Hyperledger Fabric

Introduction
介绍

In general terms, a blockchain is an immutable transaction ledger, maintained within adistributed network of peer nodes. These nodes each maintain a copy of the ledger by applyingtransactions that have been validated by a consensus protocol, grouped into blocks that includea hash that bind each block to the preceding block.

一般来说，区块链是不可变的交易分类账，维护在对等节点的分布式网络中。这些节点都通过应用已由共识协议验证的事务来维护分类账的副本，这些事务被分组到包含将每个块绑定到前一个块的哈希的块中。

The first and most widely recognized application of blockchain is the Bitcoin cryptocurrency, though others have followed in its footsteps. Ethereum, an alternative cryptocurrency, took a different approach, integrating many of the same characteristics as Bitcoin but adding smart contracts to create a platform for distributed applications. Bitcoin and Ethereum fall into a class of blockchain that we would classify as public permissionless blockchain technology. Basically, these are public networks, open to anyone, where participants interact anonymously.

区块链的第一个也是最广为认可的应用是比特币加密货币，尽管其他人也追随它的脚步。另一种加密货币以太坊采用了不同的方法，整合了许多与比特币相同的特征，但添加了智能合约，为分布式应用程序创建了一个平台。比特币和以太坊属于一类区块链，我们将其归类为无公共许可的区块链技术。基本上，这些都是公共网络，对任何人开放，参与者在那里匿名互动。

As the popularity of Bitcoin, Ethereum and a few other derivative technologies grew, interest in applying the underlying technology of the blockchain, distributed ledger and distributed application platform to more innovative *enterprise* use cases also grew. However, many enterprise use cases require performance characteristics that the permissionless blockchain technologies are unable (presently) to deliver. In addition, in many use cases, the identity of the participants is a hard requirement, such as in the case of financial transactions where Know-Your-Customer (KYC) and Anti-Money Laundering (AML) regulations must be followed.

随着比特币、以太坊以及其他一些衍生技术的普及，人们对将区块链、分布式账本和分布式应用平台的底层技术应用于更具创新性的企业用例的兴趣也随之增长。然而，许多企业用例需要无权限区块链技术（目前）无法提供的性能特征。此外，在许多用例中，参与者的身份是一个很难满足的要求，例如在金融交易中，必须遵守了解客户（KYC）和反洗钱（AML）法规。

For enterprise use, we need to consider the following requirements:

对于企业使用，我们需要考虑以下要求：

- Participants must be identified/identifiable
- Networks need to be *permissioned*
- High transaction throughput performance
- Low latency of transaction confirmation
- Privacy and confidentiality of transactions and data pertaining to business transactions

1、参与者必须被识别
2、网络需要许可
3、高事务吞吐量性能
4、事务确认延迟低
5、与商业交易有关的交易和数据的隐私和保密

While many early blockchain platforms are currently being adapted for enterprise use, Hyperledger Fabric has been designed for enterprise use from the outset. The following sections describe how Hyperledger Fabric (Fabric) differentiates itself from other blockchain platforms and describes some of the motivation for its architectural decisions.

虽然许多早期的区块链平台目前正在适应企业使用，但 Hyperledger 结构从一开始就设计用于企业使用。以下各节描述了 Hyperledger Fabric（Fabric）如何与其他区块链平台区分开来，并描述了其体系结构决策的一些动机。

一、Hyperledger Fabric

一、Hyperledger Fabric

Hyperledger Fabric is an open source enterprise-grade permissioned distributed ledger technology (DLT) platform, designed for use in enterprise contexts, that delivers some key differentiating capabilities over other popular distributed ledger or blockchain platforms.

Hyperledger Fabric 是一个开放源代码的企业级许可分布式账本技术（DLT）平台，设计用于企业环境，它提供了一些关键的区分能力，比其他流行的分布式账本或区块链平台。

One key point of differentiation is that Hyperledger was established under the Linux Foundation, which itself has a long and very successful history of nurturing open source projects under **open governance** that grow strong sustaining communities and thriving ecosystems. Hyperledger is governed by a diverse technical steering committee, and the Hyperledger Fabric project by a diverse set of maintainers from multiple organizations. It has a development community that has grown to over 35 organizations and nearly 200 developers since its earliest commits.

