CrowdPoint

Vogon Deep Dive

Author: Sean Michael Brehm
January 2023
Executive Leadership

Questions? Email hello@crowdpointtech.com or visit us at https://crwdunit.com.

Vogons are described as officiously bureaucratic, a line of work at which they perform so well that the entire galactic bureaucracy is run by them."

— Douglas Adams, Hitchhiker's Guide to the Galaxy

Table of Contents

What is the big idea?

Collective Intelligence. Collective intelligence (CI) is when a group comes together to share their knowledge and skills to solve a problem or accomplish a goal. It is a way for people to work together to combine their different perspectives and create something more than any one person could do on their own. Working together is a more inclusionary way to set goals.

CI technology has the potential to improve decision-making and provide insight into customer behavior for both consumers and businesses. Consumers benefit from better and more accurate information, while businesses gain a greater understanding of their customers, helping them improve their products, services, and marketing. By leveraging collective intelligence, businesses can better meet customer needs and expectations, increasing customer satisfaction, loyalty, and sales.

CI helps the economy and creates more financial inclusion because they both seek to empower individuals and communities to participate in the global economy. CI is the idea that online and offline collaboration can help people make better decisions and solve problems. On the other hand, financial inclusion enables people to access and use financial services, such as banking, credit, and insurance.

Together, these two concepts are important because they provide a way for people to become more informed and empowered to make the best decisions for their financial futures, thus helping to reduce inequality.

How Does a Decentralized Cloud Support C.I. Better?

A decentralized cloud is a network of computers connected over a distributed peer-to-peer network, eliminating the need for a centralized server. This inversion means that data is stored in multiple locations rather than on a single server, allowing quicker access and more secure storage. Additionally, a decentralized cloud allows CI to be developed more quickly and inexpensively than a centralized cloud, as it does not rely on a single provider or server.

CI is generated by the collective data and insights from nodes on the network, increasing the accuracy and reliability of the data. Furthermore, since a centralized provider does not control the collective, the data is more secure and less susceptible to attack or manipulation.

Lastly, because the collective is not reliant on a single entity for its operation, it is more resilient to outages, making it more reliable and efficient. Centralized cloud companies can shut down applications if they disagree with them; they maintain rigid control and enforce their values, beliefs, and interests on their customers. A decentralized cloud democratizes control and therefore is more likely to generate discovery.

The CI can best identify it and make recommendations even when users want to spread content that could be considered offensive or inflammatory. In some cases, when freedom is limited because laws or regulations are repressive in certain countries, it can surface alternatives for both parties more efficiently.

The Current Cloud Market is Growing, Why Change it?

The global cloud market is estimated to be worth around $260 billion, and analysts project it will grow to $623 billion by 2023.

It is difficult to predict what percentage of the global cloud market decentralized cloud companies will capture in the next five years. However, analysts predict that the decentralized cloud market will continue to proliferate, with some estimates expecting it to reach 10-15% of the total global cloud market share by 2025.

Increasing demand for cloud-based services, such as Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS), is mainly driving growth. Companies are increasingly leveraging these services to reduce their IT costs, improve scalability and agility, and enable the deployment of applications faster and more efficiently. Additionally, the growing trend of "digital transformation" has driven businesses to migrate their operations to the cloud, leading to increased demand for cloud-based services.

Cloud offerings are growing in popularity and adoption in first and third-world nations for a few reasons.

- First, providers typically offer cloud services on a subscription basis, which makes them more accessible and cost-effective for businesses of all sizes in any country.
- Additionally, cloud services provide businesses with access to the latest technology, which can help them remain competitive in their respective markets.
- Finally, cloud services are more secure than traditional IT solutions, which makes them attractive to businesses in any country.

If the current centralized cloud providers work fine, what are the top three reasons we need a decentralized cloud?

1. Increased Security: Decentralized cloud networks can provide greater security for data storage and transactions by decentralizing the data, making it harder for hackers to target a single point of failure.
    - With more devices connected to the internet, cybercriminals have more opportunities to exploit vulnerable systems.
    - Cyber attackers can breach even the most secure systems as their skills have become more sophisticated.
    - As more people use technology, the lack of understanding about the potential risks associated with online activities and the lack of proper cyber security measures leave many systems vulnerable to attack.
2. Increased Privacy: Decentralized cloud networks can help protect users' privacy by reducing the risk of data breaches and allowing users to control their data.
3. Lower Costs: Decentralized cloud networks can reduce costs associated with cloud storage and computing by eliminating the need for expensive hardware and software infrastructure.

In summary, a decentralized cloud offers greater security, privacy, and reliability than a centralized cloud because the protocol distributes data and files across multiple independent servers and computing nodes with no single point of failure. It also eliminates the need for a third-party provider, allowing users to store their data directly on the network.

Additionally, decentralized cloud storage is more affordable than centralized cloud storage since users can pay for only the storage space they need without paying for the infrastructure of a centralized cloud provider. Finally, decentralized cloud storage allows users to have greater control over their data, as they can choose which nodes store their data, who has access to it, and how others use it.

