Cloud ICO White Paper

DISCLAIMER - IMPORTANT NOTICE: *Please read the following notice carefully before proceeding to read this White Paper document prepared by the Company (the "**White Paper**"). This notice applies to all persons who read this document. Please note this notice may be altered or updated. The White Paper has been prepared solely in respect of the Company's crowdsale of the Cloud tokens ("**ICO**"). No shares or other securities of the Company are being offered for subscription or sale in any jurisdiction pursuant to the White Paper. The White Paper is being made publicly available for information purposes only and does not require any action to be taken by the general public or shareholders of the Company. The White Paper does not constitute an offer or invitation to any person to subscribe for or purchase shares, rights or any other securities in the Company. The shares of the Company are not being presently offered to be, registered under the Securities Act of any country, or under any securities laws of any state. Company reserves the right to price the Tokens at its sole discretion and sell them at any discount or premium in relation to the Tokens ICO prices or any market price. The Company reserves the right to sell its reserve Tokens to any eligible buyers, including but not limited to, buyers of Company's on-boarding cloud services, cryptocurrency buyers, Company's shareholders and any affiliates of the above. The Company reserves the right to, at its sole discretion, create and/or activate the maximum amount of Tokens (specified in the White paper), including the reserve Tokens, and make them commercially available to the general public either directly or any via eligible agents, including cryptocurrency exchanges and trading facilitators worldwide. The Company states that all Token market capitalization and inventory calculations and valuations will be performed independently by 3rd parties and it shall not assume any responsibility whatsoever for such calculations and valuations.THE TOKENS REFERRED TO IN THIS WHITE PAPER HAVE NOT BEEN REGISTERED, APPROVED, OR DISAPPROVED BY THE US SECURITIES AND EXCHANGE COMMISSION, ANY STATE SECURITIES COMMISSION IN THE UNITED STATES OR ANY OTHER REGULATORY AUTHORITY NOR HAVE ANY OF THE FOREGOING AUTHORITIES EXAMINED OR APPROVED THE CHARACTERISTICS OR THE ECONOMIC REALITIES OF THIS TOKEN SALE OR THE ACCURACY OR THE ADEQUACY OF THE INFORMATION CONTAINED IN THIS WHITE PAPER UNDER, THE U.S. SECURITIES ACT OF 1933 AS AMENDED, OR UNDER THE SECURITIES LAWS OF ANY STATE OF THE UNITED STATES OF AMERICA OR ANY OTHER JURISDICTION. PURCHASERS OF THE TOKENS REFERRED TO IN THIS WHITE PAPER SHOULD BE AWARE THAT THEY BEAR ANY RISKS INVOLVED IN THE PURCHASE OF TOKENS, IF ANY, FOR AN INDEFINITE PERIOD OF TIME.*

FORWARD-LOOKING STATEMENTS: *Some of the statements in the White Paper include forward-looking statements that reflect the Company's and/or the Management current views with respect to product development, execution roadmap, financial performance, business strategy and future plans, both with respect to the Company and the sectors and industries in which the Company operates. Statements that include the words "expects", "intends", "plans", "believes", "projects", "anticipates", "will", "targets", "aims", "may", "would", "could", "continue" and similar statements are of a future or forward-looking nature. All forward-looking statements address matters that involve risks and uncertainties. Accordingly, there are or will be important factors that could cause the Group's actual results to differ materially from those indicated in these statements. These factors include but are not limited to those described in the part of the White Paper entitled "**Risk Factors**", which should be read in conjunction with the other cautionary statements that are included in the White Paper. Any forward-looking statements in the White Paper reflect the Group's current views with respect to future events and are subject to these and other risks, uncertainties and assumptions relating to the Group's operations, results of operations and growth strategy. These forward-looking statements speak only as of the date of the White Paper. Subject to industry-acceptable disclosure and transparency rules and common practices, the Company undertakes no obligation publicly to update or review any forward-looking statement, whether as a result of new information, future developments or otherwise. All subsequent written and oral forward-looking statements attributable to the Company or individuals acting on behalf of the Company are expressly qualified in their entirety by this paragraph. Prospective buyers of the Cloud token should specifically consider the factors identified in the White Paper that could cause actual results to differ before making a purchase decision. No statement in the White Paper is intended as a profit forecast and no statement in the White Paper should be interpreted to mean that the earnings of the Company for the current or future years would be as may be implied in this White Paper. By purchasing the Cloud token I hereby acknowledge that I have read and understand the notices and disclaimers set out above.*

ICO Passport

Token name	*Cloud (the "Token")*
Token ticker	*CLD*
Token symbol	
Token owner	*Cloudwith.me, 39 Northumberland Road, Ballsbridge, Dublin 4, D04 H1F3, Ireland*
Financial Auditors	*Baker Tilly and Hughes Blake, Joyce House, 22/23 Holles Street, Dublin 2*
Legal Advisors	*Sean Wallace and Alan Ryan, Wallace Corporate Counsel, 39 Northumberland Rd, Ballsbridge, Dublin 4*
Token type	*Ethereum ERC20*
Total Tokens issued	*60,000,000 Tokens Max. Final number of tokens created will be calculated according to contributions demand, it will be fixed and publicized by the end of the ICO day.*
Mining	*No mining or any other means of Tokens amount increase will apply to the Cloud Token post ICO*
Use of proceeds	*Funds from Token sale will be predominately used for the global Deployment of GridNodes infrastructure for migrating the web into a decentralized cloud. See "Use of Proceeds" section in this document.*
Bonus	*At the pre-order period Cloud Tokens will be available with a bonus structure as publicized at the official Token ICO site (token.cloudwith.me)*
Tokens distribution	*50% Public (of all tokens created)* *46% Company's reserve* *4% Management & Employees ("Employee Tokens")*
Lockup period	*Management and employees undertake a 12-month lockup period. No sale, transfer or pledge of Employee Tokens will be permitted*
Tokens transfer	*Tokens will be transferred to buyers upon payment confirmation. Purchased tokens will not be active during the pre-order period (i.e. sale or transfer of tokens will not be possible until the ICO is complete at the end of ICO Closing Date. Tokens will become activated automatically upon ICO completion date and can be freely transferred or exchanged)*
ICO timeline	*Pre-Order Period* — *July 25th - August 24th* *Token Generation & Redeem by Pre-Order Buyers* — *August 25th - Aug 28th* *Discounted Cloud Services payable by Cloud Tokens* — *August 28th* *Official Sale of Tokens by Company Begins* — *August 29th* *First day of Tokens Trading* — *September 21st*

Table of Contents

Executive Summary

Cloudwith.me (the "Company") has three main objectives:

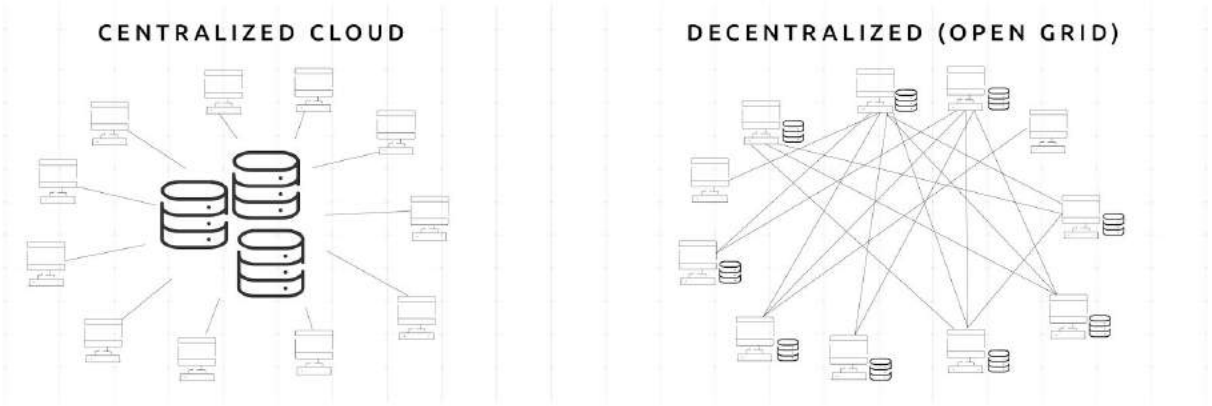
1) **To decentralize** the provisioning of cloud services, migrating from a global infrastructure operated by a small group of hyperscale service providers to a blockchain-based peer-to-peer infrastructure operated by millions of small individual providers and contributors of cloud resources.

2) **To simplify** the use of professional cloud services for all levels of web users turning such services into a commodity product that is universally accessible and utilized by the general public.

3) **To monetize** the new decentralized cloud services ecosystem with a dedicated cryptocurrency governed by a smart contract enabling automated and trusted reconciliation of payments between all peer-to-peer cloud services providers.

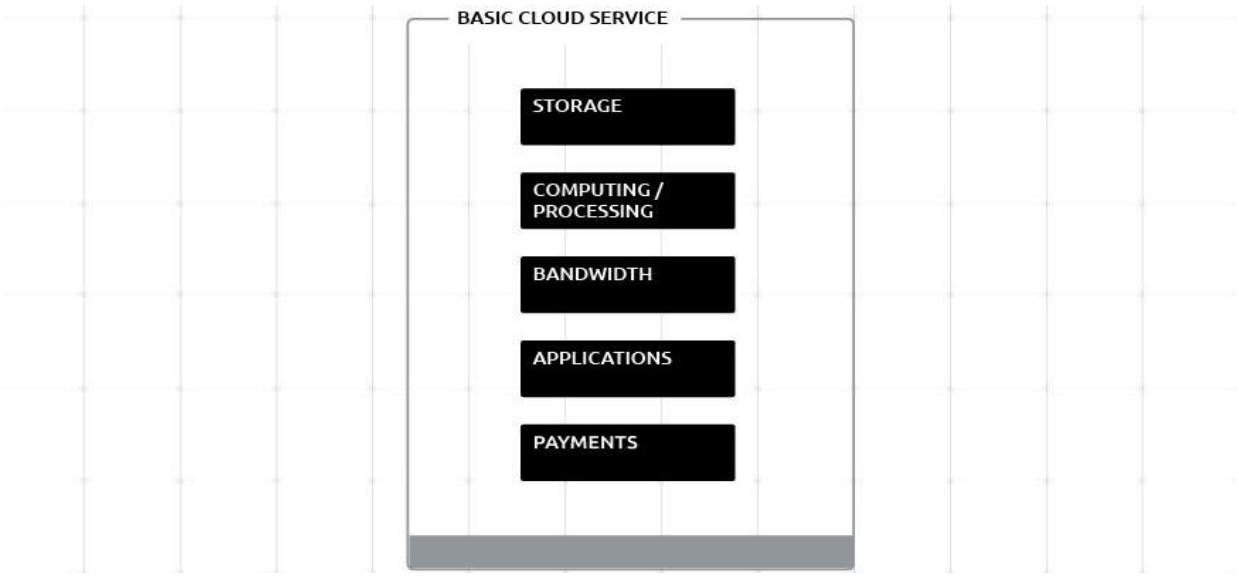
The Company has already achieved the first milestone by establishing its simplified provisioning and setup of cloud services that can be easily operated by even the most novice web users.

As a second phase, Cloudwith.me intends to facilitate the creation of a globally decentralized cloud hosting grid, allowing cloud-based applications to automatically scale across millions of cloud service providers who are in fact individuals contributing small portions of their redundant home/office computer resources to become a part of the blockchain cloud grid (thus effectively creating the new Crowd Cloud) and benefiting from continuous Provider income.



The blockchain Ethereum-based Cloud token is intended to become the standard currency for the emergent decentralized cloud services ecosystem, governing contribution-based pro-rata payments, clearing and distribution of financial benefits to the new role players of the Crowd Cloud.

On a naive schematic level, each cloud service is comprised of the following building blocks: (1) Storage (2) Computing/Processing (3) Bandwidth (4) Cloud Applications (5) Payments and billing



High-level schematic of the building blocks of an average cloud service

By encapsulating the basic building blocks of the cloud service (namely a "GridNode") and creating a smart contract that considers the key aspects of the relationship between a cloud-service provider and application providers, the Company intends to make it possible for virtually anyone to join the hosting grid as a service provider/contributor, thereby establishing a virtual infrastructure that is operated by millions of cloud-service providers across the globe, and making it possible for each cloud-based application to elastically scale to nodes that meet its own technical, geographic, regulatory and business requirements.

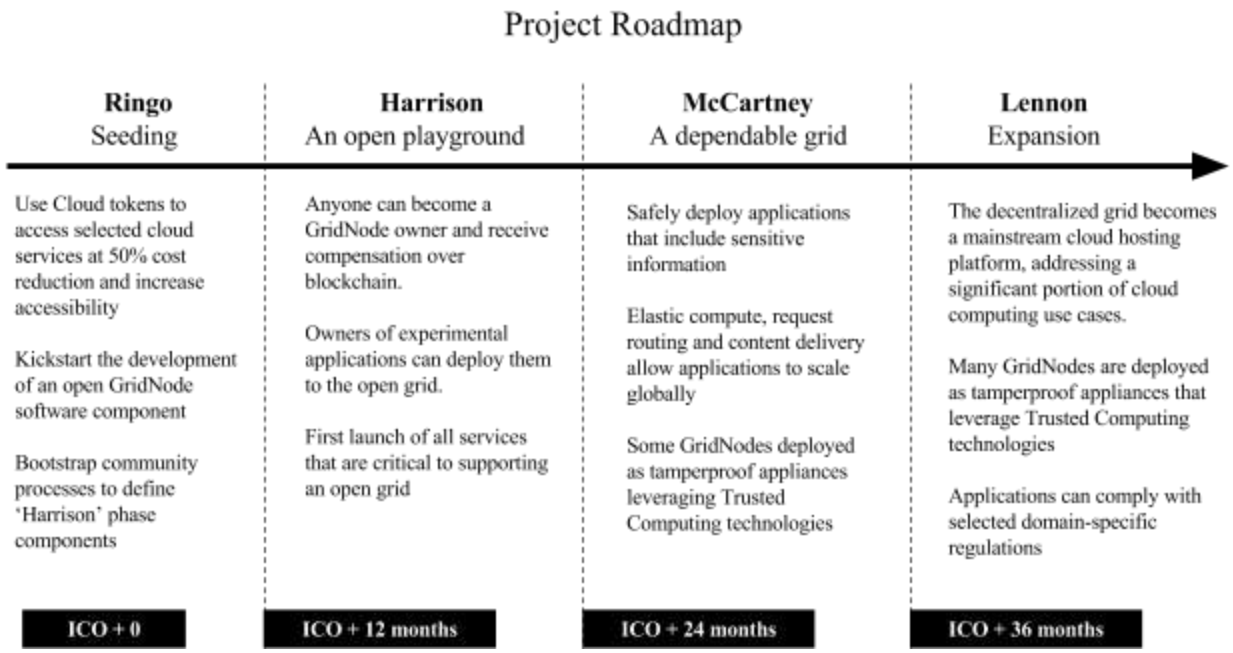
The newly issued Cloud cryptocoin will be used to automatically distribute payments to the GridNode owners and other Crowd Cloud participating parties as well as to provide important Grid Admin facilitation such as entity description, performance metrics, etc.

The Company intends to use the proceeds from the ICO to deploy a critical mass of GridNodes on a global scale and actively promote the recruitment of GridNode Service Providers/Contributors thus effectively decentralizing cloud services.

The key benefits expected from achieving cloud decentralization are:

- Reduction of cloud services costs (expected circa 94% cost reduction on average service criteria)
- Unparalleled data security and privacy with tamper-proof GridNode network components
- Near fail-safe operation with significantly higher integrity and service availability
- High degree of cloud services automation enabling to deal with the "What" instead of the "How"
- Taking pro-cloud services out of the IT expertise zone making it fully accessible for the "rest of us" by utilizing the Company's blockchain DAP technology

The DAP, or Decentralized Application Protocol, is a blockchain application threading layer. Based on the Ethereum infrastructure, the DAP will facilitate easy design and launch decentralized peer-to-peer cloud applications such as media services (music & film), social peer-to-peer insurance, decentralized banking, financial services and other applications without limitations or central governance—all monetised by a single uniform currency—the Cloud token. The protocol will allow seamless setup of self-governing smart contracts for an unlimited range of social applications.