区分的一个关键点是在 Linux 基金会下建立了超分类帐，它本身在开放治理下培育开源项目的历史悠久，而且非常成功。hyperledger 由一个不同的技术指导委员会管理，hyperledger 结构项目由来自多个组织的不同维护人员组成。它有一个开发社区，自最早提交以来，已经发展到 35 个组织和近 200 个开发人员。

Fabric has a highly modular and configurable architecture, enabling

innovation, versatility and optimization for a broad range of industry use cases including banking, finance, insurance, healthcare, human resources, supply chain and even digital music delivery.

Fabric 具有高度模块化和可配置的架构，能够为银行、金融、保险、医疗保健、人力资源、供应链甚至数字音乐交付等广泛的行业用例提供创新、多功能性和优化。

Fabric is the first distributed ledger platform to support smart contracts authored in general-purpose programming languages such as Java, Go and Node.js, rather than constrained domain-specific languages (DSL). This means that most enterprises already have the skill set needed to develop smart contracts, and no additional training to learn a new language or DSL is needed.

Fabric是第一个分布式的总帐平台，用于支持以通用编程语言（如Java、GO和NoDE.JS）编写的智能合同，而不是受约束的特定于域的语言（DSL）。这意味着大多数企业已经具备了开发智能合约所需的技能，并且不需要额外的培训来学习新语言或DSL。

The Fabric platform is also permissioned, meaning that, unlike with a public permissionless network, the participants are known to each other, rather than anonymous and therefore fully untrusted. This means that while the participants may not fully trust one another (they may, for example, be competitors in the same industry), a network can be operated under a governance model that is built off of what trust does exist between participants, such as a legal agreement or framework for handling disputes.

Fabric 平台也被允许使用，这意味着，与没有公共许可的网络不同，参与者彼此都是已知的，而不是匿名的，因此完全不受信任。这意味着，虽然参与者可能无法完全信任彼此（例如，他们可能是同一行业的竞争对手），但网络可以在基于参与者之间存在信任的治理模式下运行，例如处理争议的法律协议或框架。

One of the most important of the platform's differentiators is its support for pluggable consensus protocols that enable the platform to be more effectively customized to fit particular use cases and trust models. For instance, when deployed within a single enterprise, or operated by a trusted authority, fully byzantine fault tolerant consensus might be considered unnecessary and an excessive drag on performance and throughput. In situations such as that, a crash fault-tolerant (CFT) consensus protocol might be more than adequate whereas, in a multi-party, decentralized use case, a more traditional byzantine fault tolerant (BFT) consensus protocol might be required.

平台最重要的区别之一是它支持可插拔共识协议，使平台能够更有效地定制以适应特定的用例和信任模型。例如，当部署在单个企业中或由受信任的权威机构操作时，完全拜占庭式的容错共识可能被认为是不必要的，并且会对性能和吞吐量造成过度的拖累。在这种情况下，crash fault-tolerant（CFT）共识协议可能会更充分，而在多方分散的用例中，可能需要更传统的拜占庭容错（BFT）共识协议。

Fabric can leverage consensus protocols that do not require a native cryptocurrency to incent costly mining or to fuel smart contract execution. Avoidance of a cryptocurrency reduces some significant risk/attack vectors, and absence of cryptographic mining operations means that the platform can be deployed with roughly the same operational cost as any other distributed system.

Fabric 可以利用不需要本国加密货币的共识协议来鼓励昂贵的采矿或推动智能合约的

执行。避免使用加密货币可以减少一些重要的风险/攻击向量，并且没有加密挖掘操作意味着可以以与任何其他分布式系统大致相同的操作成本部署平台。

The combination of these differentiating design features makes Fabric one of the better performing platforms available today both in terms of transaction processing and transaction confirmation latency, and it enables privacy and confidentiality of transactions and the smart contracts (what Fabric calls "chaincode") that implement them.

