VMWARE: A CTO PERSPECTIVE

How Enterprises Will Leverage Blockchain

vmware

Table of Contents

AUTHOR:

DAVID TENNENHOUSE ET AL.

SVP & Chief Research Officer, VMware

PUBLICATION DATE:

JANUARY 2019

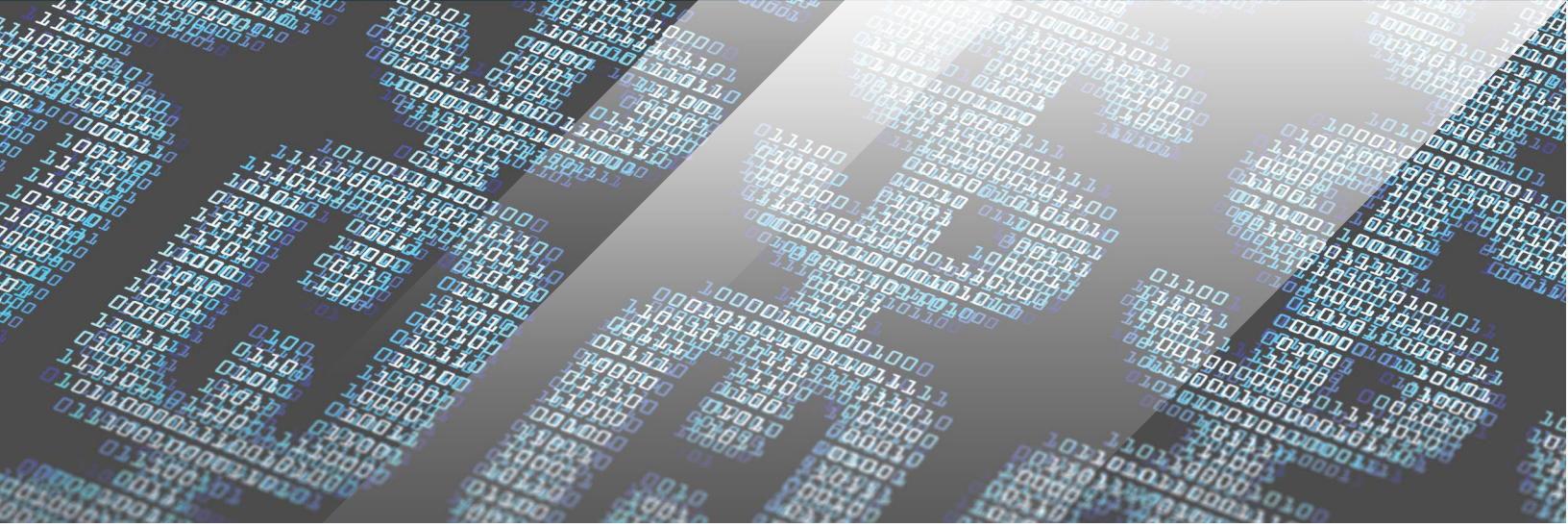

BLOCKCHAIN

Bottom Line

Shared, decentralized ledgers have the potential to enable trusted business-to-business interactions – without the need for an intermediary. We think of this generalization of blockchain technology as enabling a *decentralized trust infrastructure* that can power the digitalized exchange of value.

As with any emerging technology, our role at VMware is to be the Enterprise champion. With that in mind, we have systematically identified and addressed the key technical barriers to Enterprise adoption of this technology. For example, public blockchains sacrifice Enterprise-grade throughput, latency, and environmental sustainability to achieve scale. Similarly, legacy private blockchains sacrifice Enterprise-grade scale in order to achieve better throughput, latency, and energy consumption. They also fail to meet Enterprise requirements with respect to audit, management, governance, regulation, and compliance.

VMware has been actively working to create the next generation of decentralized trust infrastructure that addresses these issues and delivers on the promise of decentralization. One concrete output of that work is Concord, an open source package for the development of Enterprise-grade shared, decentralized ledgers.