Why Vogon Will Redefine the Decentralized Cloud

- Vogon is a real-time, high-performance decentralized crypto-cloud workhorse
- Vogon is infinitely scalable through consensus group mitosis (cell splitting)
- Vogon does not perform useless work and consumes energy normally
- Vogon has blockchain-installable microservices for the decentralized cloud
- Vogon solved the dApp update vulnerability
- Vogon is superscalar and extremely fast
- Vogon is written in Java
- Vogon is a fresh rethinking of decentralized consensus
- Vogon uses a new model for mediating byzantine fault tolerance
- Vogon makes the world better through collective intelligence

Secure Transactional Distributed and Decentralized Database

For several reasons, a secure, transactional, distributed, and decentralized database is necessary for a decentralized cloud.

- Firstly, having a decentralized database allows data to be stored securely and is not vulnerable to single points of failure. If a single data center or server fails, the data remains secure and accessible from other locations.
- Secondly, having a distributed database ensures that the protocol stores data in multiple servers, which helps to prevent data loss or corruption.
- Thirdly, having a decentralized database ensures that no single entity or central authority controls data, which helps to protect data privacy and security.
- Finally, having a distributed and decentralized database also allows for faster processing, as applications can access data from multiple servers in different locations.

The underlying technology for a secure, transactional, distributed, and decentralized database must be able to house an organized collection of data stored within a distributed document system.

a. It must be able to be accessed, manipulated, and updated in a structured way on a decentralized platform without the need for a central authority.

b. This data store must support an immutable and transparent ledger to become a foundation for automated agreements with the ability to support decentralized applications that execute themselves when the agreements meet certain conditions.

Automated Secure Contract Negotiation

The decentralized cloud needs an automated secure contract negotiation capability because it is a distributed system and requires trustless interactions between multiple parties to ensure that transactions and data remain secure. The decentralized cloud can ensure that all parties involved in a transaction agree with the rules and regulations governing the transaction using an automated secure contract negotiation capability, thus reducing the chances of any disputes arising. Additionally, this capability helps ensure that agreements between the parties are secure and cannot be tampered with or changed.

Technologists design scalable computer protocols to digitally facilitate, verify, or enforce the negotiation or performance of a contract. Like the current smart contracts of today, this protocol must enable the performance of credible transactions without third parties. These transactions are traceable, transparent, and irreversible.

To be widely adopted, smart contracts must evolve in terms of their security, scalability, and usability. They need to handle increasingly complex interactions while ensuring that the program can correctly enforce the terms of the contract. Additionally, they need to be more user-friendly so that individuals can easily understand and interact with them. Finally, the evolution of a smart contract needs to be secure and protected from malicious actors. This function could include incorporating security features such as encryption and authentication protocols to ensure the integrity of the contract.

File Storage

A distributed and decentralized file system must integrate with a decentralized database that allows users to store and access data in individual files without relying on a single server or centralized cloud storage provider. This technology helps ensure that data within these files are secure and immutable, even if the data in the files get stored across multiple devices.

This process leverages distributed document stores to identify a file in relation to a document and integrates with a distributed file storage system by using unique hashes. When the protocol adds a record to the document store, Vogon generates a unique hash for the document. Within the document, another hash can then identify the file related to the document in the distributed file storage system.

This function not only allows the document to access the location of the file store but also allows the file store to retrieve the document as needed. Since the hashes are unique and immutable, the distributed file store can be sure that the data stored is the same as the data retrieved, even if or when stored across multiple nodes. This approach ensures data integrity and security.

Decentralized Applications (DAAPS)

dApps are applications traditionally run on a decentralized network, such as a blockchain. These applications are secure, reliable, and are usable for a variety of tasks, such as creating digital tokens or issuing digital assets.

The orders of magnitude of efficiency achieved by a decentralized app accessing a decentralized database operating on a decentralized cloud with a polyglot VM deployed are incredibly high. Integrating a polyglot VM with a database offers a unique approach to distributed computing, allowing for developing low-level applications in multiple languages deployable on various platforms.

The introduction of a dedicated virtual machine for a decentralized cloud allows for faster and more efficient execution of code, as well as allowing for scalability and fault tolerance. In addition, decentralized databases offer greater security and privacy than their centralized counterparts, and using a decentralized cloud would further increase the efficiency of the distributed application.

Compaction Technology

Compaction technology shrinks, secures, and speeds up data transmission. The process significantly reduces the data size of a file and includes built-in ultralight security, thus reducing the size of files stored in a decentralized database. It does this by taking a large file (like a JSON file), shrinking it, and securing it into a smaller size. This technique helps reduce the amount of space the file takes up, making it easier to store and transfer between database nodes.

Compaction technology allows the same amount of data to be stored in less space, making it easier to access and share. It also helps speed up the sending and receiving of data, making it faster to use. Compaction technology is instrumental in decentralized databases, as it helps reduce the amount of data stored in each node while allowing the user to access the same information if they have the code book.

Internet of Things (IoT)

IoT is a network of connected devices that can interact with each other and transfer data over the internet. This technology can create a secure and decentralized cloud infrastructure, allowing devices to connect securely with each other and share data without needing a central server.

Vogon Decentralized Cloud Explained

Vogon is Unique & Forward Thinking

Vogon is a unique decentralized cloud because the technology incorporates a powerful virtual machine with microservices architecture. Vogon makes it easy to access cubes of distributed data stores that are interconnected in a deterministic manner using consensus groups.

Consensus groups are beneficial for replicating nodes using a concurrent deterministic approach for a database because they ensure that all nodes in the network reach a consensus on the data stored in the database.

A concurrent deterministic approach is like when a group of friends works together to solve a problem or build something. All the friends agree on the steps needed, and everyone works together simultaneously to finish it quickly. It's a team effort.