The Cloud token is a key component governing the financial ecosystem surrounding the new Crowd Cloud. The Company believes Cloud Token is expected to gain significant liquidity due to its immediate implementation in the Company's services.

Centralized Cloud Services Overview

The 21st century has seen a massive surge in data and cloud applications economy. The demand for both storing and processing data is on an exponential rise [1]. The cost of cloud services for small and medium-sized businesses (SMBs & SOHO) become significant as the need to develop and deploy cloud applications becomes an essential commodity.

Software applications that offer remote storage have been made accessible to compensate for the limitations of on-premises computer storage. However, the real players in this industry—the 'hyperscale' cloud services providers—go well beyond the scope of mere storage. Centralized cloud service providers offer a wide array of services including storage, backup, web hosting, remote computing power, content delivery, database management and remote software applications.

Over 90% of businesses utilize cloud services in some form, benefiting from a range of cloud applications management and analytics without the upfront investments necessary on infrastructure, specialty staffing, server equipment and maintenance. The rise of cloud services has been one of the strongest influencers on a five-year low of capital expenditure in IT budgets [2]. While capital expenditures of businesses are being significantly reduced, Intel Security predicts that by mid-2018, operational expenditure for cloud services will account for 80% of IT budgets [3] and IDC forecasts global public cloud services to reach over $195 billion in revenues by 2020, more than doubling last year's figure [4].

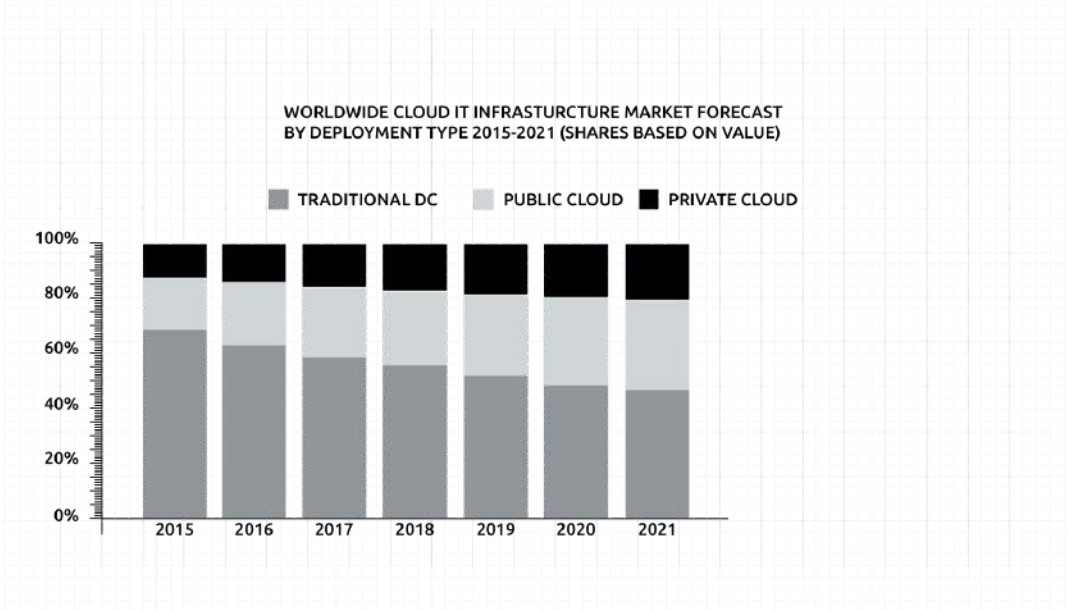


Fig: Adapted from *Spending on IT Infrastructure for Public Cloud Deployments Will Return to Double-Digit Growth in 2017, According to IDC [Press Release]*, *International Data Corporation, April 2017. Available at: http://www.idc.com/getdoc.jsp?containerId=prUS42454117*

For smaller businesses scalability is key. Investment in virtual servers via cloud services endows clients with scalable server capacity, so that a high number of server instances can be utilized during peak demand, and a lower amount during quiet periods. This way, wasted server capacity doesn't drain company resources and server time can be costed on a pay-per-use schema.

Centralized cloud services rely on a number of large data centers to serve their customer base. These data centers are physically segregated to avoid total service failure wherever possible. While providers offer services that enable customers to run instances on multiple server systems to balance traffic, these ultimately come at an additional cost as insurance in case of system failure in one data center. The reliance on physically connected local server systems is precisely what's meant when it is said that their services are 'centralized'. Users of centralized cloud services must place significant amount of trust in their providers to be secure, accessible and reliable.

The Synergy Research Group reported in 2017 that the top 24 hyperscale companies currently operate over 300 data centers around the world. They also report that hyperscale providers account for 68% of the global market share for cloud infrastructure services and 59% of cloud-hosted software services [5]. For these hyperscale providers, cloud services are accounting for an increasing percentage of their revenue. Cisco reported in 2015 that hyperscale data center traffic accounted for 34% of total Internet traffic, a figure expected to rise to 53% by 2020 [6].

It's no surprise that the only competition in the cloud service industry is between well-established corporations as the cost of setting up a viable centralized cloud service is simply too much for newcomers.

Many individuals and small businesses struggle with the necessary steps to understand and maintain the cloud service infrastructure. For businesses in particular, setting up cloud servers can take months of work by trained professionals. Some managed hosting solutions like Cloudwith.me offer automated solutions to cloud service setup and management removing barriers to cloud services setup and allowing such businesses to remain focused on their core operations.

A 2014 study by IDC found that 65% of SMBs worldwide made use of cloud storage, with the main disincentive being privacy and security concerns [7]. Security concerns have been a highly discussed issue for potential users of cloud services (especially cloud storage) since their popularization [8]. Rightfully so, it would seem. Nearly 60% of SMBs go out of business within six months of a hack [9].

¯What is Blockchain?

In 2008 Bitcoin presented an elegant solution to the problems posed by centralized administration in the finance sector: distributed databases [21]. Using a peer-to-peer architecture, this *blockchain* technology allows users to send transactions without a central authority to accommodate them. In this paradigm, every user of the service holds a copy of all the transactions sent through the network. This eliminates any need to trust a third party to maintain services or verify transactions. A transaction on a blockchain service takes place immensely fast and securely.

A blockchain-based system makes use of a blockchain to link a 'block' of transactions to the set of all past transactions in the system. Every user—or *node*—in the network is given a full copy of the blockchain. This enforces transparency between users, as each new block on the chain is synchronized among all nodes. Each transaction must be verified by a number of users. Verifications are usually performed by nodes randomly guessing inputs to a cryptographic hash algorithm until the result matches the hash ID of the last block in the chain.

Despite the transparency of transactions in a blockchain, users are awarded privacy because their identity to the system is simply a public key (a long number hash). A corresponding private key is used to sign off transactions. Users must keep the private key private; it is the only means for the system to prove authenticity. No confidential information on users is held by the blockchain except those details they decide to send out (which is not recommended).

Users who opt to try to verify transactions make automated verifications. To add incentive to do so, those that succeed are given a monetary reward. Therefore blockchain technologies allow transactions to be executed incredibly quickly. The fees for sending transactions through the distributed network are driven down compared with those in centralized services due to the ease of processing transactions and the absence of a third party to oversee the validity of a contract. If a transaction breaks the terms of agreement, the contract isn't valid and it cannot be passed down the chain.

Since the advent of Bitcoin, many other cryptocurrencies have been conceived with the same basic framework. The most notable of these is Ethereum, which generalizes the idea of a blockchain currency to a contract token that could represent any hard-coded agreement between parties. These 'smart contracts' have been utilized by platforms for crowdfunding, prediction markets (Gnosis), music licensing and distribution (Ujo), social media (AKASHA), and even cloud services (Storj, Golem), all at a discount to similar, centralized services. They are flexible enough for developers to add a wide range of conditions and functions based on which conditions are met in deployment. Yet in deployment, this flexibility vanishes, as their properties become strict terms that must be met by users to decide whether the contract is valid and precisely what it will be used for.

Bitcoin and Ethereum are examples of public blockchains. Anyone can access a public blockchain to view, send, or validate transactions. Private blockchain platforms, on the other hand, require explicit approval from an administrator to gain access to these actions [22]. This equips private blockchains with more functionality for localized use, e.g. database management within a business [23]. Part of the power of the Ethereum platform is that it allows developers to create private blockchains that still retain the increased security, privacy, and distributed self-governance that the public platform permits.

Complementary Cloud Decentralization Solutions

There are a few cases of decentralized cloud initiatives entering the market right now, most notably Storj [31] and Golem [32]. These stand to prove the strength of peer-to-peer networking to delegate utilities. Both projects are equipped with Ethereum-based smart contracts to commodify the services provided from one user to another at the lowest possible cost.

Storj

Storj concentrates its efforts on cloud storage. In essence, the service allows users to rent out unused storage space on their devices and rent such space from others. With this service, users can encrypt and fragment their files and then distribute them among its network, thus protecting their content from access by storage providers. Copies of the file fragments can also be distributed among the network to guarantee the availability of data in case of a failure on one of the hosting servers. By dealing with file fragments instead of whole files as in centralized cloud storage, the need for trust between users is minimized.

Cloud storage in this manner dissuades targeted attacks on file stores by anonymizing a user's data and distributing them across the network. In contrast, attackers will find it relatively simpler to locate the data centers that specific businesses employ in a centralized cloud service. Of course, whole-network attacks still prove to be an issue for distributed cloud service providers. Increasing the scale of the network is an effective defense against most types of attack.

Cloud storage can protect against data loss in ways other than just sending redundant copies of file fragments across the network. Erasure coding algorithms allow data owners to recover a file even if a few fragments are corrupted or lost. This way, if a storage provider terminates its node, it becomes much easier to recover the file from the remaining fragments. Keeping track of the number of remaining file fragments is necessary to guarantee that erasure coding recovery is successful; to do so across all files stored can prove difficult for a user who doesn't have the time or technological knowledge required to understand this.

In Storj, smart contracts decided upon by a user contain all the information describing the relationship between a data owner and the storage provider. Once accepted by a potential partner, both parties need only sign the contract to access the service immediately. This contract method drives the cost of renting storage space by putting users in a competitive marketplace. Users can alternatively utilize services provided by Storj themselves. Their own storage service is offered at a discount from the major centralized services, ranging between 35% and 50% of the cost of hyperscale providers [31]. Services are paid in Storj's token cryptocurrency, so costs for the service in their currency fluctuate based on the value of the token. This protects users from volatility of their costs or profits in the market.

Data owners can specify their bandwidth demands in the smart contract, which keeps the Storj algorithms aware of whole-network demands and allows bandwidth to be evenly distributed. Slow download speeds are less likely in such a network, especially with the addition of redundant file distribution and erasure coding.

For users who would struggle with the administration that comes along with the service, Storj provides the option to use a dedicated 'Bridge' server to undertake negotiation, issuing, and verification of contracts; payments for services; and recording the state of the file fragments across the network. Of course these servers reintroduce the issue of placing trust in a third party to guarantee file protection. The Bridge servers allow developers to utilize the Storj network to host and run their applications. They also allow public file sharing by hosting encryption keys for public files.

A disadvantage of this distributed service is the lengths that users must go to in order to maintain availability of data on the network. By eliminating the need for trust in a central authority with Bridge servers, users still must manage the file states and preprocess data themselves. For businesses that need to store a lot of data this means time, knowledge, and money.

Golem

Golem allows users to rent out CPU on their devices or pay to use others' hardware for computationally exhaustive tasks. The project's ultimate goal is to become the world's first decentralized supercomputer. As in any cloud service, the distributed network offers parallel processing that allows time-consuming tasks to be completed quickly when users need it. This means that in the lull periods that Golem's service affords its users, users can rent out their own idle CPU to the network and recoup a great deal of the service's cost.

Golem also allows software developers to place their own applications on its application registry. This gives developers direct payment for their services on a use-by-use basis. An issue with their open application registry is that malicious software can be uploaded and utilized by the network. The registry offers a whitelist and blacklist feature to give users the option to run only validated applications and be aware of the malicious ones.

As with Storj, Golem users can set up their own contracts on the blockchain to form an agreement on the terms of the service. Users can therefore negotiate their own payment schemes for CPU or application rental in the smart contracts that accommodate fluctuations in Golem's currency. Payment for the service is automatically executed by the contracts on the blockchain that users have agreed to use.

With a distributed network, large-scale outages such as centralized cloud service users have experienced in the past are no longer an issue. If one server goes down in the middle of task completion, resources can be pooled from elsewhere in the network to accommodate for it. In the event of a server failure, the contract between requestor and provider will not be validated and payment transfer will not be initiated. This acts as insurance against providers who don't deliver the service they agreed to. Both of these factors highlight that a business's downtime is limited by the blockchain and the costs of any downtime are automatically minimized.

Cloudwith.me Solutions - Abstract

It can take months for businesses to implement cloud services as they stand. Breaking the cloud package into independent networks, each with their own interface and currency, means businesses will have to dedicate even more resources to get trained staff to maintain their cloud servers than they would for centralized cloud hosting.

Cloudwith.me ("CWM") already offers its clients a managed hosting solution for hyperscale cloud services. This allows business owners to adapt to their server needs without being weighed down by the technology and time that goes into setting up and maintaining their cloud server instances. By bringing the cloud down to Earth, services can be made accessible to businesses big or small, tech-savvy or not. Managed hosting eliminates much of the need for in-house IT staff and administration in a business, and lets these departments focus on mission-critical tasks.

Advancing cloud service instances with peer-to-peer decentralization has already proven popular, as can be seen from the remarkable outcome of the Storj and Golem solutions [33] [34]. The cost of such decentralized services is irrefutably lower than similar services from centralized providers. Now Cloudwith.me steps in to provide a fully functional, complete cloud service that equips its clients with all the benefits of hyperscale vendors' packages at a discount. By introducing peer-to-peer cloud services on their managed platform, clients can focus on understanding the benefits of decentralization instead of its inner workings.

Even though the security and privacy benefits inherent to blockchain eliminate a lot of the trust and pricing concerns businesses express for centralized cloud service providers, decentralization can't provide a strong service without a wide or active network. Indeed, for peer-to-peer networks it's a case of 'strength in numbers'. Cloudwith.me has already established a secure client base in managed hosting for cloud services. This makes Cloudwith.me the ideal platform to transition to a decentralized cloud service, as both of the major public cloud services can still be used where various aspects of the Cloudwith.me platform are unavailable or in development. Decentralized services that are ready for market can then be provided at a discounted price. As the platforms offers both options, clients can see the difference in price and performance between peer-to-peer and centralized systems directly.

The Cloudwith.me solution is to endow clients with their own private blockchain application in the distributed network. Given a private blockchain, the cloud package owner can choose to add devices and users to the network as they please, making it possible to have a blockchain acting within the business or between affiliates. Smart contracts on the blockchain can be personally tailored to the needs of the

business, by the business. Private networks can be employed to automatically manage databases, enable peer-to-peer file sharing within a business or even to adopt a cloud computing application to utilize the idle CPU across all of a business's devices.

From the start, Cloudwith.me platform will enable users to benefit from the largest cloud service provider services at 50% of the cost. This automatically keeps a running receipt of the services used on the cloud. The **Cloud** token grants Cloudwith.me clients the security of Ethereum payments and minimizes transaction fees for cloud service applications. In this way, clients are incentivised to join Cloudwith.me's decentralized cloud. Adoption of the **Cloud** token increases the scale and therefore the strength of the cloud network.