Technology Overview

It's all about trust

Important exchanges of assets and information typically involve a *trusted intermediary* that facilitates and attests to their validity on behalf of the participants. One side-effect of Bitcoin is that it demonstrated an alternative approach that replaces the traditional centralized trust model, i.e., the trusted intermediary, with a *decentralized trust* model, based on the notion that it would be difficult to fool or corrupt a large number of otherwise independent entities.

- The centralized model relies on the absolute integrity of the trusted intermediary, i.e., it must always be available and must never be compromised.
- In contrast, the distributed model accepts that, at any point in time, some of the entities may be compromised. If/when that happens, trust can still be preserved so long as the fraction that are compromised does not exceed some agreed threshold.

As we will see, decentralized trust has applications that go way beyond crypto-currencies. There are implications not just for the exchanges of financial instruments, i.e., fintech, but also for supply chains, dissemination of trusted information, identity management, etc.

What's wrong with trusted intermediaries?

In a digital economy, intermediaries are subject to service interruptions and to both insider and outsider attacks. Having your entire supply chain, industry, or national economy depend on a single intermediary – even a highly trusted one – is a significant risk.

From an economic perspective, placing third parties in the middle of every transaction adds friction of many sorts. The decentralized trust approach allows cooperating parties to directly transact, on a peer-to-peer basis, which can dramatically speed settlement, improve visibility, reduce downtime, etc. This approach also fosters agility in the sense that new types of value and/or information exchanges can be introduced without having to create and institutionalize a new purpose-built intermediary.

So, what's the alternative?

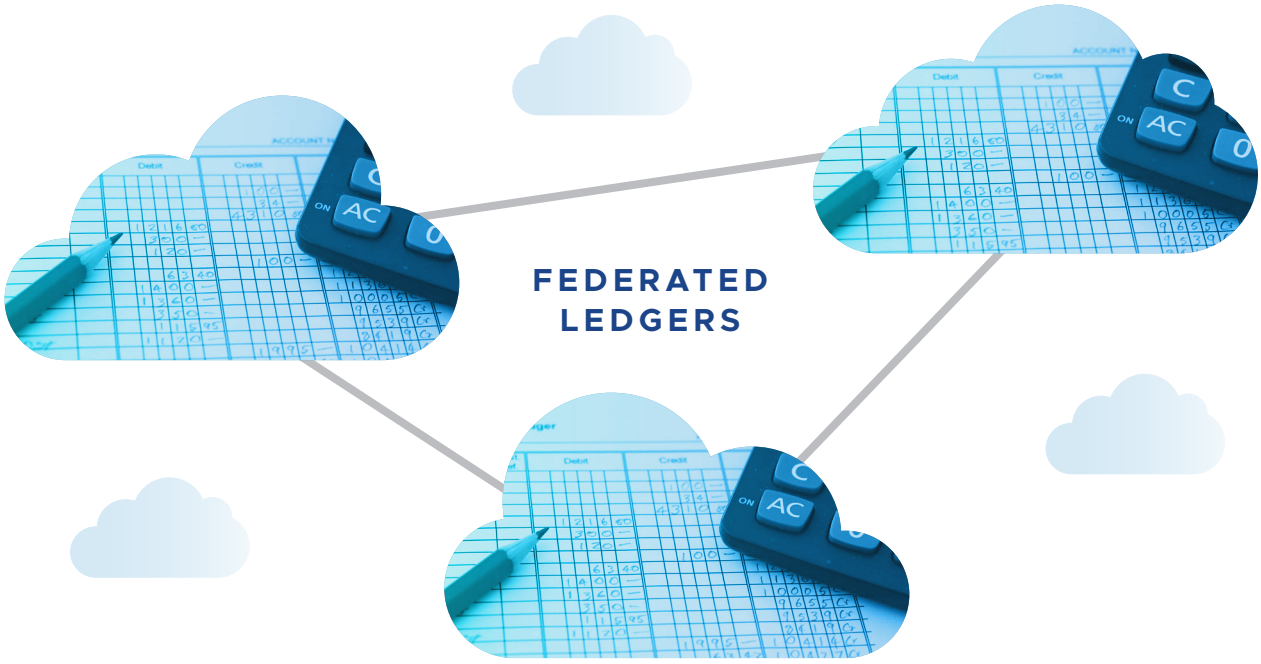
In a centralized model, records of transactions and other important types of information, such as deeds, identities, code fragments, etc. are typically maintained in a *ledger* kept by the trusted intermediary. Although each of the parties may also keep their own ledger, there is no guarantee that the ledgers will agree or even be in the same format, making it difficult to settle transactions and/or conduct audits that reconcile them.