The concurrent deterministic approach is more efficient than the proof-of-work and proof-of-stake used by energy-hogging blockchains because it requires less computing power. In a proof-of-work system, computers must solve complex mathematical puzzles to validate transactions. This method requires a lot of processing power and uses significant electricity. The most well-known blockchain that uses proof-of-work consensus is Bitcoin. Other blockchains that use this system include Ethereum, Litecoin, Dash, and Monero.

A concurrent deterministic approach is more efficient than proof-of-stake blockchains because it requires less computing power to achieve the same results. In a proof-of-stake system on a blockchain, miners must use their computing power to solve complex mathematical puzzles to validate transactions. This process is energy-intensive and time-consuming. Some popular blockchains that use proof-of-stake include Ethereum, NEO, TEZOS, Dash, Cosmos, Lisk, and EOS.

In a concurrent deterministic approach, server and hosting owners don't have to solve puzzles to validate transactions. Instead, the protocol uses a consensus algorithm to validate the transactions, which is much faster and uses less energy. In simple terms, a concurrent deterministic approach is more efficient because it does the same job as proof-of-stake but takes less time and energy.

With the concurrent deterministic approach, computers use less power to validate transactions, which are processed more quickly. This methodology makes it more efficient and cost-effective than proof-of-work and proof-of-stake.

This approach ensures that all nodes in the network have the same data, which is essential for maintaining the integrity of the database. Consensus groups also provide a mechanism for conflict resolution, allowing all nodes to agree on the most accurate version of the data. This approach helps ensure the database remains consistent and up to date.

These data stores contain machine and human-readable documents with unique IDs that enable the query of records like a database. Vogon is unique because it also has distributed ledger technology that stores data records as part of the document and is cryptographically secured. Unlike a centralized cloud server, it can store and run data across several physical locations. With the Vogon Decentralized Cloud, the data is split up and stored in multiple locations, ensuring that no single server or location holds all the data. This process makes it more secure, as the data is spread out and much more difficult to compromise.

Database vs. Blockchain

A database is like a giant filing cabinet with lots of drawers, and each drawer has information in it that is organized and searchable. A blockchain is like a big wall with lots of bricks. Each brick has information in it that connects with information in other bricks. They all work together, so if someone changes one brick, it affects all the other bricks in the wall.

Database vs. Ledger

A database is like a giant filing cabinet with many drawers that holds all your important papers. It helps you organize your information in an orderly way to find it easily. Each drawer has many folders containing paper documents with information in each folder. You can find information quickly by looking in the correct folder. A ledger is like a giant notebook. It has lots of pages with information written on them. You can look through the pages to find the information you need, but it might take a while because there are many pages. It also helps you keep track of who has access to the information.

A ledger is essentially a record book. Protocols use distributed ledger technology to keep track of cryptocurrency; however, they can also track money and other important information. It records transactions between two people. When a transaction gets recorded in a ledger, it cannot be changed or erased. This failsafe means that it is immutable. It is also transparent, meaning anyone can see the transaction in the ledger. It is like when the coach keeps track of players making the team. Everyone can see the list, and it cannot be changed.

Vogon is Better Because it Fuses Multiple Technologies

Vogon Decentralized Cloud is like a giant toolbox that holds all your tools. Those tools include capabilities like a database, blockchain, and a ledger, along with a virtual machine.

The Vogon Decentralized Cloud is like a big warehouse with many specialized polyglot virtual machines that act like containers shaped like cubes and are organized in consensus groups. The difference is that any cube can be anywhere in the worldwide warehouse. Each cube has information in it that connects to the other cubes. Each side of the cube has a role in ensuring that all the cubes work together, so if someone interacts with one cube, it affects all the other cubes in the wall.

The Vogon Decentralized Cloud enables each cube to have many special powers on the outside, and inside, there are even more things that help make it transformative. It's like having a superhero inside the cloud that can help it do extraordinary things and work faster. One of those superpowers is to store data.

Each cube can hold documents, like pieces of paper with information written on them. The documents can hold different data types, like names, addresses, and multimedia, stored as JSON files.

- JavaScript Object Notation, more commonly known by the acronym JSON, is an open data interchange format that is both human and machine-readable.
- JSON is an excellent format to store data in for a distributed document database because it is lightweight, easy to read and write, and its hierarchical structure organizes data in a way that mirrors its natural structure. Additionally, JSON is a widely supported data format compatible with many programming languages, making it easy to access and manipulate data stored in a distributed document database.
- These cubes store data without using a traditional relational database management system (RDBMS). Storing a JSON inside this cube is a good solution for many applications because it is faster and more flexible than traditional databases. It is also easier to scale, so if your application grows in popularity, you don't have to worry about the database being able to keep up. The JSON files stored in these cubes are more reliable and secure than traditional databases, so you don't have to worry about third parties compromising your data.
- A combination of references and embedded documents determine the relationships between JSON files in the cubes; with references, a document stores the unique ID of another document, which the protocol can use to retrieve the related data. Embedded documents are stored inside other documents and can represent one-to-many relationships between collections.
- To find the information you need, you can open the warehouse and look through the containers and their documents until you find the correct information. Vogon makes storing and finding information easy because the warehouse floor is an efficient virtual machine, and you can talk to any container using a microservice. Each side of the cube-shaped container performs a unique role.

