zer0

A Social Operating System

n3o

Abstract:

zer0 (Zero) is a peer-to-peer social network and operating system. A necessary and timely alternative to centralized social networks and messaging platforms like Facebook, Twitter, YouTube and WhatsApp. It enables social communication, remote collaboration, financial transactions and <u>onchain</u> governance to occur directly between content-creators, developers, and users (hereinafter referred to as Members), independent of third parties. This architecture removes the possibility of censorship at the level of code, significantly reducing the potential for security breaches and data violations. Zero moves data ownership, platform incentives and governance to individual Members, communities and decentralized autonomous organizations (DAOs). Zero is operated with Infinity, its native protocol token, which enables Members to buy and sell hosting power in the Zero Grid, a distributed peer-to-peer network run entirely by Zero Members. Similar to Bitcoin mining, Infinity creates a powerful cryptographic incentive for Members to secure, maintain, and expand the Zero Grid, creating long-term value for network participants. Zero Open System (zOS), including the Zero Name Service (ZNS), zDAO, zChain, Ethereum and Filecoin/IPFS are the primary technologies that comprise the Zero Stack. The result is a fully decentralized, hyper-scalable, and self-governing alternative to centralized social networks, societal systems, and internet platforms.



Contents/

1.	Introduction	()4						
2.	Principles)5						
3.	-	iderations(
4.	•	Zero Open System:							
т.		Layer	1.0						
	4.1. Data 1	•	1 4						
	4.1.1. 4.1.2.	Identity							
	4.1.2.	Node Message							
	4.1.3. 4.1.4.	zChain							
	4.1.4. 4.1.5.	Account & Avatar							
	4.1.6.	Network							
	4.1.7.	Cluster							
	4.1.8.	Peer Routing & Discovery							
	4.1.9.	Gossip							
	4.1.10.	zApp							
	4	1.1.10.1. Defining a zApp							
	4	1.1.10.2. zApp Store							
	4.1.11.	Dyno							
	4.2. Curre	ncy Layer2	22						
	4.2.1.	Token							
	4.2.2.	Exchange Token							
	4.2.3.	Dynamic Token							
	4.2.4.	The Infinity Token							
	4.2.5.	Credit							
	4.2.6.	The Creditor							
	4.2.7.	Dyno Matcher							
	4.3. Trust	Layer	33						
	4.3.1.	Zero Name Service (ZNS) Protocol							
	4.3.1	.1. Addressing							
	4.3.1	.2. Staking & Governance							
	4.3.1								
	4.3.1	ε							
	4.3.1								
	4.3.1								
	4.3.2.	zDAO Protocol							
	4.3.3.	DAO							
	4.3.4.	DAO Token							
	4.3.5. 4.3.6.	Omega Token DAO Mind							
	4.3.6. 4.3.7.	Choice							
	4.3.8.	Neuron							
	4.3.9.	Voting Engine							
	4.3.10.	Task							
	4.3.11.	Reputation							
	4.3.12.	Trinity							
	4.3.13.	InnerSource							
5.			18						
٥.			50						
	J. 1. IIIIIIII	i y Commigui un Oll	v						

	5.2. Tok	en Distribution	52
6.	Deployment	t	53
7.			
		Design Inspiration.	
	1.A.1.		
	1.A.2.	<u> </u>	
	1.A.3.		
	1.A.4.	. Effective Choice	
	1.B.	The Future Yesterday	62
	1.B.1.	. Introduction	
	1.B.2.	. Web 2.0	
	1.B.3.	. The Sharing Economy	
	1.B.4.	. Digital Productivity Tools	
	1.B.5.	. Automation and Asymmetry	
	1.B.6.	. Data Privacy & Security	
	1.B.7.	. Internet 3.0	
	1.B.8.	. Blockchain	
	1.B.9.	. DAOs	
	1.B.10.	. Market Makers & DEXes	
	1.B.11.	. Agent-Centric Systems	
	1.B.12.	. Immersive Reality	
	1.C.	The Zero Constitution.	76
	1.D.	InnerSource	78
	1.D.1.	. Introduction	
	1.D.2.	. License Options	
	1.D.3.	. InnerSource A	
	1.D.4.	. InnerSource B	

Free Your Mind

1.0 **Introduction**/

Digital surveillance, mass censorship, individual privacy violations, geopolitical interference, and fake news are examples of the existential challenges modern social networks and societies face. The combination of exponential tech and asymmetric incentives between platform Members and owners have led to a <u>broken attention economy</u>. The future depends on a viable alternative to current internet platforms and 'big tech'. Bitcoin, Ethereum and other blockchain architectures have proven the ability to transact crypto-assets worth tens of billions of dollars, independent of intermediaries and trusted third parties. The next major leap will be to apply these decentralized architectures to applications and societal structures that require 'internet-scale', where platform governance and economic incentives can be redistributed to those directly responsible for the creation of digital wealth.

Zero eliminates the core problematic incentives used by current internet platforms and social networks, which are designed to manipulate human attention and choice making, with advertising, gamification and opaque algorithms. Zero asserts that these problems are symptoms of a deeper structural issue that exists at the level of corporate fiduciary responsibility and code. Namely, the incentive and information asymmetry created by the legal structure of corporations and cloud-only architectures, and the resulting rent-seeking that occurs from the exchange of human attention. What is needed is a transparent structure that enables Members to retain control of their data, personal privacy and online experience, while sharing in both choice making (platform governance) and the economic upside that is generated as a result of individual and collective participation.

Zero is a community-governed alternative to traditional social networks and internet platforms that is based on open, distributed and transparent protocols. By running the Zero App on a desktop or mobile device, members receive Credits in exchange for computational resources and network bandwidth that power the Zero Grid. Credits can be redeemed for Infinity, Zero's ERC-20 token, and profits are then distributed to a shared set of community DAOs responsible for building and maintaining the Zero Grid.

The Zero Stack is at varying levels of production readiness and has been in active research and development since 2015. Prior to 2021, Zero will deploy Infinity, the Zero Name Service (ZNS), and The Zero Token System to the Ethereum Blockchain, as part of its first major public release. Zero plans to achieve full-decentralization by December 2022, as part of a four-phase release and deployment schedule that is outlined in **Section 6, Deployment**.

• • • • •	 	 	· · ·	 	 • • •	• • •	 	 	 	• • •

2.0 Principles/

Zero is designed with four core principles—Freedom, Privacy, Choice, and Stewardship:

- 1. **Freedom:** It is not possible to censor or manipulate chain entries within Zero's base-level protocols. Members maintain full ownership over their identities, personal data and private keys.
- 2. **Privacy:** All private chain entries are secured using AES256 encryption. Private keys are stored on local devices, not on central servers. Communication and collaboration can live within public (join-only) or private (invite-only) namespaces called Zero Networks.
- 3. Choice: Members and Networks have the ability to customize Zero's interface, develop custom extensions and applications at various levels of the Zero Stack. Members and Networks can select, build and configure social algorithms, reputation systems, tokenomics, and governance protocols. Zero is designed to work on the web, Mac, Windows, and Linux desktop environments, as well as iOS and Android.
- 4. **Stewardship:** No corporate entity sits between Members and their right to participate in Zero's platform stewardship. Zero is governed by InnerSource, a novel crypto-economic open-source license outlined in **Appendix D**, a founding constitution, and a collection of three community-operated DAOs referred to as Trinity.

• • • • • • • • • • • • • • • • • • • •	 	

3.0 Primary Considerations/

Zero is designed to be simultaneously *agent-centric*, *meta-consistent*, *self-healing*, and *trust-full*. These are the four primary considerations that make Zero unique when compared to purely cloud-based or blockchain maximalist architectures:

• First, Zero is 'agent-centric', which enables the preservation of data integrity across an unlimited number of peers (also referred to as Zero Nodes), without the need for global consensus. Given that no central server or shared ledger exists, Zero Nodes exchange information directly by connecting to peers through a gossip protocol. Zero Nodes store data in their own private and local append-only chain called a zChain. In aggregate, all zChains and Nodes in the network make up the Zero Grid. Within agent-centric architectures, each peer has its own unique (subjective) view of reality, compared to data-centric architectures such as blockchains, where peers require global consensus to maintain a shared (objective) view of reality. Zero's agent-centric (distributed) architecture enables each peer to operate its own individual zChain, which is then replicated across *n* nodes using eventual consistency Collectively, this architecture gives rise to the collective parallel processing capacity of the Zero Grid. Git, IPFS, SSB and Holochain are notable examples of agent-centric architectures in practice.

- Second, Zero is 'meta-consistent'. zChain enables Zero to act as a central repository for credentials and data stored in existing web3 systems and protocols. Zero does not aim to compete with traditional blockchains, and is designed to integrate both agent-centric and data-centric approaches into a single application stack called the Zero Open System (or zOS for short). For instance, while financial transactions are well suited for a shared global ledger like Ethereum, global consensus is computationally inefficient for use-cases such as sending private messages between a limited number of peers. Zero's 'meta-consistent' approach enables developers to use both 'agent-centric' and 'data-centric' approaches depending on the use-case, in a way that still prioritizes speed, security and decentralization. Zero can be thought of as the ultimate abstraction layer for the web3 ecosystem, not dissimilar to how desktop operating systems once provided a novel interface for command prompts, or how the browser provided a visual way to explore the world wide web.
- Third, Zero is 'self-healing' (and apocalypse-ready), meaning that a large portion of the system can function offline, on a local area network, or sneakernet. Zero's network transport layer is agnostic, meaning that various protocols can be used to transmit data between Nodes on different types of networks. In the event of partial internet failure, short-term disconnection, or complete loss of access, Zero intelligently propogrates new data entries across peers as they come online in what mimics a self-healing network.
- Fourth, Zero is 'trust-full', rather than purely 'trustless', and utilizes a Members social graph to determine trusted peers to replicate information from within the Zero Grid. One of the primary inventions provided by blockchain is the enablement of transactions to occur in an entirely 'trustless' fashion. This is a good catch-all where data-centric approaches are required, however highly inefficient where 'trust' is already established and can be used as a proxy to find more efficient paths for data propagation (providing of course, that decentralization is not compromised). Trust in distributed systems is a powerful concept that can be used to significantly enhance Member experience not only at the level of data exchange, but also at the level of Member experience across a wide array of applications. Imagine if any app on your mobile device automatically knew which people and devices you already trusted, without having to add new contacts, friends, or followers each and every time a new application was downloaded. Providing Member consent, Zero applications can access a Member's 'trust-graph'. Architecturally it makes sense to have trust data at the protocol level, rather than being siloed within individual applications. This opens a range of trust-related features and capabilities for applications built on top of zOS.

.....

4.0 Zero Open System/

The Zero Open System (zOS) is a fast, secure and scalable peer-to-peer application protocol, development stack and interface layer capable of handling millions of transactions per second. This scale is achieved by enabling parallel processing across Nodes and different web3 systems. Individual Nodes that make up the Zero Grid run zOS when launching the Zero desktop, mobile or web application. zOS makes extensive use of existing peer-to-peer and cryptographic protocols. Most notably, zOS leverages libp2p, Hypercore, SSB, Ethereum, IPFS and Filecoin as foundational protocols. Many of the concepts and ideas herein are both an inspiration and synthesis of existing concepts within these protocols and other successful distributed systems like BitTorrent, Git, and Bitcoin. Additionally, zOS integrates with a growing number of third-party web3 protocols and libraries, including Wallet Connect, Gnosis, DAOStack, Bancor, Synthetix, MetaMask and others. The core modules that make up zOS are defined and described in the logical sequence below, with each module serving as a foundational building-block for each subsequent module.

The zOS core system is made up of three interconnected layers; Data, Currency and Trust:

- 1. **Data Layer:** Responsible for storing, securing, and distributing Messages between Nodes in the Zero Grid.
- 2. **Currency Layer:** Responsible for the calculation, allocation, and exchange of Credits and Infinity Tokens, as well as the creation of Exchange Tokens for Zero Networks.
- 3. **Trust Layer:** Responsible for global namespaces and the management of DAO's within the Zero eco-system, including InnerSource and Trinity.

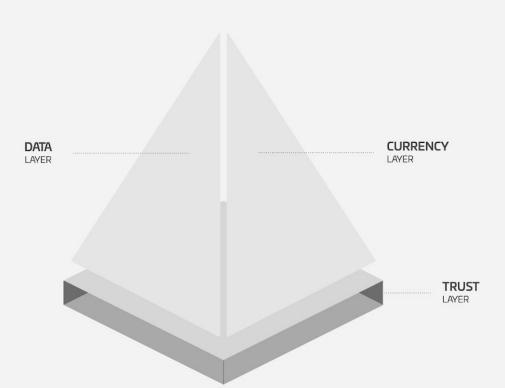


Figure: The Zero Stack

The **Data Layer** is comprised of eleven core modules:

- 1. **Identity:** A unique peer identification, generally assigned to a single Member or device.
- 2. **Node:** A peer in the network that establishes Transports, stream multiplexing and content routing.
- 3. **Message:** A canonical format and signature structure for storing system state and constructing a zChain.
- 4. **zChain:** A chain of signed Messages that is secured by a Merkle tree.
- 5. **Account & Avatar:** A representation of account information, with the enabling capacity to combine Identities and multiple devices under a parent zChain.
- 6. **Network:** A public or private namespace that is shared by Accounts or Identities.
- 7. Clusters: A Cluster that represents two or more linked Networks.
- 8. **Peer Routing and Discovery:** A shared format to ensure consistent addressing and data routing between Nodes within the Zero Grid.

- 9. Gossip: A peer-to-peer protocol used to share messages between Nodes within the Zero Grid.
- 10. **zApp:** A collection of one or more custom Message format and associated validation logic that enables the development and interoperability of custom Zero applications.
- 11. **Dyno:** A peer-to-peer service (e.g. DAEMON) that peers can run and provide to the Zero Grid.

The **Currency Layer** is comprised of seven core modules:

- 1. **Token:** An Ethereum or blockchain-based cryptographic token capable of transacting within the Infinite Economy.
- 2. **Exchange Token:** A Token that is priced through a bonding curve and backed by an underlying Reserve asset.
- 3. **Dynamic Token:** An Exchange Token with a Dynamic Reserve Ratio.
- 4. **Infinity & The Infinite Economy:** A Dynamic Token that powers Zero and the Infinite Economy, which represents all Tokens and associated transactions within Zero.
- 5. **Credit:** The atomic unit (AU) of hosting power allocated to Members and service providers within the Zero Grid.
- 6. **Creditor:** A mechanism that allocates hosting credits and debits in exchange for the operation of Nodes and Dynos.
- 7. **Dyno Matcher:** A mechanism to reliably match Members with Dynos for particular use-cases in the Zero Grid.

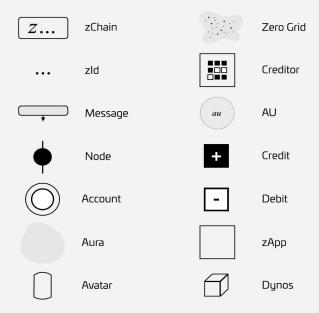
The **Trust Layer** is comprised of eleven core modules:

- 1. **ZNS:** The Zero Name Service is the set of smart contracts that enable ownership and traversal of global routes across the Zero Stack.
- 2. **zDAO**: A Distributed Autonomous Organization management protocol and set of smart contracts.
- 3. **zDAO Token:** An exchangeable economic Token controlled by a DAO.
- 4. **Omega Token:** A non-exchangeable Token issued to indicate voting power within a DAO.
- 5. **DAO Mind:** The primary contract for managing permissions and actions within a DAO.
- 6. **Choice:** A Choice that directs a DAO to execute an action, such as the issuance of a DAOs Token or Omega.
- 7. **Neuron:** A used-defined module of autonomous code that can be added and managed by a DAO.
- 8. Voting Engine: A mechanism to determine how votes are counted and Choices are passed.
- 9. **Task:** A unit of work voted on and compensated by the DAO.
- 10. **Reputation:** A mechanism to determine an Individual's global Reputation in Zero relative to skills, characteristics and behaviors across domains and DAOs.
- 11. **Trinity:** The constitutional framework and set of three DAOs responsible for governing Zero, as outlined in **Appendix C** and **B**.
- **12**. **InnerSource:** The crypto-economic open-source license that governs the Zero source-code.

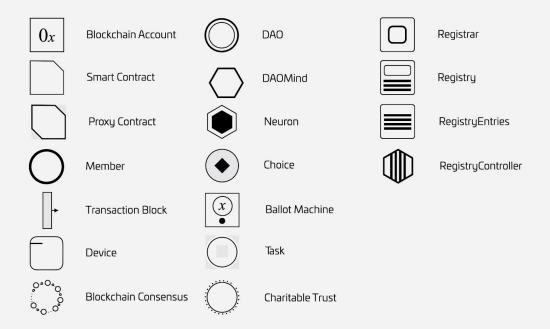
In addition to the above mentioned layers, Zero's own visual language is utilized throughout this document to make concepts easier to grasp, and to more easily show the inter-relatedness of different ideas, systems, and sub-systems. Given that Zero uses both agent-centric and data-centric approaches, it is

important to draw as distinct comparisons as possible between different approaches. Zero's visual language is organized into three categories: 1) Peer-to-Peer, 2) Tokens, and 3) Blockchain Symbols.

4.0.1 Peer-to-Peer Symbols/



4.0.2 Blockchain Symbols/



4.0.3 Blockchain Token Symbols/



.....

4.1.0 Data Layer/

4.1.1 **Identity**/

Identities in Zero are represented by a Zero Identity, or zId. A zId is a unique reference to a Node in the Zero Grid. zId's are the result of a cryptographic hash that is applied to a Node's public key. zId's utilize libp2p's <u>multihash</u> format and are constructed using a <u>base58-encoding</u>. This enables the flexible upgrading of protocol standards overtime and supports backwards compatibility. Members can generate multiple Identities per local device. A new Identity is generated by calling **generateIdentity()**, which automatically saves the keypair in a Member's home directory **.0/secret**. New Identities automatically spawn a new zChain, as outlined in **Section 4.1.4, zChain.**

4.1.2 **Node/**

A Node in Zero represents an actively running peer in the Zero Grid. Nodes establish an active Transport on a local device that is responsible for managing network traffic over specific Internet protocols such as <u>TCP</u>, <u>UDP</u>, <u>WebRTC</u> and <u>WebSockets</u>. Transports utilize the <u>libp2p-transport</u> protocol along with its standard <u>multiaddress convention</u>. E.g. /ip4/9.9.9.9/tcp/369.

Once a Node is up and running, it is capable of **listening** and **dialing** with other peers in the Zero Grid. Listening enables the Node to establish a connection for incoming traffic, whereas dialing establishes an outgoing connection, over the specified Transport layer. A Node's Transport connection is set as WebRTC by default, however is capable of supporting TCP, UDP, Websockets and other custom network protocols. Standardization on WebRTC ensures both browser-based applications and platform binaries (such as desktop or Electron apps) can easily exchange network data with one another.

Stream multiplexing (often referred to as 'muxing') is also enabled by default using the <u>mplex</u> library in order to reduce the connection overhead associated with opening multiple concurrent connections. Muxing encases connections into a single Transport medium, reducing the number of connections that need to be opened and closed at a given time, while ensuring that <u>NAT traversal</u> only needs to happen once.

Transports are encrypted by default using the <u>Noise Protocol Framework via libp2p-noise and libp2p-secio</u>, which form a <u>Diffie-Hellman key</u> agreement between peers. SECIO is a <u>TLS 1.3</u> like channel to establish an encrypted connection between two peers that was originally developed for IPFS.

4.1.3 Message/

In a zChain, state is made up of **Message** objects. Messages are the canonical format for storing and sending data between Nodes in the Zero Grid.

A Zero Message contains six fields and a signature:

- hash: a hash of a prior Message contents, including the Message signature.
- **creator:** a valid zId <u>base58-encoding</u>
- **chain:** a valid reference to a zChainId (as defined in the following section) or supported third party chain, such as Bitcoin, Ethereum, or Filecoin, utilizing zChain's multihash format.
- **seq:** the Message sequence in linear time; e.g. 0, 1, 2... etc.
- **timestamp:** the time of Message creation, represented by the number of milliseconds since 1 January 1970 00:00 UTC.
- **content:** for unencrypted data, a public JSON data field with a required type field.
- **signature:** a digital signature created from a public and private key.