What if the parties in a federation/consortium could maintain a federated, i.e. shared, ledger that could be trusted? They would be able to federate without the need for a trusted intermediary!

A blockchain is a shared digital ledger that is widely replicated and extremely difficult to tamper with. Shared ledgers, such as those used in blockchain, have two key properties:

- *Consensus*, i.e., agreement as to what transactions have been entered into the ledger. This isn't as simple as it sounds since the consensus is not just limited to the parties to the transactions. The agreement extends to the many ledger replicas, which can be spread out both geographically and organizationally. We'll return to this topic multiple times in this paper.
- *Persistence*, i.e., the interested parties have an assurance that the ledger will persist over time and cannot be tampered with. For example, their auditors will be able to examine the ledger at future points in time and conclusively see what was agreed to, making it difficult for a party to forge or *repudiate* a transaction.

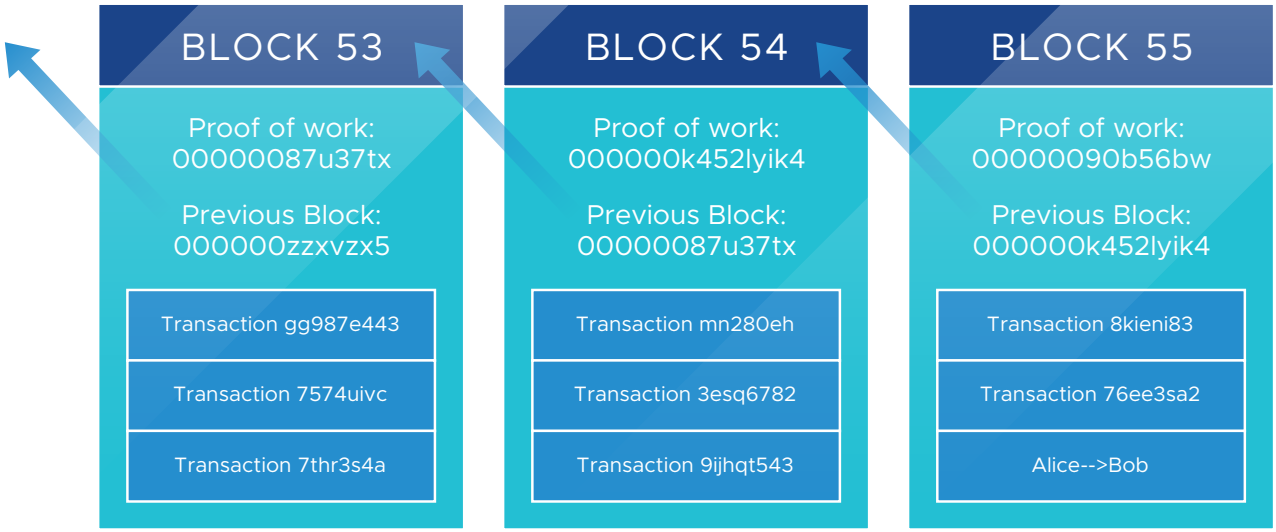
Having many widely distributed replicas also contributes to the *availability* of the ledger in the face of hardware and/or network failures.



Why do they call it blockchain?

A digital ledger stores a sequence of recorded actions that are collected into *blocks*. Each block of actions is cryptographically signed in a way that makes it extremely difficult to tamper with its contents. The signature is analogous to a cryptographic fingerprint of the block. If an adversary tries to change a record in the block, other parties will be able to detect that the signature does not match the contents.

This signature approach is also used to link each new block of recorded actions to the block that came before it. This creates a chain of cryptographically signed, tamper-resistant blocks, hence the name *blockchain*.