Vogon's virtual machine can turn code written in different programming languages into a unique language that all computers can understand. It makes it so that software written in different languages can work together and communicate with each other, like friends from different countries speaking different languages. The VM acts like a super-powered computer processor that can do more work in less time. It can do more work by looking at each task it has to do, breaking it down into smaller pieces, and then quickly solving those pieces. This process helps it use the CPU power from a server more efficiently. Think of it like a car engine that can get more miles per gallon of gasoline.

Vogons VM files help make programs run faster and use less memory. It takes all the parts of a program, like the code and images, and puts them into a special WAR file. This file can then be sent to a computer to run the program. It's like having a unique backpack containing all the puzzle pieces, so you don't have to carry them around separately. Here are five reasons Vogon's data storage with an embedded polyglot-powered virtual machine is better than a traditional database.

1. Increased Performance: Vogon's embedded polyglot-powered virtual machine is more efficient and performs better than a database that uses a non-polyglot virtual machine.
2. **Multi-language Support**: Vogon's polyglot-powered virtual machine can support multiple programming languages, allowing developers to write more efficient and robust code.
3. **Improved Security:** Vogon, a polyglot-powered virtual machine, is less vulnerable to attack, as the code is easier to audit and secure.
4. **Increased Flexibility:** Vogon's polyglot-powered virtual machine is more flexible, allowing developers to adapt their code quickly and efficiently to changing requirements.
5. **Reduced Development Time:** Vogon's embedded polyglot-powered virtual machine enables developers to develop applications quickly and easily without writing code from scratch.

Vogon microservices run faster and use less energy. By working with the VM, it leverages the compression so that it takes up less space. That makes it easier for computers to store and access the code, so it can run faster and use less energy. So, it's like making a big pile of code smaller, so it fits in a smaller computing space and is easier to get around.

A database without an embedded microservice is not a very efficient database. Without it, the database can still do things, but it may take longer and not be as robust.

Vogon is a unique decentralized cloud built on a powerful virtual machine with microservices that make it easy to access cubes of distributed data stores. The cubes are interconnected in a deterministic manner using consensus groups. These data stores contain machine and human-readable documents with unique IDs that enable the query of records like a database. Vogon is unique because it also has distributed ledger technology that stores data records as part of the document and is cryptographically secured. Unlike a centralized cloud server, it can store and run data across several physical locations. With the Vogon Decentralized Cloud, the data is split up and stored in multiple locations, ensuring that no single server or location holds all the data. This methodology makes it more secure, as the data is spread out and much more difficult to compromise.

In summary, there are three reasons why Vogon, with an embedded database using a concurrent deterministic approach through consensus groups, is a more efficient technology than a blockchain:

1. **Faster Transactions:** Transaction speed is much faster with a concurrent deterministic approach since it requires only one node to process and validate a transaction instead of the entire network. This reduction means that transactions complete much faster than with a blockchain, which requires a total consensus of the entire network.
2. **Lower Costs:** Since transactions with a concurrent deterministic approach complete faster, the cost associated with these transactions is significantly lower than with a blockchain. This saving makes the technology much more efficient and cost-effective for businesses.
3. **Scalability:** A concurrent deterministic approach can scale quicker than a blockchain because it does not require the same resources as a blockchain to process and validate transactions. This method makes the technology more efficient as businesses can quickly scale their operations without investing in expensive infrastructure.

Towels | Microservice Power!

The most glaring difference between the Vogon decentralized platform and other solutions is that Vogon hosts microservices natively.

As discussed above, a microservice is a type of computer program that does one specific task. It is essential in a decentralized cloud because it helps ensure everything runs smoothly and efficiently. Just like how a person might break down a big project into smaller pieces to make it easier to manage and complete, a microservice can break down an extensive, monolithic codebase into smaller parts so that each part can be monitored and managed separately. This architecture makes it easier to make sure that everything is running correctly.

Vogon uses this microservice or towel to break an application into smaller, loosely coupled services. A towel leverages a virtual machine (VM) to be more efficient by helping developers use resources better. It enables developers to compile code ahead-of-time (AOT) and optimize for specific platforms. This process can reduce memory overhead and boot time and increase performance. Embedded VM Towels also benefit from being part of a platform-independent runtime environment. It allows applications to run on different platforms without recompilation. This interoperability can make microservices more portable and reduce the time and effort needed for deployment.

A Towel ensures that each service is focused on a single feature or function and is independently deployable and scalable. It allows developers to develop, deploy, and manage services independently, making the overall application more flexible and easier to maintain.

Additionally, Towels are deployed across multiple locations, allowing for more distributed and decentralized cloud architectu res. This increased level of decentralization can help improve performance, reliability, and security while allowing developers to take advantage of the latest technologies and services.

Vogon Microservice Forms

A microservice is a small, highly specialized set of web services for achieving some goal. Microservices are vital to support a distributed data store on a decentralized cloud because it helps to ensure scalability and reliability.

Microservices enable scalability by allowing for the easy addition and removal of services as needed. This facility ensures that the system can handle an increase in traffic without needing to modify the entire system. Microservices also enable reliability by distributing the data stored across multiple nodes. It reduces the risk of any single node failing and breaking the entire system.

Microservices allow the management of each node independently, thus supporting the decentralization of the cloud. It helps ensure data security and privacy. With Vogon, they come in three primary forms:

1. WAR files containing micro-site static assets for web browsers and other clients.
2. Web Service API code for the handling and execution of web service APIs on Vogon.
3. Transactional objects that can manipulate the blockchain.