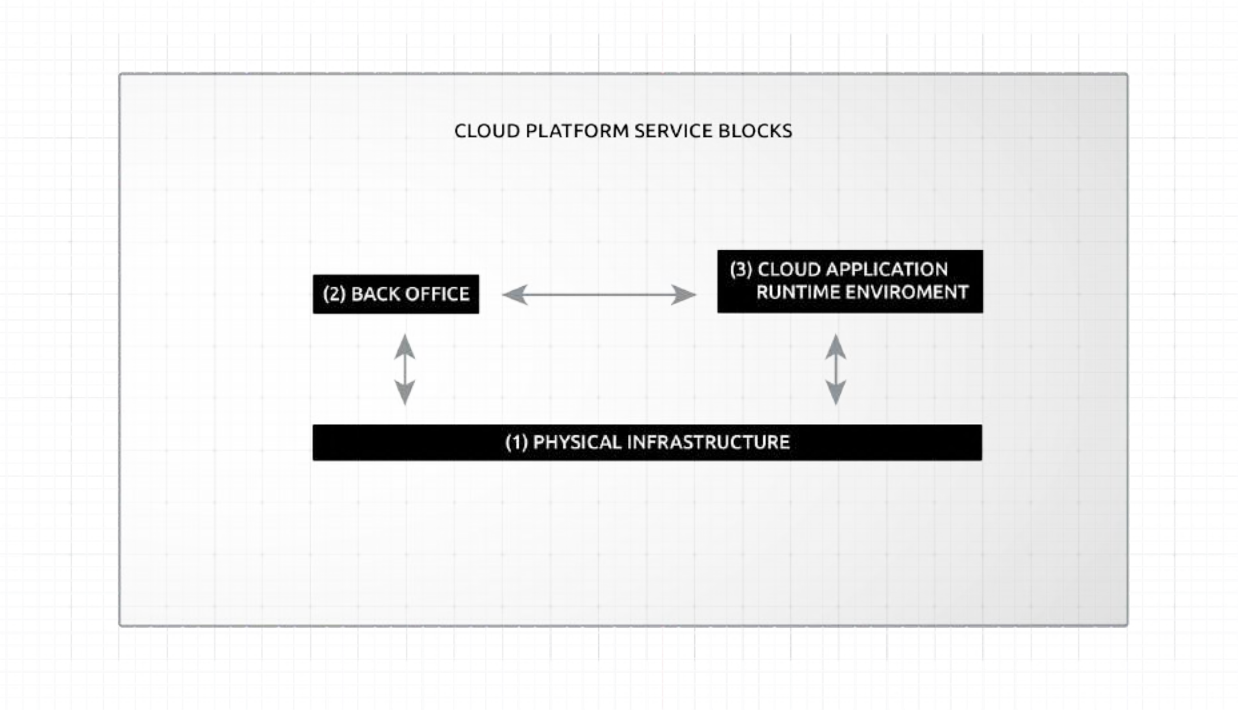
Cloudwith.me ultimately aims to create a global marketplace where blockchain-based cloud applications can be presented and traded with Ether-based payments. The introduction of the Cloud token as the sole currency for cloud services allows all applications to be experienced in the same ecosystem. Customers of Cloudwith.me can offer their services and receive payment in Cloud, which they can then use to request the services they need.

Cloudwith.me Solutions - In-depth Technical Overview

A Decentralized Cloud Hosting Environment

The Blocks of Cloud Computing Services

A distributed cloud-hosting grid must address the same needs as those are answered by today's centralized cloud hosting platform solutions. To facilitate a meaningful discussion of this topic, let's consider the following breakdown of a cloud service environment:



- The physical infrastructure consists of everything that has to do with the hardware and the physical space in which the hardware is located. Today, cloud providers operate Data Centers in which they run thousands of servers, routers and switches, power them with enough electricity, connect them to the Internet backbone, take care of cooling requirements, and secure access to them with tons of concrete, barbed wire, steel doors, alarm systems and armed guards.

- The back-office includes software services and user interfaces that allow customers of the cloud hosting provider (i.e., cloud application owners) to create accounts, get permission to access services, receive bills, pay for cloud services usage and receive support.
- The cloud application runtime environment consists of all of the software components that enable and support the runtime of cloud applications. The elements in this block provide the core functionality that differentiates today's modern cloud platforms from legacy hosting providers.

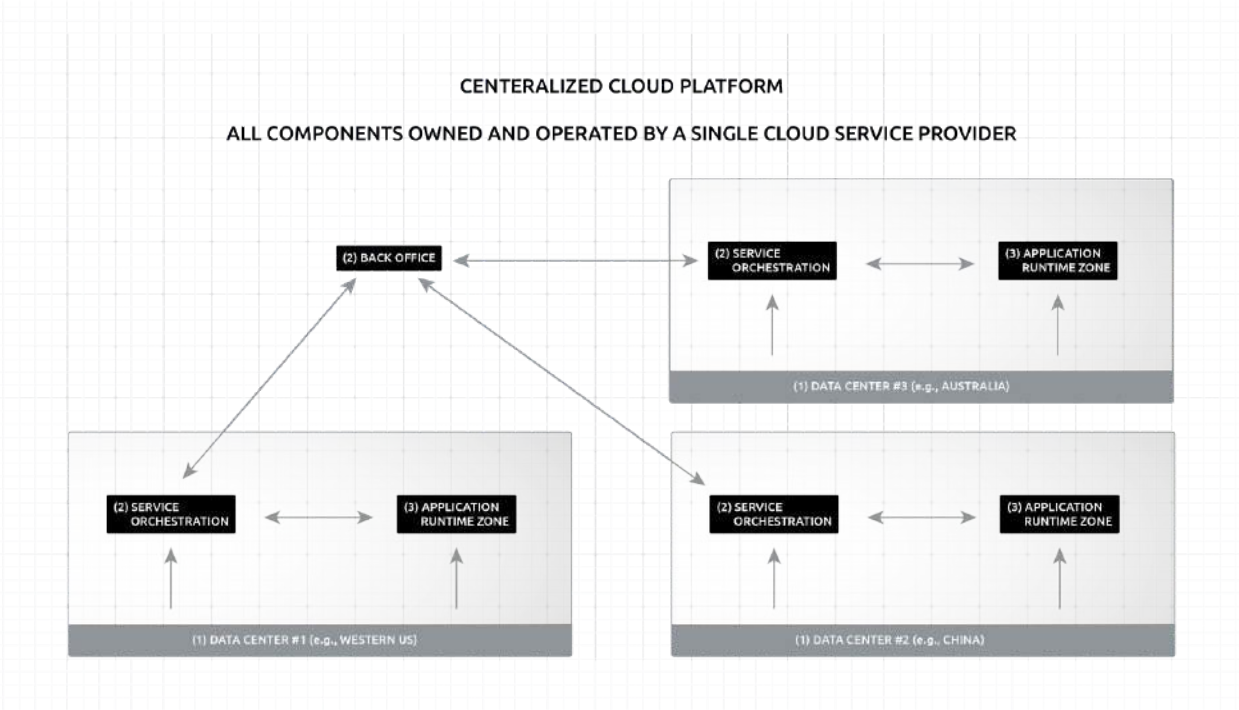
The following table further details the above by breaking each of the service blocks into layers:

<table>
<tr><th>(2) MANAGEMENT</th><th>(3) CLOUD APPLICATION RUNTIME ENVIROMENT</th></tr>
<tr><td rowspan="5">(2.2) Service Managment
[service provisioning, service orchestration, usage accounting]

(2.1) Account Management
[billing, payments, account onboarding, customer support]</td><td>(3.5) Applications</td></tr>
<tr><td>(3.4) Application Platforms
[cloud functions, node.js/php/ruby/python/etc. ruby-on-rails/django/larvel/etc., wordpress/ drupal/joomla/etc., magento/etc. and so forth.]</td></tr>
<tr><td>(3.3) Cloud Application Services
[app servers, nosql databases, sql databases, analytics, big data storage and processing, image processing, machine learning, face recognition, authentication and security, billing services, payment processing, phone messaging, mobile application backend, large scale message queueing, hybrid cloud connectivity, etc.]</td></tr>
<tr><td>(3.2) Cloud O/S
[elastic compute, load balancing, object storage, content delivery, network security]</td></tr>
<tr><td>(3.1) Operating System
[program execution and scheduling, block storage, networking]</td></tr>
<tr><th colspan="2">(1) PHYSICAL INFRASTURCTURE (DATA CENTERS)</th></tr>
<tr><td colspan="2">(1.3) Computer Hardware
[servers, hard-drive, networking equipment]</td></tr>
<tr><td colspan="2">(1.2) Hardware Enablement
[electricity, cooling systems, racks, high-bandwith internet backbone connectivity]</td></tr>
<tr><td colspan="2">(1.1) Secure Physical Space
[large buliding, barbed wire, alarm systems, armed security presonnel, etc.]</td></tr>
</table>

Centralized vs. Decentralized Cloud Platforms

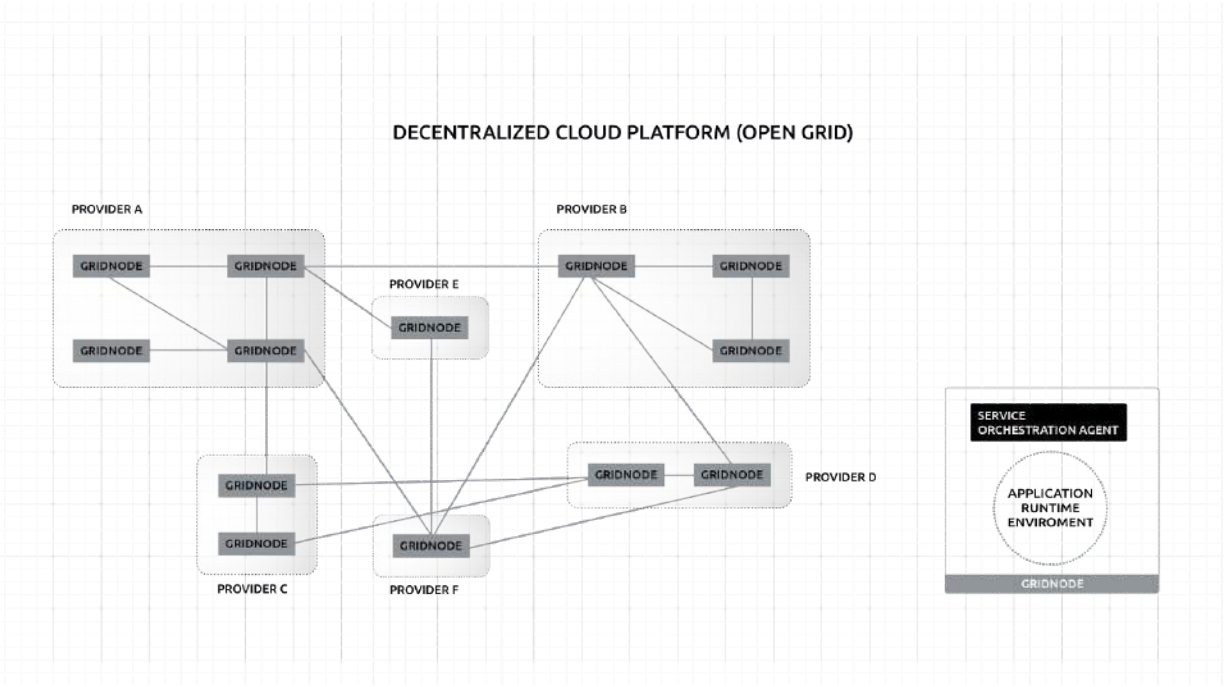
The following figure illustrates how centralized cloud platforms operate over a number of data centers (ranging from a handful of data centers in the case of small providers, up to several dozen in the case of large providers)—where those data centers are roughly exposed to blocks 2 and 3 using the concept of Zone or similar (e.g., "AWS US-West").



Key things to note here: every single component illustrated in this diagram is owned and operated by the same entity—i.e., the cloud service provider. When an application owner deploys an application to a centralized cloud platform, that application owner must do several things:

1. Establish a legal relationship with the cloud provider (i.e., open an account, provide payment information, etc.)
2. Develop expertise in the cloud provider's proprietary implementation of block #3 (cloud application runtime environment)
3. Architect and code the application to make optimal use of the cloud provider's geographical deployment and service orchestration mechanisms.

A decentralized cloud platform would transform the relationship between the application owner and the cloud service providers. To understand how, let's examine the structure of a decentralized cloud platform as illustrated below:



There are several key differences from the previous topology:

1. The basic building block of the decentralized cloud platform is a GridNode component, providing the basic cloud platform service blocks.
2. The GridNodes are interconnected to form a mesh network for service management and request routing. Each GridNode is connected to the blockchain..
3. GridNodes are operated by a variety of different service providers
4. The GridNodes expose a uniform application runtime environment, thereby allowing seamless migration of application code between service providers.
5. The application owner creates a blockchain-based accounting relationship with the decentralized platform, not with the providers.

GridNodes are intended to be installed and operated by GridNode owners, who can be independent operators as well as cloud service providers. This flexibility allows the grid to evolve from the current state and expand to a global scale. This is, of course, a long-term vision that is extremely flexible and supports a wide variety of use cases. But as explained in later sections of this document, the evolution of GridNodes can start today by supporting a limited set of scenarios and gradually evolving to fulfill the open grid vision.

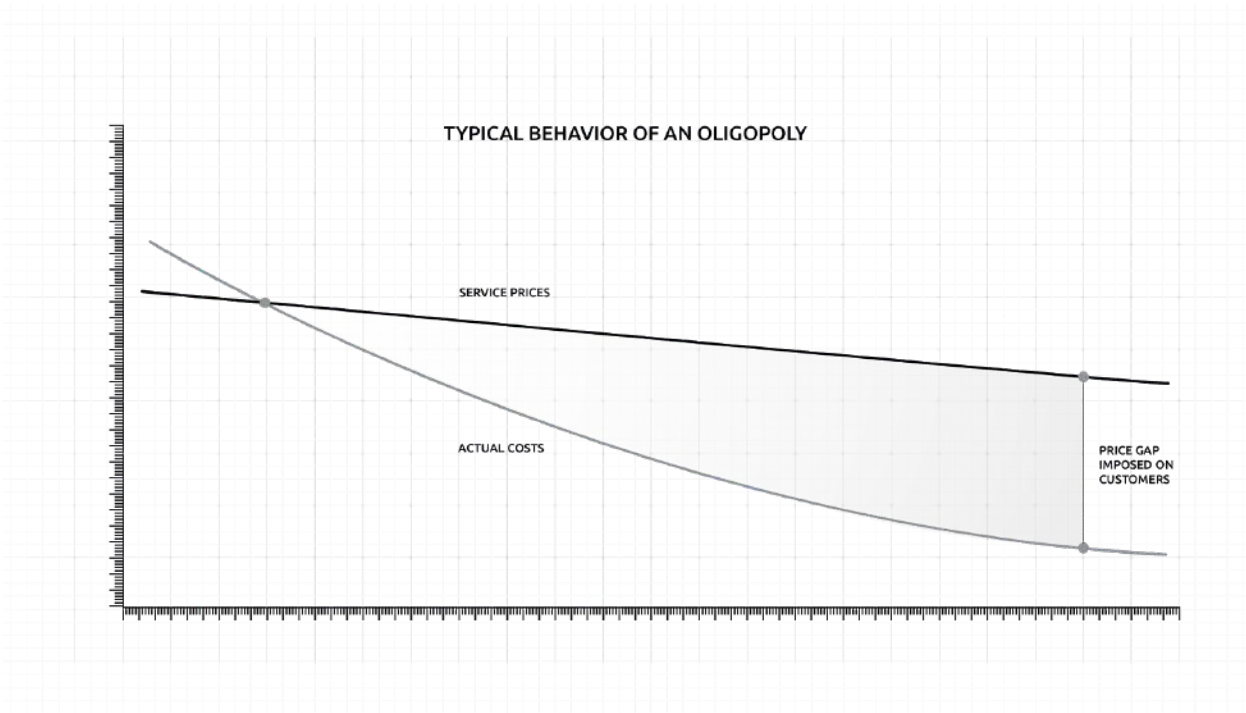
Benefits of Decentralization

Cost Reduction

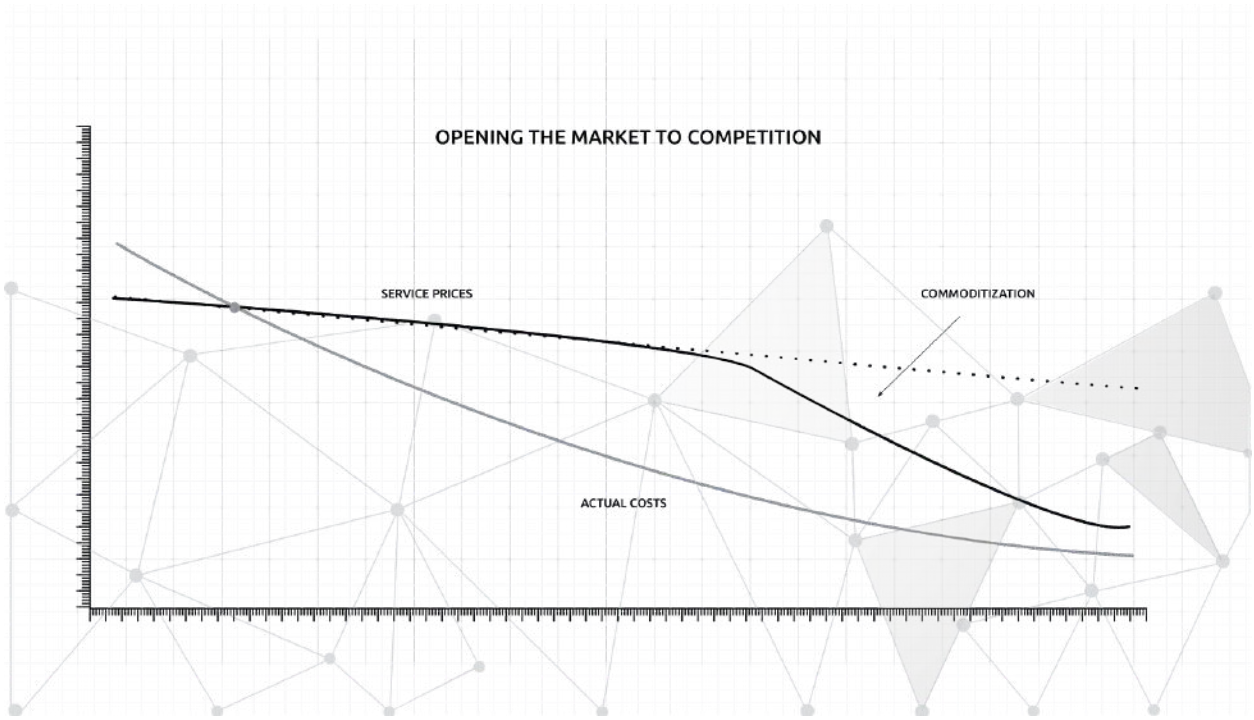
One of the realities of our economy is that new technologies eventually become commoditized, and the market moves from differentiated to undifferentiated price competition, and from monopolistic to perfect competition [link]. To quote Columbia Business School professor Bruce Greenwald: “In the long run, everything is a toaster” [link]. But until new technologies and services become commoditized, their prices are kept well above what they become when the market opens up to price-based competition.