Okay, but how do you maintain replicas of the blockchain?

As new blocks are added to the chain, the *consensus protocol* ensures that there is agreement amongst those keeping ledger replicas as to the new block's signature, which implicitly represents agreement as to its contents.

What are smart contracts?

A *smart contract* is a piece of code that can be registered in a blockchain and can be triggered by certain transactions such as the transfer of funds. This code can then create other transactions to automatically *settle* the contract, such as transferring assets. This allows some transfers to be performed without an escrow intermediary. To make it easier to write smart contracts, researchers have created smart contract languages and APIs to support them.

VMware Perspective

What's this really about?

VMware is enthusiastic about the potential to create a decentralized trust infrastructure that realizes the distributed trust model and unlocks the ability for organizations to dynamically and efficiently work together. Architecturally, we think of the distributed trust infrastructure in the following three layers:

SERVICES (Blockchain Applications)

CONTRACTS (Blockchain API)

LEDGER (Blockchain Core)

The key notion here is that tremendous efficiency can be gained if organizations can work together in a peer-to-peer, yet trusted, fashion. This also applies to how governments partner with each other and how they interact with their constituents. As described by **the World Economic Forum, benefits include:**

- ☑ OPERATIONAL SIMPLIFICATION
- ☑ REGULATORY EFFICIENCY
- ☑ COUNTERPARTY RISK REDUCTION
- ☑ CLEARING & SETTLEMENT
- ☑ LIQUIDITY & CAPITAL
- ☑ FRAUD MINIMIZATION

Today, every time a group wants to work together on something of value they need to do so through a trusted intermediary that adds overhead in terms of cost, latency, etc. to each transaction. Furthermore, for each new application, they need to create a new trusted intermediary, including processes, people, technology, governance, etc.

Imagine how much faster and more efficient this would be if we had a reusable trusted infrastructure that could support many different types of consortia and/or cross-organizational interactions. Think of how much easier it would be to work together to solve problems and improve our collective efficiency. Now, think of how auditors, regulators and governments could also be included in that infrastructure and how their operations and our interactions with them could be streamlined.

There are numerous use cases:

- SETTLEMENT-FREE EXCHANGE of financial instruments. We are not suggesting the existing third-party mechanisms for checks, stocks, bonds, etc. will go away (at least not right away) – but think of how much faster and more efficiently new financial products could be launched.
- RECORDING TRANSFERS OF PHYSICAL ASSETS, e.g., real estate, vehicles, commodities, parts, supplies, etc.
- ASSET TRACKING, including as assets move through the parties in a supply chain. This will be especially important in the case of regulated supply chains (pharmaceuticals, food, avionics parts, etc.) where regulators are implicitly party to the transactions. Even in less regulated environments, interactions with border security, customs, and those financing transactions can all be streamlined if they have direct access to a trusted shared ledger with up to date information on the location and status of assets moving through the supply chain.
- TRUSTED DISSEMINATION OF INFORMATION. This can include reliable sources for public keys, revocation lists, software updates, manuals, government publications, etc.
- VERIFIABLE CLAIMS. Sometimes it is not the transaction itself, but the metadata around the transaction that needs to be verifiable. For example, the auditable exchange of know-your-customer (KYC) documentation.
- TELECOM, UTILITIES, ETC. Mobile phone operators, electricity grids, and other others can leverage a decentralized trust infrastructure to streamline their partnerships, e.g., to facilitate mobile roaming and/or energy exchanges.
- HEALTHCARE AND PUBLIC HEALTH. The verifiable exchange of medical information amongst healthcare providers and with public health agencies can all be streamlined.
- HOLDING COMPANIES, ACQUISITIONS, ETC. For example, a multi-national bank may have different legal entities operating under different regulatory regimes. Similarly, airlines that merge may need to keep their flight operations discrete until they have merged their safety policies, training, records, etc. An internal distributed trust infrastructure may allow the parent company to achieve the benefits of federation (e.g., economies of scale, internal transparency/agility, etc.) while satisfying regulatory requirements for isolation.

What are the desirable properties of an Enterprise-grade federated ledger?