这些与众不同的设计功能的结合使 Fabric 成为当今在事务处理和事务确认延迟方面性能更好的平台之一，它支持事务的隐私和机密性以及实现它们的智能合约（Fabric 称之为"链码"）。

Let's explore these differentiating features in more detail.

让我们更详细地探讨这些差异化特性。

二、Modularity

二、模块性

Hyperledger Fabric has been specifically architected to have a modular architecture. Whether it is pluggable consensus, pluggable identity management protocols such as LDAP or OpenID Connect, key management protocols or cryptographic libraries, the platform has been designed at its core to be configured to meet the diversity of enterprise use case requirements.

Hyperledger 结构经过专门设计，具有模块化架构。无论是可插拔共识、可插拔身份管理协议（如 LDAP 或 OpenID Connect）、密钥管理协议或加密库，该平台都在其核心进行了设计，以满足企业用例需求的多样性。

At a high level, Fabric is comprised of the following modular components:

在较高的层次上，Fabric 由以下模块化组件组成：

- A pluggable *ordering service* establishes consensus on the order of transactions and then broadcasts blocks to peers.
- A pluggable *membership service provider* is responsible for associating entities in the network with cryptographic identities.
- An optional *peer-to-peer gossip service* disseminates the blocks output by ordering service to other peers.
- Smart contracts ("chaincode") run within a container environment (e.g. Docker) for isolation. They can be written in standard programming languages but do not have direct access to the ledger state.
- The ledger can be configured to support a variety of DBMSs.
- A pluggable endorsement and validation policy enforcement that can be independently configured per application.

- 可插拔的订购服务建立了对事务顺序的共识，然后将块广播给对等方。
- 可插入的成员身份服务提供程序负责将网络中的实体与加密标识关联起来。
- 可选的点对点 *gossip* 服务通过向其他对等点订购服务来传播块输出。

- 智能合约（“链码”）在容器环境（例如 Docker）中运行以进行隔离。它们可以用标准编程语言编写，但不能直接访问分类帐状态。
- 分类帐可以配置为支持各种 DBMS。
- 一种可插拔的认可和验证策略实施，可以为每个应用程序独立配置。

There is fair agreement in the industry that there is no “one blockchain to rule them all”. Hyperledger Fabric can be configured in multiple ways to satisfy the diverse solution requirements for multiple industry use cases.

业界有一个公平的共识，即没有“一个区块链来统治它们”。可以通过多种方式配置 Hyperledger 结构，以满足多个行业用例的不同解决方案需求。

三、Permissioned vs Permissionless Blockchains

三、许可与无许可区块链

In a permissionless blockchain, virtually anyone can participate, and every participant is anonymous. In such a context, there can be no trust other than that the state of the blockchain, prior to a certain depth, is immutable. In order to mitigate this absence of trust, permissionless blockchains typically employ a “mined” native cryptocurrency or transaction fees to provide economic incentive to offset the extraordinary costs of participating in a form of byzantine fault tolerant consensus based on “proof of work” (PoW).

在一个无权限的区块链中，几乎任何人都可以参与，而且每个参与者都是匿名的。在这样的背景下，除了区块链的状态在一定深度之前是不可变的之外，没有其他信任。为了缓解这种信任缺失,无许可区块链通常使用“开采”的本地加密货币或交易费用来提供经济激励，以抵消基于“工作证明”（POW）的拜占庭容错共识形式参与的额外成本。

Permissioned blockchains, on the other hand, operate a blockchain amongst a set of known, identified and often vetted participants operating under a governance model that yields a certain degree of trust. A permissioned blockchain provides a way to secure the interactions among a group of entities that have a common goal but which may not fully trust each other. By relying on the identities of the participants, a permissioned blockchain can use more traditional crash fault tolerant (CFT) or byzantine fault tolerant (BFT) consensus protocols that do not require costly mining.