All of these microservice forms install onto the Vogon Decentralized Cloud. Once installed, thesemicroservices are immediately available for use on the internet. No other blockchain solution exists with these capabilities.

For example, a developer like Tony Stark may create a set of web service APIs to represent fractional ownership of his collectibles. He may create web pages to view those collectibles, APIs to show ownership, and transactional objects used to change ownership or allow the collectibles to be bought and sold on open markets.

When Tony deploys these individual modules, which we call "Towels" [1], thousands of network nodes immediately light up and can present the website, accept API calls, and perform transactions for Tony's collectibles. Tony has deployed to the decentralized cloud.

Vogon Power Requirements & Security

Vogon uses aggregate BLS 12-381 keys, organized into a set of keychains called a consensus group, to verify its "blocks" or cubes on the decentralized cloud. These keychains are subsets of their respective blockchains, indicating consensus group membership. BLS 12-381 keys are a type of lock that uses a combination of numbers to unlock it. Instead of having a physical key, you must enter a specific combination of numbers to open the lock.

BLS 12-381 keys provide a secure way to store data in the Vogon Decentralized Cloud because they are tough to crack. This level of security means that even if someone tries to access the data stored in the cloud, their chances of succeeding are slim. Additionally, Vogon uses these keys to create digital signatures, which can prove the authenticity of a transaction or document. This signing process makes it easy for anyone to use BLS 12-381 keys to help protect their data from unauthorized access and to ensure that their private information is kept safe.

The assembly of these containers acts like a blockchain and gets secured through threshold signatures. Threshold signatures prove that data is secure and reliable when stored in a decentralized cloud. This measure means that only certain people can access the data, which is beneficial because it adds an extra layer of security so that if someone hacks the cloud, they won't be able to access it. It also makes it harder for someone to delete or modify the data without permission from everyone who has access.

In simple terms, it means that only the people with permission to access the data can do so and that it's much less likely to be lost or changed without permission.

The security guarantees of such a mechanism are as strong and even more robust than proof-of-work without any penalty associated with searching for nonces to create hashes containing the correct number of leading zeros, a process used in proof-of-work (PoW) systems.

Nonces are random numbers used in cryptographic hash functions to generate a cryptographic hash. When miners search for nonces, they try to find a combination of the data and the nonce that results in a hash with the correct number of leading zeros. This process is computationally expensive; it is how the miners prove that they have put the necessary work into creating a valid block.

The power requirements for running a Vogon are the same as running any standard enterprise-grade software. In Hitchhiker's Guide to the Galaxy, towels are the most useful items in the known universe.

Vogon | Towels vs. dApp Smart Contracts

A conventional dApp smart contract written in restrictive languages, such as Solidity, runs at some unknown time after a transaction is submitted. Solidity is a restrictive language because it has limited capabilities compared to other programming languages. It is designed to be used for creating smart contracts on the Ethereum blockchain and, as such, has a limited set of features suited to this purpose. It does not support features such as loops, inheritance, and complex data structures, which makes it more challenging to develop complex applications.

1. **Vyper:** (Cadence works with Ethereum, Hyperledger Fabric, Quorum, and R3 Corda.)
    - Vyper is a restrictive programming language designed to be as simple and secure as possible.
    - It has fewer features and capabilities than more full-featured languages like Solidity and is limited to a small set of data types and operations.
    - This limitation helps ensure that contracts are easy to read, audit, and understand, which helps reduce the potential for errors and malicious behavior.
2. **Simplicity:** (Simplicity works with various blockchains, including Ethereum, Tezos, and Polkadot.)
    - Simplicity is a restrictive programming language because it has minimal features and capabilities, making it difficult to develop complex and robust applications. Additionally, Simplicity does not allow for third-party libraries or frameworks, making it even more restrictive.
    - Simplicity is a highly secure language, so it only supports a subset of the features in more traditional programming languages. However, this limits the complexity of programs that can be written and reduces the overall flexibility of the language.
3. **Bamboo:** (Bamboo works with Ethereum and its public blockchain. It works with other public and private blockchains, such as Hyperledger, Quorum, and R3 Corda. Additionally, Bamboo is compatible with a range of blockchain ecosystems, allowing it to interact with smart contracts, digital assets, and more.)
    - Bamboo is a restrictive programming language because it is a language for teaching introductory programming concepts.
    - It has a limited set of features and does not include many of the more advanced features of more advanced programming languages. This restriction helps to keep the language simple and focused on teaching basic programming concepts.
4. **Lity:** (Lity works with the Ethereum blockchain.)
    - Lity is a restrictive programming language designed with simplicity and security in mind. It is statically typed and strongly typed, which means that the program must declare all variables must with a type, and a function may only call for that type of data.
    - It also prevents unwanted behavior by enforcing strict rules around the usage of variables and functions and ensuring that all code is well-structured and consistent. These restrictions help to prevent security vulnerabilities and maintain a high level of reliability.

5. **Manticore:** (Manticore is currently compatible with the Ethereum blockchain. It supports both Ethereum and Ethereum Classic.)
    - Manticore is a restrictive programming language because it uses strict rules designed to enforce a specific programming style. It requires that code be written in a specific way and does not allow for any deviation from its defined standards. This design means that code written in Manticore is often challenging to understand and modify, making it difficult for new developers to pick up. Additionally, the language does not support many features found in other languages, such as object-oriented programming.