New messages are signed by an Identities private key. This obfuscates Message contents and meta-data prior to being propagated to Nodes within the Zero Grid. The obscured Message disguises the content and data to ensure privacy is protected.

4.1.4 **zChain**/

Agent-centric data in Zero is stored in a local encrypted and append-only log called a zChain. zChain is designed to serve as an abstraction layer for multiple web3 protocols, including but not limited to Hypercore, Ethereum, and IPFS.

zChain is complementary to blockchain protocols and can be thought of as a 'multi-blockchain', or 'meta-chain', that aggregates identities, authentication keys, and chain data that originates outside of zOS. This enables Members to integrate third-party services, chains and dApps into a single location while maintaining data integrity, security and decentralization.

A zChain is a chain of Messages. Each Message contains a hash as a reference to the prior Message in the chain, that can be used to reconstruct the chain back to the initial Message. Each zChain maintains a unique public key for identification (zId) and holds a private key for Message signatures. zChains utilize the append-only log from the <u>Hypercore library</u>.

privateKey Message n+1 publicKey hash creator chain timestamp seq content signature ■ hash ■ Z ... privateKey Message n + 2 publicKey hash creator chain timestamp content seq signature ■ hash

Figure: zChain Message Structure

zChains are verified through a Merkle tree that utilizes the <u>BLAKE2b-256</u> hash function. This append-only structure enables Nodes to only download relevant portions of other Node zChains to their local machine, while cryptographically proving that the chain has not been tampered with.

Asymmetric cryptography is used to sign the root of a merkle tree when Messages are added to a zChain. Unlike typical append-only logs and blockchains, the introduction of asymmetric cryptography enables historical Messages to remain verifiable and be modified in the future by the owner of the private key. This is particularly useful for typical web applications where historical content editing is often required.

zChain is designed to serve as a complement and extension to existing Blockchain systems, such as Ethereum, Cardano or Polkadot. zChain can be thought of as a local blockchain that runs on a Member's device that is agent-centric, compared to existing blockchains where data is object-centric. Together, these two design patterns elegantly support the handling of local data, such as private messages and feed posts, and global data, such as namespaces and financial transactions, respectively.

The following diagram illustrates the relationship of between multiple zChains and a Blockchain:

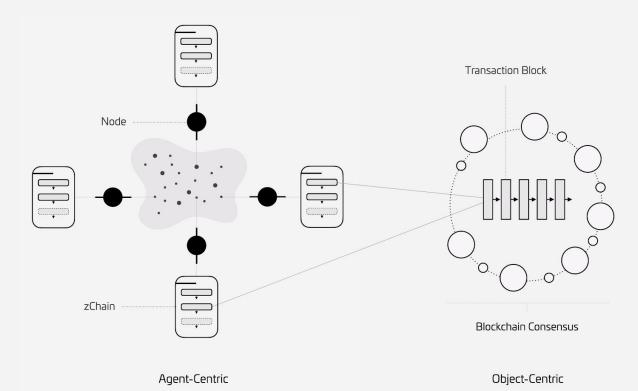


Figure: zChain and Blockchain

4.1.5 Accounts & Avatars/

4.1.5.1 **Accounts**/

Accounts enable Members to connect multiple Identities (devices or Nodes) through the use of a parent Identity. Accounts are created by having two or more Identities each sign an encrypted Message of type 'connectIdentity'. The successful execution of this method generates a new Identity, associated zId and Message for each signing zChain, cryptographically linking the chains together. The newly created Message content and type is identical between chains, and specify the newly appointed parent zId.

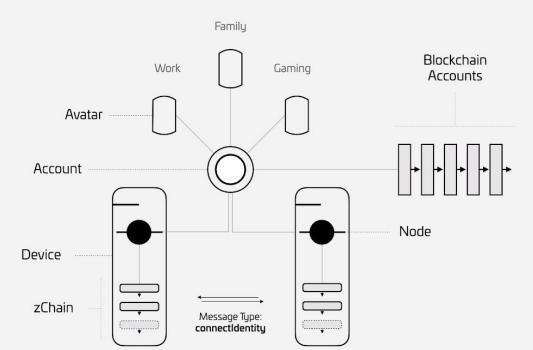


Figure: *Identities and Accounts*

Applications can retrieve a Member's linked zChains by performing a query for the most recent 'connectIdentity' Message type from any of an Account's linked chains. This enables developers to construct a Member's combined application across zChains and devices.

4.1.5.2 Avatars/

An Avatar enables a Member to decouple their public facing identifiers (such as a Member handle or earned reputation score) from linked Identities and zChains. Like Identities, Avatars are unique, and serve as the primary external mechanism for a Node to reference a Member Identity or Account. Avatars enable the Member to maintain multiple external facing personas, without the requirement to publicly disclose associated Identities, zChains or Avatars that are linked to their Account.

This enables a Member to maintain different Identities (and respective zChains) for different use-cases (such as having a separate work Avatar or personal Avatar), while maintaining a single and private set of system access credentials. Members may generate as many Avatars as they like, however can not associate an Identity with more than a single Avatar at a time. An Avatar must be selected when posting a new Message to a zChain.

Avatars are generated with a Message of type **Avatar**, and require five fields:

- ald: a unique Avatar Id
- **zIdHash:** An encrypted hash of the Avatar's associated zId. This can only be verified by the holder of the private key itself, and is checked when creating a new Message for a specific Avatar. This can include a zId that belongs to either a parent, child or standalone zChain.
- **photo:** a profile image
- banner: A background image
- **publicHandle:** a unique handle issued by ZNS, the Zero Name Service.

Globally unique Member handles for Avatars can be purchased using ZNS, as outlined in **Section 4.3.1. Zero Name Service (ZNS) Protocol**.

416 Networks/

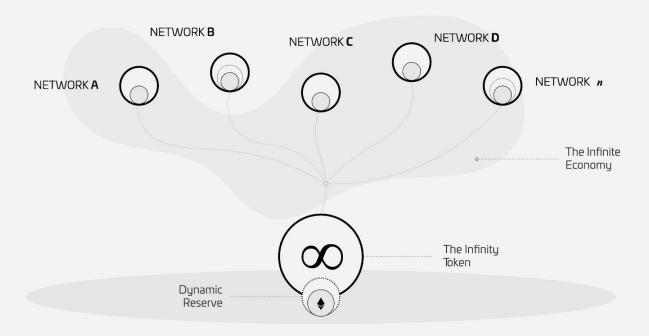
Networks in Zero represent a shared namespace that can be set as public or private. Public Networks can be joined automatically, whereas private Networks require an invitation. All Messages in public Networks are readable as raw text, whereas Messages in private Networks are encrypted. Individuals that belong to a Network are called Network Members.

Creating a Network in Zero spawns a new zChain and unique zId. Responsibilities are shared by maintaining a list of authorized Actions for each Network Member within the Network's zChain. This is stored inside of a mapping array within the content of a Message. Each Action represents something a Member can or cannot do within the Network. Individual Actions map directly to Message types. For instance, only Members that are permitted to use 'action::invite', are able to append Messages of type 'invite' to the Network's zChain. If a Member attempts to sign a Message of type 'invite' and does not have the corresponding Action and zId in the Network's permissions map, the entry will fail and return an error.

One of the benefits of having a distinct zChain for each Network, rather than including Network Transactions within a single zChain, is to provide an additional level of efficacy for Message propagation. This architecture enables a Network to choose to only gossip and store Messages with Nodes who are Members in the Network

If a Member leaves a Network, their access is simply removed by adding a new Message to the Network's zChain, and freeing up all disk space on the Member's local device that was associated with the Network. When a Member leaves a Network, the local Network chain is automatically destroyed and is no longer propagated to other Members of the Network.

Figure: *The Infinite Economy*



4.1.7 Clusters/

Networks can optionally be linked together into Clusters. Similar to how two or more linked Identities form an Account, two or more linked Networks form a Cluster. This pattern is fractal, enabling any Cluster to become associated with any other Cluster, which may contain other Networks or Clusters of Networks.

For instance, the Zero application represents Networks and Clusters as 'Worlds, Galaxies and Universes', in order to make Networks easier to represent and traverse. Worlds represent a single Network, Galaxies represent a Cluster of Networks, and Universes represent multiple Clusters of multiple Networks. This is a flexible structure that enables Members to spatially visualize the relationship and hierarchy between Networks.

Globally unique Network names, also called domains, can be purchased using ZNS, as outlined in Section 4.3.1 Zero Name Service (ZNS) Protocol.

4.1.8 Peer Routing & Discovery/

zOS comes equipped with libp2p's standard content routing and addressing libraries. Routing is necessary in order to know exactly where to send Messages in the network. In order to successfully route data from one Node to another, both a Node zId and Address (such as an IP) are required.

In the common event where a Node knows another Node's zId but not it's Address, it can begin the process of Node Discovery. Node Discovery enables a Node to leverage Address information known by the other Nodes in which they are already connected to. As more Nodes are contacted, more is learned about the Addresses of other Nodes, eventually resulting in a match. This builds a more comprehensive view of the Zero Grid, both enabling Message traversal to happen faster for subsequent connections, and to store/provide valuable routing data to other Nodes in what mimics a mesh-network.

Peer routing currently uses a <u>Kademlia</u> implementation of a <u>distributed hash table</u> (DHT). This processed lookups where each search step results in a closer Node to the Node in question, until it is found. Prior to building knowledge of any Nodes within the network, the zId and Address of one or more Nodes is needed to bootstrap the process of Network Discovery. This can be achieved by three methods:

- **Bootstrap:** A bulk list of pre-defined Node addresses that can be used to initiate data propagation.
- **Invite:** A single invitation to connect to a specific address.
- Local Network: A search for valid addresses and ports on a local area IP range.

4.1.9 **Gossip**/

Gossip is the primary <u>PubSub</u> (publish/subscribe) pattern used for sending Messages between Nodes in the Zero Grid. Publish/Subscribe is a common pattern to enable peers to converge around topics in which they share mutual interest. When a peer sends a Message to a particular topic, the Message is automatically relayed to all peers subscribed to the topic. For example, IPFS uses pub/sub topics to represent downloadable files on the IPFS network. Once uploaders and downloaders know which peers are holding what file fragments, they then connect together directly to coordinate the full-file download, outside of the pub/sub.

Zero utilizes libp2p's gossipsub modality. This enables peers to gossip which messages they have seen in order to maintain a message delivery network. Prior to subscribing to a topic, the Node must have a mechanism for finding other peers on the network using Peer Discovery, as outlined in the prior section.

Gossipsub enables peers to connect using one of two mechanisms: **full-message** peerings or **metadata-only** peerings. Together, these networks create two independant and connected virtual overlay networks. Full-message peerings maintain the full contents of Messages, and metadata-only peerings maintain only Message metadata. Gossipsub referrers to full-message networks as 'sparsely-connected', given that a much smaller number of the total Nodes in the network are connected relative to the total network size. The metadata-only network is referred to as 'densely-connected'. This network maintains the network of full-message pairings by sharing gossip about which peers have full-messages available.

Gossipsub enables the configuration of a lower and upper bound of how many full-message Nodes each Node must be connected in order to help regulate network traffic in what is called a <u>peering degree</u>. The peering degree is the central mechanism for managing the trade-offs between speed, reliability, resiliences and network efficacy. For instance, a network with more full-message Nodes will share data faster and maintain a higher level of data replication, however come with the cost of added storage space,

computational energy and network bandwidth. The <u>gossipsub-1.1</u> specification and documentation outlines additional core concepts, including grafting and pruning, the process of publishing and subscribing to topics, how messages are sent, fan-out, network-packets and state.

For public Messages, Zero utilizes a Member's social graph to determine which Message data to store and replicate. Given that each Node announces which Nodes they publicly follow, zOS can arrange the application state into a graph of who follows who. By default, Zero sets a graph depth of two, meaning that Messages will be downloaded from those followed by who you follow. This is a useful heuristic to establish trust between Peers that was pioneered by Secure Scuttlebutt.

4.1.10 **zApp**/

4.1.10.1 **Defining a zApp**/

Zero Applications (zApps) enable developers to build and deploy custom applications that can be run by Nodes within the Zero Grid. zApps in zOS can be thought of as miniature programs that store and manage custom-defined information.

zApps are defined with the specialized Message type

'zApp:[zAppId]:[ObjectType]:[PermissionHash]', with the associated application state residing in the 'content' parameter of the Message. '[]' indicates a Member-generated look-up value for a respective zApp and functionality set. The definitions of these parameters are as follows:

- **zApp:** This specifies that the Message is a custom defined zApp
- [zAppId]: Similar to a zId, a zAppId is a unique and randomly generated hash to represent a specific zApp and version number.
- **[ObjectType]:** This specifies a custom defined Object Type that is used to break up application state into small chunks of data. Example ObjectTypes could be 'Posts', 'Tasks', or 'MarketItems'.
- [PermissionHash]: This specifies a PermissionHash that is generated using a SHA256 hash of a 'permission object' by the Message signatory. Permission objects require the following format: { read: {zId_1, zId_2...}, write: {...}, delete: {...} }

Simplifying a zApp's application state into small chunks ensures that Messages remain reasonably sized and are efficiently propagated through the Zero Grid.

Given that zApps are defined purely at the level of state, clients are responsible for how to interpret this state, as well as ensuring that zAppIds are consistent between clients and relevant Messages. Adding a new zApp is as simple as signing a new Message to a zChain, which other zChains can then choose to interpret or not, depending on the application logic they have chosen to run. Given that zApps are just custom Messages, they can be added to any type of zChain, including an Identity, Account, or Network.

4.1.10.2 **zApp Store**

zApps require a corresponding interface to interpret and display state inside of a client. Zero intends to make a front-end component kit available to make it easy for developers to build visually consistent interfaces inside of the Zero Application.

Zero plans to add an zApp Store (similar to the iOS store or Google Play) that enables the installation of zApps that exist in the Zero ecosystem. To prevent malware and malicious software, application Verification will be introduced and managed using a fully decentralized process by zDAO/. Pertinent details will be released in advance of the launch of the zApp Store.

4.1.11 **Dynos/**

Nodes in Zero can choose to operate Dynos. Dynos are containerized virtual environments that make specific applications or services available to Nodes in the Zero Grid. Unlike peer-to-peer only systems like IPFS, BitTorrent or Ethereum, Dynos enable Nodes to run and deliver virtually any type of distributed environment to the Zero Grid, including cloud infrastructure.

Dynos are intended to be easily instantiated, configured, replicated, and non-persistent, meaning that they can be easily spawned or destroyed without consequence. Dynos exist to serve a particular service rather than act as long-term data stores. In this way, Dynos are analogous to dynamic application servers in traditional cloud architectures, that can scale up or down depending on network load. The main difference here being that any Node in the Zero Grid could choose to provide these services, rather than a centralized company. This enables the Zero Grid to use the speed, availability and scale of existing cloud providers for certain use-cases, without sacrificing decentralization.

For instance, a Member could choose to run a set of Video Conferencing Dynos on AWS under their own name and credit card, and recieve Credits based on the Dynos usage. A Dynos binary code is open-source and verified with a cryptographic proof, so other Members would be able to ensure that no personal data is collected at the server-level, and the service is operating according to what it is advertising.

In order for a Dyno to receive Credits its binary must be approved by the Member. Credits can then be issued in exchange for the operation and utilization of the Dyno, depending on Dyno performance, along with market supply and demand. This is outlined in more detail in **Section 4.2.6. The Creditor.**

Dynos can be thought of as similar to AWS (Amazon Web Services) services, such as EC2 instances or s3 buckets, but at a higher level of abstraction. Dynos provide a new incentivization, security and decentralization model on top of existing cloud-architectures that is agnostic. This provides flexibility for pricing, the possibility for many service types, as well as enables the network to dynamically adjust the availability of different Dynos, based on the real-time needs of the Zero Grid.

Example use-cases for Dynos include:

• **Messages Pub:** A Node that increases the speed Message propagation for other Nodes on the Network.

- **Video Conferencing:** A Node that routes encrypted peer-to-peer video traffic to reduce latency for multi-party video conferencing.
- Live Streaming: A Node that manages connection speed and bandwidth capacity for individual live streams.
- Virtual Worlds: A Node that enables low-latency and persistent virtual worlds.

Dynos are authorized and installed on a Node by signing a new Message to a Member's local zChain with type 'Dyno:[DynoId]:[binaryHash]:[filecoinAddress]':

- **[dynoId]:** A unique Id that represents the Dyno and container version on the Network.
- **dynoType:** A valid dynoType as defined by the Dyno Matcher (e.g. videoConference, virtualWorld, etc.).
- **[binaryHash]:** A sha256 hash of the Dyno's container binary that is used to ensure version consistency and that the file has not been tampered with.
- **[filecoinAddress]:** A static location to download the file from the Filecoin Network.

This process begins the remote installation process of the Dyno binary from the provided Filecoin address, and then performs a checksum on the file to ensure the binaryHash is valid. Once the Dyno has been verified, it is added to the zOS run-time environment and the Node begins to propagate Messages within the Grid, signaling that the Dyno is available to connect to.

Compared to simply propagating Messages like a normal Node, Dynos are designed to provide services that require more dedicated, consistent, and low-latency connections on a single server. Given this fact, once a Node discovers a particular Dyno, its connection generally needs to stay persistent for the duration of the process provided by the Dyno. For instance, a two-hour video conference between ten network participants requires a fast, stable, and persistent connection. This is a significantly higher bar than routing a single small sized Message between chains.

The Dyno Matcher, outlined in **Section 4.2.7 Dyno Matcher**, describes the process of matching Nodes with Dynos and the stack-ranking that occurs to ensure that Nodes connect to the highest grade Dynos within the Grid

4.2 Currency Layer/

4.2.1 **Tokens/**

Tokens in Zero represent any standard cryptographic token on a blockchain, including but not limited to Ethereum's Ether (ETH) and tokens compatible with Ethereum's <u>ERC-20</u>, <u>223</u>, <u>721</u> and <u>777</u> standards.

The Zero Token System refers to the entire set of token related protocols that make cryptographic token exchange possible in Zero.

The Zero Token System represents the architecture and set of protocols that make token exchange possible within Zero. The Infinite Economy (outlined in Section 4.2.4 The Infinity Token) is used generally to represent all Tokens, protocols and associated transactions that make up and take place within Zero's economy.

4.2.2 Exchange Tokens/

Exchange Tokens in Zero are used to facilitate the direct exchange of two Tokens, without the need for an intermediary exchange or counterparty. Zero Networks are required to launch their own Exchange Token and associated DAO that is responsible for the economic design and governance of the Network. Exchange Tokens utilize a modified version of the <u>Bancor</u> Protocol.

Generating a new Exchange Token in Zero requires the initialization of **five** parameters:

- Token Name: A name, such as 'Infinity'.
- **Token Symbol:** A ticker, such as 'IOI.
- **Initial Supply:** The Token's initial number of tokens in circulation, such as 1,000.
- **Token Reserve Currency:** A reference to the Token's Reserve contract address on the Ethereum blockchain.
- **Reserve Ratio:** The initial ratio between the Exchange Token's Reserve balance and price, denominated in the Reserve Currency.

Exchange Tokens function as automated on-chain <u>market makers</u>, which create immediate <u>liquidity</u> and price discovery for buyers and sellers, without the need for market depth or a traditional order book.

Zero refers to this mechanism as 'Autonomous Liquidity'. This feature enables tokenization for a <u>long-tail</u> of digital communities. Similar to the prohibitive cost of creating and distributing video content prior to the advent of YouTube, Exchange Tokens lower the barrier of entry for the creation of functional digital currencies for communities of any size. Autonomous Liquidity is particularly useful for open-source projects, remote or digital organizations, and DAOs, where a shared token can be used to capture and coordinate value between participants, without the need for an intermediary corporation, physical office, or jurisdiction.

Autonomous liquidity is achieved through the use of both an internal <u>bonding curve</u> (also referred to as a continuous token) and Token Reserve, which enables an Exchange Token's price to dynamically adjust based on supply and demand.

