Statement regarding Forward-Looking Statements

The Crowd Computer Whitepaper ("WP") is issued by Crowd Machine, Inc. ("Crowd Machine") and sets out the general business information for Crowd Machine. This notice applies to each recipient of this WP. This notice may be altered or updated. Crowd Machine does not have an obligation to amend, modify or update this WP (including this notice) or to otherwise notify a recipient thereof in the event that any matter stated herein, or any opinion, or forward-looking statement set forth herein, changes or subsequently becomes inaccurate.

This WP does not constitute any relations between the recipient of this WP and Crowd Machine. This WP does not constitute a prospectus or offer document of any sort, and is not intended to constitute an offer for securities or a solicitation for investment in securities in Crowd Machine or any related or associated company in any jurisdiction.

This WP is for information purposes only. The contents of this WP are not a financial promotion. None of the information or analyses presented are intended to form the basis for any investment decision, and no specific recommendations are intended. Therefore, none of the contents of this WP serve as an invitation or inducement to engage in any sort of investment activity.

This WP outlines the current and future developments of Crowd Machine. Unless expressly specified otherwise, the products and innovations set out in this WP are currently under development and are not currently in deployment. Crowd Machine makes no warranties or representations as to the successful development or implementation of such technologies and innovation, or achievement of any other activities noted in the WP, and disclaims any warranties implied by law or otherwise.

The information in this WP is given in good faith, but no warranties, guarantees or representations are made by Crowd Machine with regard to the accuracy, completeness or suitability of the information presented. Certain statements, estimates and information contained in this WP constitute forward-looking statements or information. Such forward-looking statements or information involve known and unknown risks and uncertainties which may cause actual events or results to differ materially from the estimates or the results implied or expressed in such forward-looking statements. Neither Crowd Machine nor its advisors have independently verified any of the information, including forward-looking statements, contained in this WP. No regulatory authority has examined or approved any of the information, including forward-looking statements, contained in this WP. No such action has been or will be taken under the laws, regulatory requirements or rules of any jurisdiction. The publication, distribution or dissemination of this WP does not imply that the applicable, regulatory requirements or rules have been complied with. Nor should this WP, or any part of it, form the basis of, or be relied upon in any connection with, any contract or commitment whatsoever.

Crowd Machine expressly disclaims any and all responsibility for any direct or consequential loss or damages of any kind whatsoever (whether foreseeable or not) arising directly or indirectly from: (i) reliance on any information contained in this WP, or any information which is made available in connection with any further enquiries, (ii) any error, omission or inaccuracy in any such information, (iii) any action resulting therefrom, or (iv) usage or acquisition of products. This disclaimer applies notwithstanding any negligence, default or lack of care.

This WP is not directed to, or intended for distribution to or use by, any person or entity who is a citizen or resident of or located in any state, country or other jurisdiction where such distribution, publication, availability or use would be contrary to law or regulation. The manner of distributing this WP may be restricted by law or regulation in certain states, countries or jurisdictions. Persons into whose possession this WP may come are required to inform themselves about and to observe such restrictions. By accessing this WP, a recipient hereof agrees to be bound by the foregoing limitations.
Table of Contents

1 Overview

2 Agoric Nodes
   2.1 Sibling Onboarding
   2.2 Publishing Pricing
   2.3 Managing a Community of ACNs
   2.4 Routing Activity Requests
   2.5 Managing Network Load
   2.6 Deployment of Activities
   2.7 Consolidation of Payments
   2.8 Proof of Trust and Proof of Worthiness Testing
   2.9 Settlement

3 Activity Compute Node
   3.1 Joining an Agoric Community
   3.2 Proof of Worthiness
   3.3 Proof of Trust
   3.4 Compute State of Readiness
   3.5 Executing a Request
   3.6 Settlement

4 Revenue, Security and Scalability
   4.1 Rewarding Crowd Computer Nodes
   4.2 Geo-locking Apps and Data
   4.3 Executing an App on the Crowd Computer
   4.4 Paying for App Execution on the Crowd Computer
   4.5 Proof of Reserve
   4.6 Crowd Computer Security
   4.7 Revenue, Security and Scalability
   4.8 Crowd Pay – A Secure, High-Speed Payment Network
   4.9 Extending the Crowd Computer

Conclusion
The Crowd Computer is a key element of Crowd Machine's vision to enable global participation in the decentralized technology revolution. It will achieve this aim by transforming the worldwide IT market with a new breed of applications offering lower development and runtime costs, and faster time-to-market.