另一方面，被许可的区块链在一组已知的、已识别的、经常经过审查的参与者之间运行区块链，这些参与者在一个产生一定程度信任的治理模型下运行。许可的区块链提供了一种方法来确保具有共同目标但可能不完全信任彼此的一组实体之间的交互。通过依赖参与者的身份,许可的区块链可以使用更传统的崩溃容错(CFT)或拜占庭容错(BFT)共识协议，这些协议不需要昂贵的挖掘。

Additionally, in such a permissioned context, the risk of a participant intentionally introducing malicious code through a smart contract is diminished. First, the participants are known to one another and all actions, whether submitting application transactions, modifying the configuration of the network or deploying a smart contract are recorded on the blockchain following an endorsement policy that was established for the network and

relevant transaction type. Rather than being completely anonymous, the guilty party can be easily identified and the incident handled in accordance with the terms of the governance model.

此外，在这种允许的情况下，参与者通过智能合约故意引入恶意代码的风险会降低。首先，参与者相互了解，所有的行为，无论是提交应用程序交易、修改网络配置或部署智能合约，都按照为网络和相关交易类型建立的认可政策记录在区块链上。犯罪方不是完全匿名的，而是可以很容易地被识别出来，并按照治理模式的条款处理事件。

四、Smart Contracts

四、智能合约

A smart contract, or what Fabric calls "chaincode", functions as a trusted distributed application that gains its security/trust from the blockchain and the underlying consensus among the peers. It is the business logic of a blockchain application.

智能合约或 Fabric 称之为"链码"，作为一个可信的分布式应用程序发挥作用，通过区块链和同行之间的基本共识获得其安全/信任。它是区块链应用程序的业务逻辑。

There are three key points that apply to smart contracts, especially when applied to a platform:

智能合约有三个关键点，尤其是应用于平台时：

- many smart contracts run concurrently in the network,
- they may be deployed dynamically (in many cases by anyone), and
- application code should be treated as untrusted, potentially even malicious.
  - 许多智能合约在网络中同时运行，
  - 它们可以动态部署（在许多情况下，任何人都可以），并且
  - 应用程序代码应被视为不可信的，甚至可能是恶意的。

Most existing smart-contract capable blockchain platforms follow an order-execute architecture in which the consensus protocol:

大多数现有的智能合约区块链平台遵循订单执行架构，其中共识协议：

- validates and orders transactions then propagates them to all peer nodes,
- each peer then executes the transactions sequentially.
- 验证和排序事务，然后将它们传播到所有对等节点，
- 然后，每个对等机按顺序执行事务。

The order-execute architecture can be found in virtually all existing blockchain systems, ranging from public/permissionless platforms such as Ethereum (with PoW-based consensus) to permissioned platforms such as Tendermint, Chain, and Quorum.

订单执行架构可以在几乎所有现有的区块链系统中找到，从公共/无许可平台（如以太坊（基于 POW 的共识））到许可平台（如 Tendermint、Chain 和 Quorum）。

Smart contracts executing in a blockchain that operates with the order-execute architecture must be deterministic; otherwise, consensus might never be reached. To address the non-determinism issue, many platforms require that the smart contracts be written in a non-standard, or domain-specific language (such as Solidity) so that non-deterministic operations can be eliminated. This hinders wide-spread adoption because it requires developers writing smart contracts to learn a new language and may lead to programming errors.

在与订单执行架构一起运行的区块链中执行的智能合约必须具有确定性；否则，可能永远无法达成共识。为了解决不确定性问题，许多平台都要求用非标准或特定于域的语言（如 solidity）编写智能合约，以便消除不确定性操作。这阻碍了广泛的采用，因为它要求编写智能合约的开发人员学习新的语言，并可能导致编程错误。

Further, since all transactions are executed sequentially by all nodes, performance and scale is limited. The fact that the smart contract code executes on every node in the system demands that complex measures be taken to protect the overall system from potentially malicious contracts in order to ensure resiliency of the overall system.