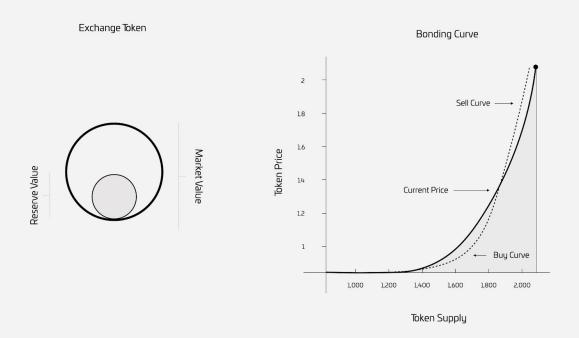
The following definitions are useful to comprehend the Zero Token System:

• **Price:** The price at which a single Exchange Token can be purchased or sold. This price dynamically adjusts with each purchase and sale.

- **Supply:** The current total number of Exchange Tokens that have been issued and are in circulation.
- **Reserve:** The total amount of Reserve Tokens used to 'back' the value of the Exchange Token. This is achieved by holding a balance one or more Tokens inside an Exchange Token's smart contract.
- **Reserve Token:** A specific cryptographic asset that is stored within a Reserve. Common reserve assets include ETH, DAI, wBTC.
- **Reserve Value:** The total value of the assets that make up a Token's Reserve.
- Market Cap.: The total market capitalization (total market value) of an Exchange Token.
- **Liquidity Pool:** A type of Reserve with a built-in fee structure to incentivize the buying and holding of Tokens in order to provide liquidity for a token. Liquidity Pools must hold two or more tokens in its reserve.

Exchange Tokens can be purchased by sending any amount of Reserve Tokens to the Exchange Token contract, where newly minted Exchange Tokens are then automatically issued into circulation and transferred to the sender. Conversely, when Exchange Tokens are sold, Reserve Tokens are automatically sent from the Reserve to settle the exchange. This mechanism enables Token Price and Token Supply to be dynamically recalculated based on supply and demand, with no need for an intermediary to facilitate the transaction

Figure: Exchange Token and Bonding Curve



Currently Exchange Tokens can hold any ERC20-compliant Token as its Reserve. This enables the onchain exchange of a growing number of blockchain-based tokens. Exchange Tokens require a statically defined Reserve Ratio at initialization, which serves as a fixed ratio between the Exchange Token's Reserve balance and market price. This ratio remains static with the purchase and sale of each Token. This Reserve Ratio is a primary factor in establishing an Exchange Token's long-term price sensitivity. Price sensitivity refers to the degree of change in price that occurs from each purchase of sale of an Exchange Token. A higher Reserve Ratio will result in lower price sensitivity, whereas a lower Reserve Ratio will result in a higher price sensitivity. The following diagram illustrates how different Reserve Ratios impact a token's price as it grows.

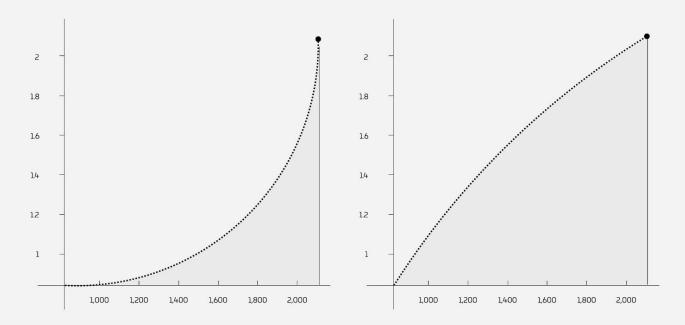


Figure: Demand Curves for Reserve Ratios of 10% (left) and 90% (right)

The following formulas outlines how to calculate an Exchange Token's price:

- Calculating a Reserve Ratio:
 - Reserve Ratio = Reserve Balance / Market Capitalization
- Calculating Market Capitalization:
 - *Market Capitalization = Price x Supply*
- Calculating Price:
 - Price = Reserve Balance / (Supply x Reserve Ratio)

4.2.3 Dynamic Tokens/

A Dynamic Token is an Exchange Token that uses a Dynamic Reserve Ratio (DRR). The DRR creates the desired effect of token stability and liquidity, while creating an alternative funding model to Initial Coin Offerings (ICOs). Dynamic Tokens replace static Reserve Ratios with an immutable function that adjusts a Token's Reserve Ratio overtime, based on specific inputs connected to the demand of the Token.

To understand the potential significance for Dynamic Tokens, it is useful to quickly review the evolution of Token Models overtime.

Initially, fixed-supply tokens, often associated with ICO-funded projects during the 2017-2018 'ICO boom', set their tokens supply at the time of token sale. These tokens often set an arbitrary 'cap' on the amount to be raised – effectively setting, or at least targeting, a large, predetermined market capitalization for the token and the project. In this model there is no reserve (backing of the token) with real world value, making the token price vulnerable to major volatility. Price swings are especially volatile if a

project utilized clever fundraising mechanics, like daily token auctions or double dutch auctions, in order to create a sense of artificial scarcity to quickly reach their ICO fundraising cap. Most importantly however, the challenge with this model is that after the completion of the initial sale, 100% of proceeds were provided to developers, irrespective of any actual usage traction or demonstrated viability.

More recently, token architectures with a *bonding curve* and a *static reserve ratio* have shown the possibility of greater liquidity and price stability by partially backing tokens with underlying reserve assets that hold real world value. Unlike with a fixed supply token, in this model there is no single funding event. The project is funded continuously over time in accordance with the growth of its token economy. While this approach supports token holder confidence, it can make it challenging for projects to receive the optimal amount of funding overtime and can negatively impact price. In this model the only way to create operating capital is to set aside an initial supply of token that must be eventually sold back into the economy, creating downward pressure on token price and potentially signaling a loss of confidence from the project's developers.

Today, we are witnessing the maturation of a vast number of DeFi (Decentralized Finance) protocols, of which 'stablecoins' and 'synthetic asset protocols' are a major component. These token models use over-capitalized reserve ratios – 150% in the case of MakerDAO, and 750% in the case of Synthetix – to combat the price volatility of crypto-assets relative to underlying currencies (such as USD or Gold). These protocols incentivize individuals to hold and maintain a certain ratio of assets in decentralized liquidity pools, which is used to back (and peg) the value of their token. While these protocols are useful for decentralized stablecoins, derivatives and loans, they are not especially useful for funding projects, due to the fact that they have no upward price mobility relative to the value of their underlying assets.

Dynamic Tokens simultaneously solve the token funding and volatility issue by combining a number of mechanics from the previously mentioned token models. Dynamic Tokens use a bonding curve that partially backs the token with one or more underlying reserve assets, which dynamically changes its reserve ratio over time based on supply and demand. This is achieved by automatically transferring 'excess reserves' to a specified smart contract, such as a multisig contract or DAO, based on the function specified in the DRR. Excess reserves in this context are referred to as the available Reserve Tokens no longer required 'to back' the Exchange Token given that a certain demand threshold has been hit based on the parameters of the DRR. Perhaps most importantly, this is all done without impacting the token's market price.

Dynamic Tokens can adjust their reserve by the contract owner calling the reduceWeight method on the Dynamic Token contract. A Dynamic Token is initialized with four parameters in addition to a regular Exchange Token:

- Minimum Reserve Ratio: The smallest possible ratio that the reserve ratio can be reduced to.
- **Step Weight:** The total amount (percentage) that a reserve ratio can decrease when calling reduceWeight().
- Market Cap. Threshold (optional): The market capitalization in which funds can no longer be transferred from the Reserve.

• Maximum Reinflation (optional): The total amount a Reserve Ratio can be reinflated (increased) to once its reserve ratio has passed that threshold.

When successfully executing the reduceWeight method, tokens held by the reserve equal to the Step Weight are automatically transferred to the owner (via msg.sender) in the transaction. Conversely, once a Reserve has been reduced past its Maximum Reinflation parameter, all future token purchases will be directly deposited into the token's reserve to restabilize the token, thereby bypassing the standard formula for token price and issuance. Once the token's reserve reaches or exceeds its point of Maximum Reinflation, token purchases will resume as normal and continue to autonomously expand the token's supply as per its standard formula.

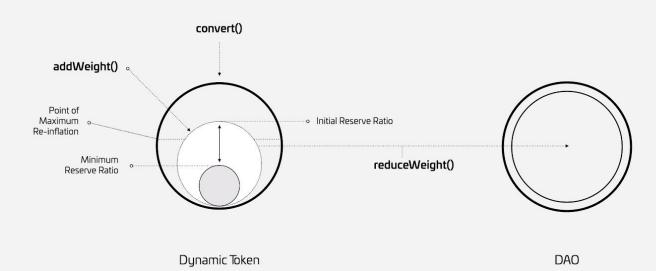


Figure: Dynamic Token Elasticity and Funding

This approach enables projects to start with a high reserve ratio, and then decrease it over time as the project matures, balancing token volatility and project funding, while bolstering the confidence of early token purchasers. This mechanism bears some similarities to the 'DAICO' model proposed by Vitalik Buterin, where funds are not transferred to developers in an upfront fundraising event, but rather are dispersed using a formula based on demand for the token *after* it has been deployed.

4.2.4 Infinity & The Infinite Economy/

Infinity (IOI, ∞), Zero's native utility token, powers Zero and the Infinite Economy. The Infinite Economy represents all Tokens, protocols and associated transactions within Zero. The Infinity Token provides three token utility models:

- 1. **Network Token Exchange:** Infinity's primary utility model facilitates the exchange of Network Tokens in Zero. Each Network is required to utilize an Exchange Token, that is fractionally backed with Infinity as a Reserve. This enables automatic price discovery and the seamless exchange between any Token within the Zero Network, simultaneously removing the need for both centralized exchanges and decentralized liquidity pools (as are required with <u>Uniswap</u>).
- 2. ZNS Registries: Infinity's secondary utility model enables the purchase and sale of public namespaces, such as Member handles and domains via the Zero Name Service. Members can stake Infinity to purchase Zero Registries and Registry Entries. Registries represent 'root' namespaces within ZNS, such as '0://rootdomain', and enable Registry owners to sell associated domains, such as 0://rootdomain::domain. ZNS is further outlined in Section 4.3.1 Zero Name Service (ZNS) Protocol.
- 3. **Infinity Mining:** Infinity's tertiary token model is to incentivize Members to power the Zero Grid. In exchange for providing network compute, bandwidth, and web services (Dynos), Members receive hosting Credits (as outlined in **Section 4.2.6 The Creditor**) that are redeemable for Infinity. Redeemed Infinity can then be sold on the open market for traditional currencies, such as ETH, BTC or USD, creating a globally distributed and decentralized computing network. Example Dynos include additional storage, high availability video conferencing, and virtual world servers. The process of receiving Infinity in exchange for powering the grid is referred to as 'Infinity Mining'.

Infinity can be thought of as the 'fractional reserve' that backs the Infinity Economy. This structure creates three powerful mutual incentives for all Members of the Infinite Economy.

First, holding Infinity as the primary Reserve asset in an Exchange or Dynamic Token facilitates the seamless exchange and automatic pricing for tokens in the Infinite Economy. Technically, a common Reserve token is necessary in order to facilitate entirely on-chain exchange between two Exchange Tokens. This increases liquidity between all Exchange Tokens in the overall economy, something that is generally a primary issue with new tokens, communities, and projects. In this way, Infinity makes a 'long-tail' of tokens possible for new upstarts.

Second, a shared Reserve Token aligns incentives between all Zero Networks and the Zero system itself. As the Infinite economy grows and experiences network effects, individual (and especially early) Networks benefit proportional to the percentage they've chosen to back their token by the Infinity Token. This leads to the increase in value of a Network irrespective (and in addition to) the growth of their individual Networks economy.

Third, Infinity eventually plans to be pegged to a basket of underlying Tokens that have been selected for long-term growth potential (such as Gold, Bitcoin, and Ethereum). This enables Networks and their associated tokens to benefit from the overall macro-transition away from fiat-based currencies that are occurring and hedge volatility risk. Irrespective of an individual Network's success, or even the Zero platform's success, Infinity stands to benefit if precious metals and crypto assets perform generally well over the long-term.

The structure and issuance of Infinity is outlined in Section 5, Governance & Token Issuance.

4.2.5 **Credit/**

Credits in Zero represent a claim on an atomic unit of hosting capacity, or 'AU', in the Zero Grid. 'Atomic' in this sense refers to the smallest possible unit of transferable (future) hosting capacity. Debits in Zero are the inverse of a Credit, and represent the obligation of a Node to provide or repay an AU. Members are credited or debited based on the Nodes AU contribution or utilization, provided to or received from the Zero Grid. Credits and Debits in Zero can be viewed as a distributed micro-transaction system based on the principles of mutual credit accounting.

Mutual credit is an alternative way of viewing money, currencies, and value. In essence, currency is not centrally issued to represent an extrinsic (objective) amount of value that is derived from a single (shared) ledger (such as a central bank of cryptocurrency), and rather, is tracked as a series credits and debits for each participant in the system. The Meta-Currency Project is an excellent resource for delving deeper into the philosophy and economic concepts behind mutual credit.

Due to the distributed architecture of Zero's peer-to-peer system, a mutual credit system is the ideal candidate for tracking hosting credits and debits. Unlike purely blockchain-based systems, for Zero's peer-to-peer network, no central authority exists to accurately keep track of the hosting credits and debits of a particular Member. When a Member runs the Zero software as a Node, a local component called The Creditor (outlined in **Section 4.2.6 The Creditor**) is responsible for rapidly issuing credits and debits for the Node. These Credits are then redeemable for Infinity, enabling an entirely distributed credit-system to exist between participants, that is also connected to all tokens in the Infinite and traditional economy, enabling a pathway to convert credits to fiat currencies and vice-versa.

When a Member provides AU to the Zero Grid they accumulate Credits, which can then be redeemed for Infinity, at an exchange rate algorithmically determined by The Creditor. The exact calculations as to what determines an AU at a given time are dynamic and are performed entirely in isolation by The Creditor. Credits and debits are issued in a distributed manner, making it near-impossible to know the full amount of credits or debits in total circulation at a given moment in time, given that the number is changing all of the time, and no single Node is aware of the entire system's state. This system architecture solves the problem of latency and high transaction fees that would exist with managing AU transactions on a blockchain, while still enabling the transferability between the Credits and cryptocurrencies.

Given that the supply of Infinity itself is determined by its own free market economics, there is no easy way of issuing new Infinity from the Creditor without either diluting Infinity or requiring the Creditor to purchase it directly like any other market participant, which would be prohibitively expensive. Alternatively, if the Creditor was given control of the issuance of Infinity, it would undermine Infinity's tokenomics, dynamic reserve structure and governance, and likely limit the flexibility for incentivization within Zero's credit system.

For this reason, The Creditor cannot issue new Infinity directly. Instead, a 'bridge-token' is utilized called 'Credit Token', which holds a percentage of the Infinity Reserve, whose supply can be dynamically expanded or subtracted within the Creditor's DAO.

The Creditor

Node

Infinity Token

Zero Grid

Figure: Dynamic Token Elasticity and Funding

The details of Infinity, it's reserve configuration and associated DAO structures are outlined in **Section 5**. **Governance & Token Issuance.**

4.2.6 The Creditor/

The Creditor runs the underlying set of algorithms responsible for managing the issuance of credits and debits on a Member's local machine, based on actual network bandwidth, compute, and Node reliability as well as the usage and performance of a Node's individual Dynos. The Creditor is responsible for the end-to-end security of this process, ensuring that a Nodes performance logs have not been tampered with to artificially inflate the number of credits it is entitled to.

Each AU is issued based on Zero's 'Credit Entitlement Algorithm', or *CEA*. The *CEA* utilizes an internal set of calculations that produce four primary scores that get associated to each Node and active Dyno:

- Uptime: A 0-100 scalar value representing historical system availability over time.
- **Utilization:** A 0-100 scalar value representing historical Node Dyno utilization over time.
- Latency: A weighted-average of time in milliseconds for processing incoming and outgoing network requests for an individual Dyno over time.
- Credibility: The reputation of the Node (or operator) as determined by other peers in the Zero Grid at the time of calculation.

Here is the generalized formula used to calculate a Node and/or Dyno's performance using CEA:

$$CEA = \left(Uptime \ x \ Utilization \ x \ Latency \right)^{Credibility}$$

Results based on these computed values are then weighted over time, with more recent uptime and latency results attributing to a higher score. This is represented as CEA(t), with t representing how different weightings are applied to different historical time intervals.

Nodes then stack rank the results from individual Node Dyno pairs that can be used the The Dyno Matcher (outlined in **Section 4.2.7 Dyno Matcher**), to support in supporting and allocating network traffic to increase system-wide reliability. This is the generalized formula used to weight *CEA* over time:

$$CEA(t) = \left(\frac{(t(<24h)(CEA)) + (t(>24h)(CEA))}{2} \right)$$

Once a new CEA has been recorded, Nodes then gossip scores to peers using the custom Message type 'CEA', which is accompanied with a cryptographic proof to ensure the CEA and CEAt calculations have been performed correctly and not been altered.

4.2.7 Dyno Matcher/

Zero's Dyno Matcher aims to ensure that Nodes requesting services are matched with the highest quality Nodes and Dynos available within the Zero Grid. Nodes and Dynos are hereinafter referred to as Node Dyno pairs, given that all matching requires a Node to be matched via a Dyno. One way to think of the Dyno Matcher is an agent-centric approach to index and connect additional services that are useful beyond zChain. The process enables peers to establish connections directly with Node Dynos that can exist at any endpoint, including centralized servers or peers in any type of distributed network.

As outlined in the previous **Section 4.2.6 The Creditor**, after CEA(t) calculations are performed, this data is propagated to other peers in the Zero Grid using a gossip protocol. As CEA(t) data is received, the Dyno Matcher stack ranks Node Dyno pairs with associated scores. Once this occurs an additional process is run that continually pings the top resulting Nodes within the stack rank that also adds network latency data to the stack rank. If latency falls below a certain threshold, the Node is temporarily removed from the stack rank until it makes it back into the ranking.

Providing a Node Dyno-pair's latency is within an acceptable range, the Node then establishes a connection from the local machine to Node running the Dyno (otherwise known as the Dyno host). Once the service is instantiated, address information and credentials are sent to other relevant peers. To illustrate this, let's explore the process of a Node establishing a video conference with multiple parties:

- 1. The Member submits a Message to the Dyno Matcher to start a new video conference from their local machine.
- 2. The Dyno Matcher pings the top ranking video conferencing Dynos according to the most recent data recorded in the Member's local CET stack ranking. This data has been previously gossiped by other Nodes on the Network.
- 3. Once the Node has found a viable Dyno, it sends a direct request to spawn a new connection instance for the relevant Dyno on the host's machine.
- 4. If the Video Conferencing Dyno instance is successfully deployed, a connection is established, and the requisite process is established on the remote machine.
- 5. A unique url and set of access credentials are created to access the host machine. The remote IP / domain, route and necessary credentials are provided, such as:

 'https://video.zer0.io/TCP/23.29.18.18/video/videoConfName'. Members are provided that link along with a password that must be entered upon arriving at the url.

As more Members place system load on the top ranking Nodes, scores are dynamically adjusted and propagated to other Nodes in the Grid using the gossip protocol. Nodes with more capacity and better performance move up in rank, creating a natural balancing effect between Node Dyno pairs within the Grid. Given that matching happens using an agent-centric approach, different clusters of Nodes maintain unique stack ranks, further aiding with network distribution, and supporting to alleviate potential congestion.

4.3.0 Trust Layer/

Zero's Trust layer is built upon two primary technologies: the Zero Name Service (ZNS) and zDAO.