VMware's focus has been on the creation of an enterprise-grade decentralized trust infrastructure. Looking beyond the top-level need for consensus and persistence, we have identified the following requirements, which are very different from those of a crypto-currency blockchain:

- KNOWN COUNTER-PARTIES. This argues for *permissioned* environments in which groups of known organizations federate with each other.
- SELECTIVE PRIVACY. The unencrypted details of a given record may be restricted to a subset of the ledger participants.
- A WORLD OF MANY LEDGERS (vs. a single public ledger).
- SCALE: Large number of replicas, with high throughput and low latency – all achieved in an energy efficient way.
- SUPPORT AUDITORS AND REGULATORS as first-class participants.
- MANAGEMENT: Support for dynamic changes in group membership, identity management, software update, monitoring of ledger health, etc.
- DEVELOPER-FRIENDLY. A successful enterprise ecosystem must attract a pool of developers.
- AGILITY. This is a rapidly moving space. The infrastructure must be able to adapt to new requirements and opportunities.

Why do Enterprises care about their counter-parties?

While anonymity has its benefits, e.g., for small cash transactions, it is not applicable to most legitimate enterprise transactions, which are between known counter-parties. This is especially true in the case of supply chains, futures trading, etc. where the parties depend heavily on each other's reputation in order to manage their counter-party risk, i.e., the risk of default on commitments. Similarly, the dissemination of information is often dependent on being able to assess the reputation of the source of that information, e.g., the publisher of a software update. Although knowing who the counter-party is does not ensure that they will act responsibly, their failure to do so comes at the expense of their reputation, i.e., their acceptability as counter-parties in the future.

Why focus on permissioned environments?

Companies often participate in federations or consortia, e.g., supply chains or clearinghouses. Each federation has its own rules as to what constituates appropriate behavior. Members are admitted to the federation based on their reputation and subject to explusion if they are caught cheating. The integrity of the federation depends on the premise that the value members derive from being in the federation greatly exceeds the benefit they might gain from cheating on any one transaction.

In the blockchain space, the term *permissioned* is used to refer to these sorts of closed environments in which only a limited group of entities has permission to participate in the consensus protocol of the ledger. In some ways the notion of permissioned blockchains is the extension of known counter-parties and reputations to the operation of the ledger as a whole (vs. its individual records).

Of course, there is no assurance that all the members of a federation will always act appropriately. Even if they are highly motivated and committed to doing so, it is possible that an enterprise's actions will be compromised through failures, malicious insiders, cyber-criminals, etc. Compromised members may engage in what researchers refer to as *Byzantine* behavior, i.e., they may take deliberate actions to undermine the operation of a federation's trusted infrastructure and even collude with each other to do so.

This is where a key observation underlying decentralized trust comes in: Trust in the group as a whole can be sustained so long as no more than an acceptable fraction of its members have been compromised. Since the membership of a permissioned group is known, the health of its trust infrastructure can be enforced, monitored, and assessed, i.e., the acceptable fraction can be determined and used by the consistency protocol to govern the recording of information, the behavior of group members can be monitored, and/or misbehaving members can be expelled until they take corrective action.

What about privacy?

Since members of a consortium may be both collaborators and competitors, corresponding parties will sometimes require the ability to exercise *selective privacy* over the visibility others have into their information. For example, two parties to a transaction could use a consortium's ledger to allows others to see the meta-data related to a transaction while encrypting its details. Alternatively, they could record only the *signature* of the transaction in the ledger and use *off-chain* mechanisms to record the details. In both cases, the ledger's decentralized trust property would preclude either party from repudiating the transaction or forging its contents. Research technologies, such as *secure multiparty communication and computation* and *Zero Knowledge Proofs*, can be used to create many interesting flavors of selective privacy. For more information and a tutorial on secure multi-party computation see: **http://u.cs.biu.ac.il/~lindell/MPC-resources.html**

Why private ledgers? Why not use Bitcoin's public blockchain?

Although other approaches to decentralized trust are possible, there are multiple reasons why we believe that permissioned environments, each supported by their own, i.e., *private*, decentralized ledger will be the vehicle of choice for Enterprises.