此外，由于所有事务都是由所有节点顺序执行的，因此性能和规模都受到限制。智能合约代码在系统中的每个节点上执行的事实要求采取复杂的措施来保护整个系统不受潜在恶意合约的影响，以确保整个系统的弹性。

五、A New Approach

五、新视角

Fabric introduces a new architecture for transactions that we call execute-order-validate. It addresses the resiliency, flexibility, scalability, performance and confidentiality challenges faced by the order-execute model by separating the transaction flow into three steps:

Fabric 为我们称之为执行订单验证的事务引入了一种新的体系结构。它通过将事务流分为三个步骤来解决订单执行模型所面临的弹性、灵活性、可扩展性、性能和机密性挑战：

- *execute* a transaction and check its correctness, thereby endorsing it,
- *order* transactions via a (pluggable) consensus protocol, and
- *validate* transactions against an application-specific endorsement policy before committing them to the ledger
- 执行事务并检查其正确性，从而认可它，
- 通过（可插拔）共识协议订购交易，以及
- 在将交易提交到分类帐之前，根据特定于应用程序的背书策略验证交易

This design departs radically from the order-execute paradigm in that Fabric executes transactions before reaching final agreement on their order.

这种设计从根本上背离了订单执行模式，即结构在对订单达成最终协议之前执行事务。

In Fabric, an application-specific endorsement policy specifies which peer nodes, or how many of them, need to vouch for the correct execution of a given smart contract. Thus, each transaction need only be executed (endorsed) by the subset of the peer nodes necessary to satisfy the transaction's endorsement policy. This allows for parallel execution increasing overall performance and scale of the system. This first phase also eliminates any non-determinism, as inconsistent results can be filtered out before ordering.

在 Fabric 中，特定于应用程序的认可策略指定哪些对等节点或其中多少节点需要保证给定智能合约的正确执行。因此，每个事务只需要由满足事务的认可策略所必需的对等节点子集执行（认可）。这允许并行执行提高系统的整体性能和规模。第一个阶段还消除了任何不确定性，因为不一致的结果可以在排序之前过滤掉。

Because we have eliminated non-determinism, Fabric is the first blockchain technology that enables use of standard programming languages. In the 1.1.0 release, smart contracts can be written in either Go or Node.js, while there are plans to support other popular languages including Java in subsequent releases.

因为我们已经消除了不确定性，Fabric 是第一个允许使用标准编程语言的区块链技术。在 1.1.0 版本中，智能合同可以写在 GO 或 NoDE.JS 中，而在后续版本中有支持其他流行语言的计划，包括 Java。

六、Privacy and Confidentiality

六、隐私和保密

As we have discussed, in a public, permissionless blockchain network that leverages PoW for its consensus model, transactions are executed on every node. This means that neither can there be confidentiality of the contracts themselves, nor of the transaction data that they process. Every transaction, and the code that implements it, is visible to every node in the network. In this case, we have traded confidentiality of contract and data for byzantine fault tolerant consensus delivered by PoW.

正如我们所讨论的，在一个公共的、无许可的区块链网络中，利用 POW 作为其共识模型，交易在每个节点上执行。这意味着既不能对合同本身进行保密，也不能对其处理的交易数据进行保密。每个事务以及实现它的代码对网络中的每个节点都是可见的。在这种情况下，我们将合同和数据的保密性交易给了 POW 提供的拜占庭容错共识。

This lack of confidentiality can be problematic for many business/enterprise use cases. For example, in a network of supply-chain partners, some consumers might be given preferred rates as a means of either solidifying a relationship, or promoting additional sales. If every participant can see every contract and transaction, it becomes impossible to maintain such business relationships in a completely transparent network - everyone will want the preferred rates!