The Crowd Computer is designed to:

1. utilize the world's spare processor and memory capacity to create a decentralized cloud network
2. execute apps built using the Crowd App Studio
3. support user-facing and real-time apps, not just batch-related solutions.

Because apps on the Crowd Computer are distributed in fragments across the network yet function in unison, the Crowd Computer can offer highly scalable processing power and network redundancy, and host sophisticated enterprise-class apps.

Enterprise-class apps usually accommodate a significant feature set and cannot be deployed onto smaller mobile devices due to their size, instead they are typically limited to centralized, server-based infrastructure.

The unique approach of building apps on the Crowd App Studio as Patterns of behavior comprised of one or more discrete logical tasks called Activities allows an app to be distributed redundantly across the network as fragments. Each app fragment can be deployed to smaller devices such as mobile phones and IoT devices yet function in unison with the other fragments.

**Figure 1.** Apps are created as sets of Patterns rather than monolithic binary executables.

The Crowd Computer's ability to deliver apps to market at speed, combined with its ease-of-use and the interoperability of its apps with any blockchain, gives it the flexibility and scalability to enable individuals and organizations to realize the industry-disrupting potential of a decentralized IT economy.
How it Works

The Crowd Computer network consists of two types of Nodes working in tandem:

1. **Agoric Node**
2. **Activity Compute Node (ACN)**

ACN owners use their devices' surplus processor and memory in the background of their normal use to execute each application Activity on the network, and Agoric Node operators are responsible for building a community of ACNs tethered to their Agoric Node. The ACN owner is rewarded by the Agoric Node operators for the use of their compute resources. Agoric Nodes form a key component of the Crowd Computer network by acting as super nodes with a range of functions, including:

- Onboarding new Agoric Nodes
- Offering competitive pricing for ACNs
- Building and managing a community of ACNs
- Routing Activity requests
- Managing network load
- Deployment of Activities
- Consolidation of Payments
- Proof of Trust (PoT) and Proof of Worthiness (PoW) testing
- Settlement

### 2.1 Sibling Onboarding

Installing a new Agoric Node in the Crowd Computer network requires an existing Agoric Node to onboard the new 'sibling', including a security verification (defined in the Security section below). Once installed, the new Agoric Node is then remotely configured by the onboarding node so it can form a direct connection with every other sibling to create an Agoric mesh network. Additionally, the new node establishes a redundant list of Activities for each app deployed on the computer in order to relay any Activity requests it is unable to execute itself.

### 2.2 Publishing Pricing

Because the price of mobile devices and data plans varies globally, the Agoric Node operators will be responsible for ensuring the price they offer for the use of ACN compute resources is sufficient to induce an ACN owner to join their community. This approach creates a free market where supply and demand dictates pricing in specific geographies. If an Agoric Node operator does not provide sufficient incentive for an ACN owner to join their community, then the ACN owner has the discretion to participate in a more competitive community. During their onboarding, the ACN will have access to the pricing being offered by each of the Agoric Nodes.
2.3 Managing a Community of ACNs

When an ACN joins an Agoric Node's community it is polled by the Agoric Node on a timed basis to ensure that it is ready to undertake Activities. If acknowledgement is not received, then the ACN is removed from the active ACN list until the ACN itself sends a notification of availability. The ACNs attached to the network execute Activities when requested by the Agoric Node. All requests to, and acknowledgements from, ACNs are managed by Agoric Nodes. Agoric Nodes also make all payments to their ACN community for the execution of Activities. They are also responsible for ensuring their ACNs comply with Crowd Machine's security and performance standards.

2.4 Routing Activity Requests

Agoric Nodes maintain a list of active ACNs and the Activities they are able to execute. When a user or process requests that an Activity is executed, the Agoric Node randomly assigns the request to an ACN that is registered as being available to undertake that particular Activity. To ensure apps perform as required, the ACN must acknowledge receipt of the request in an acceptable timeframe to be assigned the Activity, otherwise the Agoric Node will assign the request to a subsequent ACN and cancel the request sent to the prior node.

