提供者与闲置资源之间的关联，特别是绑定和解绑。
- 资源提供者对其拥有的闲置资源进行配置。
- 允许资源提供者查看他们拥有的闲置资源的详细信息。

3.4.7. Titan 客户端数据服务

Titan 客户端数据服务的主要职责是实现 Titan 协议中定义的数据服务协议，为 Titan 客户端提供数据访问和修改服务。

Titan Harbour 的资源调度服务为 Titan 客户端分配闲置资源。通过客户端数据服务，Titan Harbour 允许客户操作他们已购买的闲置资源，包括但不限于：

- 查询已购买闲置资源的信息。
- 对于 VM 类型的闲置资源，提供访问方法（例如，IP 地址/端口、账户凭证）。
- 对于容器类型的闲置资源，提供进入容器编排系统的入口点，允许客户通过这些系统

部署工作负载。

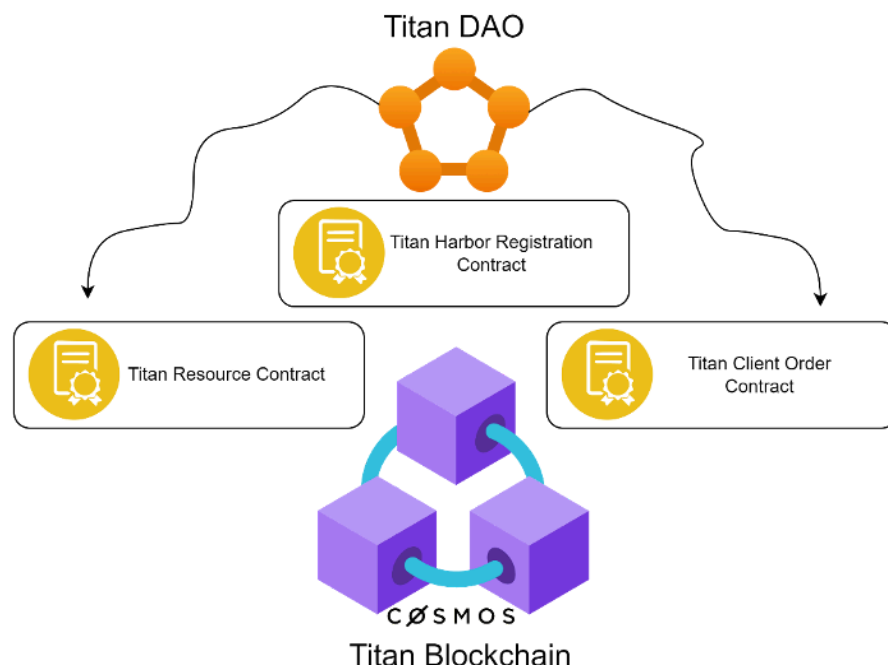
3.4.8. 运维 WEB 平台

运维 WEB 平台是一组用于 Titan Harbour 运营和管理的 WEB 服务器。通过该平台, Titan Harbour 管理员可以查看 Harbour 的当前运行状态、检查警报事件、修改运行时配置等。

结合 Prometheus 等常见的运营和监控工具, 确保 Titan Harbour 能够提供可靠的闲置资源管理和调度服务 7x24 小时。

3.5. Titan 区块链

Titan L1 区块链技术为 Titan 的去中心化治理和社区协作奠定了基础。



Titan 区块链是基于 Cosmos 构建的。在主网启动时，Cosmos 满足了 Titan 生态系统当前的 TPS(每秒交易数)要求和现有功能。然而，随着生态系统中闲置资源和 Titan 客户端数量的持续增长，对区块链 TPS 的需求将显著增加。届时，Titan DAO 将根据需要，过渡到具有更高 TPS 的区块链基础，可能采用基于 DAG 的区块同步机制来替代当前的 Cosmos 区块同步机制。

在选择要在链上记录的数据时，Titan 必须考虑以下几个方面：

- 数据量：在确保核心数据上链的同时，最小化存储需求。
- 上链频率：尽可能减少对 TPS 的需求，以避免给当前的 Titan 区块链带来压力。

3.6. Titan 智能合约

Titan 生态系统内两个最基本的智能合约是资源管理智能合约和客户任务智能合约。资源管理智能合约管理可用的资源和资源提供者，而客户任务智能合约则服务于 Titan 客户。

3.6.1. 资源管理智能合约

每个 Titan Harbour 都拥有自己专用的资源管理智能合约，以避免单个合约因大量闲置资源的数据管理压力而不堪重负。

每个闲置资源在初始化时都会创建一个 Titan 钱包。这个钱包地址作为闲置资源在 Titan 区块链上的身份代表。

闲置资源连接到 Titan Harbour 并成功加入 Titan 生态系统后，Harbour 调用资源管理智能合约的资源初始化接口，完成闲置资源在 Titan 区块链上的初始化。

只有当闲置资源的钱包有足够余额支付合约接口调用的 GAS 费用时，Titan Harbour 才会将该闲置资源注册到资源管理智能合约上。

资源管理智能合约中每个闲置资源的数据集包括但不限于：

- 闲置资源的所有者(即资源提供者)。
- 闲置资源的抵押数据。
- 闲置资源的收益数据。

一旦闲置资源完成其链上初始化，Titan Harbour 将定期将其运营数据(如每日收益)提交到 Titan 区块链。

资源提供者可以调用资源管理智能合约的抵押接口，为其闲置资源进行抵押，从而改善资源的信誉向量指标，使其能够满足更多 Titan 客户端任务的要求。

3.6.2. 客户任务智能合约

当 Titan 客户端通过 Titan 生态系统的客户服务平台购买资源服务，并且 Titan Harbour 完成资源分配等操作并成功确认订单后，Harbour 会生成一个新的客户任务智能合约，并将其锚定到 Titan 区块链上。Titan Harbour 持续监控客户任务的服务质量，并更新智能合约中的链上数据。

客户任务智能合约包括但不限于以下数据集：

- Titan 客户端标识符
- Titan Harbour 标识符
- 支付策略
- 订购的资源信息

支付策略考虑了各种选项, 包括按需付费模式和月度或定期支付方法。

支付策略还考虑了潜在的支付争议, 例如客户因服务质量问题不愿支付或要求退款。此类争议通常需要 Titan DAO 的仲裁。

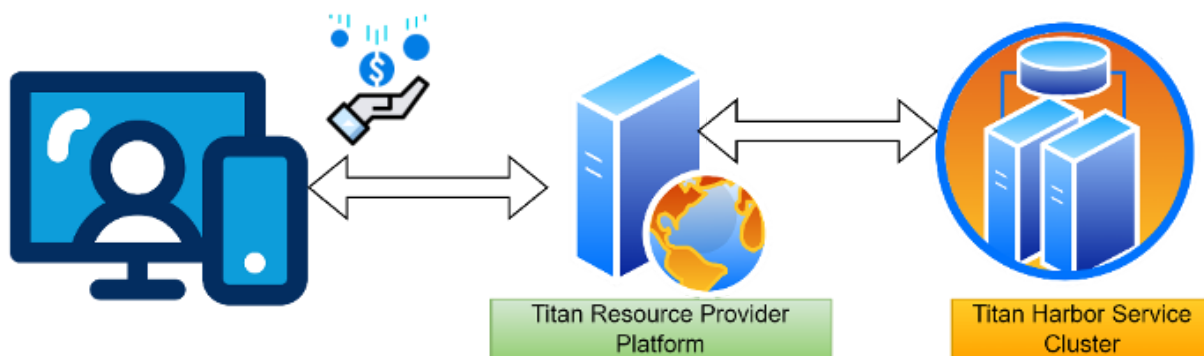
3.6.3. Harbour 治理智能合约

Harbour 治理智能合约由 Titan DAO 发布, 用于注册所有 Harbour。它包括 Harbour 的公钥、Harbour 管理的闲置资源的激励参数集以及 Harbour 管理的闲置资源的价格参数集等信息。Titan DAO 决定 Harbour 治理智能合约中的各种参数集。