The cloud-hosting space is currently very far from perfect competition. It has been described by some as an oligopoly [link]: a market dominated by a handful of major players, where no new player of any meaningful size can enter.

An oligopoly has a natural tendency to drive prices way above what they might be in a truly competitive market.



The Company believes that the decentralization process will drive commoditization of cloud services to commodity rates as low as 94% below current market prices. Essentially, the goal is to have cloud services on equal status to running water or electricity in developed countries—a utility, removed from price gaps or unnecessary markups.



As demonstrated most recently by Uber and Lyft in the ridesharing domain, enabling almost anyone to become a service provider can transform that market and bring prices down [link].

Privacy and Security

The distribution of application data and logic across millions of nodes, each one secured independently and located in a different place, makes it difficult for an attacker to obtain access to a hoard of sensitive information items.

Similarly, with the right protections and isolations in place, the GridNode owner is unaware of the data and computations performed in the GridNode.

Akin to "double blind" clinical trial protocols, data is secured from both malicious end-users and GridNode owners.

Increased Resiliency

Another important benefit of decentralizing the cloud is improved resiliency. Today, the vast majority of cloud services are being provisioned from no more than a few hundred data centers, located in strategic locations around the world. As a result, failure in a single data center can have catastrophic effects on a significant number of Internet users that rely on applications that run in those data centers. As recently as

last March, AWS Storage services in Amazon's US-East-1 region in northern Virginia had an outage, affecting, as reported in The Register [link]: "Docker's Registry Hub, Trello, Travis CI, GitHub and GitLab, Quora, Medium, Signal, Slack, Imgur, Twitch.tv, Razer, heaps of publications that stored images and other media in S3, Adobe's cloud, Zendesk, Heroku, Coursera, Bitbucket, Autodesk's cloud, Twilio, Mailchimp, Citrix, Expedia, Flipboard" and more.

A decentralization of cloud hosting would exponentially increase the number of locations that serve cloud applications, turning the effect of a failure in any one of those locations into a much less painful event.

Operational Transparency

The knowledge we have today, as a society, about the overall use of computational resources across the globe is filled with uncertainty: neither application owners nor cloud service providers are under any obligation whatsoever to release accurate and complete information about how many servers are being used, where they are located geographically, what type of cloud services are being consumed and so forth.

The decentralized grid can include mechanisms that contribute valuable knowledge about usage statistics without revealing sensitive business details, and allow all of us to better research and understand computational usage patterns and trends across the globe.

Better Network Performance

One of the exciting benefits of the decentralized grid is that it could eventually provide better performance for users that are not located near high-bandwidth trunks, such as underdeveloped regions or countries. If an application is used extensively in a certain geographic area of the globe, and the grid is architected to facilitate such behavior through appropriate transparency and bidding mechanisms, then the application would tend to migrate to that region, thereby reducing distances between server and end-user, improving performance and reducing traffic costs.

More Choice

Today the large cloud providers offer a large menu of platform and application services, yet it is a limited set, fully controlled by the provider. Subscribers of cloud services benefit from services such as managed databases, but they are locked in, with little incentive to migrate to a different provider, or to use multiple providers. The open GridNode platform will be built to accommodate third parties of building blocks from any layer. Choices are expected to significantly outnumber existing offerings by a single cloud-service provider.

In addition, decentralized cloud operating on blockchain currency payment method lowers the barrier to entry and creates accessibility of pro cloud services to developing countries where credit cards, which are the standard modus operandi for the current cloud services, are scarce. The Cloud token becomes an equalizer and helps to bring services to every country on equal grounds.

Grid Growth Expectations

As the project is put into motion, the community should be able to track progress and understand whether the entire grid, as a decentralized hosting entity, is making a significant impact and indeed transforming the way cloud services are consumed globally.

One of the things we would like to do is compare the total computing power (in FLOPS) of the decentralized grid with the total computing power offered by the major cloud service providers. Unfortunately, there is no reliable data about how much computing power is made available by the major cloud providers. However, some data exists on the total electrical power consumed by those providers, and we can also make some educated guesses about the performance per watt of the machines hosted by them and the critical power loads in their data centers.

We suggest that those metrics—of total power consumption and total computing power—could also be useful for understanding the growth of the grid as a computational entity. Those metrics also help understand efficiencies and thereby the environmental impact. As an open community, we would like to see the grid not only create more transparency, resiliency and cost reduction, but also have a positive environmental effect.

Let us examine the total set servers S_θ operated by a cloud hosting provider θ be defined as $S_\Theta = \{S_1, S_2, ...\}$. Let C_i be the maximal computing power (in FLOPS) afforded by server S_i. The total computational power CP potentially provided by cloud service θ would then be computed as

$$CP_\Theta = CP(S_\Theta) = \sum_i^{|S_\Theta|} C_i .$$

To calculate the *effective* computing power made available to applications that operate on the hosting provider's infrastructure we must also take into account any computational overhead that is expended on operational aspects of the hosting provider itself, such as the overhead of running virtual machines, deploying applications to new servers, etc. The effective computational power CP' would then be calculated as $CP'_\Theta = \frac{CP_\Theta}{\upsilon}$ where υ is the computational overhead quotient of the system Θ as a whole.

Let W_i be the power consumption in watts of S_i.

If we know the Performance per Watt (ω_i) in FLOPS/Watt for S_i, we can infer that its computational power C_i would be $C_i = W_i * \omega_i$.

To calculate the total computational power CP_Θ of provider Θ we can now use the formula

$$CP_\Theta = W_\Theta * \overline{\omega}_\Theta \text{, where } \overline{\omega}_\Theta \text{ is the average } \overline{\omega}_\Theta = \frac{\sum_{i}^{|S_\Theta|} \omega_i}{|S_\Theta|} .$$

Looking at various sources (e.g., [link] and [link]) we can probably assume that for today's servers the value of ω would be in the range of around 2-4 GFLOPS/Watt. We should also assume that data centers have a mix of newer and older servers, where the older servers have a lower energy efficiency.

To examine the power consumption (and from that infer the computational power) of Θ we must take into consideration, not just the power consumed by the servers themselves (*critical power*), but also the power consumed by ancillary systems in the data center—most notably cooling (but also lighting systems and others). Since not all data centers have the same power efficiency, let $\overline{e}_\Theta$ be the weighted average power consumption overhead of all data centers in which the servers are stored.
Let $D = \{D_1, D_2 ...\}$ be the cloud provider's set of data centers, and $S^d = \left\{S_1^d, S_2^d, ...\right\}$ be the set of servers in data center D_d, and W_i^d be the power consumption of server S_i^d.

The total critical power consumption W^d of data center D_d is $W^d = \sum_{i}^{|D_d|} W_i^d$, but the total power $\widehat{W}^d$ consumed by the data center in its entirety (including overhead of the non-critical systems such as cooling, etc.) would be $\widehat{W}^d = e_d \cdot W^d$, where e_d is the power overhead quotient for data center D_d (Based on [link] we can assume that a reasonable value for e in most data centers would be around $\frac{1}{0.36} \sim 2.8$).
The total power consumption of the provider across all data centers is $\widehat{W}_\Theta = \sum_{d}^{|D|} e_d \cdot W^d$.

$$\text{Since } \widehat{W}_\Theta = \overline{e}_\Theta \cdot W_\Theta \text{, we know that } \overline{e}_\Theta = \frac{\sum_{d}^{|D|} e_d \cdot W^d}{\sum_{d}^{|D|} W^d} .$$

Due to the heterogeneous nature of the decentralized grid, we will not be able to assume that $\overline{e}_\Theta = e_d$ for any given D_d when considering our own decentralized grid, but in the case of traditional data centers we can probably overlook this and also assume that $\overline{e}_\Theta \sim 2.8$, and assume that for a cloud provider running typical data centers, $W_\Theta = \frac{\widehat{W}_\Theta}{2.8}$.

From various public sources, we can tell the $\widehat{W}_{\Theta}$ of major cloud providers. For example, according to [link], Amazon's Virginia US-East data center is almost at 10^9 watts. Assuming Amazon has put a lot of effort into improving their data center's power efficiency, we would guess that their $\overline{e}_{\Theta}$ is likely better than the common 2.8, so using $\overline{e}_{\Theta} = 2.8$ puts a lower bound on our calculation. As such, we can suppose that W_{Θ} is better than $10^9 \times \frac{1}{2.8} = 0.36 \times 10^9$ watts, i.e., 0.36 Gigawatts.

To estimate the computational power of the data center we can now use our earlier formula $CP_{\Theta} = W_{\Theta} * \overline{\omega}_{\Theta}$. If we place a range of 2-4 on the value of $\overline{\omega}_{\Theta}$, we can guess that Amazon's US-East data center probably has a compute power in the range of $2 \times 0.36 \times 10^9 = 0.72 \times 10^9$ GFLOPS and $4 \times 0.36 \times 10^9 = 1.44 \times 10^9$ GFLOPS—i.e., between 0.7-1.4 Peta FLOPS (where 1 Peta FLOPS = 10^{15} FLOPS).

These numbers and formulas give us a sense of desired scale and a way to monitor the growth of the decentralized grid, while also providing some baseline energy efficiency targets.

In an effort to calculate the cost of bootstrapping a decentralized grid that is comparable to just 4% of the effective computing power of an AWS data center (as an example), while the actual computations are beyond the scope of this paper, we calculate an estimate that will guide the level of funds needed to achieve critical mass on the path to decentralization.

For that matter, we assume an average cost of $30K/server. This cost also factors in estimated operational costs for three years as well as the cost of the networking equipment, etc. (i.e. the effective computing power we get at this cost should not be assumed to be high). We then assume each such compute unit, which for simplicity purposes runs a single GridNode and has an effective computing power of 100 gigaflops on average which means that to get an effective computing power of 25.6 petaflops in total we would need 10,240 GridNodes.

With one GridNode per server, and at a server price of $30K, the total cost for the GridNodes critical mass bootstrap would be approximately $307M. That might seem like a high number, but it's just 1% of the $31.5B spent on capital expenses and leases by the top three cloud service providers in 2016 alone, or of Google's $30B overall cloud spending.

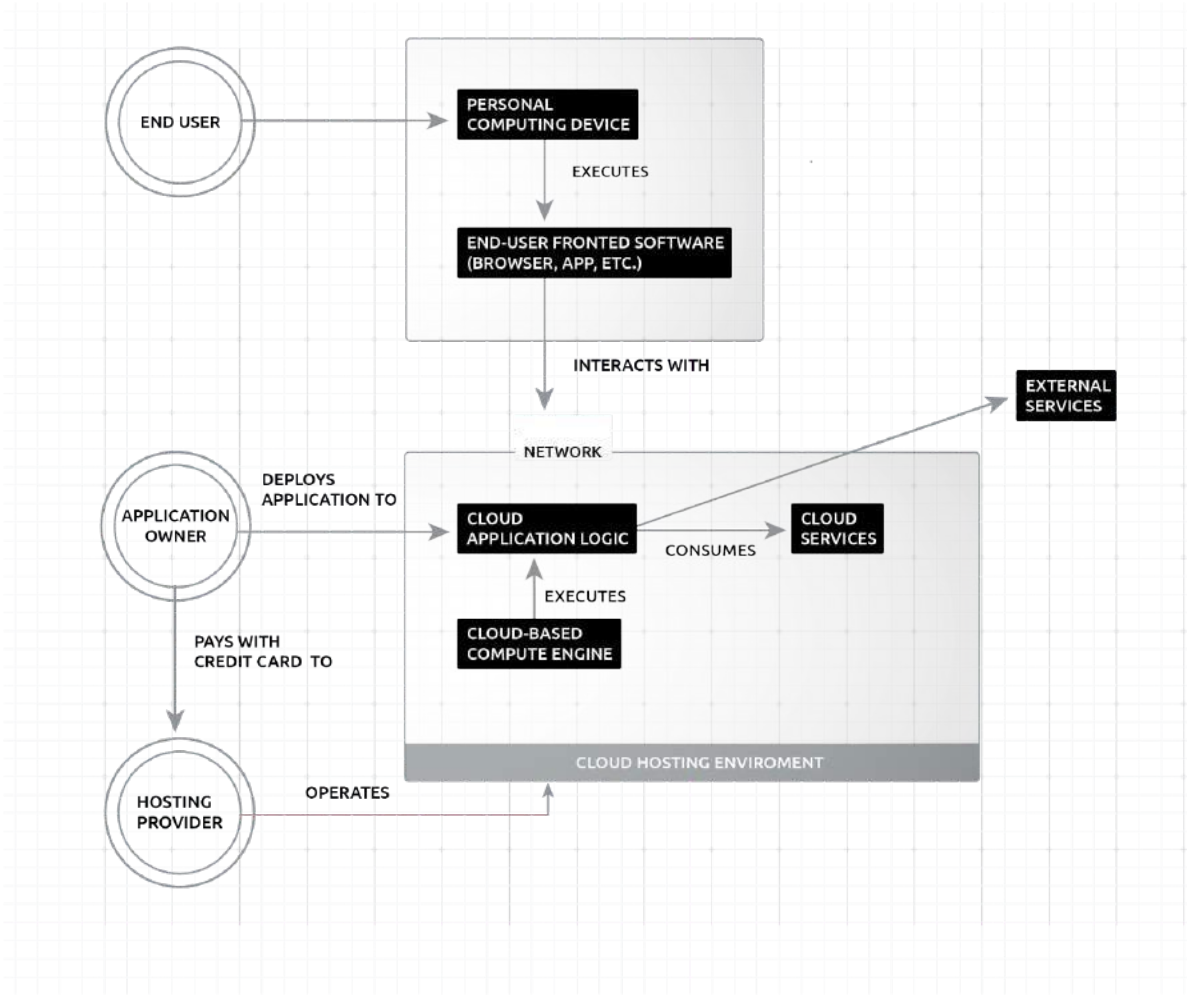
Application Use Case Analysis

Let's examine the common use case of a web/mobile application back-end, and compare the behavior of the centralized vs. decentralized approaches. In this sort of scenario, we have the following entities:

- End-users: interacting with the hosted application via a personal front-end application
- The application owner: requiring physical and logical infrastructure to host and operate the application
- The cloud services provider: furnishing the application owner with the means to run the application.