2.5 Managing Network Load

To maintain a balance between overall app execution cost, performance, and fairness of participation, Agoric Nodes share their current workload metrics so that each is aware of another's state of compute readiness and can jointly manage network load:

1. **Activity Load Balancing** – To reduce network latency and improve app performance, where an Agoric Node is experiencing heavy use, and its ability to respond to an Activity request in a timely manner is diminished, it will dispatch that request to the next nearest node that meets the performance requirements (see *Proof of Worthiness Testing* below).

2. **Income Opportunity Balancing** – To give regional or isolated geographies the same opportunity to generate income from Crowd Machine as more densely populated urban areas, Agoric Nodes relay asynchronous Activity requests (those that don’t require an immediate response) further afield to ensure balanced income opportunity across the computer network, while taking into consideration the cost of Activity execution. Synchronous Activities (those requiring an immediate response) are sent to ACNs that offer the least network latency from the point of origin.

2.6 Deployment of Activities

The Agoric Node relays the app deployment request to all other Agoric Nodes across the network and each undertakes the deployment within its own community of ACNs. The app Activities are redundantly deployed without bias toward geography, although the ACNs are selected for their ability to execute a particular Activity based upon its type and complexity. The extent to which redundancy of deployment is applied is randomized, but the process ensures coverage if Agoric Nodes or ACNs leave the network.

As an app is executed on the Crowd Computer, its points of origin are identified based upon the frequency of request. The Crowd Computer will migrate the app’s synchronous Activities (those requiring an immediate response) to those ACNs demonstrating lower latency to the points of origin, and move the asynchronous tasks further afield to balance
network load and income opportunity for Agoric Nodes and ACNs alike. To ensure apps perform as required on the Crowd Computer, ACNs are subjected to Proof of Worthiness (PoW) evaluation, and any ACN found to not meet its previous assessment may be subjected to the restriction of its Activity requests.

2.7 Consolidation of Payments

Agoric Nodes maintain a record of Activity requests sent to an ACN and whether each Activity was completed. Payments to ACNs for the completed execution of Activities are consolidated and paid by the Agoric Node on a daily basis.

2.8 Proof of Trust and Proof of Worthiness Testing

To ensure ACNs are acting in a trustworthy manner on the network and not bypassing their responsibilities to increase their throughput, Agoric Nodes randomly assign computational tasks to multiple ACNs in parallel and compare the results. If any one ACN does not match the result from all other ACNs being evaluated, then it is evicted from the network. This process enforces Proof of Trust (PoT) participation and is applied in combination with industry standard security protocols.

To ensure that an ACN can execute the Activities assigned to it, Agoric Nodes randomly send tasks of varying complexity to ACNs for execution – including when an ACN is first onboarded to the Crowd Computer. The ACN’s ability to complete those tasks in specific timeframes determines what Activities an Agoric Node will dispatch to it in the future. This evaluation process is referred to as Proof of Worthiness (PoW).

2.9 Settlement

To participate as an Agoric Node, the node operator must demonstrate that they hold a minimum reserve balance of Crowd Machine Compute Tokens (CMCT). This process is referred to as Proof of Reserve (PoR). To enforce maintenance of this reserve, the Crowd Computer will evict an Agoric Node from the Crowd Computer network if the reserve balance falls below the minimum amount required.

After an Agoric Node makes its daily payments to its community of ACNs, it submits a subsequent request for payment to Crowd Machine. Payments to Agoric Node operators are consolidated and processed every seven days and written to the blockchain.

Activity Compute Nodes are the ‘worker bees’ of the Crowd Computer network. They are responsible for executing an Activity when requested by an Agoric Node. An ACN may be any compute device including mobile phones, tablets, notebooks, IoT devices, desktop computers or data center servers.