3.7. Titan 接口

3.7.1. 资源提供者接口

Titan 的资源提供者服务平台，由 WEB 服务和 Titan 资源提供者数据服务组成，是一个为 Titan 资源提供者设计的服务集群。

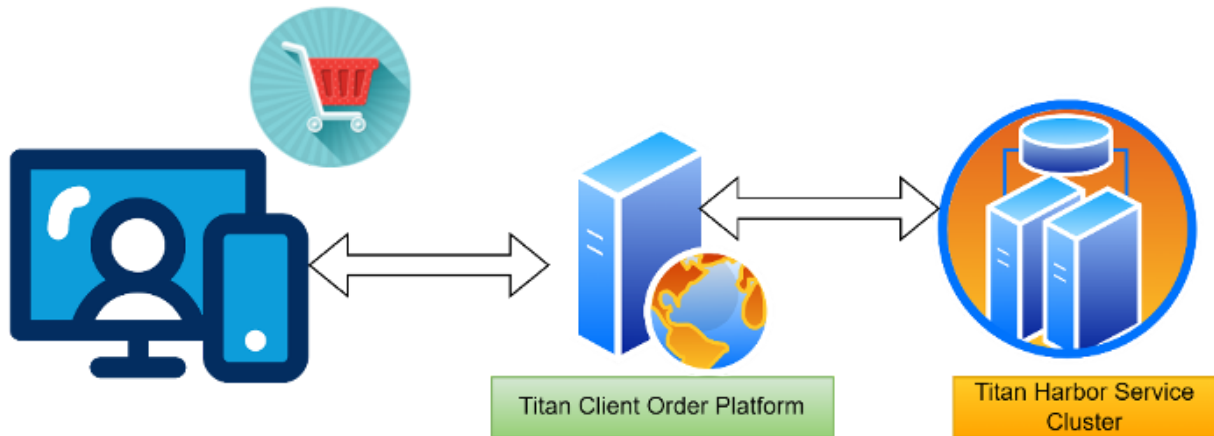


它是一个由 Titan DAO 管理的数据平台，旨在提供与资源信息、配置操作和资源收益等相关的数据服务。

- WEB 服务作为 Titan 资源提供者数据服务的图形用户界面。Titan 资源提供者通过 WEB 页面与数据服务进行交互。
- Titan 资源提供者数据服务实现了 Titan 协议的数据服务协议，整合了生态系统内各个 Titan Harbour 的资源提供者数据，以对外提供一致的数据服务。例如，通过此服务，Titan 资源提供者可以从一个统一的 WEB 页面浏览和管理他们分布在不同 Titan Harbour 的闲置资源，而无需单独访问每个 Harbour 的提供者数据服务。
- 资源提供者可以从 WEB 页面查看其所有闲置资源的信息，包括状态、收益等。
- 资源提供者可以从 WEB 页面管理和配置其所有闲置资源，例如设置闲置资源配额计划。

3.7.2. 资源消费者接口

Titan 的客户服务平台，由 WEB 服务和 Titan 客户数据服务组成，是一个面向 Titan 客户端的服务集群。



它是一个由 Titan DAO 管理的综合服务平台，作为 Titan 生态系统的客户入口和价值交换中心。它旨在为 Titan 生态系统客户提供数据浏览、订单管理、资源管理、应用部署等服务。

Web 服务功能：

- Titan 客户数据服务的图形用户界面导航
- 浏览和管理租用的闲置资源
- Titan Harbour 数据概览
- 个人账户数据概览

WEB 服务作为 Titan 客户数据服务的图形用户界面。Titan 客户端通过 WEB 页面与数据服务进行交互。

Titan 客户数据服务实现了 Titan 协议的数据服务协议，整合了生态系统内各个 Titan Harbour 的客户数据，以对外提供一致的数据服务。例如，通过 Titan 客户数据服务，Titan 客户端可以从一个统一的 WEB 页面浏览和管理他们购买的、分布在不同 Titan Harbour 的闲置资源，而无需单独访问每个 Harbour 的客户数据服务。

Titan 客户端可以在网页上查看生态系统内 Titan Harbour 的数据概览，包括闲置资源类型、数量、位置分布、使用状态等信息。

Titan 客户端可以在网页上查看与其账户相关的数据，例如当前使用资源的概览、部署的各种项目的运营状态等。

Titan 客户端可以从 WEB 页面选择并购买他们需要的闲置资源。选择基于多个维度，例如闲置资源所在的区域、CPU 类型、带宽大小等。

Titan 客户端可以从 WEB 页面使用他们已购买的闲置资源，例如，通过将项目部署到这些资源上。

4. 协议协调

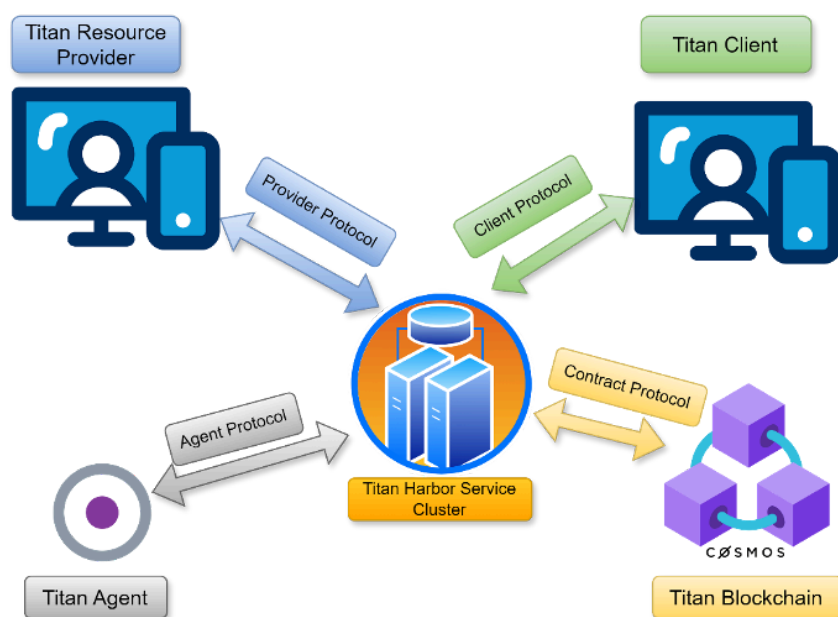
Titan 网络依赖一套协调的协议，以安全、可扩展和高效的方式将闲置的边缘资源与用户工作负载连接起来。这些协议定义了设备如何通信、资源如何验证以及任务如何在去中心化网络中调度。本章重点关注三个关键技术领域：

- **Titan 协议**: 处理 Agent、Harbour 和客户端之间的通信。
- **资源验证系统**: 确保资源真实、可用且可信。
- **资源调度系统**: 将用户需求与最合适的可用资源相匹配。

这些系统共同构成了 Titan 去中心化基础设施的技术支柱。

4.1. Titan 协议

Titan 生态系统由多个核心系统组成，例如在“系统架构”部分提到的 Titan Agent 和 Titan Harbour。这些系统需要进行通信和协作。Titan 协议定义了它们之间的协作协议，并建立了各种系统之间交互的规范。由于篇幅限制，本文仅概述 Titan 协议的几个关键子协议。



4.1.1. 闲置资源接入协议

Titan 协议定义了闲置资源为接入 Titan 生态系统必须实现的一套协议。所有类型的 Titan Agent 都需要实现这套协议以进行连接。

这部分主要关注 Agent 和 Harbour 之间的通信与协作，使 Agent 能够登录到 Titan Harbour，从 Harbour 同步管理数据，并响应来自 Harbour 的调度命令。

以下列出了一些核心协议：

4.1.1.1. Agent 登录

与 Harbour 建立网络连接后，Agent 发送登录请求。登录请求的输入集必须包括两个核心参数：Agent 设备的唯一 ID 和使用 Agent 密钥生成的签名。这两个参数是进行 Agent 身份验证所必需的。设备的唯一 ID 和密钥目前从 Titan 的资源提供者服务平台获取。

4.1.1.2. Agent 与 Harbour 的配置/数据同步

登录后，Agent 需要与 Harbour 同步数据。例如，将本地配额计划与 Harbour 上的计划对齐。如果需要从 Harbour 下载配置或数据，此步骤可能需要相对较长的时间（例如，一分钟）。