6. **Rholang:** (Rholang is a smart-contracting language that works on the RChain blockchain, a decentralized platform based on the Byzantine Fault Tolerance consensus protocol.)
    - Rholang is a restrictive programming language because it has limited capabilities compared to other languages. It specifically provides limited access to resources, making it difficult to write complex code. Additionally, its syntax is not as flexible as other languages, meaning it takes longer to write code in Rholang than in languages like Python or JavaScript.
7. **DAML:** (DAML works with various blockchain solutions, including Hyperledger Sawtooth, Hyperledger Fabric, Corda R3, Ethereum, Quorum, and DAML Ledger.)
    - DAML is a restrictive programming language because it provides a comprehensive, secure, and reliable system for creating and executing distributed applications. As such, it has a limited set of features and capabilities, allowing developers to focus on the core elements of their application while minimizing the risk of introducing vulnerabilities. This aspect also helps ensure that the applications built using DAML are secure and reliable, as the language minimizes potential errors and bugs.
8. **Flux:** (Flux works with the Ethereum blockchain.)
    - Flux is a restrictive programming language because it restricts the use of certain features commonly found in other languages. For example, it does not allow for functions, loops, or recursion. Additionally, it does not support data structures such as arrays and objects and does not allow for the use of variables. All of these restrictions make it challenging to create complex programs and can limit the potential of the language.
9. **Ink:** (Ink works with the Ethereum blockchain.)
    - Ink is a restrictive programming language because it does not allow for complex programming structures such as loops or functions. It creates interactive narrative experiences, focusing on story-telling rather than complex programming logic. As a result, it is not suitable for creating complex applications or games.

10. **Cadence:** (Cadence works with Ethereum and Tezos blockchains.)
    - Cadence is a restrictive programming language because it only supports a limited set of features. This constraint means that it is not as versatile as other programming languages and limits the functionality of the programs written with it. This barrier limits the scope of tasks that can be completed with Cadence, making it challenging for more complex projects.

To determine the effect of running any smart contract written in a restrictive language like Solidity, you must check if results wrote out to the blockchain, which is a cumbersome problem that comes with risks[3].

By comparison, Vogon transactional objects immediately run when asked to through web service APIs. They have rich semantics and can receive data from JSON objects for processing.

The benefit of web service APIs with rich semantics is that they enable developers to create more powerful and flexible applications. By allowing developers to use any data that can carry through JSON objects, developers can create applications customized to their specific needs and not limited to a specific format. This extensibility helps to create a more decentralized cloud, as the restrictions of a specific data format no longer limit developers. Additionally, using rich semantics allows developers to create more easily maintainable applications, as they can push updates without having to rewrite the application completely.

This technique means that the API can interpret and understand data presented in JSON objects, even if the data is complex. It can then use this data to perform tasks the user has asked it to do. In simple terms, the API can understand and process any data sent in an easy-to-understand format like JSON.

The benefit of curating data after its addition to the distributed data store on a decentralized cloud is that it allows users to control who has access to the data and how third parties use it. This approach is critical in applications, for example, in healthcare and finance, where data privacy and security are of utmost importance.

With data curation, users can ensure that their data is only used for their intended purposes and can revoke access to the data if necessary. Additionally, data curation allows for faster data retrieval. Users can quickly find the needed data without having to search through a large amount of data and manipulate relational database tables readable by sister APIs during API servicing.

Vogon Does Not Have the DAPP Update Vulnerability

The dApp update vulnerability is a security flaw that can allow malicious actors to modify the code of a decentralized application (dApp) without the users' knowledge. This weakness can lead to users executing malicious code or allowing unauthorized access to their personal information. The update vulnerability can significantly impact user experience, as users may be unaware that their data is accessed or manipulated. Additionally, it can lead to financial losses due to the theft of funds or other assets.

Blockchain dApps (and other related technologies) are inherently vulnerable. Because dApps were essentially grafted onto the bitcoin proof-of-work blockchain to form Ethereum, there was no built-in mechanism to mediate code updates.

As a result, dApp authors must forward smart contracts to future updates if present. This requirement allows the authors to fix bugs and update smart contracts. However, it also provides anyone with the developer key to completely alter the smart contracts and economics associated with those dApps, making it neither decentralized nor secure.

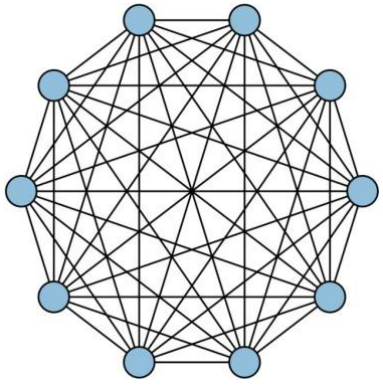
Vogon uses threshold signature enforcement on installing and updating microservices, and that enforcement can include third-party auditing bodies to ensure no malfeasance.

Using threshold signature enforcement when installing and updating microservices on a decentralized cloud can help secure dApps more efficiently. It provides a reliable way of verifying and authenticating the accuracy of the microservices and their updates. This process requires multiple parties to sign off on the microservices and their updates before they can be installed or deployed. It ensures that only verified and trusted microservices and updates get used within the decentralized cloud. This method helps reduce the risk of malicious attacks and ensures that the microservices are up-to-date and functioning correctly.