ACNs undertake a number of functions, including:

- Joining an Agoric Community
- Proof of Worthiness (PoW)
- Proof of Trust (PoT)
- Compute State of Readiness
- Executing a Request
- Settlement
3.1 Joining an Agoric Community

To assist a device owner in choosing which Agoric Node community to join, they are offered a list of Agoric Nodes ranked according to latency and financial return. Although an Agoric Node operator may offer a particularly high return to their ACN community, if the node is too remote from the ACN then the possibility of latency is increased, which potentially results in the ACN receiving fewer Activity execution requests and consequently less income opportunity. An ACN device owner can elect to move their device to another Agoric Node community at any time, but the one device can only ever belong to a single community.

3.2 Proof of Worthiness

During installation of the ACN software, a PoW evaluation is conducted to test the device's processing potential. Minimum CPU and memory thresholds are set to determine when the device can be made available to participate in the execution of Activities. The installation then determines the type of Activity and the level of complexity that the onboarding device can undertake to ensure a minimum app performance level on the network. The PoW evaluation is also conducted whenever a device owner adjusts the threshold, and randomly by Agoric Nodes to ensure the device is compliant.

3.3 Proof of Trust

An ACN is subject to random trustworthiness assessment by its Agoric Node, which assigns a task to several ACNs in parallel and then compares the results. If an ACN does not match the results from the other ACNs being evaluated, then that ACN is deemed to be acting in a malicious manner and is evicted from the network.

3.4 Compute State of Readiness

To ensure an ACN is available to fulfil Activity execution requests, each ACN reports its state of readiness on a timed basis to its Agoric Node. ACNs that do not report their state of readiness or are deemed to incur enough latency to result in poor app performance are excluded from participation. However, latency is only evaluated in those instances where a synchronous Activity (one requiring an immediate response) is being requested.

3.5 Executing a Request

An ACN acknowledges the request to execute an Activity from its Agoric Node before the Activity is executed. Where a Pattern consists of a series of synchronous Activities (one requiring an immediate response), the ACN will execute each Activity in sequence – once its PoW has determined that it can execute the Activity type and its degree of complexity. The ACN reports completion of the task to the Agoric Node that originated the request.

3.6 Settlement

Payment for ACN compute resources is managed and processed daily by the relevant Agoric Node. Each payment made to an ACN owner is written to the blockchain. The amount paid is the price offered at ACN onboarding by the Agoric Node owner.
The Crowd Computer has a number of features and functions not exclusive to either Agoric Nodes or Activity Compute Nodes, which are essential to the overall operation of the system:

- Rewarding Crowd Computer nodes
- Geo-locking apps and data
- Executing an app on the Crowd Computer
- Paying for app execution on the Crowd Computer
- Proof of Reserve
- Crowd Computer security
- Redundancy and scalability
- Crowd Pay – a secure, high-speed payment network
- Extending the Crowd Computer

### 4.1 Rewarding Crowd Computer Nodes

The Crowd Computer utilizes the combination of Agoric Nodes and Activity Compute Nodes (ACNs) to run real-time decentralized apps on a pay-as-you-go model using Crowd Machine Compute Tokens (CMCT). CMCTs function as the ‘gas’ to power the apps running on the computer. The amount of compute resource required to execute apps depends upon its complexity or its number of users. The more heavily used or compute intensive the app, the more CMCT is required to execute it.

Crowd Machine does not set the price for Crowd Computer resources. The price is set by Agoric Node operators and is subject to the normal laws of supply and demand. The Crowd Computer always seeks to optimize price performance but utilizes the geographical distribution of Agoric Nodes and their community of ACNs to balance compute workloads. See the above section ‘Managing Network Load’ for more information. Using the Crowd Machine model, network latency is less of a barrier to compute-based wealth generation.

If an Agoric Node operator establishes a price for their ACN community’s services that is not competitive with other operators (taking into consideration synchronous and asynchronous distribution of Activities as described above) then they are not likely to receive Activity execution requests. Furthermore, the Agoric Node’s ACNs device owners will migrate to an alternative node where they can better monetize their devices. Conversely, if an Agoric Node establishes a price which is too low, then the node will be overwhelmed, resulting in the Activities being relayed for execution to alternative nodes offering a higher price.