4.1.1.3. Agent 状态转换与状态同步

完成配置和数据同步后，Agent 可以将其状态切换为“就绪”，并将此新状态与 Harbour 同步。

4.1.1.4. 保持活跃(Keep-Alive)

Agent 需要定期向 Harbour 发送保持活跃数据包，以表明其在线。如果 Harbour 长时间未收到保持活跃数据包，则假定 Agent 已下线。

4.1.1.5. 资源平面格式

前几节提到需要标准化闲置资源以供 Titan 客户端使用。目前，Titan 协议规定了三种形式的资源平面：

- **VM**(虚拟机)格式。
- **K8S** 容器格式。
- **Titan** 轻量级容器(**WASM/JS** 隔离运行时) 格式。

其中，虚拟机最重，需要最多的资源，适用于像家用电脑或数据中心这样的闲置资源。K8S 容器相对轻量，不仅适用于家用电脑，也适用于资源有限的设备，如平板电脑和路由器盒子。Titan 轻量级容器最轻，需要最少的资源，但只能运行受限的 WASM 和 JS 程序。随着 Titan 协议的发展，将添加更多的资源平面格式。例如，Titan 生态系统最近为 Android 手机 Titan Agent 添加了一种基于 TERMUX 库的模拟终端格式，能够运行各种 ARM CPU 类型的控制台程序。

4.1.1.6. 事件订阅

完成登录后，Agent 需要与 Harbour 建立一个持久的 WebSocket 连接。这个长连接的主要功能是及时接收和响应来自 Harbour 的事件通知。例如，当配额计划更新时，Harbour 通过长连接向 Agent 发送事件通知。然后 Agent 需要从 Harbour 更新配额计划并相应地

调整其本地配额策略。

4.1.1.7. 业务调度

当 Agent 所代表的闲置资源被 Harbour 选中为某个客户端提供服务时, Harbour 会根据客户端的任务配置向 Agent 发出业务调度事件。

收到调度事件后, Agent 准备必要的资源, 从 Harbour 检索业务配置, 然后根据配置在本地部署业务。例如, Harbour 可能要求 Agent 创建一个具有 Harbour 指定资源规格的虚拟机; 或者, Harbour 可能要求 Agent 创建容器并运行特定的容器镜像, 同样由 Harbour 指定资源配置。

Agent 还需要定期与 Harbour 同步业务状态以保持一致。

通常, CPU/内存/带宽资源配置要求对 Agent 来说相对容易处理。然而, 网络环境要求更复杂, 可能需要特定的网络类型, 如 NAT1 或具有公网 IP 地址的网络。因此, 在 Agent 和 Harbour 之间适配网络环境需要额外的工具集, 例如执行特定的 iptables 命令或安装特定的网络插件。

4.1.2. 资源提供者服务协议

Titan 协议还定义了一套服务于资源提供者的协议。通过这些协议, 资源提供者可以与 Titan Harbour 交互以管理他们的闲置资源。

目前, Titan 的资源提供者服务平台使用这套协议与 Titan Harbour 通信, 向资源提供者展示闲置资源数据和管理操作。

以下列出了一些核心协议:

4.1.2.1. 访问授权

资源提供者使用此协议来获取访问 Titan 数据服务的授权。成功授权后, 提供者会收到一个授权令牌, 用于后续的数据访问。该令牌包含权限集、有效期等。当令牌即将过期时, 提供者可以向 Titan 的数据服务请求刷新, 以避免重复授权过程。

4.1.2.2. 绑定或解绑闲置资源

一个闲置资源有且只有一个所有者。资源提供者通过绑定操作成为所有者。只有所有者可以配置闲置资源、修改其服务状态等。如果闲置资源已被其他提供者绑定, 则不允许重复绑定。提供者使用解绑操作来放弃资源的所有权。成功解绑后, 闲置资源恢复到无主状态, 不再参与 Titan Harbour 的调度。如果闲置资源的当前状态阻止解绑(例如, 它正在为 Titan 客户端提供服务), 提供者必须首先更改资源的服务状态, 然后再进行解绑。

4.1.2.3. 配置闲置资源

资源提供者使用此协议来配置他们拥有的闲置资源。提供者应根据闲置资源所在的位置选择正确的区域, 因为区域直接决定了资源应连接到哪个区域的 Titan Harbour。跨区域连接会影响调度(例如, 更高的延迟)。提供者需要为闲置资源设置可用配额和配额计划, 以防止过度使用。

4.1.2.4. 查看闲置资源信息

通过该协议, 资源提供者可以检索有关可用闲置资源的详细信息。除了标准配置数据外, 这还包括由 Titan Harbour 提供的补充信息。这些补充数据显示了闲置资源当前的调度情况, 提供了其利用率的洞察, 例如, 不同资源类型已消耗的配额以及活动的 Titan 客户端工作负载数量。另一个关键信息是信誉向量。这是 Titan Harbour 根据来自多个维度的数据(包括 Titan 验证系统和为 Titan 客户端任务提供的服务质量)分配的分数。信誉向量直接影响闲置资源的收益。Titan 生态系统社区将提供指导, 说明提供者如何提高其资源的信誉和收益。

4.1.3. 客户服务协议

Titan 协议还为 Titan 资源使用者(即 Titan 客户)定义了一套协议。通过这些协议, 客户可以利用 Titan 生态系统的闲置资源。

目前, Titan 的客户服务平台作为客户的使用入口。该平台使用这套协议与 Titan Harbour 通信, 向客户展示 Harbour 的能力。

以下列出了一些核心协议:

4.1.3.1. 访问令牌

通过此协议, Titan 客户获得访问 Titan 数据服务的授权。成功授权后, 客户会获得一个访问令牌, 用于所有后续的数据访问请求。访问令牌包含权限范围(权限集)和其过期时间(有效期)等详细信息。为避免重复完整的授权流程, 客户可以在当前令牌过期前不久向 Titan 的数据服务请求刷新令牌。

4.1.3.2. 查询 Harbour 闲置资源概览

客户使用此协议来获取由 Titan Harbour 管理的闲置资源的概览, 例如总资源类型和数量、已使用量以及剩余可用资源。此信息有助于客户订购资源服务。

4.1.3.3. 查询自有资源信息

客户使用此协议查询他们已订购资源的信息。该协议支持查询所有资源的概览信息以及单个资源的详细信息。对于 VM 类型资源, 详细信息包括 VM 的当前状态和运行时信息, 如 CPU、内存、存储和网络使用情况。对于容器类型资源, 详细信息包括容器数量、资源使用情况(CPU、内存、存储、网络)以及容器内部署服务的运行状态。

4.1.3.4. 订购虚拟机资源

客户使用此协议从 Titan 生态系统订购 VM 类型的闲置资源服务。他们需要指定所需的 VM 特性, 如 CPU、内存、存储和网络要求, 以及可能需要的规则, 如要求抵押资源或最低信誉向量级别。成功订购后, Titan Harbour 会为此服务订单生成一个客户任务智能合约, 并将其锚定到 Titan 区块链上。