4.3.1 Zero Name Service (ZNS) Protocol /

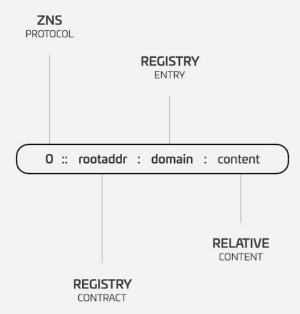
ZNS is responsible for managing the ownership and routing of globally persistent namespaces through Zero Registries and Registry Entries, which are also referred to as root domains and domains, respectively. A primary necessity within distributed systems (such as IPFS, BitTorrent and the Internet) is the requirement for a globally consistent addressing structure for universally accessing content (hence, http). Given that zChain is peer-to-peer, there is no simple way to ensure human readable addressing is globally unique, making it difficult for Members and search engines to store and index content. ZNS solves this problem by enabling Members to purchase and register traversalable namespaces, such as a Member handle or Network name, which maps directly to a persistent digital reference location. ZNS is inspired using core concepts from Internet DNS, along with Token Curated Registries, blockchain Identity providers like Civic, and crypto domain providers like Unstoppable Domains. A vertically integrated global routing and identity protocol enables Zero Networks and Members to set their own registration stake prices, and creates a universally consistent method for indexing global content.

The eventual objective of ZNS is that modern browsers and web3 projects adopt the protocol to create a more customizable, flexible, and decentralized alternative to modern DNS, which also includes the introduction of human verification and superior financial incentives.

4.3.1.1 Addressing/

Zero Addresses are made up of a hierarchical set of registries and registry entries on the Ethereum blockchain.

Figure: ZNS Address Structure



Let's review the constituent parts of a Zero Address by evaluating the following: '0::rootaddr:domain:content':

- 0:: -> is used to signify the beginning of a Zero Address path.
- **0:rootaddr** -> The first value in the address is a root domain that references a unique Registry Contract and Member-defined reference value.
- **0:rootaddr:domain ->** The second value in the address is a domain that references an Entry within the Registry, along with a Member-defined reference value.
- **0:rootaddr:domain:content** -> The third value in the address represents a path defined within the Registry Entry.

In relation to traditional DNS addressing, the 0:: is comparable to http://, the 'rootaddr' is comparable to a TLD like .com or .io, the domain is comparable to a domain, and content is comparable to a relative path on a web server.

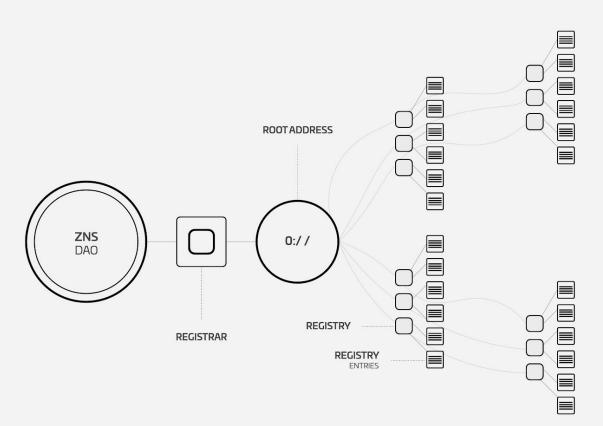


Figure: Example ZNS Registry Tree

4.3.1.2 Staking & Governance/

One of the unique properties of ZNS is the introduction of blockchain staking and optional DAO-based governance at the Registry and RegistryEntry level. ZNS enables anyone to purchase new Registries by staking Infinity equal to the stakePrice set within the ZNS Global Registrar. The stakePrice can be increased or decreased overtime via the submission and acceptance of a proposal from within the ZNS DAO. Purchasers of root domains (Registries) and domains (Registry Entries) are automatically added as DAO Members (along with voting power) to the ZNS DAO, which is the owner of the Registrar contract. This ensures that all ZNS owners (at both the root domain and domain levels) maintain the rights to set the stakePrice required to register new Registries and vote on ZNS protocol upgrades.

Given that each Registry has its own RegistryToken, Registry operators can create staking-based incentives via the issuance of governance tokens, similar to the mechanism used by modern DeFi protocols. This creates a 'market of markets'; an incentive-based directory structure for curating and traversing content-sources across web3 protocols and the Internet at large.

Governance for individual Registries operate similar to the Registrar, with the exception that the contract owner can define a stakePrice and appoint a custom DAO to manage the Registry after it is purchased. For instance, an individual or community could purchase the Registry (a root domain) at '0:shop', and sell domains, such as '0:shop.books', to fund the Registry and its associated DAOs, at a price the Registry DAO has collectively established.

4.3.1.3 Global Handles/

A Global Handle is a human readable name (such as @n3o) and associated ZNS path and url, such as 0:://n3o and https://zer0.io/n3o, respectively, that links to the ZNS reference of a Registry or Registry Entry. Global Handles can be utilized to reference either an individual Account, Avatar, Network, Cluster, or any arbitrary content source. Global Handles can exist at the ZNS root (0://), or can exist within a Registry Entry (such as 0://mynetwork/n3o).

4.3.1.4 The Universe Ontology/

ZNS is the core protocol and global index responsible for organizing Networks and Clusters (outlined in **Section 4.1.6** and **4.1.7**, respectively) into Worlds, Galaxies, and Universes within Zero. This naming structure is defined as the Universe Ontology, and is implemented at the Zero application level by interpreting entries in the global ZNS Registrar.

The Zero application interprets a Registry domain as a Universe and can interpret a Registry Entry as either a World or Galaxy, depending on the type of zId provided in the ZNS entry's

reference field. Within Zero, a World is represented using the zId of a single Network, a Galaxy using the zId of a Cluster, and a Universe using the zId of a Cluster of Clusters.

The Universe Ontology enables Members to easily traverse and understand the respective relationships between Networks in a graph data structure. These relationships can then be spatially visualized, or searchable based on attributes, creating unique, private, spatially located digital spaces. Networks within the same Universe can be organized to represent shared values, collective identity, policies, token economics, DAO governance, and custom code, built on top of zOS.

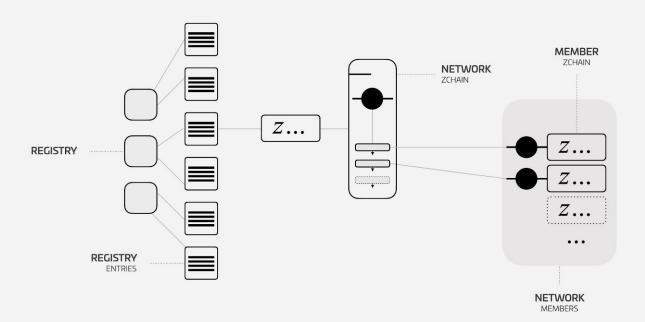


Figure: ZNS Registry Entry Referencing Network

Reference fields (e.g. where ZNS entries link to) are defined as an arbitrary string in ZNS, and do not enforce validation at the level of the protocol. This gives Registry and Registry Entry owners and DAOs full control over where ZNS connect to, irrespective of whether or not a valid may or valid zId's and associated Network type is provided. For instance, a Registry or Registry Entry could link to a file, or external web link (via http), instead of a valid zId. This is designed to enable a rich and flexible namespace service, which not only supports the above described Universe Ontology, but that also supports a wide range of relationship structures that can be defined and interpreted by developers building on the ZNS protocol.

4.3.1.5 Human Verification/

Human Verification in Zero is used to determine whether or not a Handle is operated by a real human identity. The ZNS Human Verification process, outlined herein, is an alternative to both centralized verification processes and automated verification processes, such as Twitter's infamous blue check or a <u>CAPTCHA</u> system, respectively.

Verification in Zero is loosely modeled off of the simple approach introduced by <u>HumanityDAO</u> in 2019. Unlike blue check systems utilized in services like Instagram and Twitter, verification in Zero is not intended to serve as a symbol for popularity, notoriety, or status, it is merely to determine if an individual is human or not.

Initially, there will be three tiers of identity verification in ZNS, which will result in a different level of Omega tokens being issued to a Member within the ZNS DAO (outlined in **Section 4.3.1.2**). Verification is managed by creating a Choice in the ZNS DAO to request and issue one or more ZNS DAO Omega tokens to a Member's account. Prior to starting this process, a Member must already have a registered Handle, which is associated to their zId and Ethereum account, by having purchased a new ZNS Registry or Registry Entry and staking Infinity.

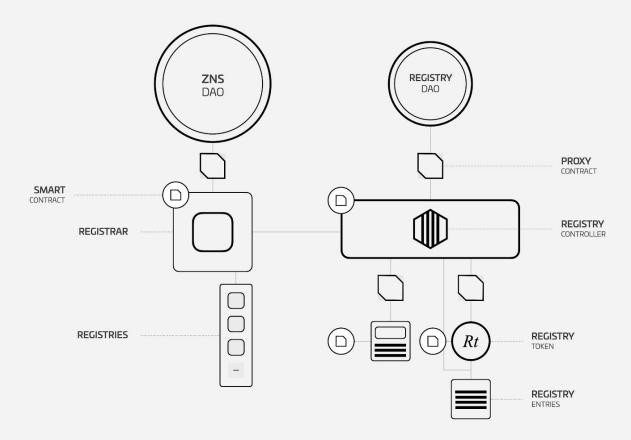
The Member may submit a Choice of type 'identity' to a DAO. By default, one Omega token is requested in the amount parameter for the Choice. The Member may choose to increase this number if they feel they deserve more Omega tokens in zDAO. Active and early contributors to the Zero ecosystem are likely to request and receive more Omega due their being less participants. A Member's total Omega within the ZNS DAO relative to the total amount of issued reputation in zDAO determines a Member's voting weight (choice making authority) within the ZNS DAO. This number can also be used as a proxy by developers to determine the probability an individual account is human.

When a verification Choice is submitted, existing Omega holders in the ZNS DAO can vote as to whether or not they believe the account is a human. Verification Choices require the Member to submit third party proof of identity. Initially, the ZNS DAO will allocate different amounts of Omega tokens for three tiers of Identity verification:

- Tier 1: Online Accounts: Access to a unique, pre-existing account for a major online service, that has not already been linked to a Handle in zDAO. Valid examples include Twitter, Facebook, GitHub, and Behance, and require the Member to post their public zId within a public post on at least one of these services.
- **Tier 2: Government Issued ID:** An encrypted and password-protected photo of a government issued ID, such as an identity card, birth certificate, or passport.
- **Tier 3: Formal KYC:** Formal KYC using a globally regulated third party service, such as everest.org.

Verification is entirely optional and not required to utilize the Zero system, but may be used by developers to create applications that require varying levels of identity verification.

Figure: ZNS Smart Contract Architecture



ZNS is made up of four primary contracts that should be deployed in the following order:

Registrar:

Registrar.sol is the root contract for creating and storing Registries. New Registries are created by calling the createRegistry() method by providing a valid:

- **domain:** a globally unique and Member-defined root domain
- **ref:** a reference to an arbitrary content source such as a url like https://zer0.io/network/zero
- **registryType:** a valid registry type to aid with indexing registries. Valid registry types are defined by the ZNS DAO (See Governance & Staking section below).

- **stakePrice:** the price of registering an entry (a domain) with the Registry such as zero:guild
- **registryTokenAddress:** the address of the registry's RegistryToken.

RegistryToken:

RegistryToken.sol is the associated token contract for a specific Registry that is based on the OpenZepplin ERC20 standard.

The RegistryToken is responsible for the financial related activity of a Registry including adding and removing stakers, minting and burning a registry's tokens, and depositing and withdrawing (via staking) funds into the RegistryToken contract itself.

A new RegistryToken is created on construction by providing a valid:

- **owner:** the owner's Ethereum address
- tokenName: the token's name such as 'Infinity'
- **tokenSymbol:** the token's ticker such as 'INI'
- tokenSupply: the token's initial total number of tokens in circulation
- **stakePrice:** the price for registering a new registry entry (a domain)

Registry:

Registry.sol is the contract responsible for managing and updating registry entries (called 'domains') within an individual Registry. A Registry must be initialized by calling the init() function after creation by providing a valid domain, ref, registryType and registryToken address.

A new RegistryEntry can be added by calling the CreateRegistryEntry() method and providing a valid domain and ref.

RegistryController:

RegistryController.sol is the main contract that validates permissible Member actions on a particular Registry, RegistryEntry, and RegistryToken. After creation and initialization, all interactions with the contract-set for a particular Registry must happen via RegistryController for security purposes.

After a new RegistryController is created it must be initialized by calling the init() method and providing a valid registry address and registryToken address.

4.3.2 **zDAO Protocol**

zDAO is responsible for the deployment, operation and governance of all Distributed Autonomous Organizations, or DAOs, within the Infinite Economy. DAOs have grown in popularity in recent years, after the concept was first realized in April 2016 with what is now infamously referred to as 'The DAO'.

A DAO is a fully decentralized autonomous organization that is managed and maintained according to executable code that exists on the blockchain. In essence, DAOs are a digital replacement for legacy corporations, including that of conventional startups, partnerships, funds and venture capitalists.

Unlike traditional organizations that are instantiated and bound by the legal framework of a particular jurisdiction, DAOs are entirely digital, borderless, and trustless organizations that transcend the notion of a legacy corporation. Similar to how the web reduced the marginal cost of information sharing and media distribution to near zero, DAOs reduce the marginal cost of complexity and coordination at a similar magnitude, enabling every aspect of a company to become digital.

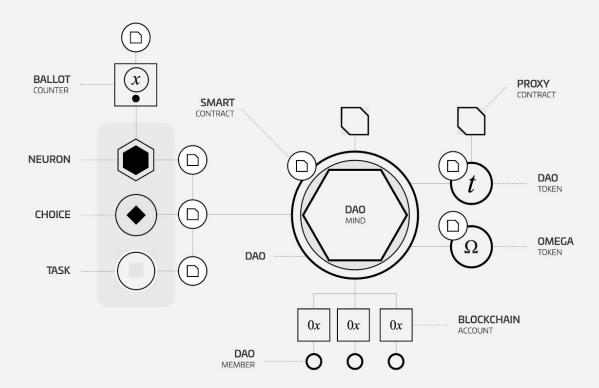
From the instantiation of corporate bylaws and governance, to the management of contracts and funding, Members can coordinate directly with one another by using a shared set of blockchain-based smart contracts.

Imagine a version of Facebook where billions of Members are paid for their data and collectively entitled to vote on the platform's future, or how their data is permitted to be utilized. Imagine a version of Pixar, where digital artists around the world are paid royalties for the submission of their 3D content, without the need for conventional management or venture capital. Imagine a version of Spotify, where individual fans and artists can exchange value directly, without a music label or platform taking the lion's share of revenue. DAOs make these ideas possible and will redefine the role of organizations in the near future.

zDAO is the underlying protocol and set of blockchain-based smart contracts that make DAOs possible within Zero. zDAO is inspired by the foundational ideas behind recent blockchain-based DAO protocols, including <u>DAOStack</u>, <u>Aragon</u>, <u>Colony</u> and <u>MolochDAO</u>. The zDAO protocol provides a flexible set of Ethereum-based smart contracts that enable onchain governance, along with zDAO.js, a nodejs library for interacting with zDAO's underlying blockchain layer. zDAO is built with modularity in mind, enabling third-party developers to easily extend its functionality for specific use-cases. The Zero Member-interface layer provides a simple abstraction using zDAO and zDAO.js to enable a fast and easy to use pre-configured DAO to govern Zero Networks.

DAOs in Zero are used to facilitate the coordination of shared responsibilities, resources and choices between DAO Members. DAO Members are distinct from Members. DAO Members are individual Ethereum accounts who hold DAO Tokens or Omega Tokens, each of which represents a percentage of total voting weight for a given DAO (outlined in the **Section 7.2** and **7.3**, respectively). Members may be humans or other smart contracts.

Figure: zDAO Protocol Architecture



4.3.3 **DAO**/

A DAO requires a valid name, Token and Omega Token to be instantiated at genesis. Generating a new DAO in Zero requires three parameters:

- **DAOName:** a string for the DAO's name
- **DAOToken:** a Token that represents economic value within the DAO
- omegaToken: a Token that represents voting weight within the DAO
- AttentionContract (optional): an AttentionContract that determines the visibility and rank of Choices within the DAO

Unlike Exchange and Dynamic Tokens, DAO Tokens have a fixed supply. DAOs can be linked to individual Networks and Tokens in one-to-one, one-to-many, and many-to-many relationships. Linking occurs when a Network's associated ZNS Registry, Registry Entry, and/or Exchange Token is linked to an individual DAO's Ethereum Address.

Linking tokens and smart contracts in this way provide a valuable path to the fully decentralized governance and operation of a company or organization. For instance, a Network DAO could be funded by using the proceeds generated from an Exchange Token's Dynamic Reserve. Similar to Zero and Infinity, this structure would enable a Network DAO to have reserve tokens automatically transferred to a DAO's ethereum contract, making the allocation of proceeds subject to a DAO's internal governance structure.

4.3.4 **DAO Token/**

A DAO Token is an ERC20-compatible Token based on OpenZepplin's <u>ERC20 library</u> that is associated with a DAO and serves in its core functionality.

A DAO Token is initialized with five parameters:

- name: a string of the token's name (e.g. InfinityDAO)
- **symbol:** a string representing the token's abbreviation (e.g. INI)
- **decimals:** a uint8 representing the total decimals permitted (e.g. 18)
- cap: a uint256 representing the maximum number of tokens allowed
- **transactions:** a uint256 store of historical transactions

DAO Tokens are controlled by the DAO Mind. A DAO Token's parameters can be updated by approved Neurons (as outlined in **Section 4.3.8 Neuron**). Neurons are autonomous code that can be assigned to manage DAO choice making and code execution.

4.3.5 Omega Token/

An Omega Token (Ω Token) is used to represent 'voting weight' in a DAO. Voting weight in this context refers to the total choice making weight ascribed to an individual Omega Token that is relative to a DAO's individual governance structure. Omega Tokens enable Members to have a non-economically motivated mechanism to determine choice making authority within a DAO. Unlike a DAO Token, Omega Tokens are non-transferable, meaning they cannot be purchased, sold, or transferred to other addresses by DAO Members.

DAOs have discretion in how they utilize Omega. A basic implementation could be to have a fixed supply of Omega and to pass (or fail) a Choice (as outlined in **Section 4.3.7. Choice**) based on the majority of Omega votes. For some or all Choices in a DAO, an 'absolute' majority of Omega might be required for a pass – i.e., greater than 50% of the entire Omega supply – while for others, a 'relative' majority might be required, which would only require 51% of the voting Omega to pass within a particular time frame. In this context, a Choice could represent the allocation of a DAO's internal resources, the issuance or Omega tokens to a Member, or the approval of new Neurons that could be assigned to autonomously manage the DAO.

As for how Omega is allocated within a DAO, there can be a variety of ways as determined by the DAO. One simple way is for a Choice to be raised to allocate a certain amount of Omega to a certain Member,

and for that Choice to be voted on just as any other Choice as described above. It is also possible for DAOs to utilize more advanced algorithms to allocate Omega, such as using input from a variety of sources like Reputation (outlined in **Section 4.3.10**, **Reputation**).

A DAO is required to have a single Omega Token that is automatically generated as part of DAO instantiation. DAO's have the optional ability to add more than one Omega Token, enabling Members to maintain different amounts of voting power for different purposes or Choice types.

Omega tokens should not be confused with an individual's 'Reputation'. Reputation in Zero refers to the community ascribed trustworthiness of an individual, relative to individual skills, characteristics, and observable behaviors that are defined by a DAO. Reputation is outlined in more detail in **Section 4.3.10**, **Reputation.**

4.3.6 **DAOMind**/

The DAOMind is the primary contract responsible for modulating decisions within the zDAO protocol. The DAOMind is managed by approved Neurons (outlined in **Section 4.3.7, Neurons**), which maintain specific 'permissions' over a DAO's *actions*.

A DAOMind is instantiated with a single parameter:

• DAO: the address of the corresponding DAO

DAOTokens can be:

Neurons can be:

- minted
- burned
- transferred
- received

- activated
- deactivated

The DAOMind has seven primary permission modifiers:

- PermissionToActivateNeuron
- PermissionToDeactivateNeuron
- PermissionToMintToken
- PermissionToBurnToken
- PermissionToTransferToken
- PermissionToTransferTokenFrom
- PermissionToApproveToken

Upon activation of a Neuron, one or more PermissionTypes that determine the Neuron's choice-making capacity are set inside DAOMind.