Centralized Cloud

The following drawing describes the trivial case of a single user accessing the cloud-hosted app:



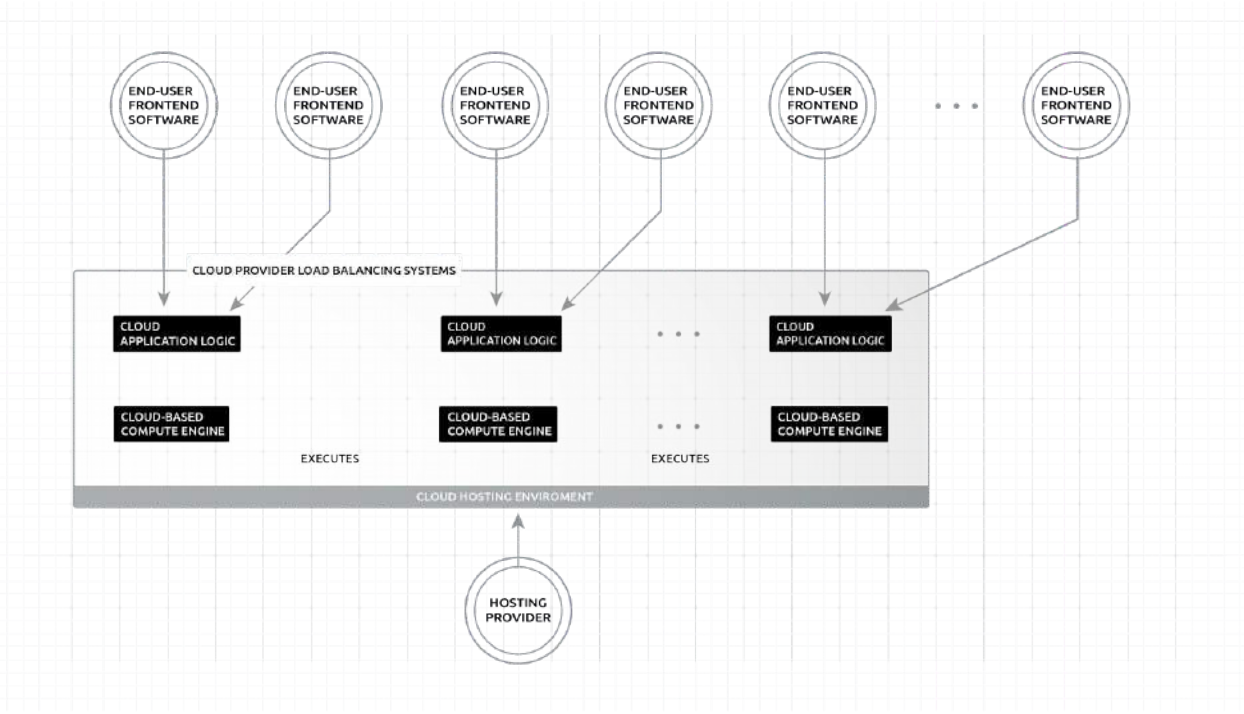
The end-user operates the personal computing device (mobile phone, laptop, gaming console or other). The front-end application (e.g., a browser or a mobile app) runs on the user's device, and interacts with a single instance of the cloud-application logic to achieve tasks such as getting the contents of web pages, storing progress in a game, sending an email message, initiating an online purchase, or any other applicative operation.

Referring to our service blocks layers (in the above table), let's describe a common configuration using the layers terminology. In this configuration, our cloud application logic (layer 3.5) is written in TypeScript using bootstrap.js (layer 3.4) and transpiled into javascript code running on node.js (layer 3.3) to implement the REST APIs and the back-end algorithms required for the applicative system to function correctly. The code leverages mongodb (layer 3.3) and an object store provided by the cloud service provider (layer 3.2), and it runs on a virtual machine (layer 3.1) that executes on a physical server (layer 1.3) in a secure data center (layers 1.2, 1.1) operated by the cloud service provider. In addition to

consuming services provided by the cloud service provider, it also leverages a set of services that it receives from external sources (e.g., a payment processing service).

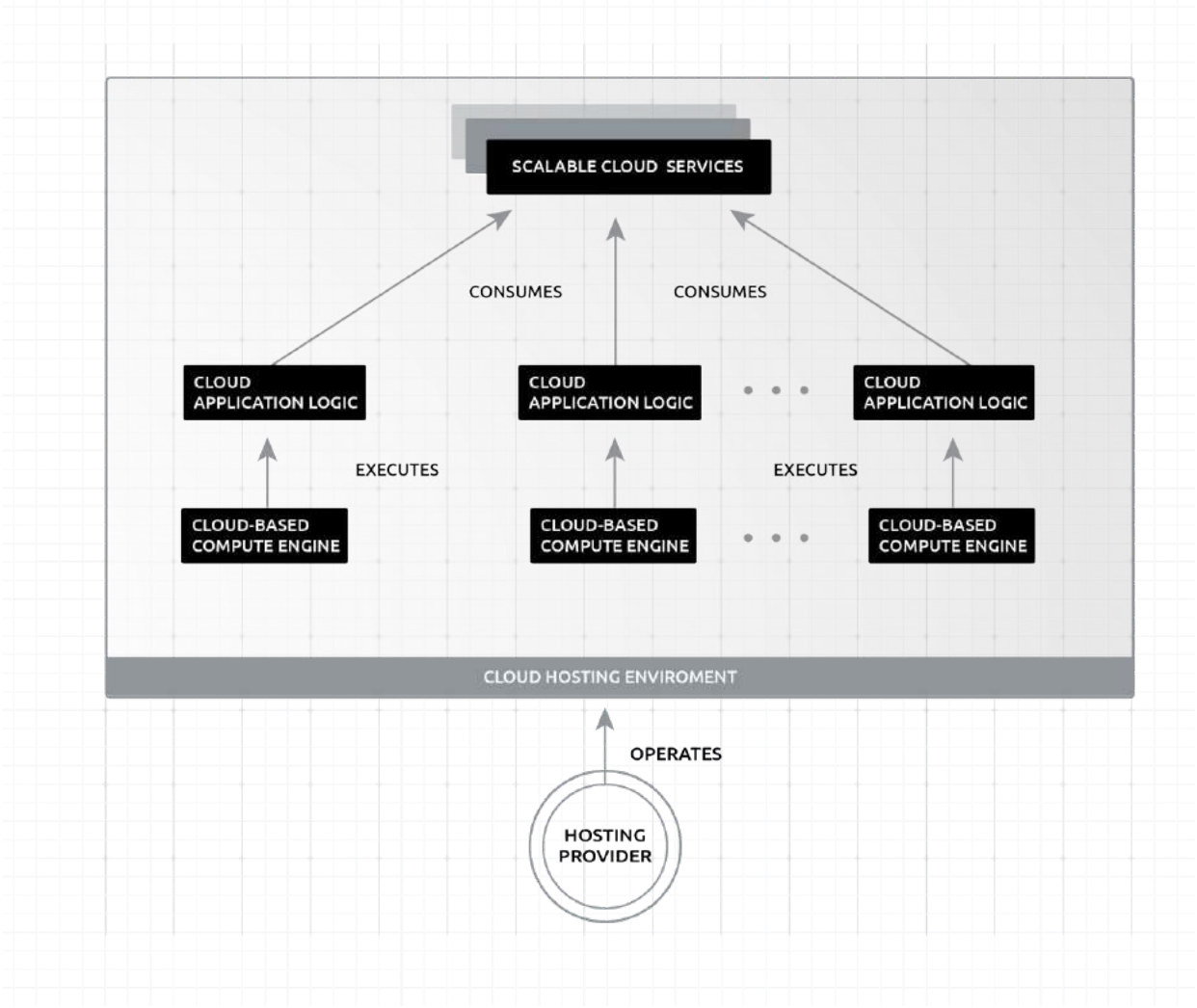
The application owner pays the hosting provider to have this entire stack execute on the provider's infrastructure. It is the responsibility of the hosting provider to ensure that the infrastructure operates seamlessly and supports the needs of the applications. As this application consumes services from external providers, it is up to the application owner to establish a relationship and settle payments with those other providers.

To support the high load generated by a multitude of users, applications need to scale. This requires that more than one compute engines be available to execute multiple instances of the application logic. Requests from different external users are routed to different instances of the application, based on factors such as resource availability, application context, geographic location, and so forth.



To support such scaling, the cloud-hosting provider also allows the application owner to rent load balancing and content delivery services in order to route front-end requests to available instances of the application that are in the general geographic area of the request origin.

In addition to deploying the application logic to multiple compute engines, the cloud services consumed by the application logic must also scale to support the high load generated by having multiple application instances access them:



The implementation and operation of a highly scalable cloud-hosting infrastructure requires a considerable amount of resources (data centers, computers, networking equipment, electricity, cooling, physical security, etc.). Only a small number of very large companies are able to provide such infrastructure on a global scale.

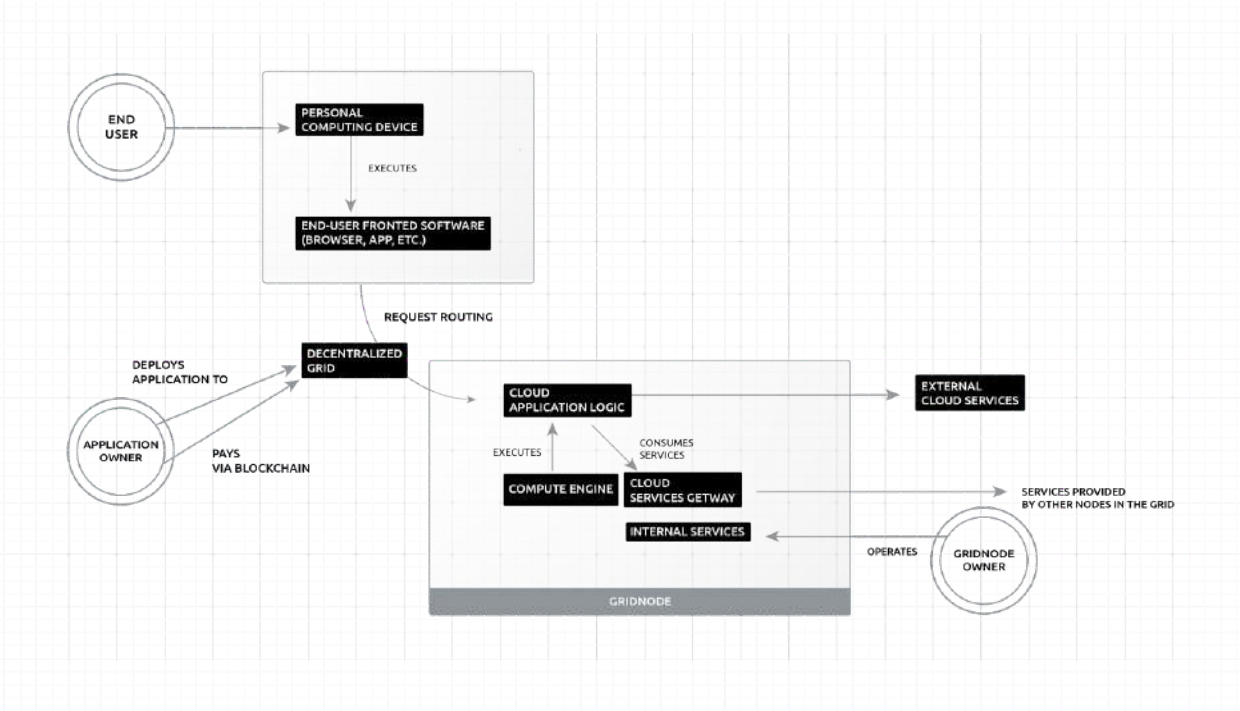
Decentralized Cloud

Returning to the trivial case of a single user accessing the web/mobile backend cloud-hosted app, let's examine how this would work in the decentralized topology, where a GridNode executes a single instance of a cloud application via a sandboxed environment such as a virtual machine or container (i.e., an element in layer 3.1).

As before, we have three primary participating entities (end-user, application-owner and cloud services provider), only now there is no direct relationship between the application-owner and the cloud services provider (i.e., the GridNode owner), as the relationship is established using a smart contract between the application owner and the grid.

The GridNode owner is responsible for setting up the GridNode, connecting it to the decentralized grid via the Internet and ensuring its ongoing operation.

A blockchain (initially Ethereum-based Cloud token, but other blockchains will also be supported) is used to send payments to the GridNode owner and other participating parties—but also for additional purposes, such as entity description, performance metrics, etc.



The fundamental interaction between the front-end software and a single application instance that is hosted on a GridNode is quite similar to that of traditional cloud hosting, with one important exception: routing of the request from the front-end device to the GridNode is achieved through a grid-based load balancing system, rather than through a load balancer operated by a single cloud services provider. The load-balancing logic is embedded into the GridNode, so that every GridNode can optionally participate in the routing of network requests.

The GridNode provides the application with a full runtime environment (block 3). Many aspects of the application's architecture would be similar to the previous case, but some aspects might differ. For example, while the app's layer 3.4 elements might still be TypeScript and bootstrap.js, and the layer 3.3 app server is still node.js, the underlying storage service (layer 3.2) might be based on Storj or a similar decentralized storage system. Conversely, an application running on a GridNode might choose to leverage

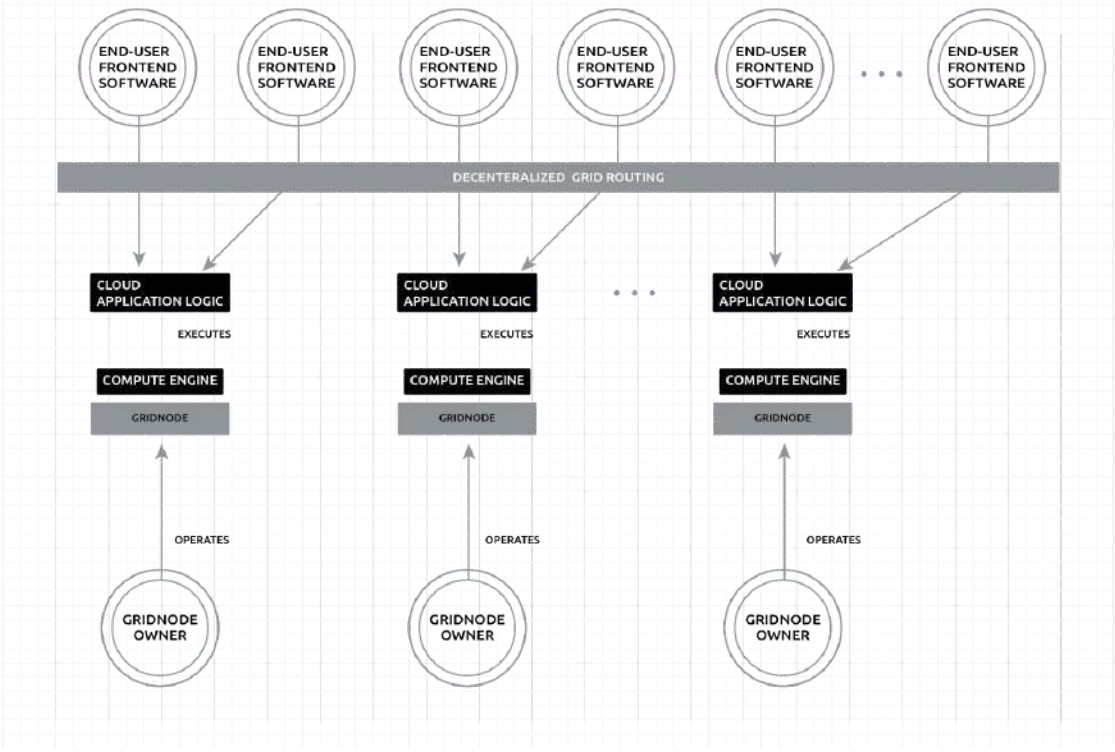
Amazon S3 or Google Cloud Storage, while running some of its compute functionality on a decentralized supercomputer such as Golem. To put this in more general terms, layer 3.2 and 3.3 (see table p.15) services such as storage, analytics, message queuing, etc. are provided to the Application Logic using three different means:

- Some services are provided directly by software components on the GridNode.

- Other services are provided by entities that are external to the GridNode, but through a standardized API that is provided by the GridNode to the application through a 'Cloud Services Gateway' component. Those external services can run on other GridNodes, or be hosted other environments such as classic cloud hosting. The Cloud Services Gateway component enables the grid to provide a blockchain-based token conversion mechanism for those external services.