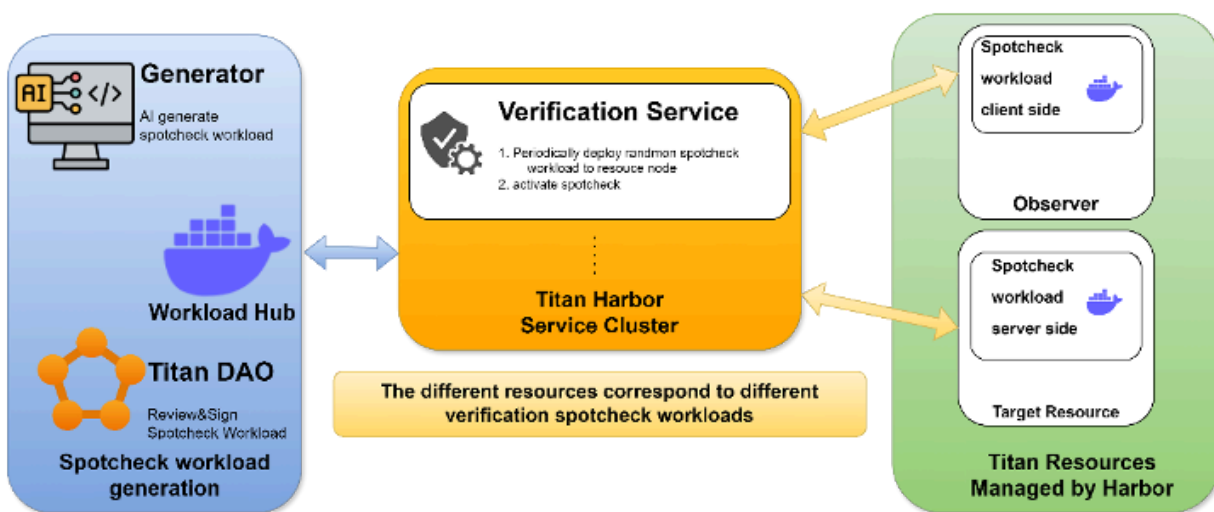
4.1.3.5. 订购容器资源

客户使用此协议订购容器类型的闲置资源服务。他们需要指定容器要求, 如运行时类型(K8S 容器、Titan 轻量级容器、可信容器等)、资源需求(CPU、内存、存储、网络), 以及可能对质押或最低信誉水平的要求。成功订购后, Titan Harbour 会为此服务订单生成一个客户任务智能合约, 并将其锚定到 Titan 区块链上。

4.1.3.6. 查询客户任务合约

客户使用此协议请求 Titan 的数据服务, 以从 Titan 区块链检索他们已订购服务合约的基本信息和当前状态。它支持查询所有服务合约的概览信息以及单个服务合约的详细信息。

4.2. 资源验证系统



Titan 生态系统中闲置资源的主要职责是为 Titan 客户服务，从而赚取这些客户支付的收入。只要闲置资源通过提供满足客户期望的服务来正确履行其职责，Titan 就认为它是诚实的。因此，Titan 的验证系统专注于验证 Titan 客户服务是否按预期运行。为实现这一目标，Titan 需要建立一个验证网络。该网络涉及 Titan 生态系统中许多闲置资源的参与，旨在验证 Titan 客户在生态系统内运行的服务是否按预期运行。

4.2.1. Titan 验证网络

Titan 验证网络涉及观察者、Titan Harbour 和 Titan 客户之间的协作。

- 客户向 Titan Harbour 提交其服务验证要求，通常在订购闲置资源时指定。
- Titan Harbour 根据客户的要求制定验证计划，并指示观察者执行。验证计划伴随客户服务的生命周期，确保服务在其整个持续时间内都受益于 Titan 的验证。
- 观察者是 Titan 生态系统中任何满足验证要求的闲置资源，包括大量的移动设备。值得注意的是，生态系统中的一些设备，如手机，可以选择退出资源调度，而仅作为验证网络中的观察者，执行验证计划并获得相应的工作奖励。
- 在执行验证计划时，Titan Harbour 随机选择若干观察者，并在其设备上部署一个验证程序。观察者的设备运行此程序。验证程序请求被验证的客户服务，记录服务的响应速度 and 内容，并将这些结果发送回 Titan Harbour。
- Titan Harbour 从多个观察者那里收集验证结果，以评估闲置资源是否按预期运行客户的服务。评估结果影响闲置资源的信誉向量并最终影响其收益。

4.2.2. AI 生成的服务负载

Titan 生态系统还利用 AI 生成的服务程序来模拟 Titan 客户端任务。AI 生成的随机性有助于防止恶意资源提供者针对特定的验证程序进行作弊。AI 生成的相似性也使得恶意提供者

更难识别和欺骗验证程序。这些 AI 生成的服务程序以类似于 Titan 客户端服务的方式部署到闲置资源上。然后, Titan Harbour 可以指示观察者对其进行验证。

AI 生成的服务负载提供了几个好处:首先,它们不依赖于实际的 Titan 客户端任务,这意味着即使当前没有为客户端调度,也可以对闲置资源进行验证。其次,它们不受 Titan 客户端的控制或影响。

至关重要的是,这些 AI 生成的服务负载会不断变化;例如, Titan Harbour 可能每天生成不同的负载。

Titan 生态系统还利用生态系统内的闲置 GPU 资源作为生成这些 AI 服务负载的部分计算能力。

4.2.3. Titan 验证系统的有效性

从恶意资源提供者的角度来看,很难区分真正的 Titan 客户端任务和验证程序。如前所述,验证程序可能就是客户端任务本身,也可能是 AI 生成的服务负载。因此,识别和篡改验证程序以逃避或欺骗验证过程是一项具有挑战性的任务。

观察者是随机选择的,他们代表了 Titan 生态系统中的任何参与者,使其数量庞大。恶意为者无法控制大多数观察者。

从恶意 Titan 客户端的角度来看,通过作弊来减少付款也很棘手,因为他们无法轻易区分请求是来自他们的实际客户还是 Titan 观察者。此外, Titan Harbour 可以通过验证 AI 生成的服务负载来证明闲置资源的有效性。

在制定验证计划和指导验证执行时, Titan Harbour 必须努力避免干扰 Titan 客户端服务的正常运行,例如,不要消耗过多资源或花费太长时间。

4.2.4. Titan 的验证流程和协议

Titan 在选择期间优先考虑该区域的观察者,以避免洲际网络延迟等因素影响验证执行,重点关注影响计划具体内容不同资源类型和客户任务类型。Titan DAO 负责建立一个长期的、可演进的验证流程和协议,持续提高验证质量,包括有关观察者执行验证计划的奖励、这些奖励的构成、计算公式等细节。