4.3.7 **Choice**/

A Choice in Zero is the primary mechanism for initiating a vote to perform a specific action within a DAO. A Choice can be for virtually any type of suggestion or initiative. A Choice must be initiated by a Member who is referred to as *the proposer*.

A Choice is a proposed voting event within a DAO, managed by the DAOMind. A Choice is initialized with five parameters:

- **DAO:** the corresponding DAO
- **DAOToken:** a DAOToken that will be utilized by the result of the Choice
- **choiceType:** the type of transaction being proposed, i.e., *to mint, burn, transfer, or increase the supply* of a DAOToken. Please note that approving a new Member is a subset of transferring DAOToken, since holding DAOToken is the criteria for being a Member of a DAO.
- **status:** one of three valid statuses: *proposed, approved, or rejected*
- **Neuron:** the associated Neuron and resulting code sequence that executed in the event of an accepted status.

A Choice has its own internal data structure to store ChoiceData, such as the proposer, an associated url for additional information, the status, vote count, and voter history.

Choices are passed based on a binary yes/no voting system and utilize weighted-Omega as a primary mechanism to determine if a Choice is approved or denied. If a Choice is accepted it will perform the Choice's corresponding action automatically, as determined by the connected Neuron within the Choice. Choices by default must be executed within a timeframe that is determined universally as part of a Neuron's interface definition.

In the event of a large DAO, such as a social network, it is easy to imagine millions of Choices being created by Members. This creates an existential problem for collective choice making: how should attention or rank be determined for new Choices? Zero enables DAO's to choose or build their own 'attention algorithm', called an AttentionContract, which defines the order and method by which Choices are displayed within a particular DAO. The AttentionContract is an optional parameter that is defined during or after DAO instantiation.

438 Neuron/

A Neuron is an autonomous and immutable smart contract that can be assigned to control various aspects of a DAO, according to parameters set and stored in the Neuron's struct once it is assigned by a DAOMind.

Neurons make calls to the DAOMind based on their own internal logic, and must be registered (approved) by the DAO Mind to execute certain methods (actions) on behalf of the DAO. Neurons may themselves add and remove other Neurons, enabling a DAO to effectively upgrade its own governance system

recursively, without the need for a third party or intermediary. Thus, a DAO can not only manage itself according to a defined set of rules, but may evolve itself over time in an autopoietic manner.

A Neuron is instantiated with the following parameters:

- **neuronName:** a string of the Neuron's name
- **neuronToken:** a DAOToken associated with the Neuron
- **votingEngine:** a votingEngine to manage Choices
- permissions: an array representing the permissions defined in DAOMind

A Neuron can be programmed to run with or without spawning a Choice that DAO Members are required to vote on. Developers can easily build custom Neuron constraints as long as the Neuron's smart contract address is approved and registered in the DAOMind.

In the event that a major technical upgrade is needed, the Mind is capable of upgrading itself by transferring the ownership of all its associated smart contracts to a new address. This transaction is non-reversible and non-transferable. Calling this method can be determined by a vote at the discretion of DAO members.

4.3.9 **Voting Engine/**

A Voting Engine defines the criteria for whether an individual Choice passes or fails. Creating a new Voting Engine requires the following **two** parameters:

- **name:** a string of the Voting Engine's name (e.g. Absolute Majority)
- **acceptanceThreshold:** a uint256 representing the percentage threshold the Omega votes for a Choice must achieve in order to be accepted.

Example Voting Engines include:

- Absolute Majority Engine: Votes are counted based on the total amount of Omega that has
 voted in favor of a Choice, divided by the total supply of Omega token that is connected to the
 Neuron.
- **Relative Majority Engine:** Votes are counted based on the relative amount of Omega that has voted in favor of a Choice, divided by the total amount of Omega that has voted within the allotted time frame.

Neurons (outlined in **Section 4.3.8 Neurons**) require a singular Voting Engine that is used to determine the acceptance criteria of Choices.

4 3 10 Task/

A Task is designed to represent the smallest possible unit of work that can be completed in Zero. Tasks can be thought of as a miniature contract, where work is performed in exchange for an amount of tokens.

A When a new Task is initiated, tokens are sent to the Task contract, which serves as a form of escrow between the proposer (task creator) and assignee (individual doing the work).

A Task is instantiated with the following parameters:

- **taskName:** a string of the Task's name.
- **owner:** the ethereum address of the Task creator.
- amount: the Task contract amount denominated in Infinity, ETH, or DAI.
- **assigned:** the ethereum address for the who the task is assigned.
- **status:** a string of a valid task status.
- url: a link to Zero Task url, which may include additional Task details and fields.

Tasks can be assigned one of six valid statuses:

- 1. **New:** No one has been assigned to the Task yet.
- 2. **Accepted:** The task has been accepted by a Member but not started.
- 3. **In Progress:** The Task has been started and is in progress.
- 4. **Ready for Review:** The Task is ready to be reviewed, such as with a design, feature or code review
- 5. **Paid:** The Task has been paid to the assigned user.
- 6. **Dispute:** The Task ended in a dispute.

Tasks are a base contract and accompanying interface that can be used to create different types of Tasks with more complicated parameters and decision flows.

4.3.11 **Reputation**/

Reputation systems are a fundamental component of modern internet marketplaces and social networks. Sharing economy apps like Uber and AirBnb rely heavily on Member generated ratings before, during and after Member events, whereas social networks like Twitter and Instagram use centralized processes for Member verification (such as the infamous 'blue check'). In most cases, algorithms track various Member interactions and outcomes to form specific opinions, and dynamically adjust a Members online experience.

More often than not, algorithm's and reputation systems utilized by major internet platforms are opaque, making it difficult (if not impossible) for Members to understand the mechanisms being used to create their online experience. In recent years fake accounts and fake news have become a fundamental problem for the Internet at large, which advances in machine learning is only likely to increase, unless a new type of architecture and business model is introduced. Greater experimentation, transparency, and innovation in reputation systems will be a critical aspect to building better online systems.

For example, a DAO that highly values timeliness might algorithmically assign Omega to Members who score highly on timeliness, while another DAO might algorithmically assign Omega to Members who score highly on quality of work. This way, instead of needing to pass Choices related to Omega on a

case-by-case basis, the DAO can pass algorithms that work repeatedly over time. It's also possible that a DAO's Omega distribution algorithms might factor in data other than a reputation, such as how long the individual has been a Member, how many tasks the Member has completed with the corresponding Network (with maybe something like bonus levels involved), or, to some extent, how much token the Member has staked in support of the Network.

4.3.12 **Trinity**/

Trinity is the central governing body of Zero and Infinity, composed of three central DAOs: The Zero Community, The Zero Guild, and The Zero Council. Trinity represents this structure of three, and includes the Zero Constitution and InnerSource License. For more information, please refer to the Zero Constitution, outlined in **Appendix C**.

4 3 13 InnerSource/

InnerSource is a novel crypto-economic license that governs zOS and the Zero Stack, located in **Appendix D**.

.....

5.0 Governance & Token Issuance/

Zero's internal token structure and governance is made up of three primary and two ancillary currencies each of which are coupled with an accompanying reputation (governance) token and DAO. Primary tokens in this context are defined as tokens that are contractually entitled to a direct contribution from the Infinity Reserve, whereas ancillary tokens are not entitled to automatic distributions. The underlying architecture and token utility of Infinity is outlined in **Section 4.2.4**, **The Infinity Token**. The following table articulates each of the currencies, along with their primary responsibility and governing DAO/Network:

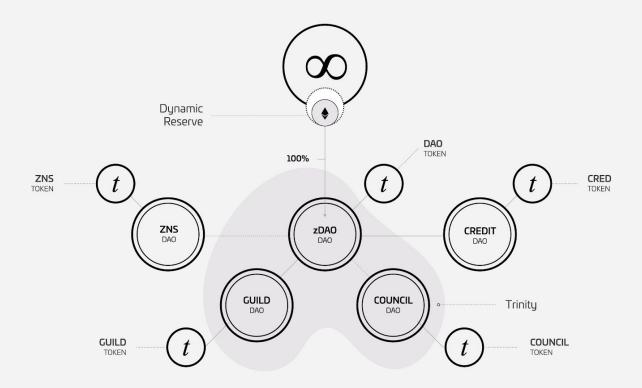
Table: Zero Currencies

Token Name	Symbol	Responsibility	DAO / Network
Infinity	IOI	Operating Zero, facilitating token exchange, purchasing domains, and Credits for the Zero Grid.	zDAO
Zero DAO	DAO	Managing the Infinity Reserve and funding	zDAO

proposals to build the Zero ecosystem.

Zero Guild	GUILD	Managing the core development of the Zero Stack.	Zero.Guild
ZNS	ZNS	Managing the primary ZNS Registrar Contract and setting the stakePrice for new Registries.	Zero.ZNS
Credit	CRED	Managing and maintaining a percentage of the Infinity Reserve to settle the redemption of Credit issued from The Creditor.	Zero.Cred

Figure: Zero's Internal Token & Governance Structure



Similar to the various functioning aspects of a government, the aforementioned currencies each represent an independent aspect of governance for the Infinite Economy. For comparison, Infinity (IOI) itself could be viewed as analogous to the United States dollar, Infinity's Reserve to the Federal Reserve and central

bank, the DAO to the Congress and Senate, the GUILD to the Executive Branch, ZNS to federal and state registries, and CRED to the banking system responsible for the issuance of credit. This is an imperfect comparison, however still useful in understanding the various components Zero provides to support in the operation of a functional society.

Each of these DAO's are fully sovereign and entirely responsible for managing their respective resources and responsibilities according to their own internal governance structure, and with the principles established in the Zero Constitution and InnerSource License.

5.1. **Infinity Configuration**/

Infinity (IOI, ∞) is a Dynamic Token that utilizes ETH as its singular Reserve Token. Once deployed, Zero Members zDAO, Zero's primary governing DAO, may vote to change Infinity's underlying reserve currency or create a more stable basket of currencies using a multi-asset liquidity pool.

Infinity is a Dynamic Token (as outlined in **Section 4.2.3 Dynamic Token**) that will begin with an initial Reserve Ratio of 99%, and decrease at a rate of 3% (the Step Weight), to a minimum Reserve Ratio of 3%. The reduction of the Infinity Reserve requires a successful vote by zDAO, until the Infinite Economy reaches or exceeds a market capitalization of \$9 billion (the Market Cap. Threshold). With each purchase of Infinity from its smart contract, the value of its reserve increases, stabilizing the economy and generating available capital that can be allocated by the community to fund the ecosystem.

The initial configuration of Infinity is as follows:

Reserve Token: Ethereum (ETH)Reserve Ratio: 90% initially

Step Weight: 3%
Initial Supply: 99,999
Initial Reserve: 0.001 ETH
Initial Market Cap: 0.003 ETH

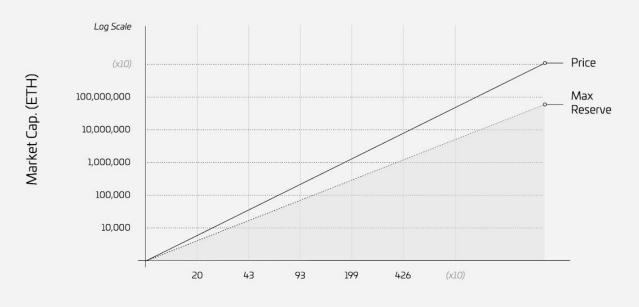
• **Initial Price (for 1 ETH):** 0.00000003 ETH

As illustrated by the initial configuration, the starting market capitalization and purchase price of Infinity is effectively nil (hence, Zero), enabling the fair price of Infinity to be discovered by the market using genuine price discovery. The hope here is to operate in stark contrast to ICOs, where early token holders were taken advantage of using adversarial pricing mechanics.

Infinity's dynamic reserve model plays a key role in Zero's model for incentivizing project and platform contributors, which include the various developers and stewards of Zero Networks, the Zero Stack, Zero Applications, Smart Objects, and other projects governed by the InnerSource license. Given that Zero will be able to draw on Infinity's dynamic reserve to fund its operations as tokens are sold, the project does not need to rely on liquidating an initial supply of token for cash to pay contributors (or on paying contributors with liquid token, which amounts to the same thing economically).

The following illustrates Infinity growth overtime based on Infinity's initial configuration, as defined above.

Graph: From Zero to Infinity*



Infinity Supply (in millions)

Table: Infinity Pricing and Reserve Conditions*

USD Purchased*	3,400,000	34,000,000	340,000,000	3,400,000,000	34,000,000,000
In ETH	10,000	100,000	1,000,000	10,000,000	100,000,000
Market Cap.	\$10,303,031	\$103,030,304	\$1,030,303,031	\$10,303,030,304	\$103,030,303,031
Ending Token					
Price	\$0.505	\$2.360	\$11.040	\$51.638	\$241.529
Reserve (at 90%)	10,000.00	100,000.00	1,000,000.00	10,000,000.00	100,000,000.00
Infinity Supply	20,417,175.95	43,651,146.85	93,324,496.86	199,524,236.24	426,575,253.01

^{*} Using an estimated ETH spot price of \$350.00. Please note that this diagram does not indicate the estimated or future growth of Infinity Tokens, and rather illustrates the algorithmic price that results from different inputs based on the architecture and initial configuration of Infinity. <u>Click here</u> for a link to this tables worksheet.

5.2. Token Distribution/

The zDAO's DAO token will be issued to reward existing and future contributors of the Infinite Economy. DAO tokens holders are responsible for governing excess funds from Infinity's dynamic reserve to develop the Zero Stack and Infinite Economy. All DAO tokens issued for contributions prior to the deployment of Infinity will be locked for a period of thirty-six (36) months.

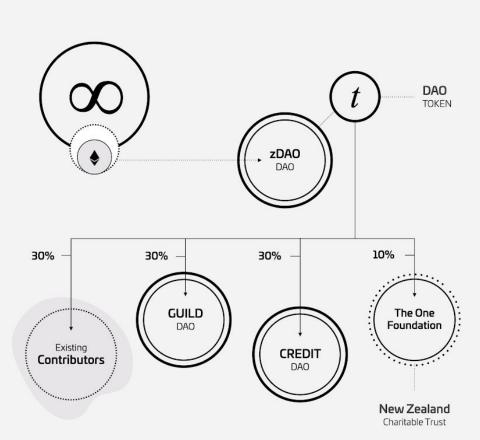


Figure: DAO Token Distribution

A fixed total of **9,000,000** DAO Tokens will be issued as follows:

30% for Existing Contributors: including founders, investors, and advisors, for the many individuals that have supported the development of Zero over the past five years.

30% for the Zero Guild: to incentivize new development projects, including app developers, protocol developers, and early Zero Network operators.

30% to the Credit DAO: to bootstrap the Zero Credit system to enable the redemption of internal hosting Credits for Infinity, in exchange for operating the Zero Grid.

10% to the One Foundation: Ten percent of DAO tokens will be reserved by The One Foundation, a New Zealand-based Charitable Trust, that will in its sole discretion have authority to spend its tokens and fund projects according to its charitable charter.

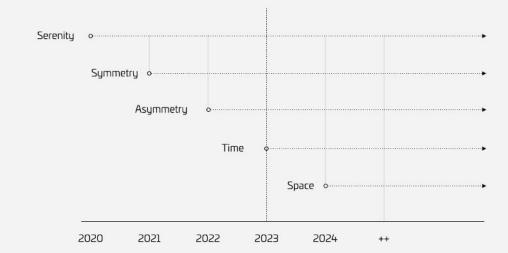
Future token issuances, including amounts, payout criteria, and locking schedules, will be determined solely by the internal governance of each respective DAO and corresponding token governance model, through the successful execution of proposals using Zero's DAO-based governance system.

.....

6.0 **Deployment/**

Zero will be deployed in four core releases, *Serenity, Symmetry, Asymmetry, and Time*, followed by a bonus release, *Space*. These releases will be rolled out in succession between the end of 2020 and beginning of 2024.

Figure: Zero Core Releases



Releases one through four will focus on deploying the core components of the Zero Stack and moving successively towards increased community-based governance, until the entire Zero Stack and associated Networks reach full-decentralization.

Zero's first public release, Serenity, will be deployed on *November third*, 2020 and include the deployment of the following core components of the Zero Stack:

1. The Zero Web application.

Release Schedule

- 2. The Zero Token System and Infinity Token, as outlined in **Section 4.2.1** and **4.2.4**, respectively.
- 3. Trinity, as outlined in **Section 4.3.12** and **5.0**.
- 4. The Zero Name Service, as outlined in **Section 4.3.1**.
- 5. The initial peer-to-peer version of 'the ZeroSphere', Zero's public social network.
- 6. Zero Desktop for Mac, Windows, and Linux.
- 7. Zero Mobile for iPhone.
- 8. The launch of the zer0.io public website, which will include the ability to purchase Infinity, see network stats, register ZNS names, and launch Zero Networks and Tokens.

All source-code as part of this release will be open-sourced under the InnerSource license and be publicly available on GitHub.

Zero's second public release, Symmetry, will include:

- 1. zDAO, Zero's DAO-based governance system, and Tasks, as outlined in **Section 4.3.2**, and **Section 4.3.10**.
- 2. Trinity, Zero's constitutional governance framework and set of DAOs, as outlined in **Section 4.3.12** and **Appendix C**.

3. Zero Mobile for Android.

Zero's third release, Asymmetry, will include:

- 1. The production deployment of zChain (as outlined in **Section 4.1.4**), for both the ZeroSphere and Zero Networks.
- 2. The Creditor, Credits/CRED Token and Credit DAO, as outlined in **Section 4.2.6**, **4.2.6**, **4.3.3**, respectively.
- 3. The Reputation System, as outlined in **Section 4.3.11**.

Zero's fourth release, Time, will include:

- 1. Zero Developer SDK and App Store, as outlined in Section 4.1.9.
- 2. Dyno System, as outlined in **Section 4.1.10**.

Zero's final release, Space (or 'zSpace'), will be the release of a persistent, photorealistic and distributed virtual world. zSpace has been in prototype development in Unreal Engine and will operate entirely through the use of virtual world dynos that are operated by the community within the Zero Grid. zSpace represents the underlying concepts of Zero, such as networks, worlds and universes, in immersive virtual space, that serves a higher fidelity extension of the Zero Member interface.

.....

.....

APPENDIX A: DESIGN INSPIRATION

A.1 Collective Intelligence & Coherence/

Dating far back as Aristotle's <u>Politics</u>, concepts such as <u>Vladimir Vernadsky's 'noosphere'</u> and <u>H.G. Wells' 'world brain'</u> have long considered the presence of a collective form of intelligence. From this lens, everything from the invention of language and writing, to the printing press, is the product of collective intelligence. While the meaning of 'intelligence' itself is highly contested among scholars, establishing a definition is a useful first step in endeavoring to explore the design of more intelligent systems. For the purpose of Zero we will define Intelligence as:

"The totality of possible states within a particular system, coupled with its adaptive capacity to make choices—given a particular set of goals, possible outcomes and enabling constraints."

In other words, Intelligence can be thought of as what humans (and presumably other beings) use to make sense of the world and guide decision making. The nature of Intelligence is in itself abstract—and yet meaningful progress has been made to better understand it empirically. A well-known example is the Intellectual Quotient Test (IQ Test). Ignited by the French Ministry of Education, the first IQ Test was developed by Alfred Binet and Theodore Simon in 1904, in order to better understand the differences between unmotivated and intellectually challenged children. Standardized intelligence testing is a primary pillar of the western academic system, and yet despite its wide-spread utilization, it wasn't until nearly a century later that a group of MIT scientists began to consider a similar question aimed at groups: 'Can we measure collective intelligence?' (Woolley, Chabris, Pentland, Hashmi, & Malone(2010) 'Evidence for a Collective Intelligence Factor in the Performance of Human Groups').

Malone and his team define Collective Intelligence simply as: "Groups of individuals acting in ways that seem intelligent." Across the study of 192 groups, their research found a single statistical factor and set of consistent characteristics for the groups that possessed the highest degree of Collective Intelligence:

- 1. Social perceptiveness—the collective capacity to discern the emotional state of other group members. This spanned interactions using both synchronous and asynchronous communication mediums.
- 2. Equality of participation—the degree to which speaking time was equally shared between group members.
- 3. The proportion of women in the groups—potentially explained by a greater degree of social perceptiveness and emotional intelligence when compared to men.