### 4.2 Geo-locking Apps and Data

Apps and their associated data can be locked to specific geographical territories by the app engineer at any time during or after deployment. During the initial deployment process, the deployment request is only sent to those Agoric Nodes that meet the geographical requirement. After the initial deployment, a request is sent to each Agoric Node residing outside those geographies to have them remove the specific app's Activities from their community of ACNs. Any ACN that transits from its original network and is found to be no longer located in its original geography is excluded from executing any Activities restricted from that geography.
4.3 Executing an App on the Crowd Computer

All apps exist as a combination of sets of behaviors with varying levels of complexity. When an app engineer builds an app on the Crowd Computer and deploys it for execution, it is redundantly distributed as Activities to ACNs via Agoric Nodes. Apps exist on the Crowd Computer as linked sets of Activities, and synchronous Activities in the same Pattern are deployed, where possible, to the same ACN.

Figure 2. Apps on the Crowd Computer are distributed in fragments, replicated across the network. Agoric Nodes jointly manage network load to shift processing to where it's most performant and cost effective.

As an app is executed, a request is made to an Agoric Node in the form of a URL and can be made to any node on the network. A request can be made by a user, an external process or even another Activity in the same or another app built with the Crowd App Studio.

Given that apps are made up of Patterns and Patterns are made up of Activities, then upon receipt of the request to execute an app, an Agoric Node will identify an ACN within its community that is ready to execute the first Activity in the Pattern. The Agoric Node will dispatch a request to the ACN to execute the Activity. If the Activity following the previous Activity is synchronously linked, then the ACN will continue to execute each Activity in the sequence until it either completes the Pattern execution or the next Activity is not in its library. At this point, the ACN concludes its work and returns control back to the Agoric Node.

The Agoric Node notes that the work has been completed by the ACN and records an entry in its ledger for payment. In parallel to this process, the Agoric Node, if appropriate, will either launch the next Activity to be executed or respond to the originating request point. The execution of Activities can occur in parallel on the Crowd Computer to allow apps to run multiple threads simultaneously.
4.4 Paying for App Execution on the Crowd Computer

Using the Crowd App Studio, anyone can build and test apps for free on the Crowd Computer. They only pay when the app is deployed to production. To execute an app in production, the app engineer must purchase Crowd Machine Compute Tokens (CMCT) in advance. The CMCTs act as gas to power apps on the Crowd Computer. The Agoric Node operator pays the ACNs in their community that participated in executing app Activities, and Crowd Machine subsequently pays the Agoric Node operator a fee, inclusive of payments to ACNs. The amount that the app engineer pays for app execution is dependent upon the pricing set by the Agoric Node (as outlined above in ‘Rewarding Crowd Computer Nodes’).

![Figure 3. App owners buy Crowd Machine Compute Tokens (CMCT) from Crowd Machine to run their apps on the Crowd Computer. Agoric Nodes pay their network of attached devices CMCT to run the apps, and are then reimbursed with CMCT by Crowd Machine.](image)

As the number of Agoric Nodes and apps running on the Crowd Machine network increases, demand for CMCT will increase and circulating supply will further reduce. When the price of CMCT required to pay for app and Agoric Node reserves approaches the price of competitive alternatives available to app owners, more CMCT will be released to keep the price below those alternatives. The amount of CMCT released will be contingent upon maintaining that price point. This competitive pricing model promotes product adoption resulting in increased CMCT liquidity.
4.5 Proof of Reserve

The Crowd Computer Proof of Reserve (PoR) model requires Agoric Node operators to pre-purchase a CMCT reserve before their node is operational to ensure they can pay their community of ACNs before receiving payment themselves. The requirement to maintain a minimum reserve of CMCT at all times and enforce an upfront financial commitment to the Crowd Computer is a strong disincentive to exploit the network.

Another disincentive is that the fact that asynchronous tasks (those not requiring immediate execution) are dispatched to alternate nodes removed from more densely populated points of origin. Most apps are likely to have a balance of asynchronous to synchronous Activities, so even if Agoric Node owners create their own field of ACNs removed from the Crowd Machine community, the wide distribution of Activities makes gaming the network less attractive.