为了进一步解释,重点介绍了两个验证案例。

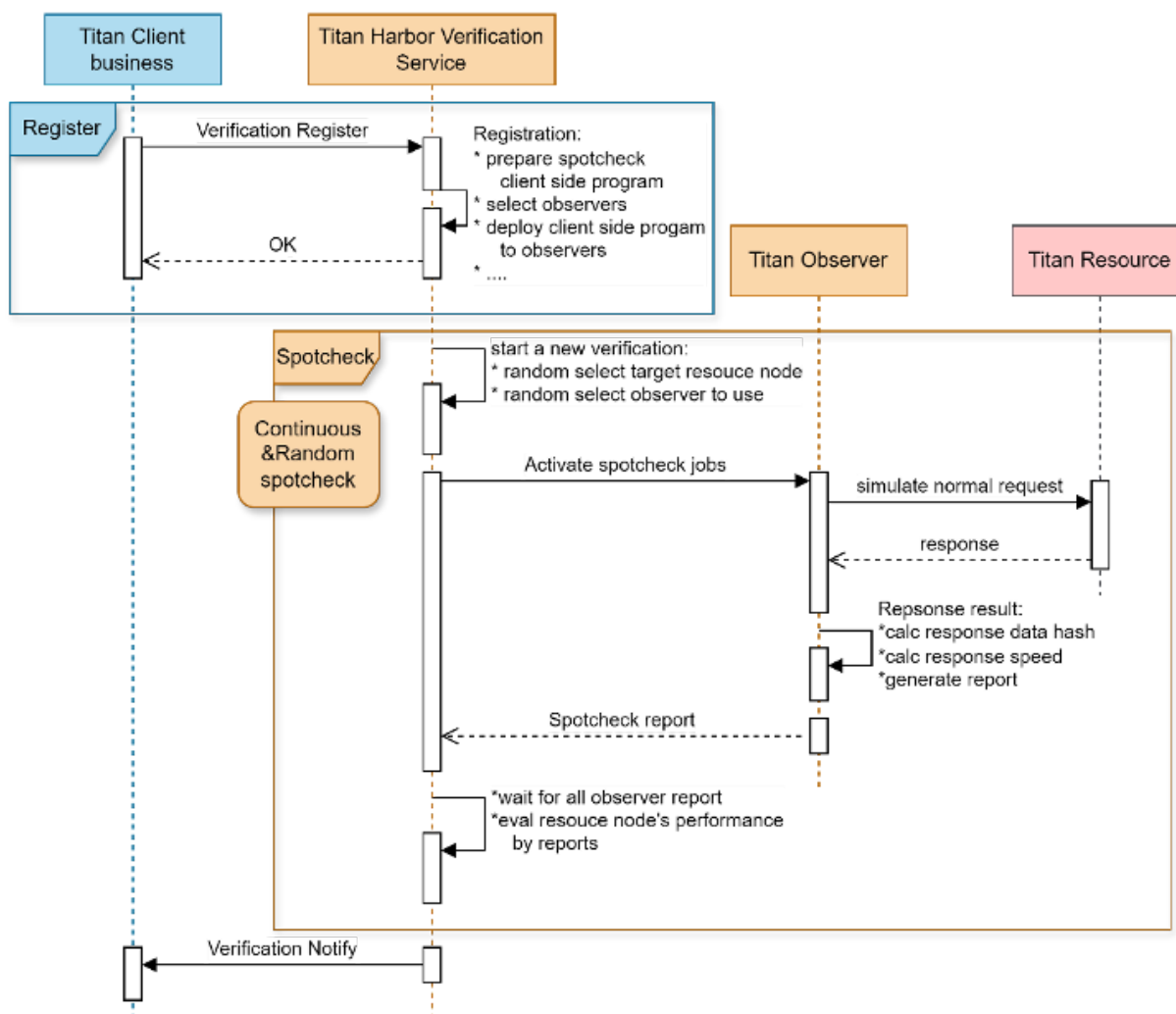
4.2.4.1. Titan 验证系统应用案例 A

项目 A 是一个去中心化存储项目,利用 Titan 生态系统的闲置资源作为其基础。它将内容存储在 Titan 内的多个闲置资源上,利用纠删码等技术来防止因单节点故障导致的数据丢失。

项目 A 利用 Titan 的验证系统对其存储服务进行定期验证。此过程既验证了闲置资源的有

效性，也验证了项目 A 服务的有效性。例如，它可以检测到因闲置资源作弊导致的服务质量下降，以及由存储服务程序崩溃引起的服务故障。

验证流程如下：



1. Titan Harbour 按计划执行验证。它首先请求项目 A 的服务器，以获取待验证的节点集，以及存储内容的名称和哈希值。
2. Harbour 规划验证配对。例如，存储节点集 X 包含 10 个存储相同内容的节点。Harbour 选择包含 100 个节点的观察者集 Y。观察者集 Y 负责验证存储节点集 X。
3. 在观察者集 Y 中，每 10 个观察者节点随机检查存储节点集 X 中的一个节点。
4. 观察者节点模拟真实客户端请求存储节点。存储节点无法区分来自观察者和实际客户端的请求。存储节点发送指定内容，完成请求。
5. 接收后，观察者节点计算内容哈希并测量带宽，将这些结果发送给 Titan Harbour。
6. 验证完成后，Harbour 收集结果，根据结果评估存储节点的服务有效性和质量，对存储

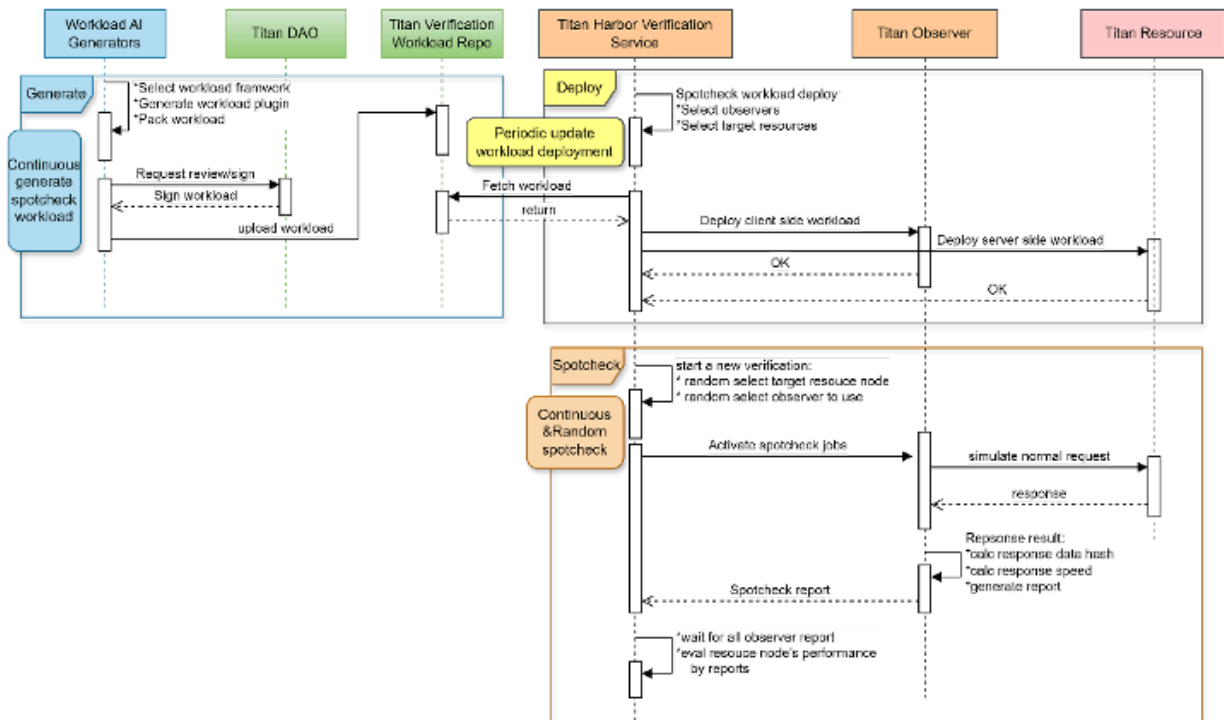
节点的闲置资源进行评分，并更新其信誉向量。

7. Harbour 将验证结果通知项目 A。根据结果(例如，识别出故障节点)，项目 A 需要进行调整。
8. Harbour 记录观察者的贡献，以便后续奖励。

4.2.4.2. Titan 验证系统应用案例 B

在上一个案例中，项目需要遵循 Titan 的验证流程和协议，与验证系统合作。优点是同时验证了资源和服务的有效性；缺点是给项目带来了额外的工作量。

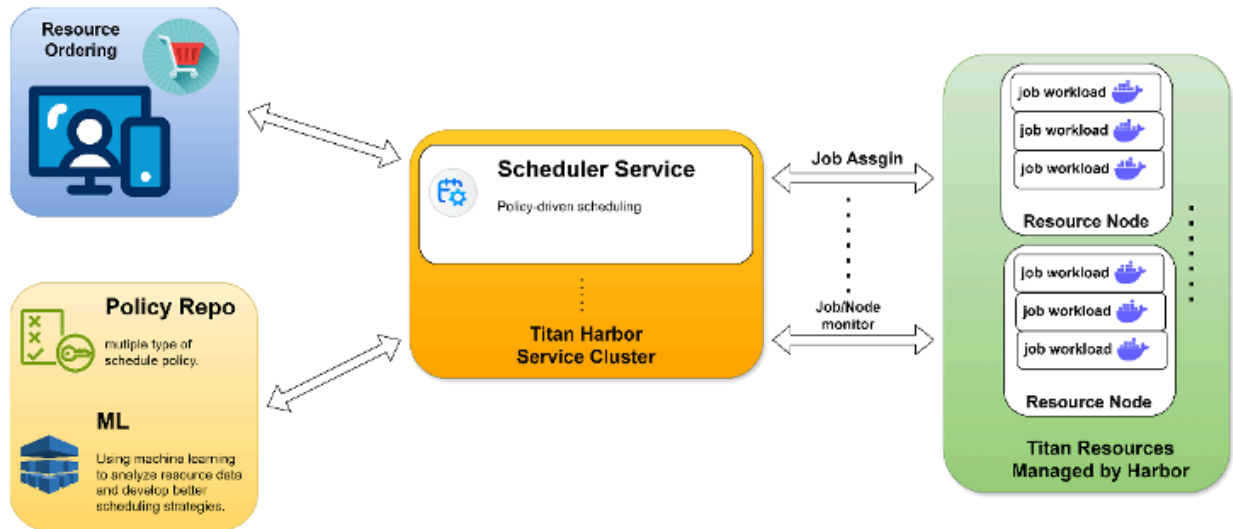
另一种方法是项目不参与。Titan 生态系统独立完成验证过程，仅考虑资源的有效性。流程如下：