Finally, Vogon's distributed document store mechanism can store the code for a dApp and its updates. The protocol stores the code on multiple nodes in a network of computers using a distributed document store, making it more difficult for an attacker to target and manipulate the code. Additionally, the code can be versioned and tracked, so if an attacker can access the code, an administrator can restore the original version of the code. This lineage system would help to minimize the impact of the vulnerability.

Vogon | Deterministic Concurrence

Deterministic concurrency is a way of making sure that all the computers in a decentralized cloud architecture work together in harmony. It's like having a team working together on a project, but each person has their part and must do it in a specific order. This way everyone's work is done consistently without hiccups or problems. It's desirable because it ensures that all the computers in a decentralized cloud architecture always work together in the same way, making the whole system more efficient and reliable.



The foundational unit of consensus in the Vogon protocol is a consensus group. A consensus group is a group of nodes in a decentralized network that agree on the validity of data and transactions.

This agreement is called consensus, which allows for deterministic concurrence to create a more efficient and secure decentralized cloud. Having nodes reach a consensus on data validity eliminates the need for a centralized authority or intermediary to verify the data, making it much more efficient.

This design creates a fully interconnected high-performance mesh network containing hundreds of individual Vogon.

Consensus is a deterministic process where a consensus group collectively stitches together the cubes on the decentralized cloud. It is optimally efficient,

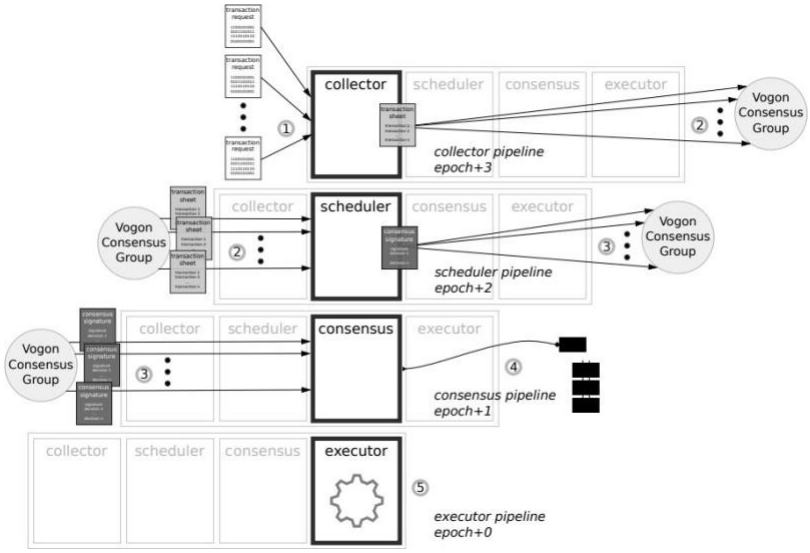
with each Vogon contributingtransactional material to every other Vogon such that the cumulative network load for each Vogon is one cube or "block" in and one cube or "block" out.

A cumulative network load is the total amount of traffic passing through a decentralized cloud network. It includes both inbound and outbound traffic over a given period and is essential to optimize to ensure that the decentralized cloud network remains responsive, secure, and reliable.

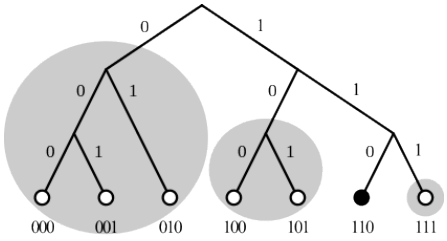
Optimizing the cumulative network load can help to reduce latency, improve performance, and reduce the risk of network overload or outages. BLS 12-381 threshold signatures act to seal and determine consensus with full byzantine fault tolerance. BLS 12-381 threshold signatures are a type of digital signature scheme that uses a threshold of participants to generate a single signature.

This type of signature scheme is vital for deterministic concurrence because it allows multiple parties to securely sign off on a single transaction without requiring all participants to be present or online simultaneously. By using a threshold signature scheme, all participants can securely sign off on a single transaction without worrying that an individual participant will compromise the security of the transaction. This method makes it much easier to achieve deterministic concurrence

The diagram below illustrates the stages of consensus and block forming.



Vogon consensus is superscalar, with the ability to simultaneously keep four new blocks intended for adding to the blockchain in flight. Superscalar consensus technology is a consensus algorithm designed to achieve high throughput and scale. It can process more transactions in a shorter period than traditional consensus algorithms while still ensuring the security and accuracy of the ledger. Distributed ledger technology, like blockchain applications, uses a superscalar consensus algorithm. This high-performance design processes thousands of transactions per second.



Vogon constantly considers the trade-off between communication cost and the system's fault tolerance. Increasing the size of the consensus group can increase the system's fault tolerance, but it also increases the cost of communication. Therefore, the optimal size of the consensus group is determined based on balancing optimal performance and the desired level of fault tolerance.

As the number of Vogons grows past the optimal size for a consensus group, the consensus group splits into two

consensus groups, each taking responsibility for half of the key space in the blockchain. The Vogons with keys starting with a binary 0 go to one group, and those starting with a binary 1 go to the other group.

Vogon essentially becomes a block graph. A block graph is a graph data structure used to store and manage data in a decentralized cloud. Block graphs are necessary for decentralization because they allow for distributed data storage and management, which means that the data is not owned or controlled by a single entity, such as a company or government.

This protocol governance ensures that data is secure and that anyone with permission can access it. A block graph is a data structure that stores data in blocks linked together using cryptographic hashes. It is an alternative to traditional blockchain technology.