- Application Owners can also pay external providers for consumption of additional cloud services by the application, just like they can in the traditional cloud hosting approach. Such payments would not necessarily be via blockchain-based currency.

The big difference between the traditional cloud hosting approach and the open grid becomes apparent when we examine how the grid supports scaling of applications. In the traditional approach, we see multiple instances of the cloud application logic being executed by multiple compute engines that are operated by the same cloud services provider. In the decentralized cloud, we still see multiple instances of the application logic—only now they are served by compute engines that are operated by many different GridNode owners.



Cloud-Hosting Contracts

When designing the smart contracts for the decentralized service, we must consider the needs and expectations of both the GridNode owners and the application owners. As the objective of the Cloudwith.me Open Grid is to provide a service that is comparable to traditional cloud hosting, let's first examine how contracts between application owners and cloud service providers are built today, and see how that would be translated to the decentralized cloud.

How Things Are Done Today

Most cloud-hosting contracts include at least the following elements:

- The cloud-hosting provider commits to provide the application owner with a certain predefined package of services in return for payments that are calculated to take into account a number of factors. Factors may include such things as the amount and type of allocated resources, expected availability of the resources, expected response time in case of faults, and so forth.

- The application owner commits to pay for the services, and use them only according to certain terms set by the hosting provider. Almost always, the hosting provider would require that the application owner will only use the service for lawful purposes (including such things as avoidance of copyright infringement, using only software that the application owner is allowed to use, not supporting any illegal or immoral activities, and so forth), and that the application owner will not abuse any APIs provided by the hosting provider or disrupt the services of the hosting provider in any way.

Hosting providers almost always commit to a Service Level Agreement (SLA) that defines what level of performance the customer can expect from the hosted environment, and what would be the compensation in case the hosting provider is unable to live up to that promise (many cloud-hosting providers give 'service credits'-i.e., they provide a limited discount on future services—to make up for downtime or slow performance suffered by the customer).

The type of service in question (e.g., compute, network, data backup, etc.) determines which measurements are possible on that service. The specific definition of metrics and service levels varies across cloud service providers, but they usually refer to uptime (often measured as a monthly percentage), and in some cases the response time as well.

Smart Contracts

Supporting the Relationship Between GridNode Owner and Application Owner

The relationship between a GridNode owner and an application owner is very similar—in terms of mutual commitment and expectations—to the relationship between an application owner and a cloud services provider:

- The GridNode owner would commit to provide the application owner with a predefined package of services and capabilities, and expect to receive appropriate payments in return for providing those services as promised.
- The application owner would commit to pay for the services, and to use those services in a way that does not disrupt the operation of the GridNode or for any unlawful or immoral purposes.

However, a key conceptual problem that an open grid service needs to deal with is trust between GridNode owners and application owners. Unlike classic cloud hosting, the application owner is not interacting with a well-known brand, and cannot simply rely on the name of the GridNode owner to decide whether to trust that entity to live up to those commitments. Similarly, the GridNode owner might not have the required financial resources to deal with application owners who do not live up to their commitments.

The Trust Issue

GridNode owners do not inherently trust the application owners, but they want to be sure those application owners will pay them for the resources they consume and will not misuse the service.

On the other hand, application owners do not inherently trust the GridNode owners, but want to be sure that if they pay (or commit to pay) for grid resources, then they will actually get those resources from the GridNode owners.

In many cases, Blockchain smart contracts aim to remove the need for trust by completely automating the conclusion of a contract through objectively verifiable means. Unfortunately, in the Cloud Services domain, the reliable and accurate provisioning of computational services cannot be 100% verified by automated means [link].

This situation creates an evident conundrum:

1. If the Application owners commit to pay before getting access to the resources: as there is no fully-automated way to verify that the cloud resources behaved as promised, how can application owners know that the GridNode owners won't just take the payment and then not provide the promised resources?
2. If the GridNode owners commit to provide the resources before getting paid: as there is no fully-automated way to prove that they actually provided the correct resources, how can GridNode owners know that the application owners won't use the resources and then refuse to approve payment (even if placed in escrow), claiming that the resources provided were not as promised?

A Blockchain-Based Reputation Mechanism

The established mechanism for overcoming this problem—of enabling a trust-based transaction between anonymous buyers and sellers on the Internet—is a reputation system. This mechanism is used by companies like eBay and Amazon to facilitate trust-building for the sale of physical goods. Companies like eBay and Amazon serve as a trusted intermediary, managing a central database of the reputation of both buyers and sellers, and providing a dispute resolution service to overcome disagreements when those arise.

Cloudwith.me intends to create a blockchain-based reputation system that is based on a smart contract for subscribing to cloud resources (Cloud tokens).

A primary concept of Cloud tokens is GridReputation. GridReputation is stored (both for GridNode owners and for application owners) on the blockchain (initially the Ethereum blockchain, but a mechanism will be developed to transfer GridReputation to other blockchains in the future).

GridReputation is a measure of how trustworthy the participant has been so far: application owners gain GridReputation when they demonstrate that they are well-behaved systems of the grid and quickly pay for services they have used, and lose GridReputation if they use grid resources and then fail to pay for them, or otherwise fail to abide by the terms of service.

GridNode owners gain GridReputation by providing reliable hosting services, and lose GridReputation if the hosting services they provide prove to be unreliable (e.g., the environment crashes, network bandwidth is too low, etc.).

Reputation elements for GridNode Owners: uptime, consistent performance as promised (compute, network, data r/w access), data storage reliability, trustworthiness (i.e., they only charge for services rendered), data security.

There are two modes whereby the grid grants access to GridNode resources:

- Option 1: pay now, use later (like buying a prepaid SIM or an Amazon gift card)
- Option 2: use now, pay later (like giving your credit card number to a hosting provider)

Pay-now-use-later will be favored when the application owner does not yet have a strong reputation.

Use-now-pay-later will be favored in situations where a GridNode owner does not yet have a strong reputation.

Performance-monitoring Agents

Performance-monitoring agents can be introduced into the grid for the purpose of allowing GridNode owners to have a third party periodically assess their level of service, publicize the results, and objectively influence their GridReputation over a range of service metric dimensions.

A trivial example of a performance-monitoring agent could be an “uptime tracker”, paid to periodically check the uptime of nodes in the grid.

Other performance monitors could assess aspects such as compute engine performance, bandwidth reliability, firewall setup, hardware configuration reliability, and so forth. Such performance monitors would send compute jobs to selected GridNodes and assess the performance of those nodes by running benchmark jobs and comparing the actual result to what is expected, given the platform attributes published by the GridNode owner.

Entities that run such performance monitoring tasks have could be incentivized through smart contracts to maintain their objectivity, either by acting as Oracles (where multiple independent monitors would perform independent assessments on the same GridNode, and a monitor who deviates significantly from the majority finding would risk losing a deposit), or by creating a mechanism whereby the payment for publishing the results of the investigations would come from the GridNode owner if the results were positive, and from the application owner if the results were negative.

Dispute Resolution

As with any reputation-based mechanism, disputes are bound to arise. To facilitate dispute resolution, the infrastructure and smart contract for the decentralized grid will take into account the need to transfer payments to arbitrators (possibly as Oracles on the blockchain), and to expose to them operational logs that can help them make decisions.

Ensuring Integrity of Code and Data

A naive design of a hosting grid would be vulnerable to malicious parties who join the grid as GridNode owners—with the intent to steal or modify sensitive data that is processed by cloud applications.

Traditional cloud providers do not have an incentive to steal customer data: as organizations, their reputation depends on their ability to increase confidence that applications are not tampered with, and they are also potentially liable to legal action should they cause the leak of sensitive information that they are supposed to protect. However, some employees (or partners) of those organizations might potentially abuse the access they have to data. As such, providers advertise the use of physical security mechanisms to prevent access to the machines, and they fire employees who are found to tamper with data. While this approach is essentially trust-based and is not entirely foolproof, it does make it difficult for attackers to reach the data,.

As anyone can join the grid as GridNode owner, the grid cannot assume that all GridNode operators are trustworthy custodians of data.

A malicious GridNode owner might attack the grid using various means. Two prominent examples include:

- Operating a modified GridNode that steals information from the applications it executes, or modifies data in some way that benefits the malicious party.
- Installing a shadow (RAID) drive on the machine that runs the GridNode, and physically removing it from the machine in order to copy sensitive information from it.

As an application owner, you cannot by any means deploy the SSL certificate with your TLS secret key to a web server hosted on an unknown machine that you cannot reasonably trust.

While there are means that a Cloud application might employ to obtain some level of confidence that it is not running on a modified GridNode, no software mechanism can self-verify with 100% certainty that it is running on an uncompromised operating system. Even worse, there is no means whatsoever by which a software application could detect the existence of a shadow hard drive that is installed on the computer.

The conclusion from the above is that there is a need for external means to increase confidence in the trustworthiness of the computing environment operated by the GridNode owner.

There are essentially three approaches that the grid could combine to provide such means:

1. Cryptographically assure the real-world identity of any GridNode operator. Application owners will be able to whitelist or blacklist specific operators based on their identity.

2. Perform spot inspections of the GridNode operator’s physical infrastructure

a. GridNode owners who wish to increase their 'data safety' reputation will sign up for surprise inspections by independent inspectors. The smart contract for data safety will assure that such owners will escrow a fraction of their Cloud tokens revenue to pay the inspectors who inspect their site.
b. To reduce the likelihood of the independent inspectors colluding with malicious GridNode owners, some GridNode owners will be rewarded for running honeypot sting operations to uncover unreliable inspectors. This reward will come from an escrow of a portion of the fees paid to the inspectors—also assured by the smart contract.

3. Support Trusted Execution Technology with cryptographic signatures for attestation of the authenticity of the GridNode application and the underlying Operating System, as well as the physical architecture of the computer. We envision that as the number of GridNode owners increases, the availability of such platforms will increase, and we intend to work directly with partners who wish to create 'trusted data-center-in-a-box' appliances.

GridNode operators that wish to maintain a very high data-safety reputation level would combine at least two if not all three of the above.

Compliance Considerations

Regulatory Compliance

Another important aspect of the decentralized grid is regulatory compliance. There are two primary dimensions to take into consideration: geographic regulations, and domain-specific regulations.

Two prominent examples of geographical regulation include the upcoming European General Data Protection Regulation (GDPR) and the recently imposed Chinese regulations on cloud service providers. GridNode owners who fall under the jurisdiction of such regulation must comply with it or risk being subjected to very high fines. (Note that the EU GDPR applies to any entity that handles personal information—including IP addresses—of European citizens, regardless of where the servers are physically located [link]). This means that the GridNode component must include mechanisms for data security and privacy to assure such compliance.

Domain-specific regulations—for example HIPAA (for health [link]) or PCI-DSS (for credit card information [link])—impose even higher security requirements both on the application owner and on the operator of the infrastructure running the application. Regardless of whether the regulation comes from governments or from industry associations, applications that fall into categories covered by such regulation will not be able to run on an infrastructure (i.e., a GridNode) that doesn't comply with those regulations.

Uniform Terms of Service for Application Owners

Since GridNodes are intended to be hosted in a wide variety of locations—including a multitude of cloud-hosting providers—each such hosting provider requires that application owners using the hosting provider's services comply with the hosting provider's legal terms of service. This means that application owners who run their applications on multiple GridNodes distributed across several hosting provider facilities need to abide by the terms of service of each and every one of those hosting providers.

One of the goals of the open grid is to define uniform terms of service that all hosting providers see as sufficient, so that once application owners agree to those uniform terms of service, they will be free to have their applications operate on GridNodes that are hosted on the facilities of any cloud-hosting provider.

Deploying Application Instances

Deployment Parameters

To create instances of an application, the application owner needs to deploy an application to the decentralized grid. There are several important aspects to consider here:

1. The grid will contain a mix of many variants of GridNodes from many different providers. When deploying an application instance to the grid, the application owner will specify certain criteria, and only GridNodes that meet those requirements will be eligible to bid on the right to deploy a copy of this application. The following is a non-exhaustive list that illustrates the types of parameters that the application owner might choose to specify. Note that some parameters are optional, and since we are dealing with a highly heterogeneous grid environment, the parameter values need not be exact and might be a range of acceptable values from the point of view of the application:
    - Layer 1.1: geographical location, physical security requirements from the machine (e.g., is it required to be hosted in a secure data center, is it expected to employe Trusted Computing technologies for tamper-proofing, etc.)
    - Layer 1.2: hosting location bandwidth
    - Layer 1.3: hardware configuration (cpu speed, memory, max # of VMs per bare metal machine)
    - Layer 3.1: type of OS, allocated memory for the VM/Container, network bandwidth assigned to the VM/Container, amount of local storage available, type of local filesystem
    - Layers 3.2, 3.3: a list of cloud APIs (and API versions) that are expected to be supported directly by the GridNode, firewall requirements

- Layer 3.4: platform components installation requirements (Does the application owner expect certain packages (e.g., Ruby-on-Rails or WordPress) to be preinstalled on the machine, or does the owner already include the entire set of platform components in the application bundle for deployment?)
- Layer 3.5: is the application expected to stay on the machine even after the execution period is over, or can the application be removed from local storage once execution is completed?

2. In addition to technical requirements from the GridNode, the application owner can specify business criteria, including:
    - A minimal reputation for the GridNode owner
    - Integrity requirements (e.g., frequency of audits)
    - Historical results from performance monitors
    - Regulatory compliance requirements
    - Payment mode (a ratio of use-now-pay-later to pay-now-use-later)
    - Resource allocation duration—i.e., for how long are the computing resources going to be reserved?
    - Agreed dispute resolution mechanisms
    - Black-listed and/or white-listed GridNode operators

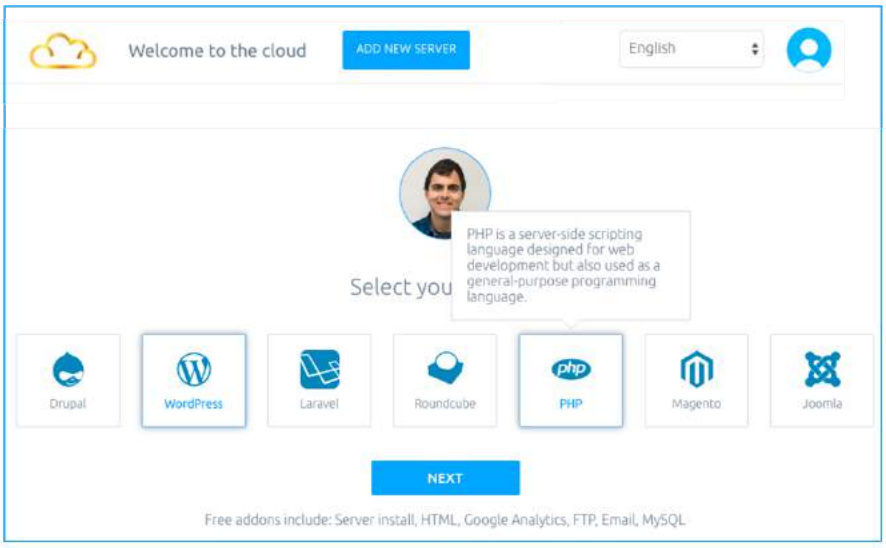
3. Pricing. When an application instance is deployed to the grid, the application owner specifies the maximum price to be paid for every type of service that the application consumes. This might be specified for the entire application, but might also be specified on a per-component basis, in the case where the application consumes multiple resources.
4. User experience. It is important to allow people who are not cloud experts to be able to deploy applications to the grid. As such, the user experience for performing all of the above specifications should be made clear and intuitive.
5. Decentralization assurance. In order to ensure that the grid does not become controlled by a small number of large GridNode owners, the grid will incorporate mechanisms to incentivize (or even require) that when a large number of instances of an application are deployed, they will be distributed between multiple GridNode owners.