1. Titan Harbour 按计划执行验证。它首先将被验证的 AI 生成的服务负载部署到闲置资源节点上。
2. 闲置资源节点无法区分服务负载是为验证而由 AI 生成的，还是真实的 Titan 客户端服务负载。
3. Harbour 规划验证配对，选择观察者集 Y(例如 10 个节点)负责验证此闲置资源节点。
4. 观察者节点模拟真实客户端，请求闲置资源节点上的 AI 生成服务负载，该节点完成请求并返回响应。
5. 收到响应后，观察者节点计算所需的验证结果。
6. 验证完成后，Harbour 收集结果，根据结果评估闲置资源的资源有效性和质量，对闲置资源进行评分，并更新其信誉向量。

7. Harbour 记录观察者的贡献，以便后续奖励。

4.3. 资源调度系统



Titan 生态系统中的闲置资源是多种多样的。调度不仅要考虑类型和容量，还要考虑稳定性。

来自家庭设备的闲置资源的稳定性受多种因素影响：消费级设备固有的较低稳定性、与 IDC 相比不稳定的家庭电源供应，以及不同时间资源可用性的潜在变化。

如果 Titan 客户端订购 VM 资源，所提供的闲置资源必须足够稳定；否则，VM 可能无法满足客户端的要求。因此，在为 VM 类型的客户端任务调度资源时，系统会选择 IDC 服务器级别的设备或具有大量质押和高信誉向量的家庭设备。

如果客户端订购容器资源，对于难以迁移的有状态容器，系统会选择高度稳定的设备。对于易于迁移的无状态容器，系统可以选择稳定性较差的设备，因为问题可以通过从可用池中找到替代品并快速完成迁移来快速解决。

当大量稳定性较差的闲置资源（例如家庭设备）为 Titan 客户端服务时，调度系统必须持续监控它们的可用性状态。当它们变得不可用时，系统必须迅速从可用池中找到替代品。

4.3.1. 调度策略的公平性

每个 Titan 客户端都希望选择质量最高的闲置资源。一个简单的策略可能是为每个任务从池中分配可用的最佳资源。然而，这会导致问题：后来的客户端往往会收到比早期客户端质量更低的资源，而质量较低的资源可能会长时间闲置。

为了满足客户的质量要求，同时确保调度的公平性，当前系统的做法是将闲置资源根据质

量上下限分为几个等级。同一等级内的资源获得平等的调度机会。在为客户任务进行调度时，系统会按一定比例从每个等级的池中选择资源。

例如，假设一个客户订购需要 30 个容器且可靠性为 99% 的容器资源。调度系统以 99% 的整体可靠性为目标，可能会从顶级（每个可靠性 >99%）中选择 10 个资源，从第二级（每个可靠性 >90%）中选择 30 个（3x10），从第三级（每个可靠性 >80%）中选择 40 个（4x10）。从统计学上讲，这相当于为客户提供了 30 个达到 99% 可靠性的资源。

闲置资源调度系统将实施各种调度策略，以更好地匹配不同的客户任务需求。然而，公平性必须是每个策略的考量因素。

系统为每个调度决策生成详细的日志，以单个文件的形式存储。每个文件的哈希值都锚定到 Titan 区块链上。Titan 社区可以使用其哈希值从 Titan Harbour 检索日志文件，以进行审计和追溯。

社区对调度日志的审查和分析，有助于监督和改进调度系统的策略。

4.3.2. 调度策略的特殊性

现实世界中的 Titan 客户端任务可能有独特的调度要求，而系统现有的自动化策略无法满足。在这种情况下，需要一条人工干预的途径。

因此，资源调度系统也支持一种混合模式，将半自动化与人工干预相结合。

一旦 Titan 生态系统社区对特定类型的特殊调度需求有了充分的了解，就可以为其设计和实施新的策略，从而在未来实现对此类调度需求的全自动化处理。

5. Titan 网络用例

Titan 网络的灵活性和可扩展性使其能够支持 Web 2.0 和 Web 3.0 中的各种现实世界应用。通过将闲置的边缘资源聚合成一个由人民驱动、可编程的云, Titan 可以满足多样化的计算、带宽和本地化需求, 通常成本更低, 地理覆盖范围比传统基础设施更广。本章重点介绍 Titan 架构能够提供明确价值的实际用例, 从边缘计算和 CDN 到 AI 代理、区域测试和保护隐私的数据服务。

5.1. 智能边缘 CDN

视频流、在线教育和游戏下载等服务依赖内容分发网络(CDN)来减少访问延迟。然而, 传统的 CDN 提供商节点覆盖范围有限, 导致偏远地区成本高昂。此外, 突发的流量高峰(例如, 体育赛事直播)很容易使中心节点过载。

企业可以利用 Titan 生态系统聚合大量分布式闲置带宽资源(例如, 家庭带宽、闲置 IDC 带宽), 以构建智能边缘 CDN。与传统集中式 CDN 相比, 智能边缘 CDN 具有以下优势:

- 降低成本: 建立在 Titan 的低成本闲置资源之上, 显著降低了企业的内容分发成本。
- 灵活性和智能性: Titan 的资源标准化平面(容器、虚拟机)允许灵活部署和执行企业特定的调度和分发逻辑, 使内容交付更及时、更智能。例如, 它可以感知企业中心服务集群上的实时带宽压力和热门内容, 自主缓存和分流热门内容的分发。
- 邻近性和性能: 边缘 CDN 提供“最后一英里”的内容交付, 比传统的集中式 CDN 更接近用户。这可以带来更优的网络延迟指标, 提升用户体验。

企业可以利用智能边缘 CDN 作为其集中式 CDN 的补充, 从而降低成本并提升分发质量。

拥有现有智能边缘 CDN 应用程序的公司可以直接将其部署到 Titan 生态系统内聚合的地理上分布广泛的闲置带宽资源网络上。相反, 没有自己智能边缘 CDN 应用程序的企业可以利用 Titan 提供的开发框架。这使他们能够通过仅添加最少量的代码来创建自己的智能边缘 CDN 解决方案。

5.2. 边缘计算

物联网(IoT)、智能可穿戴设备和自动驾驶等场景产生了对边缘计算的大量需求。传统的云计算中心, 由于物理距离和高传输延迟, 难以满足毫秒级的响应要求。此外, 将所有数据传回云端进行处理会产生巨大的带宽和计算成本。

企业可以通过将计算服务部署到 Titan 聚合的多区域、近用户的闲置资源的容器/VM 资源平面上, 来构建自己的专用边缘计算网络。来自物联网设备的请求、用户数据处理任务等, 都可以路由到这个基于 Titan 的边缘网络进行处理。

基于 Titan 构建的企业专用边缘计算网络, 不仅降低了企业的数据中心成本(带宽、计算)

，而且由于其靠近用户端，提供了更低的网络延迟和更快的服务响应，从而改善了整体用户体验。

5.3. IP 代理

如今许多应用程序根据用户的 IP 地址限制访问，有效地造成了地理限制。当用户需要绕过这些地理封锁时，使用 IP 代理是一种常见的方法。

Titan 生态系统聚合了来自全球各地的闲置资源。这些资源遵守 Titan 协议的资源平面规范，将其标准化以运行各种客户端应用程序，特别是 Titan 的轻量级容器。一个关键特性是，这些容器仅在主动使用时才加载到闲置资源上。

Titan 客户端可以在这些轻量级容器内运行他们的 IP 代理应用程序。当访问目标应用程序时，连接通过此代理路由，利用底层 Titan 闲置资源的 IP 地址和网络带宽。这有效地避免了目标应用程序基于 IP 的限制。

由于 Titan 的轻量级容器利用闲置资源并按使用时长付费，因此它们提供了显著的成本节省。客户避免了全职 IDC VPS 的费用，后者通常需要为 24/7 的可用性付费，即使他们每天只需要短时间的代理访问，比如一小时。这使得 Titan 的轻量级容器成为一个极具成本效益的解决方案。