对于许多业务/企业用例来说，缺乏机密性可能是个问题。例如，在一个供应链合作

伙伴网络中，一些消费者可能会被给予优惠价格，作为巩固关系或促进额外销售的一种手段。如果每个参与者都能看到每个合同和交易，那么在一个完全透明的网络中维持这种业务关系就变得不可能了——每个人都会想要优先的价格！

As a second example, consider the securities industry, where a trader building a position (or disposing of one) would not want her competitors to know of this, or else they will seek to get in on the game, weakening the trader’s gambit.

作为第二个例子，考虑证券行业，一个交易者建立一个位置（或处置一个位置）不希望她的竞争对手知道这一点，否则他们将寻求进入游戏，削弱交易者的游戏。

In order to address the lack of privacy and confidentiality for purposes of delivering on enterprise use case requirements, blockchain platforms have adopted a variety of approaches. All have their trade-offs.

为了解决在交付企业用例需求时缺乏隐私和保密性的问题，区块链平台采用了多种方法。他们都有自己的权衡。

Encrypting data is one approach to providing confidentiality; however, in a permissionless network leveraging PoW for its consensus, the encrypted data is sitting on every node. Given enough time and computational resource, the encryption could be broken. For many enterprise use cases, the risk that their information could become compromised is unacceptable.

加密数据是提供保密性的一种方法；但是，在一个利用 POW 达成共识的无权限网络中，加密数据位于每个节点上。如果有足够的时间和计算资源，加密可能会被破坏。对于许多企业用例来说，它们的信息可能被泄露的风险是不可接受的。

Zero knowledge proofs (ZKP) are another area of research being explored to address this problem, the trade-off here being that, presently, computing a ZKP requires considerable time and computational resources. Hence, the trade-off in this case is performance for confidentiality.

零知识证明（zero knowledge proof，ZKP）是另一个正在探索解决这个问题的研究领域，这里的权衡是，目前计算一个 ZKP 需要相当长的时间和计算资源。因此，在这种情况下，权衡的是保密性能。

In a permissioned context that can leverage alternate forms of consensus, one might explore approaches that restrict the distribution of confidential information exclusively to authorized nodes.

在一个可以利用其他形式共识的授权环境中，可以探索限制机密信息仅分发给授权节点的方法。

Hyperledger Fabric, being a permissioned platform, enables confidentiality through its channel architecture. Basically, participants on a Fabric network can establish a “channel” between the subset of participants that should be granted visibility to a particular set of transactions. Think of this as a network overlay. Thus, only those nodes that participate in a channel have access to the smart contract (chaincode) and data transacted, preserving the privacy and confidentiality of both.

作为一个被授权的平台，HyperledgeFabric 通过其渠道架构实现了保密性。基本上，结构网络上的参与者可以在参与者子集之间建立一个“通道”，该通道应被授予对特定事务集的可见性。把它看作是一个网络覆盖。因此，只有参与通道的节点才能访问智能

合约（链码）和事务处理的数据，从而保护两者的隐私和机密性。

To improve upon its privacy and confidentiality capabilities, Fabric has added support for private data and is working on zero knowledge proofs (ZKP) available in the future. More on this as it becomes available.

为了提高其隐私和保密能力，Fabric 增加了对私有数据的支持，并正在致力于将来的零知识证明（Zero Knowledge Proof，ZKP）。更多信息。

七、Pluggable Consensus

七、可插拔共识

The ordering of transactions is delegated to a modular component for consensus that is logically decoupled from the peers that execute transactions and maintain the ledger. Specifically, the ordering service. Since consensus is modular, its implementation can be tailored to the trust assumption of a particular deployment or solution. This modular architecture allows the platform to rely on well-established toolkits for CFT (crash fault-tolerant) or BFT (byzantine fault-tolerant) ordering.