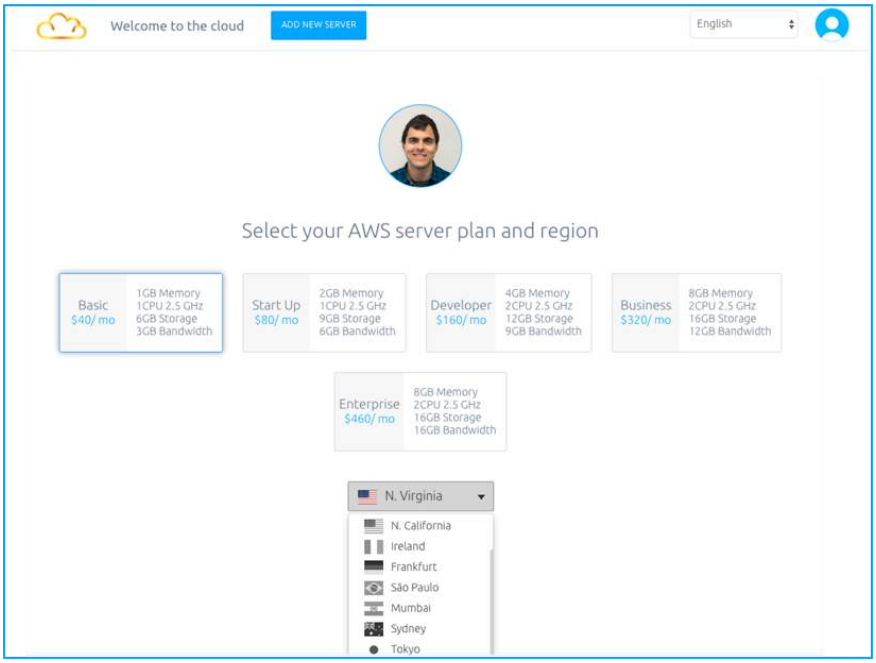
Cloudwith.me's Existing Interface for Traditional Cloud Providers

Today's cloud hosting providers offer a very wide range of execution environment types and configurations, and the selection of the right configuration is a highly technical task that requires unique knowledge and expertise.

Cloudwith.me's existing product already simplifies this task by providing a friendly user interface that makes it a breeze to set up compute environments on AWS, Azure and Google to support common application use cases.

The two screenshots below are taken from Cloudwith.me's existing interface, and demonstrate how easy it is use our existing product to provision such environments in the classic cloud. Our intention is to bring the same level of simplicity and friendliness to the process of deploying applications to the decentralized grid.





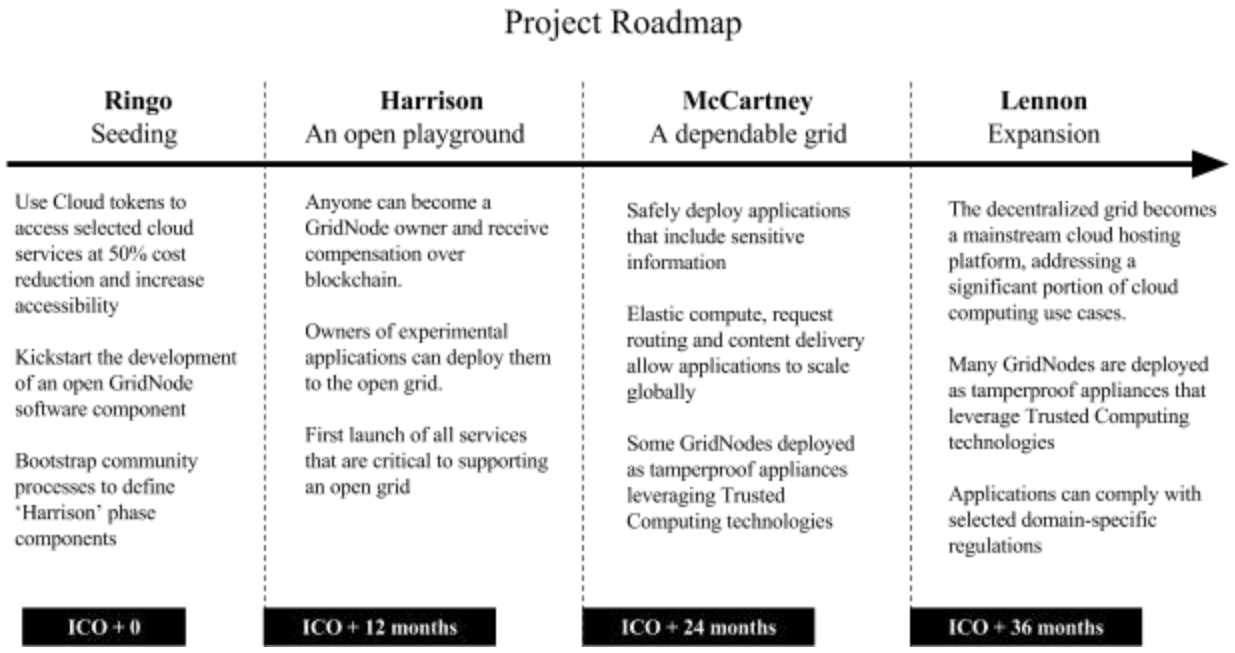
Additional Resources

- On regulation and compliance:
    - *https://www.cloudindustryforum.org/content/cloud-and-eu-gdpr-six-steps-compliance (outlines 6 steps to take as a cloud consumer who serves European customers, to ensure your compliance with the EU's pending GDPR)*
    - *http://www.computerweekly.com/feature/EU-Data-Protection-Regulation-What-the-EC-legislation-means-for-cloud-providers*
    - *https://iapp.org/resources/article/eu-data-protection-law-and-the-cloud/*
    - *http://searchcloudprovider.techtarget.com/essentialguide/Navigating-cloud-computing-regulations-and-compliance-requirements*
    - *https://uk.practicallaw.thomsonreuters.com/w-007-4744?originationContext=document&transitionType=DocumentItem&contextData=(sc.Default)&firstPage=true&bhcp=1*
    - *https://www.dlapiper.com/en/australia/insights/publications/2016/12/stepping-up-regulation-of-cloud-services-in-china/*
- On using blockchain to create a reputation system:
    - *https://medium.com/mit-media-lab/certificates-reputation-and-the-blockchain-aee03622426f*
    - *https://cointelegraph.com/news/digital-word-of-mouth-how-blockchain-reputation-system-would-work*
    - *https://medium.com/topl-blog/divine-a-blockchain-reputation-system-for-determining-good-market-actors-7c47a0308ae8*
- On Trusted Computing:
    - *https://en.wikipedia.org/wiki/Trusted_Computing*
    - *https://en.wikipedia.org/wiki/Trusted_Execution_Technology*
    - *https://invisiblethingslab.com/resources/bh09dc/Attacking%20Intel%20TXT%20-%20paper.pdf*
- About data centers and reputation
    - *http://www.tia-942.org/*
    - *http://www.opusinteractive.com/request-a-tour/*
- On vulnerability of data held by traditional hosting providers:
    - *https://techcrunch.com/2010/09/15/google-needs-to-do-a-lot-more-than-fire-employees-who-abuse-user-data/*
    - *https://security.stackexchange.com/questions/109359/can-personnel-who-manage-aws-datacenters-access-my-ec2-instances-and-monitor-dat*
- Cloud hosting SLA examples:
    - *https://cloud.google.com/compute/sla*
    - *https://aws.amazon.com/ec2/sla/*
    - *https://azure.microsoft.com/en-us/support/legal/sla/*
    - *https://azure.microsoft.com/en-us/support/legal/sla/storage/v1_2/*

Roadmap and Bootstrapping

The development of the open grid is expected to take place in four major phases. In each phase, we evolve the GridNode technology and the associated smart contracts to provide more functionality, support more use cases, increase the number of deployed GridNodes, and improve the dependability of the grid itself.

The following describes our anticipated progress through those phases.



Phase 1 (codename: 'Ringo') - Seeding

Objectives of this phase:

- Provide immediate value by allowing the use of Cloud Coins tokens to access selected AWS, Azure and GCS cloud services (and possibly also services from other cloud platforms, such as IBM Bluemix)
- Kick start the development of an open GridNode software component
- Explore the viability of various technological approaches
- Bootstrap community processes to define Phase 2 components

Primary activities:

- Develop the initial GridNode software component and smart contract.
- Adapt existing Cloudwith.me functionality to work on GridNodes.
- Orchestrate a community process for developing the detailed requirements of reputation and trust building in the open grid.
- Initiate work on Uniform Terms of Service in collaboration with hosting providers.

Grid deployment:

- Only Cloudwith.me will serve as GridNode owners.
- Hosting will be over Amazon, Azure and GCS infrastructure, and possibly additional platforms.

Cloud platform service blocks:

- Layer 3.5: applications leveraging the predefined platforms provided in Layer 3.4
- Layer 3.4: limited to a subset of the applications currently supported by Cloudwith.me (Currently supported applications are Drupal, WordPress, Laravel, Roundcube, PHP, Magento, Joomla)
- Layer 2:
    - Service management and account management capabilities to support deployment across major cloud providers
    - Usage reporting and blockchain token billing reports
- Layer 1: relying on the physical infrastructure of major cloud service providers

GridNode functionality:

- GridNodes can only run predefined applications.
- Load balancing and elasticity are achieved using the infrastructure of the major cloud-service providers (traditional load balancers, CDNs).

Phase 2 (codename: ‘Harrison’) - An open playground

Objectives of this phase:

- Anyone can become a GridNode owner and receive compensation over blockchain.
- Owners of experimental applications can deploy them to the open grid.
- First launch of all services that are critical to supporting an open grid (i.e., reputation over blockchain, GridNode capability advertisement, application deployment to the grid, grid-based load balancing, cryptographic assurance of GridNode owner identity, etc.)
- Establish initial relationships with providers of Trusted Computing hardware platforms.
- Allow software that runs on GridNodes to leverage other distributed services for storage, compute, etc.

Primary activities:

- Develop the grid protocols, algorithms and smart contracts
- Provide the capability to operate a GridNode and to deploy applications to the open grid
- Orchestrate a community process for developing the Trusted Computing certification and grid interoperability requirements
- Recruit additional hosting providers for participation in the project and in the Uniform Terms of Service initiative

Grid deployment:

- Most infrastructure still provided by the large cloud hosting companies
- Some infrastructure provided by independent providers

Cloud platform service blocks:

- Layer 3.5: open to experimental applications beyond the predefined ‘Aardvark’ set
- Layer 3.4: add support for additional application platforms based on the Layer 3.3 set of application services
- Layer 3.3:
    - Application servers: candidates include node.js, JVM-based servers, ruby-based, python-based, others.
    - Integration with decentralized services such as Storj, Golem, others.
    - Integration with common cloud services such as object storage (e.g., S3), etc.
- Layer 3.2:
    - Grid load balancing
    - Elastic application deployment
    - Content delivery
- Layer 2.2:
    - Application deployment services
    - GridNode management UI
    - Service usage and blockchain token reporting

- Layer 1: relying on the physical infrastructure of major cloud-service providers

GridNode functionality:

- GridNode owners join the grid by downloading the GridNode software and running it on their computers.
- Initial implementations of Load Balancing and Elasticity functions added to the GridNode software
- GridNodes evolve to provide more types of services and support additional use cases.
- Initial prototypes of tamperproof GridNode appliances

Phase 3 (codename: ‘McCartney’) - A dependable grid

Objectives of this phase:

- Application owners can safely deploy to the grid applications that include sensitive information (such as personal information and their SSL/TLS certificate secret key).
- The grid’s high performance mechanisms (elastic compute, request routing, content delivery) allow applications to scale globally.
- Some GridNodes are deployed as tamperproof appliances that leverage Trusted Computing technologies.

Primary activities:

- Security audit of the GridNode code and grid algorithms
- Performance optimizations to the algorithms and the code
- Implement mechanisms to enable compliance with privacy regulations (such as GDPR)
- Onboarding of new partners of all types (hosting provides, decentralized service providers, cloud service providers, Trusted Computing platform manufacturers, grid agents such as performance monitors)

Grid deployment:

- Over 30% of nodes are operated by independent providers.

Cloud platform service blocks:

- Layer 3.5: supports custom production-grade applications
- Layer 3.4: a continuously growing catalog of application platforms
- Layer 3.3: continue to add more application services and application platforms
- Layer 3.2: improved security and performance
- Layer 2.2: service provisioning and orchestration, accounting for partner services
- Layer 2.1: customer support, detailed billing information
- Layer 1: trusted computing nodes provide reliable security outside of high security data centers

Phase 4 (codename: ‘Lennon’) - Expansion

Objectives of this phase:

- The decentralized grid becomes a mainstream cloud hosting platform
- A Large number of GridNodes are deployed as tamperproof appliances that leverage Trusted Computing technologies
- Improved performance
- Enabling compliance with selected domain-specific regulations (possibly PCI-DSS, HIPAA, etc.)
- GridNodes provide a wide range of services, addressing a significant portion of cloud computing use cases.

Primary activities:

- Work to bring the decentralized grid into the mainstream
- Onboard Enterprise customers
- Continue activities of the previous phase on a larger scale

Market Opportunity

IDC and Gartner have forecast that the global public cloud service industry will reach over $195 billion in revenues by 2020, doubling figures from last year [4] [35]. Within this industry, the cloud-managed service market is expected to grow at an expected CAGR of 14.6% [36]. As more businesses and individuals take the leap into the cloud, the providers they're drawn to will be decided by: cost, range of services, availability, ease of integration with their existing computer framework, transparency of the service, and risk mitigation. The combination of blockchain technology with an automated managed solution addresses all of these concerns.

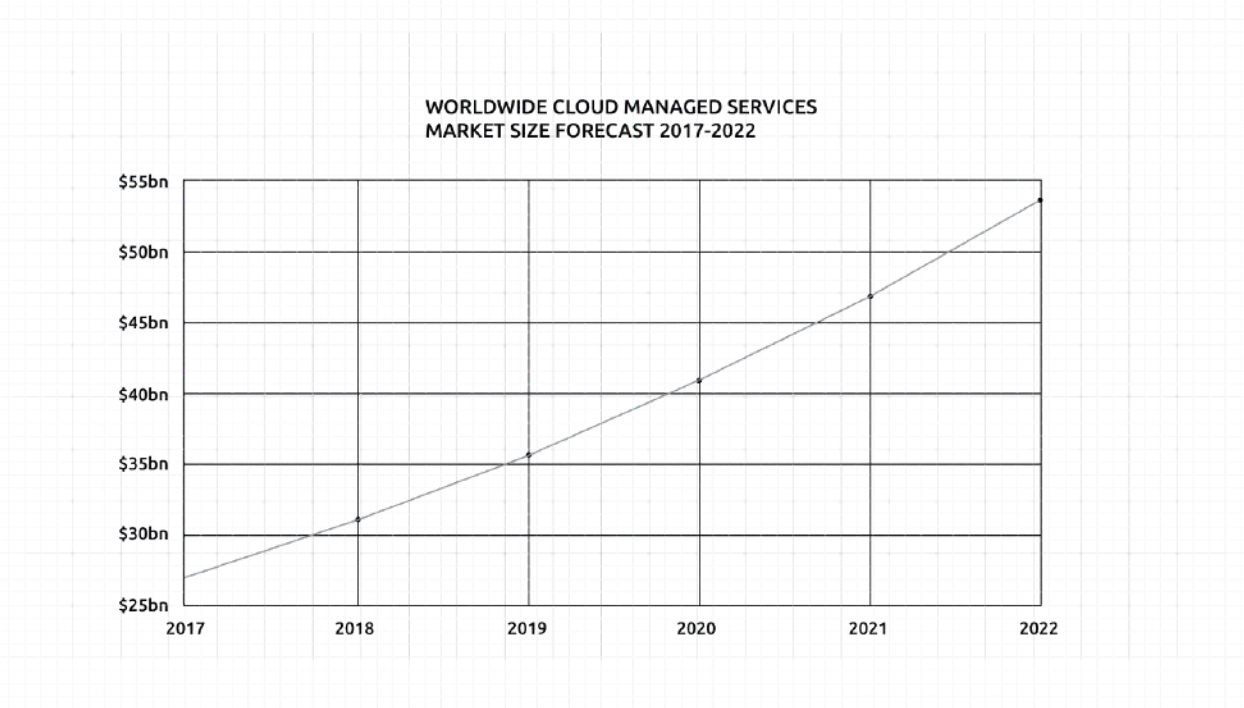


Fig: Plotted using data from [36]

Adoption of public cloud services by SMBs has been rising rapidly over the past years. This is reflected in Cloudwith.me's success with its current managed hosting platform. A 2014 study by IDC found that 65% of SMBs worldwide made use of cloud storage [37]. Their 2016 report on cloud adoption summarizes that by then, over 70% of small businesses and over 90% of medium-sized businesses had adopted cloud services [38]. Despite this, nearly 50% of SMBs still experience cyber attacks [9].