Collective Intelligence was found to only be moderately correlated to the IQ of individual group members and actually decreased when a group included too many highly intelligent individuals. Collective Intelligence shows up as a well-coordinated, intellectually balanced, and sensory perceptive group.

The groups that possessed a high degree of Collective Intelligence exhibited a high level of 'Coherence'. Definitions of Coherence include: 'having the quality of holding together' and 'acting as a unified whole' Within the research, one interpretation could be that individual sensory perception coupled with balanced participation resulted in greater coherence among members, unlocking greater capacity for group Intelligence. In physics, Coherence is the result of two wave sources being in perfect synchronization, maintaining a constant phase difference, frequency, and waveform. The emerging field of quantum biology has begun to provide clues that are enabling us to better understand how nature uses Coherence to seamlessly coordinate vastly complex biological processes. For instance, the process by which photosynthesis converts the electromagnetic energy of the sun into chemical energy has long been a scientific mystery. It turns out this process is the result of 'quantum coherence', which takes place within the chemical process of chlorophyll. Researchers discovered that thousands of chlorophyll molecules vibrate in perfect synchronization with one another by forming a liquid crystal within the cell DNA. The result is a perfect symphony of a distributed set of biological processes emitting the same phase space and frequency, not dissimilar to a group of rowers stroking in exact synchronization. These liquid crystal structures turn all of the chlorophyll molecules into a type of biological superconductor, thereby enabling the incredibly efficient transmission of sunlight quanta to all necessary molecules. This is referred to as 'resonant energy transfer', and happens when all molecules come into a state of quantum coherence at the same time. These are just some of the ideas available when endeavoring to design systems capable of new levels of Collective Intelligence.

A.2 Artificial Intelligence & Complexity/

Machine learning techniques that are broadly referred to as 'AI', do not account for the complexity that emerges within Intelligent living systems. Neural Networks such as Deep Learning and Generative Adversarial Networks (GANs) are based on a deterministic model that reduces the causal relationship between individual neuron weights and corresponding system state. 'A complex adaptive system is a system in which a perfect understanding of its individual parts does not automatically convey an understanding of the whole system's behavior.' As a result of changing conditions, environment and/or relationships between individual parts, phase transitions within a complex system lead to the creation or destruction of emergent capabilities, not visible from within any constituent part (Alicia Juarrero (1999) 'Dynamics in Action').

A prototypical example of a complex system is a living cell—composed of individual components such as the nucleus, mitochondria, and membrane—each of which is responsible for the execution of specialized tasks. The nucleus stores the DNA, the mitochondria produces ATP, and the membrane is responsible for securing the perimeter of the cell body. Where cells depart from current AI paradigms is the irreducible and nonlinear complexity that is generated by relationships between individual cell modules. For instance, two mutations inside the DNA of a nucleus do not necessarily produce twice the global effect as a single mutation. Complex systems give rise to emergent, irreducible and non-linear system effects.

To better understand how these concepts apply to the field of AI, it is useful to construct a distinction between that of Specialized Artificial Intelligence, Artificial General Intelligence, and Superintelligence:

- Specialized AI: Specialized AI (SAI) represents 'narrow forms of machine intelligence' that succeed within well-defined boundary conditions. Notable examples of SAI based systems include Google's AlphaGo and IBM's Watson. 'Using deep learning techniques with a combination of supervised learning from human expert games, as well as reinforcement learning from self-play, AlphaGo achieved a 99.8% winning rate against existing Go programs and defeated the human European Go champion by 5 games to 0. '(Multiple authors (2016) 'Mastering the game of Go with deep neural networks and tree search)'. Despite the significance of this achievement within the field of AI, modern SAI systems pale in comparison to the most basic aspects of human Intelligence—including abstract reasoning, understanding, and intuition.
- AGI: AGI is the threshold at which AI systems begin to outperform humans across a wide spectrum of tasks in adaptive, complex and unpredictable environments. While many prominent AI researchers believe in the technical feasibility and inevitability of AGI, there exists an ongoing debate as to its practical timeline.
- Superintelligence (SI): Nick Bostrom defines Superintelligence as 'an intellect that is much smarter than the best human brains in practically every field, including scientific creativity, general wisdom, and social skills. This definition leaves open how the superintelligence is implemented: it could be a digital computer, an ensemble of networked computers, cultured cortical tissue or what have you. It also leaves open whether the superintelligence is conscious and has subjective experiences' (Nick Bostrom (1998). "How Long Before Superintelligence?" International Journal of Futures Studies, 2).

As we explore the design of increasingly intelligent systems, the distinction between <u>complex and complicated systems</u> is important. Complicated systems are linear and deterministic. A car is a complicated system; traffic patterns generated by cars and drivers are complex. A thermostat is a complicated system; temperature and weather patterns are complex. From the view of complex systems science, SAI, and AGI/SI provide an elegant mapping to complicated and complex systems, respectively. While there is no simple way to assert this formally, it is difficult to imagine Intelligence as not having some degree of intrinsic complexity. Therefore, current SAI based systems, while useful in specific applications, are unlikely to succeed in getting us to AGI. In order for meaningful progress to occur, we must first ask ourselves how we can improve our understanding of complex systems—including everything from the climate and global economy to modern social networks and the human brain.

A.3. Cybernetic Networks/

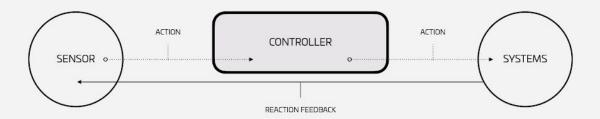
One possible approach to better understanding Intelligent systems is by developing methods to rapid prototype, scale, and analyze cybernetic systems. From here, we may be able to better empirically understand the effects of complex systems under different sets of selective pressures. Norbert Wiener coined the term cybernetics in 1948 as 'the scientific study of control and communication in the animal and the machine.' 'Cybernetics is a trans-disciplinary approach for the exploration of regulatory systems—their structures, constraints, and possibilities.' Its focus is how anything (digital, mechanical or biological) processes information, reacts to information, and how changes occur to better accomplish processing and reacting to information. While these ideas have long been discussed in the realm of

computer science, biology, and social systems, cybernetic theory may hold a unique key to the doorway of AGI.

"The literal definition of cybernetics is of Greek origins (kybernētiké), meaning 'governance'; loosely translated 'to steer, navigate or govern'."

Cybernetics can be thought of as the specific mechanisms to 'govern local and collective aspects of group decision making'. Below is an illustration of a simple cybernetic system, where sensory input, control functions, and actions can be executed by either human or machine agents.

Figure: A Cybernetic Loop



Here 'sensors' gather information from the environment, 'controllers' evaluate the appropriate actions based on internal and external sense-data, and 'actions' lead to making changes in the external environment that result in a feedback loop. As we imagine the various approaches that may lead us towards an AGI type of Intelligence, it is useful to consider why the field seems to have arrived at a local maximum within the domain of neural networks and linear regression techniques. Perhaps the most obvious place to start is with the definition of AGI itself—'Artificial General Intelligence'—and to assess whether the definition in any way artificially constrains the possible search space for new advances.

First, the notion of AGI is incredibly limited in scope. There is no reason to assume that a superior intelligence to that of current human intelligence needs to be in any way 'artificial'. It may be that intelligence is simply an emergent phenomenon produced by complexity—and whether we are operating on the scale of bacteria, cells, or human brains, nature employs the same process to achieve the same result, albeit at different scales of reality. Second, it is not clear what constitutes intelligence as being sufficiently 'general'. Where does 'general' end and where does it begin? More than being general, what makes human Intelligence useful is its 'adaptive' quality. Humans can react to a variety of ever-changing circumstances as well as possess the capacity to continually improve. For these reasons, it may be useful to shift our focus from the goal of AGI to exploring the creation of intelligent systems that include a finer set of minimum requirements:

- 1. **Complexity:** Systems where global effects are not reducible by an understanding of constituent parts.
- 2. **Adaptability:** Systems capable of acting Intelligent within an ever-adapting landscape.
- 3. **Self-Evolution:** Systems capable of autopoiesis--the ability to self-govern and continually improve.

The term 'Adaptive Evolving Intelligence', or AEI, is both more specific and more general than that of AGI, and likely provides us with a more useful waypoint in which to measure progress towards more intelligent systems.

A.4 Effective Choice/

Modern technology has gifted us with a near-omniscient capability—and yet we continue to utilize this power from the shallow confines of our animal instincts. Only recently has a public discourse started regarding the potential negative effects of runaway exponential technology. Epidemics such as the rise of anorexia in young girls (Richard M. Perloff (2014) 'Social Media Effects on Young Women's Body Image Concerns: Theoretical Perspectives and an Agenda for Research'), the proliferation of fake news (Multiple authors (2018) 'The Science of Fake News'), and scandals like Cambridge Analytica—the company responsible for the data breach of 50 million Facebook profiles—are indicators of the broken incentives that underpin the design of modern technology systems.

Our attention is our most precious resource—and any system capable of capturing it will also succeed in seizing its downstream effects—including the belief systems through which we perceive the world, the resulting choices we make, and ultimately who it is we become. Systems that aim to de-dimensionalize reality into oversimplified metrics (such as likes, followers and time-on-site) will invariably lead to unintended externalities. The future is a byproduct of how we choose to allocate our collective present attention. As tools to change minds and alter reality become available to more and more people, we must begin to shift our attention from that of purely 'if' questions to 'should' questions. Just because we may be able to create Intelligence that exceeds human Intelligence, should we? Just because we can develop and distribute gene-editing technology, should we? Just because we can create new monetary systems and governance structures, should we? Current technology systems do not include processes for such questions at the level of architecture. In evolutionary biology, it is a well-observed phenomenon that species generally evolve to meet the fitness payoffs of a given landscape, rather than act on empirical truth (Donald Hoffman et al (2017) 'Fitness Beats Truth in the Evolution of Perception').

"Evolution has shaped us with perceptions that allow us to survive. But part of that involves hiding from us the stuff we don't need to know. And that's pretty much all of reality, whatever reality might be." —**Dr. Donald Hoffman**

On an evolutionary timescale, the emergence of new forms of Intelligence may be a result of discarding or compressing information within preceding layers of the evolutionary stack, in turn giving rise to more efficient capabilities for achieving particular tasks. Applied to the average computer Member—it is far easier to employ the use of a modern graphical Member interface (GUI) rather than to write machine code

or solder electrical circuits for everyday work. It may be that the further we move up the evolutionary stack, the further we move from the way reality actually is. Emerging digital platforms and AEI-based systems may be the next layer of evolutionary abstraction that will fundamentally alter human perception.

This fundamental change to how we make sense of reality poses an existential threat to collective choice-making if not designed and deployed in harmony with nature. Never in recorded history have we possessed the capacity to alter reality in fundamental ways by an increasing number of people. Up until now, nature has provided the circumstances to guide evolutionary choice-making. As exponential technology enables smaller and smaller groups of people to make choices that impact everyone, including other humans, beings and ecosystems, we must learn to deeply consider what is, in fact, an 'Effective Choice' when designing systems capable of such far-reaching consequences (Forrest Landry (2001) 'The Effective Choice'). What may be called for is a 'meta-psycho technology'—a tool to determine how it is we build tools—to ensure our tools do not lead to unintended consequences. Similar to how institutional religious values underpinned the sub-structure of modern law and society, a new set of ethics and corresponding technological architecture is required to design, govern and improve systems capable of deeper responsibility.

"We need to create a meta-psycho technology for evaluating how we are designing our psycho technologies." —**Jordan Hall**

A psycho-technology can be thought of as a type of 'meta-heuristic'. 'In computer science, a meta-heuristic is a higher-level procedure or heuristic designed to find, generate, or select a heuristic (such as a partial search algorithm) that may provide a sufficiently good solution to an optimization problem, especially with incomplete or imperfect information or limited computation capacity.'

Psycho-technologies on the scale of modern exponential systems, including social networks, blockchains, virtual reality, AGI, and crisper, put us on an entirely new type of evolutionary fitness landscape with a novel set of characteristic dynamics. Part of the process of designing new types of Intelligent systems not only need to include a new set of ethics and methods for Effective Choices but also needs to be capable of responding to the rapidly changing reality we are finding ourselves in. New axioms must be created that enable us to step outside of current design paradigms, social systems, and beliefs to bring about non-rivalrous forms of collaboration and coherence within hierarchies, democracies, markets, communities, and ecosystems.

APPENDIX B: THE FUTURE YESTERDAY

B.1 Introduction/

This section, titled The Future Yesterday, serves as a historical overview of key technologies, trends and concepts upon which Zero is built. Key areas include: Web 2.0, The Sharing Economy, Digital Productivity Tools, Automation & Asymmetry, Reputation Systems, Data Privacy & Security, Internet 3.0, Blockchain, DAOs, Market Makers & DEXes, Agent-centric Systems, Immersive Reality.

B.2 Web 2.0/

Web 2.0 can be largely attributed to three technology advances: centralized 'cloud' computing, Member-generated content, and mobile computing. Powered by the proliferation of bandwidth and cheap compute, Web 2.0 enabled startups to take on multi-nationals to reinvent global industries. Public clouds created a wave of on-demand services far more efficient than traditional client/server technologies. Cloud-based systems moved the burden of backups, security, and network management from the Member to the service provider, and they enabled new conveniences such as automatic software updates and remote access. 'Free' advertising-based websites and software-as-a-service (SaaS) business models made content and software far more accessible for average Members and businesses. These systems created a free alternative to newspapers, magazines, and analog content, and SaaS removed expensive implementation and licensing fees associated with traditional software.

The speed with which Web 2.0 services took hold is reflected in the rapid growth and consolidation of digital advertising. In the US, for example, digital ad spend grew from \$130 million in the first quarter of 1997 to \$29.9 billion in Q2 of 2019 and totaled \$107.5 billion for full-year 2018, a 21.8% YoY increase. Facebook and Google accounted for 77% of this spending, illustrating the exponential power of network effects and Member-generated content intrinsic to Web 2.0 architectures. These systems brought with them a new set of technology-powered unit economics, vastly reducing the cost of generating content and harnessing Member attention. Imagine the per-unit cost to produce a single piece of content for a modern search engine or social network compared to that of a traditional newspaper or TV station. By leveraging the vast power of Member-generated content—such as emails, personal profiles, photos, videos, social media shares, and even search queries—companies were able to build platforms orders of magnitude more efficient than any prior medium, while in the process making our personal data among the world's most valuable resources. At the time of this writing, Apple, Microsoft, Amazon, Alphabet (Google) and Facebook represent nearly \$5 trillion USD in collective market capitalization and represent five of the most capitalized six companies in the world).

B.2 The Sharing Economy/

Technology breakthroughs are often the result of new and improved mechanisms for coordination. The printing press coordinated news more efficiently than the postal system and handwritten letters, television

and radio coordinated information more efficiently than the printing press, and today social networks and digital productivity tools coordinate information far more efficiently than TV and radio. Even low-level innovations consistent with <u>Moore's law</u>—such as the transistor or microchip—are the result of the more efficient coordination and flow of information.

From the foundation laid by Web 2.0, the most recent wave of fast-growth startups have created what is often referred to as the <u>sharing economy</u>. Broadly speaking, the sharing economy can be thought to encompass the collaborative and 'gig' economy, as well as crowdfunding, crowdsourcing, and coworking. PriceWaterhouseCoopers estimates the sharing economy will grow from \$15 billion in 2014 to around \$335 billion by 2025. Whereas Web 2.0 focused predominantly on the coordination of content within applications like search, encyclopedias, shopping, social media, music, and video, the sharing economy focuses on the coordination of latent and often physical network resources such as transportation, housing, education, and work (think Uber, Airbnb, Udemy, and Upwork). Unlike Web 2.0 companies that focused on providing primarily digital services to end-Members, companies in the sharing economy move real-world assets such as vehicles, accommodation, teachers, and workers into the digital realm. Instead of selling software platforms that serve to digitize industry incumbents as with SaaS, companies in the sharing economy connect service providers directly to consumers, greatly reducing coordination costs through the entire supply chain (as A16Z partner Chris Dixon notes in Full stack Startups').

The sharing economy is an example of how embedded the digital world is becoming within everyday life. With the ubiquity of digital connectivity, the recent advent of blockchain, and the age of AI on the horizon, there is a near-infinite number of vectors that can be used to more intelligently coordinate collective resources and activity. Technology advances historically have enabled humans to work less and with greater access to opportunity and resources. In 1929 as compared to 1890, the average urban worker put in one less day of work per week while bringing home three times as much in pay. The number of families burdened by farm life declined by half, and families received access to luxuries that were once unimaginable: nine out of ten Americans suddenly had electricity and indoor plumbing, four-fifths had automobiles, two-thirds had radios, nearly half owned refrigerators, nearly half owned phonographs, infant mortality fell by two-thirds, and life expectancy increased by 20 years. (2019 Digital History) As the flow of information became more efficient, the less work produced greater returns and radically reduced the cost of new capabilities. Similarly, in the 21st century, as systems of coordination become more efficient for all types of resources, assets previously unused or idle will become liquid, providing investors and creators alike with new income streams and flexibility.

B.3 Digital Productivity Tools/

Software companies formed during the SaaS era focused primarily on the *coordination of information* through the development of systems of records (think <u>Salesforce</u>, <u>Workday</u>, and <u>Intuit</u>). Software companies formed during the rise of the sharing economy focused primarily on the *coordination of activity* (think <u>Github</u>, <u>Slack</u>, <u>Telegram</u>, <u>Asana</u>, <u>Trello</u>, and <u>Zoom</u>). Both types of tools not only make organizations more operationally efficient, they also give organizations the novel capacity for high-quality remote coordination, enabling people to work together across the globe and asynchronously, no longer confined to single points in space and time. This has major implications for global productivity. One is the

collective productivity gained by no longer needing to lease central offices, engage in daily commutes, or build data-driven coordination processes. Another is access to new talent pools: it used to be that not being located in a center of excellence, such as New York for finance or Silicon Valley for technology, put companies at a significant disadvantage in terms of their capacity to attract top talent. This requirement is declining as it becomes easier to coordinate remotely. Workers, too, benefit from the possibility of remote work: in addition to flexible hours and locations, high-quality talent is no longer bound by the salaries available within the geographical proximity of a daily commute.

The promise of such productivity gains have created a lucrative market for companies focused on collaboration. The enterprise collaboration market was valued at \$32.74 billion in 2018, according to one analysis, and is 'expected to register a CAGR of over 10.7% during the forecast period (2019 - 2024).' Slack, a team communication tool founded in 2009, became the fastest-growing business application in history, growing to 8 million daily active Members and 3 million paid Members as of early 2020, with a valuation of roughly \$12 billion. More recently, Notion, a simple collaborative note-taking app founded in 2016, reported an \$800 million valuation and more than one million Members. Given the weight human productivity has on GDP, even minor improvements to the coordination of human activity can have significant effects on the global economy.

And we're still at the very beginning of the digital productivity revolution, much as Web 2.0 was in 1997 according to digital ad spend. Remote work still represents a small fraction of how work happens today: in 2018 there were 4.3 million remote workers in the USA, or just 3.2% of the total workforce, and only 7% of companies made remote work available to the majority of their employees (Global Workplace Analytics (2019) Telecommuting Trend Data). Most economic activity still happens primarily within physical space, such as an office headquarters or factory, and remains largely bound by industrial principles like being employed by a single company at a time, receiving a time-based salary, and working for a company that has unilateral ownership over all intellectual property. As work and creative output become more distributed and more measurable, new compensation and coordination systems will change the nature of how workers are incentivized and rewarded, as well as how they share in collective choice-making.

B.4 Automation & Asymmetry/

"The future is already here -- it's just not very evenly distributed."-- William Gibson

The forthcoming era of automation will likely supersede the transition from the industrial to the digital age, in terms of magnitude of change. This is a time in which autonomous technology will replace a vast array of tasks currently performed by humans within the global workforce. Much as the microchip, the web browser, and the smartphone served to usher in the digital and Web 2.0 eras, the convergence of software powered by AI, the internet of things (IoT), and fifth-generation (5G) wireless infrastructure are creating a threshold that is about to move us firmly into the automation era. The AI market is expected to grow from \$16.06 billion in 2017 to \$190.61 billion by 2025, that IoT will grow from \$190 billion in 2018 to over \$1.1 billion by 2026, and that the number of 5G connections will grow to between 20 million and 200 million by next year.