此外，Titan 生态系统社区经常共享常见 IP 代理应用程序的模板。这使得客户只需选择他们所需地理区域的闲置资源，并一键部署代理应用程序，即可立即使用。

5.4. 网络爬虫应用

通过在 Titan 的多区域架构上部署分布式爬虫，利用其众多闲置资源，可以满足网络爬取的需求。

基于 IDC 服务器的传统爬虫在带宽和计算能力方面存在限制。此外，它们的 IP 地址范围通常属于特定的 IDC 提供商，很容易被目标网站完全屏蔽。

Titan 的闲置资源具有多区域、靠近用户端和数量众多的特点。这些特性使得基于 Titan 构建的分布式爬虫能够拥有更大的带宽、计算和存储能力。它们靠近用户的 IP 地址不太可能被目标网站屏蔽。

基于 Titan 构建的分布式爬虫，在数据抓取后，还可以利用闲置资源的计算能力进行数据预处理，最后将结果压缩并有序地传输回 Titan 客户端的中心服务进行进一步处理。

Titan DAO 要求对分布式爬虫应用进行 KYC 流程和审查，以防止恶意的 DDoS 攻击程序伪装成爬虫来攻击目标网站。

5.5. 各种 Web2 和 Web3 项目的资源基础

由 Titan 生态系统聚合的庞大、多样化、多区域的闲置资源池，可以作为各种区块链项目的基础资源。

例如，主动验证服务 (AVS) 类型的区块链项目需要大量多区域节点来验证服务有效性。他们可以利用 Titan 丰富的多区域闲置资源作为验证节点。

值得注意的是，在 Titan 内部，Harbour 的验证系统使用生态系统内的闲置资源作为验证器，来验证目标闲置资源的有效性。

另一个例子包括 DePIN 项目，例如那些需要 GPU 资源的项目。他们可以利用 Titan 聚合的大量闲置 GPU 作为其资源池。这样的 DePIN 项目只需将其程序部署到 Titan 的闲置资源上，即可访问这些 GPU。

5.6. 区域性测试与主动监控

确保在多个目标区域提供出色的用户访问体验，对于企业网络应用服务至关重要。被动地等待用户反馈访问问题，往往意味着负面影响已经发生。因此，在用户发现问题之前主动监控以检测问题至关重要。

Titan 生态系统聚合了来自不同地区的闲置资源，包括大量天然靠近用户端的设备，如家用电脑和个人手机。企业可以将其监控程序部署到 Titan 内的这些近用户设备上。这些程序定期运行，监控企业应用服务的访问体验，并将聚合的结果数据发送回企业服务器。这些数据帮助企业了解每个地区的服务性能并及时发现问题。

Titan 上的多区域闲置资源为企业构建多区域、全天候的主动监控系统提供了一个成本效益高且庞大的资源基础。企业可以通过将监控程序实现为 Titan 轻量级容器应用来进一步降低成本，从而受益于按使用付费的定价模式。

同样，企业可以利用 Titan 的多区域闲置资源来构建自己的多区域智能测试系统。这些系统可以运行企业特定的测试程序，模拟来自多个地区的用户交互来测试应用服务。

5.7. 边缘 GPU 计算和 AI 代理

许多家庭用户购买高端 GPU 以获得最佳游戏体验。然而，游戏玩家可能一天中的大部分时间都在学习或工作，只有一小部分时间用于游戏。即使在游戏结束后，这些 GPU 也常常处于闲置状态。

这些闲置的 GPU 可以通过 Titan Agent 安全地聚合到 Titan 生态系统中。

企业可以利用 Titan 中聚合的大量多区域 GPU 资源来构建边缘 GPU 计算能力，将 AI 推理

请求路由到这个边缘网络。例如，一些最先进的 AI 模型，经过开源社区的大量优化，已经可以在家用电脑的 GPU 上运行推理任务。

基于 Titan 构建的企业专用边缘 GPU 计算网络降低了数据中心成本(网络带宽、GPU 资源)。此外，Titan 的闲置 GPU 资源更接近用户，提供更低的延迟和更快的响应，从而增强了整体用户体验。

5.8. 数据主权存储

传统的分布式数据存储解决方案大多是企业部署和管理的、位于数据中心的设施(例如，来自谷歌和 AWS 等云提供商的存储服务)。这些服务由企业控制，用户未来可能面临审查或服务终止的风险。

即使是一些基于区块链的存储服务，其矿工也主要在 IDC 中运行，面临类似的审查和关闭风险(例如，IDC 终止对矿工的服务)。

Titan 生态系统聚合了大量多区域的存储资源。虽然有些在 IDC 中，但大多数来自家庭和个人，这使得它们天生具有抗审查和抗集中强制关闭的能力——这是 Titan 生态系统的独特特性。

行业和社区可以利用 Titan 广泛的闲置存储资源，构建具有数据主权的抗审查存储服务。

基于 Titan 构建的数据主权存储不仅仅是存储本身。结合 Titan 的多区域、丰富的闲置网络带宽资源，它为客户提供了真正可用且有效的数据主权存储服务。

这与一些现有的区块链存储项目不同，后者通常只关注存储而没有提供足够的网络带宽，使得它们对于真实的客户需求不太实用。

5.9. 内容和计算本地化

随着世界各国政府实施更严格的数据驻留和基础设施本地化政策，企业和开发者越来越被要求确保内容和计算工作负载保留在特定的地理或司法管辖区边界内。传统的云提供商通常缺乏满足这些不断变化的本地化需求的粒度、灵活性或可负担性——尤其是在没有附近数据中心的地区。

Titan 网络，凭借其全球分布的闲置资源池——其中许多位于住宅或非 IDC 环境中——提供了一个独特的解决方案。通过使用该网络能够将工作负载直接部署到目标区域或国家内的资源上，Titan 可以在保持低延迟和成本效率的同时，遵守数据主权法律。

例如，一家在巴西运营的企业可以选择性地将内容交付、计算或监控任务部署到物理上位于这些国家内的 Titan 节点上。这种本地化执行模式确保了法律合规性，为区域用户提升

了性能, 并降低了跨境数据传输成本。

6. 未来发展

6.1. 更高的公开透明度

Titan 的核心能力之一是基于 Titan 区块链的数据开放性和透明性。

在当前阶段, Titan 主要专注于标准化闲置资源的资源平面、其管理以及客户端任务。具体来说, 为了快速验证 Titan Agent 和 Harbour 等核心系统的设计和实现, 并验证生态系统内闲置资源的价值并建立价值转换链, Titan 选择了几种典型的客户端业务类型(例如, PCDN)。这些业务已在 Titan 生态系统内成功实施。Titan 社区识别满足这些业务需求的闲置资源, 使用 Titan Agent 对其进行标准化, 由 Titan Harbour 进行管理, 然后将这些资源调度给客户端任务, 提供具有成本效益的资源服务。

未来, Titan 需要分析每个流程, 将关键步骤和结果上链, 包括但不限于:

- 闲置资源验证过程。
- 资源调度决策过程。
- 业务收益分配过程。
- 客户端任务服务质量监控。

目前在 Titan Harbour 服务中实现的一些代码可能需要作为智能合约重新实现, 以使整个过程公开透明。Titan 设想, 在未来具有极高 TPS 的 Titan 区块链的支持下, Titan Harbour 内的所有流程都可以用智能合约实现, 并且所有结果数据都通过智能合约进行锚定。最终, 整个 Titan 生态系统的核心系统将以真正去中心化和公开透明的方式运行。

6.2. 更好的 Titan 区块链核心

Titan 生态系统未来将需要更多的数据上链, 对 Titan 区块链的 TPS 要求越来越高。当前基于 Cosmos 的区块链, 由于其共识协议和区块传播/同步机制, 将难以满足未来的 TPS 需求。

Titan 生态系统需要引入更高效的共识协议和区块传播机制。例如, 采用基于 DAG(有向无环图) 的共识协议并重组当前的区块链基础, 可以使新的 Titan 区块链满足高 TPS 的需求。

预计随着 Titan 生态系统的发展和成熟, 许多 Web3 项目将部署到其上。这些项目也将需要区块链服务, 而具有超高 TPS 的 Titan 区块链将是它们的理想选择。Titan 可以为每种类型的 Web3 项目构建专用的智能合约 SDK 和区块链访问 SDK, 提供一站式、高质量的服务。