4.6 Crowd Computer Security

The Crowd Computer features a ‘defense in depth’ approach to security. In addition to the PoR, PoT and PoW implementations (described above), the Crowd Computer employs industry standard protocols layered over a series of proprietary algorithms to guard against malicious behavior. All Agoric Nodes run DDoS mitigation and intruder detection software. All network transmissions are encrypted using the latest versions of their respective underpinning protocols. External providers are engaged to undertake penetration testing of the network technology.

When onboarding an Agoric Node, numerous measures are undertaken to mitigate the risk of introducing a Trojan actor to the network. In the first instance, the node trying to join the network must undertake a series of exchanges with established nodes using rotating hashing algorithms. Thereafter, any transmission between nodes on the Crowd Computer requires continuously changing hashing algorithms with key secured agreement on the algorithm between parties. The result is a constantly evolving and self-managed defense mechanism that reduces the risk of network attacks.

ACNs wishing to join the Crowd Computer are subject to the same onboarding approaches as an Agoric Node. However, the Agoric Node to which the ACN is joining manages the onboarding process. Additionally, (as described above) ACNs are randomly evaluated using the Crowd Machine PoT algorithm to ensure their ongoing integrity.

4.7 Revenue, Security and Scalability

Due to the nature in which apps are deployed to the Crowd Computer, numerous devices can act as compute nodes on behalf of an app. The app’s degree of complexity can vary from very simple to extremely large and complex solutions, and yet exist on small and large devices alike. Additionally, the nature of the technology allows the Crowd Machine to scale on-demand to meet any compute load.

When an app engineer wishes to deploy their app to the Crowd Machine network, the Crowd App Studio generates entry point URLs for each Agoric Node on the network. If an Agoric Node is unavailable, the process allows an alternate Agoric Node URL to be used to launch the app. To ensure that apps on the Crowd Computer are always available, all apps are redundantly deployed to leverage the participating nodes’ geographical distribution.

Combined with the rewards mechanism to encourage participation, this approach results in the ability for all mobile phones, IoT devices, notebooks, desktop and server compute devices to act in unison to form the world’s largest decentralized app processing computer. The Crowd Computer’s decentralized nature, and its ability to leverage surplus public compute resources, means it can offer industry-leading scalability and redundancy by design.
4.8 Crowd Pay – A Secure, High-Speed Payment Network

Crowd Machine technology is in production with a growing number of Fortune 500 companies, several of which are in the banking sector and use a complex asset-based lending solution created and deployed using Crowd Machine technology.

The Crowd Computer will soon use the ACN software to offer access to a high-speed, peer-to-peer payment network that uses a 100 percent commodity-backed crypto-coin to conduct credit card-like payments without any network transaction fee.

Crowd Machine anticipates its community will create many apps to leverage the Crowd Pay network, and already offers integration into several commercial payment gateways. By using the Crowd App Studio, app developers will be able to quickly create secure and highly integrated apps that leverage the payment network to offer financial services and payment-based solutions to their customers.

4.9 Extending the Crowd Computer

The Crowd Computer is not limited to its native function. It can be extended by third parties to offer specialized functionality created using either the Crowd App Studio or traditional development tools and integrated into the Crowd Computer for private use or public consumption. Furthermore, the Crowd Computer offers RESTful integration into any external service and can also be used as a ‘compute as a service’ platform where external products can execute behaviors that are expressed as RESTful APIs, resulting in an infinitely extensible compute platform that meets any app requirement.

Conclusion

The new kind of ‘crypto-economy’ supported by decentralized infrastructure like the Crowd Computer exists as a complex dynamic between app owners, developers, device owners, app consumers and the system that binds them all together. They each have a personal stake in the shared enterprise, collecting rewards for their contribution through the flow of tokens, with trust guaranteed by the immutability of blockchain-based records.

The infrastructure for this economy already exists and is growing rapidly: the myriad smartphones, tablets, laptops, desktop PCs, servers and IoT devices. The participants are already hard at work, but siloed, without the necessary platform to bring them together in cooperation.

We have already seen in the past decade the rapid shift to a centralized model of infrastructure outsourced to the cloud and the new industries created by the prevalence of mobile devices. Networks like the Crowd Computer are the logical next step in the evolution of global IT, and Crowd Machine is one of the pioneers that will clear the way for the early adopters.

Today it’s not a question of if the move to a decentralized model of IT will become mainstream, but when.