The inevitability of autonomous driving is here and yet it remains unclear how this will impact the working class and fabric of modern society. Truck driving, for example, is <u>currently the most popular occupation in 29 US states and accounts for 3.5 million jobs</u>. But savings related to automated freight delivery are expected to total \$168 billion, including \$70 billion in reduced labor costs (as well as \$36 billion in reduced accident costs and \$27 billion in other efficiencies such as equipment utilization). These are strong incentives for automation and yet represent only a single use case for how it might be applied. As AI, IoT, and 5G technology becomes more advanced, automation has the potential of replacing or at the very least augmenting not just basic tasks like driving, making purchases, or writing a simple email, but even more complex processes such as developing software or making art.

Non-linear improvements to coordination create non-linear shifts to the allocation of resources within an economy and, as a result, demand the reconsideration of the fundamental premises that underpin societal structures. Core assumptions need to be rethought, new challenges need to be addressed, and fundamental questions need to be asked: What do we value? How do we define progress? What is the optimal model for governance? How do we make choices? How should capital be allocated? A seismic shift in technology without corresponding reconsiderations in societal patterns has the potential to result in massive power asymmetries and the creation of complacent, prideful, "dominant minorities", as the historian Arnold Toynbee noted in his review of the downfall of civilizations in 1961's *A Study of History*. (Clayton Christensen describes similar phenomena from a different angle in the "Innovator's Dilemma") The automation era has the potential to result in a better future for all or lead us toward untenable power concentrations that risk societal dissolution.

For a preview of this kind of power concentration, we need look no further than the asymmetries that have emerged from Web 2.0 systems. Modern creative minorities such as Facebook and Google have generated giant profits that have not been distributed proportionally to those who have contributed value to those projects, including employees and content-generating Members.

In the case of the employees, Web 2.0 companies still use a fairly linear formula to account for contributions, much as companies did in the industrial era: employees get paid primarily on the basis of their time, and differences between employees in terms of their contributions per time are assumed to be relatively small, as it was in the industrial era when productivity was bound by a person's physical limits. One person can only screw on so many, say, bottle caps in an hour, and it made sense for all of the outputs – the bottles with caps screwed on – to be entirely owned by the company. But what if, by way of analogy, an employee's contribution is not to screw on bottle caps, but to invent a technology that assists in bottlecap-screwing and amplifies everyone's output by double, infinitely moving forward? Should that individual be compensated differently, perhaps by retaining some stake in the intellectual property he or she developed? Despite the recent transition from a primarily industrial to a digital economy, companies have continued to utilize linear-based compensation structures, irrespective of the fact that individuals and teams now have the capacity to produce exponential output.

The Members of Web 2.0 platforms are in a similar position as employees, who've contributed value in the form of Member-generated content and Member-data that financially rewards platform owners and investors

The arbitrage opportunity created by low costs, such as 'free' Member data, cheap servers, and fixed-salaried knowledge workers, coupled with relatively novel exponential outputs, such as software, are what have made technology companies the largest and most profitable entities in the history of the world. And, the automation era promises even more potential for the concentration of power. This expanding potential for arbitrage by platform owners from the value created by others – and the resulting potential for an increasing power asymmetry – is what Zero calls the 'asymmetry problem'. This idea underscores the necessity for new models of governance, societal structures, and value distribution in order to ensure that technological progress evolves in harmony with nature and society.

B.5 Reputation Systems/

For automated systems and societies to transcend the asymmetry problem, effective reputation systems are essential. The level of 'automation' able to be sustained by a particular system can be determined by the degree to which the system is capable of increasing: 1) the speed at which it can make choices, 2) the quality of its choices related to specified goals, and 3) the capacity to improve itself over time. Thus, the ability to evaluate prior experiences and make judgments regarding future events lies at the core of a viable automation system. Reputation systems help fulfill that requirement by serving as a proxy for 'trust' in digital systems. They are common in Web 2.0 communities, social networks, and marketplaces, such as when you purchase an item on Amazon, rent a house on Airbnb or flag inappropriate content on Reddit. Reputation systems generally work by annotating choice-making events with digital scores. These algorithms play a crucial role where trust is required to enter into a digital relationship or agreement, whether that is interacting with a stranger on Twitter or purchasing an item on eBay. Reputation systems are the rails that enable Member-generated content to be effectively coordinated in online systems. Google, Facebook, and Instagram use hidden reputation algorithms to determine what content is sequenced in your search results and newsfeed, intentionally concealing their mechanics. Amazon and AirBnb use relatively simple public review systems for rating products, services, and providers.

Trust is not just important in determining whether or not to take a particular action; it may be the rate-limiter that determines how quickly choices can be made in relationships, autonomous systems, and business. A classic book to make this case is 'The Speed of Trust' by the late Steven M.R. Covey, which suggests that progress moves at the speed in which trust is established between relevant parties. More recently, Harvard neuroscientist Paul J. Zack found that, 'compared with people at low-trust companies, people at high-trust companies report: 74% less stress, 106% more energy at work, 50% higher productivity, 13% fewer sick days, 76% more engagement, 29% more satisfaction with their lives, 40% less burnout'.

In order to understand the challenges with current online reputation systems, it is worth examining how trust is gained or lost in reality. Trust grows when individuals have positive experiences consistent with their expectations and decays under the opposite conditions. A fundamental requirement for trust to grow is some degree of investment in the first place in the relationship, such as goodwill or faith – some kind of starting assumption that one will not be deceived or harmed relative to one's expectations, which might be called a kind of trust in itself. ('Trust, but verify,' goes the Russian proverb, ironically appropriated by

Ronald Reagan regarding the Cold War's nuclear disarmament treaty.) Trust cannot be easily established in a starting climate of non-trust, and therefore can only be established via the extension of trust.

On the measure of mimicking this aspect of trust-building in the real world, Web 2.0 reputation systems generally fall short in that they incentivize participants mainly toward avoiding bad scores rather than extending 'starting trust'. Forrest Landry writes: "In a [typical modern] reputation system, the basis of trust is in proportion to the degree that people are afraid of getting bad scores, of becoming ineffective and impotent -- unable to participate in the process of choice -- and so there is an incentive to be honest. There is something deeply wrong about a person who is only being honest -- doing the right thing, or being super careful, empathetic, etc. -- because they are afraid for their reputation, for imposed consequences, for punishments, etc. Such unconsciousness or secrecy as to their true motivations, their true basis of action, is a kind of sociopathology. Rules of law, prisons, etc. are about forcing purely self-oriented people (non-community-oriented people) to be slightly more community oriented, but they do not create trust."

Web 2.0 reputation systems also fall short in their oversimplification of value metrics compared to real-world scenarios. They may be useful for providing enough data to make simple choices like ordering an Uber or selecting a vendor on eBay, where the vectors of trust associated with a particular choice are limited in scope, but they are unlikely to sufficiently support more complex choices such as whether to rely on someone's expertise around a sophisticated topic or whether to involve someone in founding a venture.

Yet another challenge is the high degree of variability in each participant's interpretation of the point system itself: 'four stars' means one thing to one person and something else to someone else.

Zero considers all of the above challenges to be, collectively, the 'reputation problem'.

The innovation of blockchain seeks to solve the reputation problem by performing a kind of end-around, creating what are in essence trust-less systems that replace the need for trust with code-based protocols. This kind of architecture works well to enable the trustless processing of transactions in a ledger, but it does not provide immediate solutions to the reputation problem at the level of application for human coordination. In order to increase the effectiveness of distributed choice making, as well as to enable effective individual choices to be made within distributed autonomous systems, entirely new reputation paradigms are required for increasing the speed of trust as we enter the age of automation.

B.6 Data Privacy & Security/

After decades of 'free' Internet services, Members are beginning to understand the actual cost of 'free', waking up to the value of their personal data and how in many cases it is being used against them to distort the information they receive and manipulate the choices they make. A primary business model of Silicon Valley and many technology companies has been to give Members access to 'free' services, such as is the case with Facebook and Google, in return for vast amounts of personal information and even special rights over personal photos, videos, emails, documents, and posts.

The extent to which exponential technology is impacting our 'sensemaking' is something that we are only beginning to gain an awareness of. <u>Tristan Harris</u>, a privacy advocate and Silicon Valley technology entrepreneur who was previously a Design Ethicist at Google – along with other Silicon Valley technologists such as <u>Jaron Lanier</u>, <u>Aza Raskin</u> and Justin Rosenstein – have very publicly begun to talk about the ethical challenges with the design of modern Internet services, and how this technology may be in fact 'downgrading humans'.

Many in the Western world are familiar and concerned with privacy issues through some of the revelations of the early 2010s, such as when US intelligence operative Edward Snowden revealed the extent to which the government was unlawfully collecting data from its citizens in the form of recorded data, phone calls, and online transactions. But what Harris and others touch on is potentially more dangerous and disconcerting than the Snowden release. They reveal not only that data is often collected far in excess of what is required to provide an online service, but that modern technology companies are using that information to directly interfere with human choice making. Similar to how it was not until recently that a large number of people began to appreciate the true value of their online privacy and identity, these kinds of secretively manipulative algorithms are poised to cause increasing alarm as individuals begin to understand how they alter our programming and neuro-biology.

Whilst the notion of 'mind control' or a mass surveillance state has generally only taken place within the realm of science-fiction (such as Orweill's 1984), these stark realities are in many ways already here (but not yet distributed), and can only be altered if through a transition to omni-considerate digital systems capable of managing a delicate balance of power. New distributed technologies and business models, along with accompanying new values and ways of accounting for value, are required, with the collective result being a transition from digital systems based on control and value-extraction to ones based on liberty and self-determination.

B.7 Web 3.0/

The web has had remarkable success in reliably connecting a large percentage of humanity, and yet many technology leaders acknowledge its architecture to contain <u>fundamental flaws</u>. These issues relate to a broad range of security and reliability challenges, coupled with Member privacy, protocol governance and general incentive issues. <u>Tim Berners-Lee</u>, the creator of the worldwide web himself, as well as many others, are actively working to actively solve these issues with a variety of approaches. A primary critique of the modern-day web is how centralized power and influence has become, with the majority of collective attention being funneled through and directed by only a handful of corporations. This level of power is unprecedented for private-owned corporations and represents a threat to the underlying systems of governance that have enabled modern constitutional liberties, as we see in the interference of elections around the world through disinformation spread systemically on social media. While few would say that the Web 2.0 companies themselves had malicious intent, their architectures have nonetheless created asymmetries of power across a wide variety of relationships, such as between creators and companies, and now, increasingly, between citizens and governments. In China, for example, the <u>social credit system</u> has made it easy for the government to track a tremendous number of citizen's interactions.

Members and creators alike are beginning to recognize the economic potential for the design of a new internet and associated technologies for communicating and exchanging value. What's known as Web 3.0 can be understood as a collection of such technologies combined with corresponding ideological shifts in how systems are governed. AI, blockchain, DAOs, dApps, distributed/peer-to-peer systems, agent-centricity, and zKSnarks are examples of possible components within an emerging stack that is capable of transitioning the world into a safer, more private, and more egalitarian web. While there is no formal definition or boundary on exactly what Web 3.0 is, it generally describes the shift from highly centralized services to decentralized or distributed systems characterized by security, privacy, individual ownership of data, and self-determination in choice making.

B.8 Blockchain/

Blockchain is one of the fundamental layers of the Web 3.0 stack whose inception was marked by the release of Satoshi Nakamoto's 2009 <u>Bitcoin whitepaper</u>. Bitcoin illustrated the technical feasibility for individuals to create a viable alternative to government-backed monetary systems. Bitcoin served as the foundation for the first major 'chord', or important act, in blockchain innovation, which came to quickly include a number of 'alt-coins' based on the original Bitcoin protocol, such as <u>Namecoin</u>, <u>Litecoin</u>, and <u>Monero</u>. The significance of Bitcoin was not just the creation of a new monetary system, but rather a much deeper unlock in possibility space: the power to directly exchange value independent of third-party intermediaries. Similar to how the <u>TCI/IP</u> protocol underpinned an 'internet of information', the consensus protocol presented in Bitcoin has provided the foundation for a new type of internet—an 'internet of value'.

The second major chord in blockchain innovation began with Vitalik Buterin's invention of Ethereum in 2014 and served to further illustrate the possibility of an internet of value. Ethereum's decentralized, consensus-based world computer allows developers to create and run immutable programs, or 'smart contracts', entirely through a network of decentralized nodes. Similar to Bitcoin's alt-coins, a number of Ethereum-based projects and protocols soon followed, collectively raising billions of dollars in 'initial coin offerings' (ICOs). The bubble crashed in 2018, when by September the MVIS CryptoCompare 10 index had lost 80 percent of its peak value. Many considered this bust analogous to the dotcom bubble around the turn of the millenium, where retail investors rushed into the market before organizations had proven their business models or released products. Despite the crash and vast number of ICO failures, many companies born during that time have created a thriving blockchain industry, creating billions of dollars of value for protocol creators and Members in the process.

The invention of decentralized money (Bitcoin) and more recently decentralized computing (Ethereum) are both powerful examples of trustless coordination, but they still represent only the beginning of what blockchain can enable. Just as it took decades following the development of personal computers to move from machine code through stages of <u>unix-like</u> operating systems, to <u>DOS</u>, <u>GUIs</u>, and <u>web browsers</u>, blockchain has not even yet achieved widespread adoption beyond a primarily technical audience, and it remains uncertain what blockchain-based products or 'killer-apps' will create the next major chord in the Blockchain story. Regardless, at this point there is little question that blockchain technology will be an essential part ushering Web 3.0 into reality.

B.9 DAOs/

Decentralized Autonomous Organizations (DAOs) may very well be the next major chord in the blockchain progression. Whereas Bitcoin illustrated the possibility of a decentralized money and Ethereum a decentralized world computer, DAOs represent the possibility for fully decentralized and autonomous coordination (or work). DAOs are an alternative to the legal, capital, and operating structure of the corporation, a form of organizing that has remained meaningfully unchanged since its rise in dominance nearly half a millennium ago. Instead of existing as a legal entity within a physical jurisdiction, such as with a limited liability company or a not-for-profit, DAOs are governed by rules instantiated solely within their code that live within a decentralized network. Chris Burniske, the author of Cryptoassets, states that DAOs and crypto networks, in general, represent "supranational protocols in which human activity can be governed without nation-states, without corporations."

One reason for the excitement of DAOs is their potential to solve longstanding game-theoretic issues in economic systems, such as 'The Tragedy of The Commons', <u>initially proposed</u> by the British economist William Forster Lloyd in 1833. The tragedy of the commons occurs when individuals disregard the welfare of the whole in pursuit of their own personal interests, similar to the incentives that emerge in 'Prisoner's Dilemma'-type problems. These kinds of problems, in essence, stem from the fact that rational acting and individuated incentives prevent people from doing what is best for everyone, which eventually impacts everyone negatively, including the person acting from their own rational self-interest in the first place. The combination of these intrinsic motivators (i.e., human nature) and extrinsic motivators (e.g. economic systems) can lead to massive-scale problems like the near-collapse of the global financial system due to collateralized debt obligations (CDOs) in 2008, or the potentially irreversible harm to earth's ecosystems caused by industrial pollution.

Not unlike how written law enabled more sophisticated models of governance, including that of modern capitalism, socialism and communism, DAOs offer the promise of a 'more programmable' substrate for the creation of new economic systems and finely-grained incentivization mechanics more closely attuned to effect positive outcomes.

For a simple example of how a simple change in mechanics can fundamentally change the risk profile of certain choices, we can look at the central innovation of <u>Kickstarter</u>, a Web 2.0 fundraising platform for startups. A key challenge for early participants, purchasers and backers in startup projects is the fact that the project may never generate enough cash to meet the true costs of development and manufacturing, creating the very real possibility of losing everyone's funds in the process. Kickstarter decreases the risk of this type of event via their 'fundraising goal' mechanic, which ensures a reasonable funding amount is met in order for any funds at all to be released to the project creators. It is not hard to imagine this mechanic being written directly into the code of a DAO, where community members of the DAO vote to allocate funds to different ideas that must meet a funding viability threshold; this mechanic alone would cause a DAO and its participants to behave quite differently than how most government programs and corporations work today. But even more so, the Kickstarter example serves to illustrate how simple mechanics can unlock significant 'coordination potential' within networks and communities, suggesting

that DAOs – given the unlimited potential for building new mechanics into their smart contracts – can play a central role in the innovation of Web 3.0 organizational forms.

B.10 DEXes & Market Makers/

Blockchain invites the possibility of an 'internet of value' using cryptographic tokens as stores of value and mediums of exchange, operating free from middlemen and third parties. It's become a common exercise to Imagine the tokenization – i.e. the crypto-economically enabled shared ownership – of almost everything, from art to source code to the stewardship of earth's natural resources. Such thought experiments may seem fanciful, but it's worth considering how crazy it might have sounded in the mid-1990s to imagine a future where most every business, product, and community had a website. Two key mechanisms are required in order to establish a functioning market for such a crypto-economic future: exchanges and liquidity. An exchange is a physical or digital location in which buyers and sellers are able to connect, trade and transfer the ownership of assets. Examples of exchanges include everything from the New York Stock Exchange (NYSE) to eBay and your local farmers market. Liquidity is a measure of how quickly convertible a particular asset is into useful goods and services, without meaningfully impacting the underlying price of the asset. Government-issued currencies such as the US dollar are highly liquid since there are a large number of individuals and businesses willing to accept dollars in exchange for whatever they might be selling. Private company stock, on the other hand, is generally illiquid, given that there is usually not a large enough pool of buyers and sellers to easily facilitate its purchase and sale.

One of the fastest-growing areas of the blockchain industry, ironically enough, has been 'centralized' exchanges selling 'decentralized' crypto-currencies. <u>In just 143 days, Binance</u> became the third-largest cryptocurrency exchange after the release of its whitepaper on June 22, 2017, making it one of the fastest private companies in history to achieve a private valuation of \$1 billion. Despite the popularity of centralized crypto exchanges like Binance, their architectures are antithetical to the core tenets of decentralization: centralized servers and centralized organizations store sensitive personal data, including crypto-wallet information, and they also face regulatory risks associated with being headquartered within particular legal jurisdictions. For these reasons, centralized exchanges likely only represent a temporary stop on the road to decentralization.

Decentralized Finance (DeFi) protocols are already on the rise, already accounting for roughly 3% of Ethereum's entire supply (over 3 million ETH), thanks to the maturation of decentralized exchange(DEX) protocols like 0x, Uniswap and Bancor, lending protocols like Compound, stable currencies like MakerDAO's DAI, and synthetic asset protocols like Synthetix. DEXs function similarly to centralized exchanges but they differ significantly in that their transactions are executed entirely within Smart Contracts. This enables the exchange of cryptocurrencies to take place entirely within the distributed computing environment of the blockchain, avoiding the risks of centralized exchanges.

The step beyond DEXs may be a future where exchanges are no longer required at all. Decentralized Automated Market Makers (DAMMs) are on-chain mechanisms that create liquidity directly between tokens, removing the need for an exchange or order book altogether. To better understand DAMMs, it is worth taking a look at one of the fundamental problems that exchanges solve, sometimes called the

problem of the double coincidence of wants. This problem describes the issues that arise in a barter system, namely the low probability of buyers and sellers 'wanting' the same thing at the same time. Modern economic systems solve this by creating a 'medium of exchange' to store the value for wants and needs, as well as by creating financial exchanges, meeting places where items can be priced and purchased, using the medium of exchange as a currency. Even with functioning mediums of exchange and markets, however, it's still not necessarily the case that sellers of a certain asset can find buyers of that particular asset, or that buyers can find sellers. Enter the market makers: companies or individuals that offer to both buy assets from those looking to sell and sell the same assets to those looking to buy, hoping to make money on the difference between the two. Market makers provide an essential service to financial markets and especially to assets with low trading volumes, such as is often the case with new projects and currencies.