例如, 对于 Web3 游戏, Titan 不仅可以提供计算、存储和带宽资源, 还可以提供基于 Titan

区块链的预言机服务。

6.3. 防止资源被滥用于恶意活动

恶意行为者利用 Titan 丰富且成本效益高的闲置资源对目标发动 DDoS 攻击，将对生态系统造成巨大损害。

未来，Titan 可以从两个方面解决这个问题：

首先，需要有效的补救措施。当 Titan 社区收到投诉时，必须立即响应。Titan Harbour 必须能够迅速关闭恶意行为者的工作负载，停止攻击，并禁用其 Titan 客户端账户。这要求 Harbour 能够快速识别恶意工作负载，可能通过分析特定期内闲置资源的网络日志，查看工作负载是否访问了目标。

其次，需要建立和完善预防措施，例如：

- 访问黑名单：Titan 建立一个敏感、受保护目标的黑名单。禁止闲置资源访问此列表上的目标。
- 大额订单的 KYC：涉及订购大量闲置资源的客户业务需求，必须经过 Titan DAO 的 KYC 流程并获得批准后才允许订购。
- 对新和未验证客户的使用限制：新客户和未经 Titan DAO KYC 批准的客户在使用资源场景方面受到限制，特别是在采用出站白名单机制时，只允许访问已声明的目标。如果 Harbour 检测到大量工作负载声明相同的访问目标，可能会怀疑其为潜在的攻击目标，并采取行动以防止事件发生。

随着 Titan 的发展，它将不可避免地吸引更多试图利用其资源进行非法活动的恶意行为者。与滥用行为的斗争将是持续的，需要不断完善对策以保障生态系统的发展。

6.4. Titan 客户端数据安全

Titan 协议的资源标准化（容器、虚拟机、轻量级容器）隔离了 Titan 客户端程序，确保了资源提供者闲置资源及其他资产的安全。这种保护对提供者至关重要，使他们有信心将资源聚合到 Titan 中以赚取收入。

然而，从客户的角度来看，对他们的数据被外部入侵、盗窃或篡改的担忧也是合理的。例如，如果客户的程序在提供者闲置的家用电脑上的容器中运行，容器隔离可以保护提供者的电脑免受任何潜在问题的影响。但是，如果电脑上存在恶意软件，该恶意软件可能会通过调试等方式侵入容器环境，并危及客户的程序和数据。

目前，Titan 通过聚合具有 TEE（可信执行环境）能力的闲置资源来缓解客户的担忧，为对数据敏感的客户可信的容器和虚拟机环境。TEE 的缺点是其中执行的程序性能会显著下降，影响服务质量。为了保持质量，客户可能需要订购许多这样的可信环境。

另一种可能的方法是 Titan 发布一个开源的 hypervisor。一个透明的 hypervisor，在电脑

上 TPM 等可信组件的辅助下，可以在启动时执行自我完整性检查，然后加载提供者闲置资源的主机操作系统。由于 hypervisor 拥有最高权限，确保其可信性使其能够通过技术手段保证其管理的容器和虚拟机的可信性。

6.5. 聚合更多类型的闲置资源

在相当长的一段时间内，Titan 将主要聚合基于设备的闲置资源，如家用电脑、路由器、平板电脑、手机、IDC 服务器等。这些设备拥有计算、存储和网络资源。聚合起来，它们形成一个巨大的能力集群，满足多样化的客户需求，为个人和企业级任务提供具有成本效益的服务。

未来，Titan 可以聚合新型的基于设备的资源，如温度/湿度传感器、空气质量传感器等。这些传感器分布在全球各地，可以为生态系统提供丰富的数据服务。

除了设备，Titan 还可以聚合其他类别的闲置资源，如人力资源。利用其区块链基础，Titan 可以为闲置的人力资本和相关需求建立透明的价值转移渠道。

6.6. 支持丰富的交易模式

目前，Titan 客户通过客户服务平台订购资源。未来，Titan 将支持更多的交易模式，如招标和投标。客户可以通过平台发布招标，多个 Titan Harbour 可以提交投标。

实施新的交易模式和支持新的资源分配需求，可以吸引更多客户加入 Titan 生态系统。

6.7. 落地更多类型的客户业务

Titan 生态系统未来需要支持具有更多样化需求类型的客户业务。这些业务将对 Titan 的闲置资源提出新的要求。例如，Web3 游戏需要以数据中心集群的形式综合使用计算、存储和网络带宽。这些资源需要高速协作，只有通过同一设施内的低延迟、高带宽局域网连接才能实现。Titan 必须能够汇集和管理同一 IDC 内未使用的资源以满足这些需求。

Titan 还必须考虑客户希望将 Titan 资源集成到其私有云管理系统中的场景。例如，一个客户从 Titan 订购闲置资源，这些资源作为 K8S POD 加入客户自己的 K8S 容器编排系统，而不是强迫客户使用 Titan 提供的系统。

未来，Titan 应为客户业务提供资源订购和管理 SDK。客户可以使用这些 SDK 直接订购和管理 Titan 资源，而无需登录服务平台。

Titan 需要主动捕捉和探索业务需求，从社区和各行各业收集需求信息。它应积极构建特定的 SDK、订购和部署模板以及其他“需求粘合剂”，为新业务利用 Titan 的闲置资源提供最便捷的途径。

一个例子是 Titan 构建的智能边缘 CDN 开发框架, 使有内容分发需求的企业能够在此框架之上以最少的代码添加来实现自己的边缘 CDN 程序。然后, 他们可以将其部署在 Titan 聚合的资源上, 并构建其专用的边缘 CDN 网络。

7. 结论

Titan 网络是一个去中心化的资源平台，通过将全球的闲置资源与高需求地区相匹配，弥合了计算、存储和网络能力的全球不平衡。通过将日常设备中的异构资源众包到一个统一的云中，Titan 提供的网络服务为 AWS 和 Google Cloud 等传统提供商创造了替代或补充的解决方案。这独特地将 Titan 定位为引领下一波去中心化物理基础设施网络 (DePIN) 的领导者，有效地将未使用的容量转化为全球范围内的宝贵基础设施。

Titan 架构的核心是 Titan Agent，这是一款开放、轻量级的软件客户端，通过开放协议和虚拟化技术标准化各种资源。通过容器化和虚拟机，Titan 抽象了硬件差异，在每个设备上隔离工作负载，并在整个网络中强制执行统一的 Titan 协议。这种方法提供了强大的安全隔离和一致的服务接口，确保任何贡献的计算、带宽或存储无论其来源如何都能被可靠地利用。

Titan 生态系统被构建为一个模块化、多层次的网络，能够有效地大规模协调资源。Titan Agent 在用户设备（从智能手机、个人电脑到服务器）上运行，在网络边缘贡献闲置容量，而“Harbour”则在整个系统中协调和管理调度。在此基础上，Titan 的服务平台通过利用汇集的资源提供类似云的服务——从去中心化内容交付到 AI 推理工作负载。这种分层设计允许每个组件独立而和谐地运作，从而形成一个灵活高效的去中心化云基础设施。

Titan 还直接解决了分布式网络中信任、安全和公平的关键挑战。一个基于区块链的治理层支撑着所有的协调：资源贡献和使用在 Titan 的账本上得到验证和跟踪，智能合约自动处理任务分配和奖励分发。这种方法确保了资源可用性证明、工作负载调度和结果验证等关键功能在没有集中控制的情况下透明地处理。值得注意的是，Titan 的虚拟化沙箱模型通过严格隔离客户端工作负载与主机设备来加强安全性，并且所有交易和指标都记录在不可篡改的账本上以供审计。因此，资源提供者根据其硬件的实际使用情况获得补偿，而客户只需为他们消费的部分付费——这种经济模式保证了贡献者的公平货币化和消费者的成本效益（Titan 的部署已经为企业客户节省了高达 75% 的成本）。

总之，Titan 为将闲置容量转化为可用基础设施提供了一个实用、可扩展和透明的基础，将自己定位为 Web3 及以后时代的主要 DePIN 基础设施层。它解决了新时代资源供需的结构性矛盾，重新定义了数字时代的生产关系，将“闲置”转化为“生产力”。它赋能个人和企业从被动的资源消费者转变为价值的共同创造者。