Cloud Token ICO

The Ethereum-based **Cloud** token ICO is intended to create the first block in the vision of Decentralized Applications Internet by providing a standard currency to be used by all cloud services and service providers.

From the start, the Cloudwith.me platform will enable users to benefit from the largest cloud service provider services at 50% cost reduction. Already making a difference, the Company is determined to change the way people use and consume cloud services.

The Cloud token will evolve during its lifetime into several instances where a new version of cloud with smarter contracts and more efficient cloud services payments governance will match the growth and expansion of the GridNodes both in volume and sophistication.

Generation I**, Cloud** will enable trusted payments for cloud servers, storage, bandwidth and processing power.

Generation II, **Cloud** aims to become the token governing exchange of decentralized peer-to-peer cloud services

A Cloud token holder has the inherent right to replace (by choice) each existing coin of previous generation into a new one at no cost.

Periodically, new Clouds will be announced and offered to existing holders who can choose to send their old tokens back to the company in exchange for the new ones.

All legacy generation tokens will be destroyed and annulled once replaced. All generation tokens will retain their attributes of independence from the Company itself and will bear no rights in relation to the Company itself or its activities.

Use of Proceeds

The use of proceeds will be predominately for the global deployment of the GridNode infrastructure as per the model shown in the above "Grid Growth Expectations" section.

The company believes that this amount is sufficient in order to bootstrap the decentralized cloud nodes infrastructure growth.

Self-deployment of GridNodes is planned globally from the first successful run of the CWM blockchain GridNode module. In this quantity, the Company believes a critical mass of nodes can be reached within circa 12-18 months from which point the growth of the decentralized web will become viral-positive and the adoption of the GridNodes by new peer-to-peer Providers will accelerate at a reasonable rate for gaining momentum, leading into complete Cloud migration to blockchain within circa five years from the initial Ringo seeding stage.

In an effort to calculate the cost of bootstrapping a decentralized grid that is comparable to just 4% of the effective computing power an AWS data center (as an example), while the actual computations are beyond the scope of this paper, we calculate an estimate that will guide the level of funds needed to achieve critical mass on the path to decentralization.

For that matter, we assume an average cost of $30K/server. This cost also factors in estimated operational costs for three years as well as the cost of the networking equipment, etc. (i.e., the effective computing power we get at this cost should not be assumed to be high). We then assume each such compute unit, which for simplicity purposes runs a single GridNode and has an effective computing power of 100 gigaflops on average, which means that to get an effective computing power of 25.6 petaflops in total we would need 10,240 GridNodes. With one GridNode per server, and at a server price of $30K, the total cost for the GridNodes critical mass bootstrap would be approximately $307M. That might seem like a high number, but it's just 1% of the $31.5B spent on capital expenses and leases by the top three cloud service providers in 2016 alone, or of Google's $30B overall cloud spending.

In addition to the main use of funds, the Company will be financing the active promotion of GridNodes contributions via a "Become a Provider" campaign and ongoing development of its GridNodes and DAP technologies.

The overall amount raised has acceptance flexibility as lower amounts simply dictate longer adoption cycles and slower deployment rates, leading to longer lead times to full migration of cloud services to blockchain decentralization.

Risk Factors

The following are the risk factors in relation to Cloudwith.me ("CWM") business in general and the token sale event in particular:

CWM Token may not ever become a crypto currency

There is no assurance that at any time in the future the CWM Token (i) may be exchanged for goods or services, (ii) may have any known uses outside the CWM platform, or (iii) may be traded on any known exchanges.

Risk of failure to reach target sale amounts or risk of insufficient funds

CWM may not reach the target sale amount and may not have the sufficient funds to execute its business plan.

Risk of market trends

The CWM token may be significantly influenced by digital currency market trends and Cloud value may be severely depreciated due to non-Cloud-related events in the digital currency markets.

Risk of regulation/legislation

The Cloud services market and/or the token market may be or may be coming under global or local regulation/legislation that may render the Cloud trade impossible and/or may limit the use of tokens as a payment method and/or limit, prevent and/or sanction the sale and re-sale of tokens.

COI and relevant technologies have been the subject of scrutiny by various regulatory bodies around the world. The functioning of CWM and Token could be impacted by one or more regulatory inquiries or actions, including but not limited to restrictions on the use or possession of digital tokens like Token, which could impede or limit the development of CWM.

Risk of software not meeting expectations

The GridNodes and DAP are presently under development and may undergo significant changes before release. Any expectations regarding the form and functionality of Token or the GridNodes and DAP may not be met upon release, for any number of reasons including a change in the design and implementation plans and execution of the GridNodes and DAP.

GridNodes and DAP are complex software platforms and their launch may be significantly delayed due to unforeseen development barriers.

Risk of alternate technology

There is no guarantee that there are no other solutions or technology, whether being developed or to be developed in the future, that will severely depreciate the value of CWM as well as of its products and services.

Competition may introduce same or better solutions and cause CWM to lose market share and eventually fail to deliver on its business goals.

Risk of high volatility

Digital currencies are extremely volatile and Cloud token may suffer from said volatility.

Risk of taxation

The ownership of Cloud tokens may fall under existing and/or new and unpredicted taxation laws that will erode Cloud benefits.

Risk of low to no liquidity

Cloud may not succeed in creating the necessary momentum and acceptance, which may result in low liquidity and depletion of trades.

Risk of theft and hacking

Token sales and ICOs have been known to come under malicious attacks from hackers and criminal parties, resulting in theft of tokens, and massive losses to Token may be inflicted on buyers and the Company.

Hackers or other groups or organizations may attempt to interfere with the CWM activity or the availability of CWM Token in any number of ways, including, without limitation denial of service attacks, Sybil attacks, spoofing, smurfing, malware attacks, or consensus-based attacks.

Risk of security weaknesses in the CWM network core infrastructure software

There is a risk that the CWM team, or other third parties may intentionally or unintentionally introduce weaknesses or bugs into the core infrastructural elements of the CWM software network, interfering with the use of or causing the loss of CWM Tokens.

Risk of weaknesses or exploitable breakthroughs in the field of cryptography

Advances in cryptography, or technical advances such as the development of quantum computers, could present risks to cryptocurrencies and the CWM platform, which could result in the theft or loss of Tokens.

Token sales and ICOs have been known to come under malicious attacks from hackers and criminal parties resulting in theft of Tokens and massive losses may be inflicted on buyers and the Company

High-risk purchase

There is no guarantee that the CWM Token you purchase will increase in value. It may—and probably will at some point—decrease in value.

The activity of CWM is highly speculative, as CWM is a private and growing company with no regulatory approvals and there is no assurance such approvals, if they will be required, will be obtained or that any income shall be generated or any products shall be successfully developed.

Risk of insufficient interest in the CWM activity

It is possible that the CWM software will not be used by a large number of businesses, individuals, and other organizations and that there will be limited public interest in its creation and development. Such lack of interest could impact the development of CWM software and services and therefore the potential uses or value of Tokens.

Risk of uninsured losses

Unlike bank accounts or accounts at some other financial institutions, funds held using the CWM network are generally uninsured. In the event of loss or loss of value, there is no public insurer or private insurer to offer recourse to the purchaser.

Risk of dissolution of CWM

It is possible that, due to any number of reasons, including without limitation, the failure of business relationships or the emergence of competing intellectual property claims, CWM may no longer be a viable business and may dissolve or fail to launch.

Risk of malfunction in the GridNodes and DAP

It is possible that the GridNodes and DAP malfunction in an unfavorable way, including but not limited to one that results in the loss of data and information.

Unanticipated risks

Cryptocurrency and cryptographic tokens are a new and untested technology. In addition to the risks set forth here, there are risks that the CWM team cannot anticipate. Risks may further materialize in the form of unanticipated combinations or variations of the risks set forth here.

References

[1] K. Cukier, “Data, data everywhere,” *The Economist*, Feb-2010. [Online]. Available: http://www.economist.com/node/15557443.

[2] “IT Spending and Staffing Benchmarks: 2017/2018: IT Budget/Cost Metrics and Other Key Performance Indicators by Industry and Organization Size,” Computer Economics, 2017.

[3] “Building Trust in a Cloudy Sky,” Intel Security, Jan. 2017.

[4] “Worldwide Public Cloud Services Spending Forecast to Reach $195 Billion by 2020, According to IDC,” International Data Corporation, Aug. 2016.

[5] Synergy Research Group, Reno, and NV, “Hyperscale Operators Continue Ramping Up Share of Cloud Markets [Press Release],” Apr-2017. [Online]. Available: https://www.srgresearch.com/articles/hyperscale-operators-continue-ramping-share-cloud-markets.

[6] “Cisco Global Cloud Index: Forecast and Methodology, 2015–2020,” Cisco, Nov. 2016.

[7] International Data Corporation, “New Research Finds That 65% of Companies Are Using Cloud-Based Storage for Remote Location Disaster Recovery,” Jun-2014. [Online]. Available: https://www.acronis.com/en-us/pr/2014/07/09-14-35.html.

[8] K. Giannakouris and M. Smihily, “Cloud computing - statistics on the use by enterprises - Statistics Explained,” *Eurostat*, Dec-2016. [Online]. Available: http://ec.europa.eu/eurostat/statistics-explained/index.php/Cloud_computing_-_statistics_on_the_use_by_enterprises.

[9] SBM Group, “SMB Group’s 2017 Top 10 SMB Technology Trends,” 2017. [Online]. Available: http://www.smb-gr.com/wp-content/uploads/2017/01/2017_top_10_final.pdf.

[10] “Cost of Data Center Outages,” Ponemon Institute LLC, Jan. 2016.

[11] “Summary of the Amazon S3 Service Disruption in the Northern Virginia (US-EAST-1) Region,” *Amazon Web Services, Inc.* [Online]. Available: https://aws.amazon.com/message/41926/.

[12] P. Smith and Y. Redrup, “Amazon’s storm outages serve as a warning to businesses,” *Financial Review*, Jun-2016.

[13] J. Zander, “Update on Azure Storage Service Interruption,” *Microsoft Azure*, Nov-2014. [Online]. Available: https://azure.microsoft.com/en-us/blog/update-on-azure-storage-service-interruption/.

[14] Infosys, “Businesses Willing to Trust Mission-Critical Apps to the Cloud: Infosys Study [Press Release],” Nov-2014. [Online]. Available: https://www.infosys.com/newsroom/press-releases/Pages/mission-critical-applications-cloud.aspx.

[15] L. Tomkiw, “Amazon Web Services Experiences Outages Sunday Morning, Causing Disruptions On Netflix, Tinder, Airbnb And More,” *International Business Times*, Sep-2015.

[16] C. O’Brien, “Crash at academic cloud service Dedoose may wipe out weeks of research,” *Los Angeles Times*, May-2014.

[17] D. O’Sullivan, “The RNC Files: Inside the Largest US Voter Data Leak,” *UpGuard*, Jul-2017. [Online]. Available: https://www.upguard.com/breaches/the-rnc-files.

[18] D. Cameron, “Top Defense Contractor Left Sensitive Pentagon Files on Amazon Server With No Password [Updated],” *Gizmodo*, May-2017. [Online]. Available: http://gizmodo.com/top-defense-contractor-left-sensitive-pentagon-files-on-1795669632.

[19] S. Nichols, “AWS’s S3 outage was so bad Amazon couldn't get into its own dashboard to warn the

world," *The Register*, Mar-2017. [Online]. Available: https://www.theregister.co.uk/2017/03/01/aws_s3_outage/.
[20] Gartner, "Gartner Says Worldwide Public Cloud Services Market to Grow 18 Percent in 2017 [Press Release]," Feb-2017. [Online]. Available: http://www.gartner.com/newsroom/id/3616417.
[21] S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," *bitcoin*, Oct-2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf.
[22] V. Buterin, "On Public and Private Blockchains," *Ethereum Blog*, Aug-2015. [Online]. Available: https://blog.ethereum.org/2015/08/07/on-public-and-private-blockchains/.
[23] G. Greenspan, "Ending the bitcoin vs blockchain debate," *MultiChain*, Jul-2015. [Online]. Available: https://www.multichain.com/blog/2015/07/bitcoin-vs-blockchain-debate.
[24] Gartner, "Gartner Says the Internet of Things Installed Base Will Grow to 26 Billion Units By 2020 [Press Release]," Dec-2014. [Online]. Available: http://www.gartner.com/newsroom/id/2636073.
[25] Gartner, "Gartner Says Consumers in Mature Markets Will Use and Own Three to Four Devices By 2018 [Press Release]," Dec-2015.
[26] "Cisco Visual Networking Index: Forecast and Methodology, 2016–2021," Cisco, Jun. 2017.
[27] BLOCKCDN, "BLOCKCDN White Paper." [Online]. Available: http://www.blockcdn.org/en/file/BlockCDN%20Whitepaper%201.2.pdf.
[28] K. Kerpez, Y. Luo, and F. J. Effenberger, "Bandwidth Reduction via Localized Peer-to-Peer (P2P) Video," *International Journal of Digital Multimedia Broadcasting*, vol. 2010, Jan. 2010.
[29] K. W. Ross, J. Kangasharju, and D. Turner, "Optimizing File Availability in Peer-to-Peer Content Distribution," presented at the 26th IEEE International Conference on Computer Communications, 2007.
[30] Y. Li and M. G. Gouda, "Sources and Monitors: A Trust Model for Peer-to-Peer Networks," presented at the 17th International Conference on Computer Communications and Networks, 2008.
[31] "Storj: A Peer-to-Peer Cloud Storage Network," Dec-2016. [Online]. Available: https://storj.io/storj.pdf.
[32] T. G. Project, "The Golem Project | Crowdfunding Whitepaper," Nov-2016. [Online]. Available: https://golem.network/doc/Golemwhitepaper.pdf.
[33] A. Wagner, "Storj Crowdsale Conclusion," *Bitcoin Magazine*, Oct-2014.
[34] Smith+Crown Research Team, "Golem Token Sale (ICO): The World Supercomputer," *Smith + Crown*, 26-Sep-2016. [Online]. Available: https://www.smithandcrown.com/sale/golem/.
[35] Synergy Research Group, Reno, and NV, "Cloud & Hyperscale Growth Forecast Presents both Opportunity and Major Challenges [Press Release]," Jul-2017.
[36] "Cloud Managed Services Market by Service Type (Managed Infrastructure, Managed Network, Managed Security, Managed Data Center, and Managed Mobility Services), Deployment Type, Organization Size, Industry Vertical, and Region - Global Forecast to 2022," MarketsandMarkets, TC 3428, May 2017.
[37] Acronis, "New Research Finds That 65% of Companies Are Using Cloud-Based Storage for Remote Location Disaster Recovery [Press Release]," Jul-2014. [Online]. Available: https://www.acronis.com/en-us/pr/2014/07/09-14-35.html.
[38] "State of the SMB Cloud: 2016 U.S. Small and Medium-Sized Business Cloud Adoption Survey," International Data Corporation, Jan. 2016.
[39] Storj, "Token Migration Plan Pt.2," 08-May-2017. [Online]. Available: http://blog.storj.io/post/160448088948/token-migration-plan-pt2.
[40] E. Azar, "Golem — Building The World's Most Powerful Supercomputer… On Blockchain," *The Golem Project*, Oct-2016. [Online]. Available: https://blog.golemproject.net/golem-building-the-worlds-most-powerful-supercomputer-on-blockchain-4ccb44c328a.

[41] “Cryptocurrency and Blockchain Market Trends 2016-2022,” Infoholic Research.
[42] “Global Blockchain Technology Market 2017-2021,” Technavio, Feb. 2017.
[43] “Blockchain Technology Market - Global Industry Analysis, Size, Share, Growth, Trends, and Forecast 2016 - 2024,” Transparency Market Research, Jan. 2017.
[44] J. Dubey and V. Tokekar, “P2PCS - A Pure Peer-to-Peer Computing System for Large Scale Computation Problems,” presented at the 2011 International Conference on Computational Intelligence and Communication Networks (CICN), 2011.
[45] O. Babaoglu and M. Marzolla, “The People’s Cloud,” *IEEE Spectrum*, Oct. 2014.