DAMMs utilize an internal market maker to make their tokens themselves convertible into other assets. They achieve this through the use of 'bonding curves', mathematical formulas that pre-define the price of a token precisely according to its and supply. This concept began to generate interest with the release of Bancor's whitepaper in early 2017 and with Simon de la Rouviere's 'Exploring Continuous Token Models: Towards a Million Networks of Value'.

Among the benefits of bonding curves is that they enable 'continuous price discovery', eliminating the need for projects to establish a fixed valuation prior to token launch. This solves a critical problem illuminated by the 2017/2018 ICO bubble: projects generally sold a fixed amount of tokens at a fixed price, and in many cases used a 'hard cap' to entice early investors to move quickly, artificially propping up demand then crashing afterward. In the world of venture capital investment, funding stages with names like – seed, series A, series B, and so on have emerged as an attempt to mitigate price risk at the various stages of company maturity. While this system is partially helpful for investors to determine quickly a company's stage and specialize in investments into companies at various stages, it's still an imprecise process for determining a company's value, and it is subject to the same kind of speculation and fever that occurred during the ICO craze. Bonding curves take a very different approach: they decouple fundraising entirely from single ICOs or individual venture rounds, instead allowing the price to increase continuously as more of the asset is purchased (or decrease as more of it is sold back into the curve and removed from the market), according to an immutable formula fixed by a smart contract.

Now we can begin to see how DAMMS and bonding curves have the potential to eliminate the need for exchanges: with tokens effectively self-liquidating, they can be easily exchanged for other tokens that are also self-liquidating. This enables conversion to happen directly, seamlessly, and within the security and trustless environment of the blockchain, without the need for a centralized exchange or even a DEX.

B.11 Agent-Centric Systems/

Blockchains first and foremost exist to solve the problem of validating transactions in an objective and reliable way – of achieving 'consensus' within a distributed ledger – without needing to place trust in any centralized third parties. The recent growth of DeFi applications provides long-overdue evidence for the practical utility of blockchain protocols, but questions actually remain as to whether or not blockchains

will be the best architecture for scaling distributed applications. Arthur Brock and Josh Zemel, of the 'agent-centric' project Holochain, provide a sobering view of blockchain architectures in '<u>Unenclosable Carriers and the Future of Communication</u>'. They make the case that mining and staking pools are ultimately subject to the same power asymmetries that blockchains propose to solve, since they necessarily empower individuals outside of any given interaction – and necessarily empower some individuals more than others – to determine what is considered valid and added to the shared record.

Unlike blockchains that require global consensus for every transaction, agent-centric systems empower participants to manage their own transaction history locally, allowing them to opt in (or out) to syncing data with other agents to build shared spaces of coordination and information. What makes the information in shared spaces reliable is not consensus but rather protocols that make it impossible to alter information on one's own transaction history after it's been written.

One promise of agent-centricity is to make it especially easy for Members to control and manage their own personal data. This is very different from the paradigm of centralized cloud-computing, where Members are generally required to give control of their personal data utilized or generated within an application in exchange for usage of the application. While a goal of blockchain has been to transition out of this centralized architecture to a decentralized ledger on which Member data resides and can be accessed according to protocols set by smart contracts, blockchain still requires that Member data be stored in third-party space outside of one's own control – with control ceded to however the blockchain's consensus mechanism distributes power. Agent-centric systems represent the space beyond centralized and decentralized architectures, having Members and trust groups store their own information locally, rather than having to rely on other people's computers or public infrastructure.

Peer-to-peer (p2p) technologies combined with blockchain technologies may create the architecture needed for a truly distributed future. A notable example of an agent-centric system is <u>Git</u>, the distributed version control system that forms the basis for <u>GitHub</u>, the world's largest online source-code repository. Through the use of an encrypted private key, Git enables Members to append their own version 'of version history' through a chain of 'commits' that exist within a source-tree structure called a 'repository'. Each commit includes a <u>diff</u> of the files modified by the commit, and, similar to a blockchain, stores a hash of all prior commits via a <u>Merkle-tree</u>. When a Member is satisfied with their changes to their local version of the source code, they can 'push' their changes to a shared source-code repository, which is administered by the rules determined by the project contributors and opted-into by its participants.

Another example of agent-centric design is modern peer-to-peer file-sharing systems such as BitTorrent and the Interplanetary Filesystem (IPFS). Created by <u>Juan Benet</u>, IPFS enables Members to run a local IPFS DAEMON on their machine and store files on behalf of other Members within the network, creating a distributed network of file backups, instead of relying on a central cloud-storage service like Dropbox or Google Drive. Both IPFS and torrenting applications are made possible through the use of a <u>Distributed Hash Table</u> (DHT) and a <u>gossip protocol</u>. Unlike Git which relies on a local hash chain to record modified commits, IPFS and BitTorrent utilize a shared DHT to store and share the locale of files that are stored on various nodes. These systems generally utilize a hash of the file binary as a global index and a DHT search traversal algorithm to find the 'nearest neighbor' storing a particular file. This illuminates one of

the downsides of DHTs, however, in that they require a large degree of coverage to create fast and fluid Member experiences. With IPFS, it still takes considerable time for updates to 'propagate' across the network, which limits the potential for applications that require low latency.

Another example of an agent-centric system is <u>Scuttlebutt</u>, founded by <u>Dominic Tarr</u>, which is creating a decentralized social network and gossip protocol. Scuttlebutt is not a blockchain in the traditional sense and uses a local hash chain private key signature for social networking entries. When Members connect, Scuttlebutt automatically syncs chains with those that you trust, enabling them to share information without even being connected to the internet.

Finally, <u>Holochain</u>, introduced in 2017, is designed as an agent-centric direct alternative to blockchain-based architectures, based on the patterns used in Git and IPFS. Members run a local Holochain node that stores a personal chain of their transactions known as a 'source chain', then have the option of interacting with others through various Holochain applications (hApps) according to validation logic specific to each application – logic that serves as the basis for both how Members interact within the hApp as well as how and whether data is propagated over an application-specific DHT.

B.12 Immersive Reality/

AI and blockchain-based technologies are likely to engender meaningful progress in the growing field of Immersive Reality (IR). Artificial intelligence has always had a natural place in gaming and is likely to be one of the primary areas where major advances in the field occur. Since the dawn of pong, the dream of every hardcore gamer was to one day experience full photorealistic experiences. Recent advances in ray tracing, a technique that utilizes machine learning and hyper-fast GPUs to more accurately portrait how objects refract virtual light, show us that photorealism is no longer a distant fantasy. AI is already being used to generate vast open and procedurally generated virtual worlds and is able to more closely simulate the nuances of reality, such as atmospheric fog, complex weather patterns, and crowds. Augmented and virtual reality are a natural force multiplier for photorealism, not only offering visuals that more closely match reality, but offering full-range sensory experiences.

As games become a closer simulation of reality, in-game economies are likely to become increasingly real as well, and blockchain is very likely to become a core aspect of managing these <u>virtual economies</u>. Not only can blockchain technology enable greater in-game liquidity and underpin robust virtual economies, it also offers the possibility of exchanging tokens across games, as well as inside and outside of virtual space.

People in the US spent over \$43 Billion on video games in 2018, and the video game industry's forecasted revenues are predicted to reach \$230 Billion by 2022. Not only is gaming growing in the ways it historically has – like sales of games and virtual goods – but gaming itself is a fast-growing profession (known as eSports) with champions now earning more than top some traditional athletes. Twitch, a site dedicated to the live-streaming of video gameplay, was purchased by Amazon in 2017 for \$1.1 billion.

Similar to the shift we saw in Web 2.0 from upfront software licensing fees to SaaS and 'free' ad-supported services, games have been changing their business models to become accessible to more people, as exemplified by Fortnite and Star Citizen. Created by Epic Games, Fortnight has become a major phenomenon among young players with a free-to-play game that generates hundreds of millions of dollars a year through in-game transactions. Star Citizen, created by video game pioneer Chris Roberts, has been in development since 2011 and is scheduled for release in the summer of 2021. Over 2 million people have helped crowd-fund Star Citizen by pre-purchasing game gear, to the tune of a combined \$288 million as of early 2020. Both Fortnite and Star Citizen provide valuable insight into the growing capacity for online communities to crowd-source fundraising and activate digital marketplaces.

Another burgeoning frontier within the gaming space is <u>augmented reality</u> (AR). <u>Pokèmon Go</u> launched in 2016 and by 2019 grew to \$894 million in sales. With over a billion downloads worldwide and 5 million daily players who collectively have walked 8.7 billion kilometers, Pokèmon Go illuminates the influence AR is capable of.

Immersive-reality technologies are rapidly accelerating to provide cheaper, faster, and higher-quality experiences, including virtual-reality devices such as Facebook's <u>Oculus</u> and the <u>HTC Vive</u> as well as AR devices such as <u>Microsoft's Hololens</u>, <u>Magic Leap</u> and modern smartphones. Apple, Google, and Snapchat have released AR platforms and created an array of opportunities for gaming developers to produce apps and games. As AI, blockchain, augmented reality, virtual reality, and other emerging technologies unify, the dimensions of our relationships within the virtual world will expand. As innovations such as <u>epidermal VR</u> and <u>haptics</u> technology come into focus, the resemblance between the virtual world and the physical world will become even more indistinguishable.

APPENDIX C: THE ZERO CONSTITUTION

Zero's governing body is made up of three independent governing DAOs, collectively referred to as *Trinity*. This three-part structure distributes power in order to support the governance of Zero in operating within the domain of Effective Choice. Whereas two represents duality, three represents balance:

- The yin, yang, and that which unites the two.
- The irrational, the rational, and the irrational rational.
- The zero, one, and infinity.

For the purpose of Zero we express this pattern using the Body, Mind, and Spirit:

- The Zero Community (the Body): Comprising every Member across every Network.
- The Zero Guild (the Mind): Guild members who contribute work to the Zero Stack and Zero Community.
- The Zero Council (the Spirit): A council of wisdom keepers to ensure the principles of The Constitution are upheld.

Each DAO in Trinity is responsible for making its own choices, including its own Fundamental Choices, which are choices prescribed by the Constitution as needing to be made.

Three different 'levels' of Fundamental Choice require three different degrees of alignment among Trinity's DAOs:

- 1. **Local:** Choices made within an individual DAO, not requiring the authority of any other DAOs.
- 2. Global: Choices that require a 2/3 majority among Trinity's DAOs.
- 3. Universal: Choices that require a 3/3 majority among Trinity's DAOs.

LEVEL 1 (LOCAL) CHOICES:

- 1. Establishing and amending DAO's own Mission, Principles, and Values.
- 2. Establishing and amending the DAO's own parameters for collective choice-making within the DAO.
- 3. Establishing and amending the parameters of its own Reputation System.
- 4. Adding and removing Members from the DAO, as well as establishing and amending the process by which members are added and removed from the DAO.
- 5. Choosing whether to create a Network in Zero, and, if created, managing the Network's Control Panel within Zero.
- 6. Issuing new DAO Token and changing its token protocol.
- 7. Allocating its own budget

LEVEL 2 (GLOBAL) CHOICES:

- 1. Printing new Infinity tokens
- 2. Extending the Infinity token cap
- 3. Issuing new Infinity tokens
- 4. Materially amending the Zero front-end website
- 5. Approving and removing new Networks in Zero
- 6. Making material changes to the DAO Protocol

LEVEL 3 (UNIVERSAL) CHOICES:

- 1. Amending The Zero Constitution
- 2. Amending The InnerSource License by which Zero IP is bound.

In addition to its authority to make the Fundamental Choices described above, **The Guild** has the following additional characteristics:

- The Guild is responsible for the design, development, branding, community management, architecture and maintenance of The Zero Stack.
- Anyone has the right to apply to become part of The Guild. New members are accepted by the existing Guild Members based on skill level, work ethic, and character.
- The Guild's Reputation system is based on actual peer-reviewed work created for the Zero System based on its values. These values are consistent with the values expressed in the Zero Constitution, although focus on specificity for the task at hand -- building a world-class and ethical technology stack.
- Members of the Guild must contribute in a way that is both concrete, practical and meaningful. They must produce tangible work, whether it be actual code (developers), designs (designers), architecture diagrams (system architects), or written and recorded recommendations (strategists).
- They must choose to dedicate at least twenty (20) hours of work per week directly to the Zero Stack, or one or more Networks in Zero.

In addition to its authority to make the Fundamental Choices described above, **The Council** has the following additional characteristics:

- The Zero Council is the primary steward and governing authority for Zero
- Its Members are initially selected by The One Foundation
- The goal is to create diverse, international, capable and wise group of members

APPENDIX D: INNERSOURCE

- 1. Introduction
- 2. License Options
- 3. INNERSOURCE A
- 4. INNERSOURCE B

D.1 Introduction/

Zero and The One Foundation present the INNERSOURCE LICENSE (the "License"), a modified license for the benefit and sustainability of open-source projects. The purpose of this license is to simultaneously ensure that (a) Zero software remains open and free and that (b) there is a sufficient economic model to reward the ongoing time and effort of project contributors.

While the majority of software utilized online today is open-source (Linux, Apache, TCP/IP, JavaScript, etc.), all too often companies simply take software from the open-source community, add a paywall or ad-model, and keep the majority of the profits generated. InnerSource is an attempt to solve this by transcending the 'open/closed' debate and providing a legal framework that enables software to remain both free and open, as well as ensure that project contributors have a mechanism to be rewarded for their hard earned efforts. Our hope is that by providing a framework for open and distributed projects, we will see the migration from full-time employment at centralized institutions like Google, Facebook and Bank of America, to individuals who gain their own creative and economic sovereignty by contributing to open-source projects. Anyone is free to utilize this License for the benefit of their projects, irrespective of whether or not they are connected to Zero.

The following includes some legalese that we need to present to You as part of this License. The Definitions section below outlines key terms for what is used herein.

D.2 License Options/

INNERSOURCE provides two options for Open-Source Software ("OSS"); INNERSOURCE A and B.

INNERSOURCE A enables the free and unencumbered use of software with limited restrictions.

INNERSOURCE B encompasses additional protections specifically related to the monetization and reward of well-defined pieces of source-code through the use of cryptographic tokens.

D.3 License A/

OSS which is made available to You under our INNERSOURCE A License, similar to the MIT License, with the primary modification that You are required to contribute Software back to the Zero Stack.

The INNERSOURCE A License states: Copyright <YEAR> <COPYRIGHT HOLDER>

You are hereby granted permission to obtain a copy of this software and associated documentation files (the "Software") for free and without restrictions (except those that are explicitly stated here), including, but not limited to the right to: use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

When using the Software, any modifications You make must be committed back to the relevant Zero Stack, providing they are accepted by the repository administrators, and this modified Software will also be governed under the same respective INNERSOURCE A License.

In plain English, if you use any of the Software to build anything, you have to contribute what you built back to the Zero Stack.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

D.4 License B/

Software which is made available to You under INNERSOURCE B is licensed under the INNERSOURCE B License which is provided in its entirety in this section 4:

4.1 Grant of Copyright License.

Subject to the terms and conditions of this License, each Contributor to the Zero community hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

In plain English, this means that You have the right to use, reproduce, make modifications and changes, and create other works from the things that You access and use under this License which have been contributed by others.

4.2 Grant of Patent License.

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

In plain English, this means that You have the right to freely use, sell, import, or transfer, works that have been contributed by others under this License, even if those contributed works are covered by a patent.

4.3. Contributor Monetization.

The grant of the INNERSOURCE B License to You is conditioned on and subject to your agreement to use the Infinity Tokens as stated here: When using OSS under the INNERSOURCE B License, you are required to utilize the Infinity Token on the websites and platforms you develop in accordance with the design specifications established by The One Foundation. By way of an example only, in the Zero UI, a button to purchase Infinity Tokens is required to be labeled either as "buy" or "purchase" and marked with the corresponding Infinity logo, Infinity name, and a ticker (INI), which must be shown at all times in the core UX. These standards are subject to change at any time at the sole discretion of The One Foundation, as determined by the collective governance processes outlined in The Zero Constitution.

4.4. Modification.

- 1. You may modify Your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of this License, provided that You also meet all of these conditions:
 - 1. The modified work must itself be a software library.
 - 2. You must cause the files modified to carry prominent notices stating that You changed the files and the date of any change.
 - 3. You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

- 4. If You make any modifications to the licensed OSS, You must contribute Your changes back to the main OSS repository.
- 2. These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when You distribute them as separate works. But when You distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.
- 3. Thus, it is not the intent of this section to claim rights or contest Your rights to work written entirely by You; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.
- 4. In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

6. Redistribution.

- 1. The copyright notice "Copyright < YEAR > < COPYRIGHT HOLDER > " in this Section 8.1 and the permission notice consisting of the entire Section 8.2 below shall be included in all copies or substantial portions of the Work.
- 2. Permission is hereby granted, free of charge, to any person obtaining a copy of this Work and Derivative Works, to deal in the Work and Derivative Works without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Work and Derivative Works, and to permit persons to whom the Work and Derivative Works are furnished to do so, subject to the following conditions:
- 3. You may copy and distribute the Work and Derivative Works in Object Code or executable form under the terms of this License provided that You also do one of the following:
 - 1. Accompany it with the complete corresponding machine-readable Source Code, which must be distributed under the terms of this License on a medium customarily used for software interchange; or,
 - 2. Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than Your cost of physically performing source distribution, a complete machine-readable copy of the corresponding Source Code, to be distributed under the terms of this License on a medium customarily used for software interchange; or,
 - 3. Accompany it with the information You received as to the offer to distribute corresponding Source Code. (This alternative is allowed only for noncommercial distribution and only if You received the program in Object Code or executable form with such an offer, in accord with Section 8.3.2 above.)
- 4. If distribution of executable or Object Code is made by offering access to copy from a designated place, then offering equivalent access to copy the Source Code from the same place counts as

distribution of the Source Code, even though third parties are not compelled to copy the source along with the Object Code.

- 5. When the Work is Distributed as Source Code:
 - 1. it must be made available under this License, and
 - 2. a copy of this License must be included with each copy of the Work.
- 6. Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ('notices') contained within the Work from any copy of the Work which they distribute, provided that Contributors may add their own appropriate notices.

4.6. Submission of Contributions.

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement You may have executed with Licensor regarding such Contributions.

4.7. Trademarks.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

4.8. Disclaimer of Warranty.

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

4.9. Limitation of Liability.

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer

failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

4.10. **Termination**.

- 1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.
- 2. In the event of termination under Sections 6 (institution of patent litigation) or 14.1 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

4.11. General.

- 1. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.
- 2. All Your rights under this License shall terminate if You fail to comply with any of the material terms or conditions of this License and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Your rights under this License terminate, You agree to cease use and distribution of the Work as soon as reasonably practicable. However, Your obligations under this License and any licenses granted by You relating to the Work shall continue and survive.
- 3. Except as expressly stated in this License, You receive no rights or licenses to the intellectual property of any Contributor under this License, whether expressly, by implication, estoppel or otherwise. All rights in the Work not expressly granted under this License are reserved. Nothing in this License is intended to be enforceable by any entity that is not You or a Contributor. No third-party beneficiary rights are created under this License.

5. Zero Terms and Conditions

Your use of any Software or Works under either the INNERSOURCE A or B Licenses are subject to your compliance with Zero's Terms and Conditions.

6. **Definitions**.

1. "Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic

- mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "**Not a Contribution**."
- 2. "Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.
- 3. "Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.
- 4. "**Infinity Tokens**" shall mean those crypto currency tokens developed by the One Foundation.
- 5. "Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.
- 6. "Library" shall mean a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables. "Library" refers to any such software library or work, which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)
- 7. "**License**" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 11 of this document.
- 8. "Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.
- 9. "**Object**" or "**Object Code**" shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.
- 10. "Source" or "Source Code" shall mean the form of a program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.
- 11. "Work" shall mean the work of authorship, whether in Source or Object form, made available under the License.
- 12. "You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License
- 13. "**Zero Stack**" shall mean the Source Code repository provided and managed by Zero and the One Foundation.

The InnerSource License can be found at <u>zer0.io</u>.