交易的排序委托给一个模块化组件以达成共识，该模块在逻辑上与执行交易和维护分类账的对等方分离。特别是订购服务。由于共识是模块化的，它的实现可以根据特定部署或解决方案的信任假设进行定制。这种模块化架构允许平台依赖于成熟的工具包来进行 CFT（崩溃容错）或 BFT（拜占庭容错）排序。

In the currently available releases, Fabric offers a CFT ordering service implemented with Kafka and Zookeeper. In subsequent releases, Fabric will deliver a Raft consensus ordering service implemented with etcd/Raft and a fully decentralized BFT ordering service.

在当前可用的版本中，Fabric 提供了一个用 Kafka 和 ZooKeeper 实现的 CFT 订购服务。在随后的版本中，Fabric 将提供一个使用 etcd/Raft 实现的 RAFT 共识订购服务和一个完全分散的 BFT 订购服务。

Note also that these are not mutually exclusive. A Fabric network can have multiple ordering services supporting different applications or application requirements.

还要注意，这些不是相互排斥的。一个结构网络可以有多个订购服务，支持不同的应用程序或应用程序需求。

八、Performance and Scalability

八、可扩展性

Performance of a blockchain platform can be affected by many variables such as transaction size, block size, network size, as well as limits of the hardware, etc. The Hyperledger community is currently developing a draft set of measures within the Performance and Scale working group, along with a corresponding implementation of a benchmarking framework called Hyperledger Caliper.

区块链平台的性能可能受到许多变量的影响，如交易大小、块大小、网络大小以及

硬件限制等。Hyperledger 社区目前正在性能和规模工作组内制定一套措施草案，并相应实施基准 FRAM。这项工作被称为“超账本卡尺”。

While that work continues to be developed and should be seen as a definitive measure of blockchain platform performance and scale characteristics, a team from IBM Research has published a peer reviewed paper that evaluated the architecture and performance of Hyperledger Fabric. The paper offers an in-depth discussion of the architecture of Fabric and then reports on the team's performance evaluation of the platform using a preliminary release of Hyperledger Fabric v1.1.

虽然这项工作仍在继续发展，并应被视为区块链平台性能和规模特征的决定性衡量指标，但 IBM Research 的一个团队发表了一篇同行评审的论文，评估了 Hyperledger 结构的架构和性能。本文对 Fabric 的体系结构进行了深入的讨论，并使用 HyperledgeFabricv1.1 的初步版本报告了团队对平台的性能评估。

The benchmarking efforts that the research team did yielded a significant number of performance improvements for the Fabric v1.1.0 release that more than doubled the overall performance of the platform from the v1.0.0 release levels.

研究团队所做的基准测试工作为 Fabric v1.1.0 版本提供了大量的性能改进，使平台的总体性能比 v1.0.0 版本高出一倍多。

九、Conclusion

九、结论

Any serious evaluation of blockchain platforms should include Hyperledger Fabric in its short list.

任何对区块链平台的认真评估都应该在其短名单中包括 Hyperledger 结构。

Combined, the differentiating capabilities of Fabric make it a highly scalable system for permissioned blockchains supporting flexible trust assumptions that enable the platform to support a wide range of industry use cases ranging from government, to finance, to supply-chain logistics, to healthcare and so much more.

结合起来，Fabric 的独特功能使其成为一个高度可扩展的系统，用于支持灵活的信任假设的许可区块链，使平台能够支持从政府、金融、供应链物流、医疗保健等广泛的行业用例。

More importantly, Hyperledger Fabric is the most active of the (currently) ten Hyperledger projects. The community building around the platform is growing steadily, and the innovation delivered with each successive release far out-paces any of the other enterprise blockchain platforms.

更重要的是，hyperledger Fabric 是十个（当前）hyperledger 项目中最活跃的。围绕平台的社区建设正在稳步发展，每一次连续发布所带来的创新远远超过其他任何企业区块链平台。

十、acknowledgement

十、确认

The preceding is derived from the peer reviewed “Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains” – Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolic, Sharon Weed Cocco, Jason Yellick

上述内容来源于同行评审的“超级账本结构：许可区块链的分布式操作系统”——Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolic, Sharon Weed Cocco, Jason Yellick。