Some of the reasons for our multiple ledger viewpoint are:

- AGILITY. This is a rapidly moving space. Having separate ledgers will make it easier for consortia to adopt new technologies and capabilities on their own timelines.
- ROBUSTNESS. A single ledger represents a single point of failure with potentially wide-reaching economic implications.
- SELF-ENFORCEMENT. A consortium can use non-technical means to ensure good behavior – at least by the vast majority of its members. Conversely, using a public permissionless blockchain means sharing one's trust infrastructure with parties that have very different motivations. Furthermore, sharing a blockchain with parties engaged in criminal activities, such as money-laundering, involves the risk that the infrastructure (and one's business) might be unexpectedly shut down by government agencies.
- SOVEREIGNTY AND REGULATORY. A global public blockchain may not be desirable from a sovereignty and/or regulatory perspective. For example, national governments may look for ways to partition it. Similarly, it will be subject to scrutiny by many different types of regulators (financial, health, transportation, border security, etc.).

Cross-ledger references and interoperability

Although the above obstacles to a single ledger can be overcome, as they have with the Internet, we believe that a better path forward is one in which there are many ledgers and mechanisms that support cross-ledger references. For example, a part vendor who is a participant in multiple supply chains may participate in multiple ledgers and sometimes want to enter records that reference each other.

Note that an ecosystem involving many ledgers can still benefit from economies of scale with respect to software reuse, i.e., the same software can power an unlimited number of ledgers. Over the long term, standardized ledger platforms are likely to emerge. The nearer term is likely to be built on the emergence of a small number of popular platforms that become de-facto standards (at least within specific industry verticals) and ad-hoc mechanisms for their interconnection.

Finally, there will also be a role for public/permissionless ledgers. For example, systems like Ethereum could be used to facilitate the widespread dissemination of information, provided there are off-ledger mechanisms, such as a Public Key Infrastructure, to authenticate the identity of the publishers.

VMware's researchers are exploring ways to facilitate interoperability amongst a range of platforms both private (e.g., Concord, Quorum, Hyperledger) and public (e.g., Ethereum).

How does everyone reach agreement on the ledger contents?

Shared ledgers depend on a *consensus* protocol to ensure that new records, or blocks of records, are authoritatively entered into the blockchain. The key to decentralized trust is that there is a mechanism for a group of entities to reach agreement, i.e., *consensus*, as to the cryp-to-signature of each new block, and that only a tolerable fraction of those entities is compro-mised and/or colluding with each other.

In Bitcoin, *miners* participating in the consensus protocol compete for a financial reward by solving computationally intensive crypto-puzzles. This is sometimes referred to as a *proof-of-work* protocol since the solution to each puzzle is proof that the miner has invested computational effort. The basis for decentralized trust in this sort of system is that an adversary would have to invest in significant computational resources, i.e., some large fraction of all of the resources in use by all of the miners, in order to reliably compromise the system. Although this appears to be a strong basis for trust when there are many independent miners with roughly equivalent resources, we have learned that a subset of the miners will invest in acquiring out-sized levels of computation and that miners will also pool their resources in order to statisti-cally share the financial rewards. Thus, there is reason to be believe that a relatively small number of colluding players could dominate the system. Furthermore, since the system is permissionless, it is difficult to police miner behavior.

Bitcoin's proof-of-work approach has several other drawbacks that limit its suitability for enterprise operations:

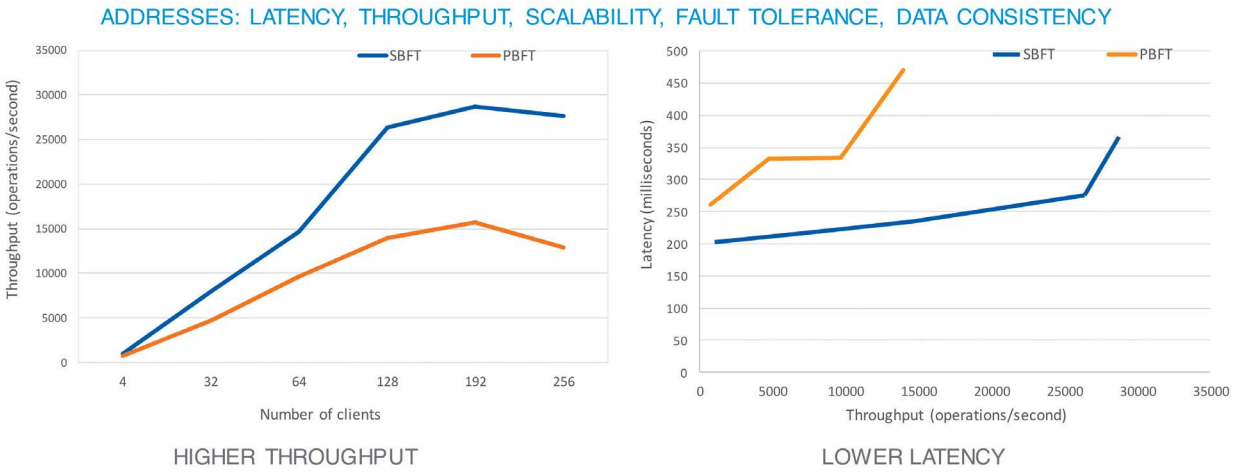
- ENERGY CONSUMPTION. The computational *work* being done consumes energy with minimal social or economic benefit. From a VMware perspective, this goes against our core values and our focus on environmental sustainability.
- THROUGHPUT. The bitcoin protocol implicitly throttles the rate at which blocks can be added to the chain. This limits the overall transaction rate of the system as a whole.
- LATENCY. Bitcoin invokes a complicated tie-breaking strategy to deal with cases where multiple miners solve a crypto-puzzle at approximately the same time. Without going into the details, suffice it to say that participants may need to wait hours or even days before they can be assured that their entry has been *finalized*, i.e., reliably recorded in the ledger.

Is there a better approach to consensus?

Yes, there is. Most members of a consortium will value their reputations and behave appropriately – and no more than a tolerable fraction of them will engage in *Byzantine* behavior. In these *permissioned* environments consensus can be achieved by using a *Byzantine fault tolerant* (BFT) protocol that only admits blocks to the ledger after a subset of the replica operators have voted in favor of them. The minimum number of participants in a BFT subset must allow for a combination of hardware, software, network and byzantine failures.

On the surface, BFT appears to address the challenges listed above: it avoids wasting computation; its throughput is not artificially throttled; and its time to finalization is typically measured in milliseconds. Unfortunately, past BFT implementations, the most well-known of which is PBFT, suffer from scaling challenges. For example, PBFT's aggregate throughput tapers off as the number of replicas in the system goes up. Furthermore, its time to finalization dramatically increases across the limited throughput range that is achievable.

To address these and other issues, VMware researchers have pioneered a scalable consensus protocol, SBFT, that addresses these challenges and is suitable for enterprise-grade operations. **The figure below illustrates some of the benefits of SBFT over PBFT.**



SBFT is at the heart of Concord, an open source package being made publicly available by VMware. Although this paper has only touched lightly on SBFT and Concord, they are the key to unlocking the full potential of decentralized trust. The researchers who developed SBFT have decades of experience with consensus algorithms, which are notoriously difficult to design and implement correctly. In the process of developing SBFT they identified and addressed flaws in PBFT that had gone undetected for many years.

I've also heard of *proof-of-stake*. What's that about?

As an alternative to proof-of-work, some permissionless systems rely on *proof-of-stake*, i.e., they require entities to place cash-like assets (often in the form of cyber-currency) in escrow in order to participate in certain types of smart contracts, mining, etc. The limitation of this approach is that modern economies are highly dependent on the leverage and amplification achieved through the reputation-based financing of economic activity. If the participants in a supply chain are always required to operate on a cash-like basis then, collectively, they will only be able to realize a fraction of the economic activity that they do today.

What are the implications for auditors and regulators?