Block graphs are more efficient than block directed acyclic graphs (DAGs) because they offer increased scalability, faster and more secure transactions, and the ability to store more data in the same space.

Additionally, they provide better privacy and security features than block DAGs.

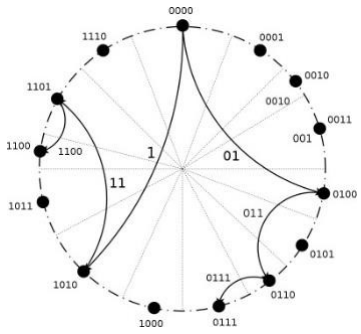
Block graph technology also allows for faster data processing and consensus on data, making it an ideal platform for decentralized applications. This mechanism helps to identify the new groups with the leading bits generated by the split. As this process continues, one group becomes two. It is identifiable with one leading bit; two groups become four and are identifiable by two leading bits. Four groups become eight, identifiable by three leading bits, and so on. The diagram below visualizes this routing.

These leading bits that identify the consensus groups act to segment or "shard" the keyspace. Addresses automatically route to the consensus group identified with the same leading binary digits as the address. This process happens for cryptocurrency wallet addresses and developer scopes that identify where microservices install.

Invocation of microservices and other operations route through this mechanism to the correct consensus group. This type of routing is very similar to Kademlia routing, a technology that allows computers in a decentralized network to communicate and share data. It is preferred for a decentralized cloud because it is much more secure than a traditional centralized cloud. With a decentralized cloud, no single point of failure exists, meaning that if one computer in the network fails, other computers can still access the data. This failsafe makes it more reliable and secure than a centralized cloud. Additionally, it is faster and more efficient than traditional cloud solutions.

Kademlia uses a distributed hash table (DHT) to store and retrieve data in a peer-to-peer network. The DHT is a distributed data structure that maps data keys to their associated values. In Kademlia, the DHT stores the node's public key and associated contact information. When a node wants to find a particular node, it broadcasts a search request with the node's public key. Other nodes in the network search their local DHTs for the node's public key and contact information. Once the protocol discovers the node's contact information, the requesting node can then contact the requested node directly.

This process is like P2P protocols such as BitTorrent.



Here in this diagram, you can see the consensus groups visualized as residing around a circle, sorted by their leading binarydigits. We can quickly locate any consensus group through a Kademlia "similar" routing protocol.

It is not the same, as Vogon consensus groups contain up to hundreds of members, and each is fully interconnected, allowing the leaf nodes to act as a group, exhibiting a much more intelligent and faster routing fabric than pure Kademlia.

Some transactions must cross address boundaries between multiple consensus groups, such as a transaction to exchange one cryptographic asset for another, where their addresses have sharded them into different consensus groups. When this happens, Vogon performs "meta consensus," a two-stage "atomic" operation that merges consensus decisions from two or more consensus groups. All concerned consensus groups simultaneously initiate the atomic transaction, and the results get shared between them. The first step authorizes the transaction on the blockchain of all concerned consensus groups and that authorization commits when each concerned consensus group mathematically combines the results.

Vogon Design and Engineering

Daniel Guinan, an accomplished computer scientist, designed Vogon. He worked at Visa on the first cryptographic protocol implementations for securing credit cards on the internet and at Javasoft/Sun Microsystems, running the Java Commerce team. There Daniel worked on early cryptocurrencies and technologies like Java Card. He sold his company, Verisign, which focused on XML Web Services and cryptography. Later, he started TRUSTe/TRUST Arc and built one of the world's most significant ad techs to mediate consumer privacy options in advertising, AdChoices.

Vogon was designed from the ground up with the discipline and experience that comes from such endeavors; it was designed to work at an ad scale and potentially supplant or replace traditional PKI such as digital certificates. The vision is even more significant. The vision is to create a safer internet where the playing field is level, and massive internet companies cannot monopolize the lives of everyone through tyrannical practices such as censorship and de-platforming.

20 Issues Surrounding the Decentralized Cloud

There are twenty issues associated with centralized cloud technologies, ten are technical and process-oriented, and the remainder are people-oriented:

Technical Issues:

1. Excessive dependence on a single provider.
2. Security and privacy concerns.
3. Lack of control over infrastructure and data.
4. Limited scalability options.
5. High cost of ownership.
6. Vulnerability to outages and disasters.
7. The difficulty of migrating to other providers.
8. The difficulty of integrating with existing systems.
9. The complexity of managing multiple cloud services.
10. The difficulty of auditing and compliance.

Cultural Issues:

1. Companies that own centralized cloud technology can access and store data about people without their permission.
2. Companies that own centralized cloud technology can track user activities online.
3. Companies with centralized cloud technology can censor information, preventing people from accessing certain content.
4. Companies that own centralized cloud technology can use it to manipulate public opinion by spreading false information.
5. Companies that own centralized cloud technology can use it to target ads to people, which can be intrusive and annoying.
6. Companies that own centralized cloud technology can use data collected to target vulnerable or marginalized groups of people.
7. Companies that own centralized cloud technology can use it to spread hate speech and other malicious messages.
8. Companies that own centralized cloud technology can use algorithms to make decisions that could be biased or unfair.
9. Companies with centralized cloud technology can create digital monopolies, where one company controls most of the market.
10. Companies with centralized cloud technology can influence political decisions, leading to a lack of transparency and accountability.