Auditors and regulators play important roles in modern business operations, but they often add friction to transactions and/or their settlements. There is an opportunity to revolutionize their roles. They could perform their duties more efficiently and securely – and less intrusively – if the federating parties have an authoritative shared ledger that is equipped with the appropriate selective privacy safeguards. We are investigating the creation of tools and technologies that will embed audit functionality in the decentralized trust infrastructure.

Does decentralization mean there is no trusted third party?

Conceptually, it may be possible for a federation to operate without any trusted third party. In practice, we believe that most federations will look to a third party to maintain, manage, and vouch for the health of their decentralized trust infrastructure. For example, there must be a trusted source of software updates to deal with changes to the consensus algorithm and security patches to cryptography modules that the infrastructure depends on. Similarly, it will be important to monitor the health, patch currency, and connectivity of the individual replicas and to actively ensure that there are always a sufficient number of healthy replicas available.

Taking the above a step further, we believe that many federations will contract with a trusted third party to provide a comprehensive service that involves the remote installation, configuration, monitoring, and management of all of the replicas and other components of their decentralized trust infrastructure. This could involve the cloud-based hosting of some number of replicas in addition to those hosted by the individual participants. It could also involve management of the federation's identity infrastructure, admission/expulsion of participants, etc.

Note that this infrastructure management role is very different from the role played by today's centralized intermediaries:

- The infrastructure manager does not have visibility into the transaction data and is not a participant in, or a centralized point of trust for the transactions themselves, i.e., they do not resemble a clearinghouse operator.
- Since the manager's activities are agnostic to the type of transactions involved, the same service can be used to support a very diverse range of federations, i.e., there is an economy of scale involved.

What about developers?

To be successful, an enterprise blockchain platform will need to attract developers, especially at levels higher up in the "stack", i.e., at the application layer. We see the following ingredients to building a successful developer ecosystem at each layer of the architecture that was introduced earlier in this paper:

- MARKETPLACE. A third-party market for value-added services will help developers monetize their expertise, e.g., by offering business analytics and prediction capabilities.
- SOLIDITY / EVM COMPATIBILITY. At the smart contracts layer, Ethereum's Solidity language and its underlying EVM have been embraced by developers as de-facto standards. Enterprise-grade solutions that are EVM-compatible will be able to harvest the interest of those developers.
- OPEN SOURCE LEDGER LAYER. An open source "core" will engender trust and attract developers higher in the stack.

Architectural agility: Beyond blockchains

This is a rapidly moving space and many of its requirements/opportunities have yet to emerge. This calls for an architecture that is modular and amenable to the introduction of new capabilities. For example, the mechanisms used to achieve consensus and persistence should be architecturally distinct.

As we look into the future, we can imagine the extension of the decentralized trust infrastructure to support services other than shared ledgers. For example, consensus protocols such as SBFT could be used to create other types of shared data structures, such as key-value stores, with a wide range of interesting properties that differ from those of today's shared ledgers.

Recommendations

We have the following recommendations for customers contemplating blockchain-based applications:

IDENTIFY FEDERATION OPPORTUNITIES

Invest time identifying a few use cases where compelling business value could be derived if the barriers to federation were lowered. How is business done today? How could ease of federation disrupt the market? Most importantly, who would your partners be in such a federation? Invest time in getting to know your peers in those organizations.

BE WARY OF *BLOCKCHAIN-WASHING*

There is tremendous hype around blockchain right now and many people are proposing blockchain-based solutions for problems that do not require decentralized trust and/or could be addressed with an existing database. Look for those simpler and proven alternatives first.

START WITH POC'S

Get started with a few PoC's. This is a new space and there is no substitute for *learning by doing*.

PLAN FOR SUCCESS

In selecting PoC's, try to focus on projects that could someday be put into production. That may mean anticipating scaling, governance, regulatory, and compliance challenges. For each PoC application, think of its ledger as an in-house platform that could someday support numerous applications involving the same federation members. Then think about the next level up, i.e., how you will participate in multiple federations. Take all of these factors into account when selecting a blockchain platform – and choose wisely!

vmware®

VMware, Inc.
3401 Hillview Ave., Palo Alto, CA 94304
www.vmware.com