**The Fluence ABCs: A Fluence Explainer Article**

**Introduction**

Hey there.

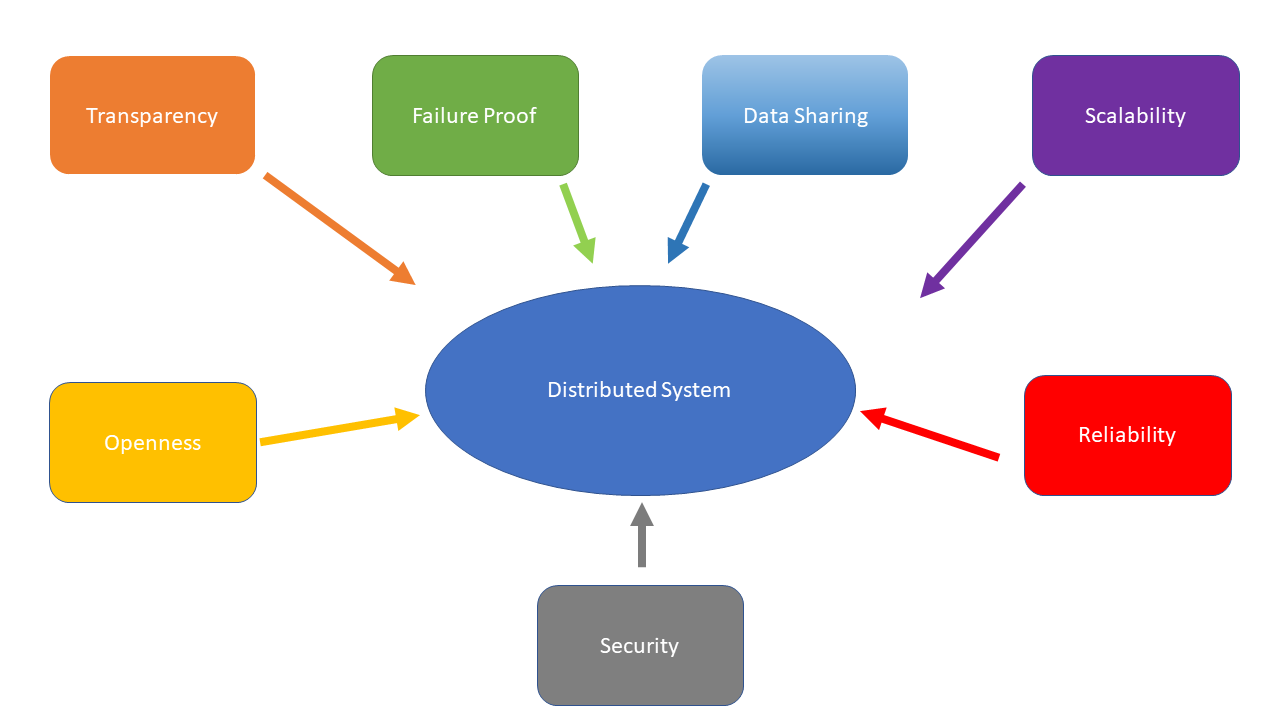
I bet you’re pretty new to the concepts that make up Web3, or you’ve stumbled on something called off-chain, the waves it’s making, and to top it off, Fluence is swimming around in the mix somewhere and you’re not sure how it all fits.

I’m here to help.Let's start with the Web3 ecosystem and how [Fluence](http://fluence.network) is a part of it.

**WEB3**

A group of forward thinkers thought how unfair it is that all the information we have and all of the data we post, and use, are centralized. That is, controlled by a specific group of companies, like Google and Facebook (Meta), and thought it best to develop a digital environment that makes people feel a lot safer and more in control of their personal information.

You can see Web 3 as an evolution from the centralized web where you can only access and post information, but not own it, which is, the basic principle of the internet today, also known as Web 2, into the Decentralized Web. In the sea of Web3, you can now have rights to your data giving you the ability to have seamless interactions between websites, decentralized platforms and users.



The evolution kickstarted in the financial industry with the emergence of cryptocurrency and later on, non fungible tokens (NFTs) which are use cases of the blockchain technology, thereby creating decentralized finance (Defi).

Web3 relies heavily on cloud computing methods to function. Cloud computing gives you immediate access to resources used on the internet, ranging from development tools, to networking capacities, data storage, applications and more.

Considering the amount of transactions that occur on the blockchain (on-chain), there are a few issues that have come up.

Some of those issues are lags, and security breaches, and here’s why.

The blockchain works by scrambling your data into something unreadable (encryption) and making sure that there's a secure system to confirm the validity of data (verification). There are several points, systems or nodes in the blockchain network that data input has to go through before it gets the stamp of approval, and if it skips any point, there’s an error and the data doesn’t go through or is unverified. The more the transactions that need to be verified on-chain are, the more time it takes, and the easier it is for people who want to steal your data to get to it.

It’s also leading to some companies compromising on decentralization. That is, they're taking out the option of running it on an open ledger so that not everyone has access to it and it's like another cloud that belongs to someone, or a server that is controlled by someone. They do this in order to keep the transactions and on-chain computing working well.

I can tell it's starting to feel like the blockchain is creating the same problems it's trying to fix as we go into Web 3. It's still in its early stages and developers and other personnel are making use of these new tools, working round the clock to come up with ways to make this work better.

A solution that has come up is off-chain computing. Off-chain computing can deliver transactions and data very quickly and it does so outside of the blockchain, abiding by the rules of the smart contract, accessing higher scalability, cost-efficiency, and privacy, as opposed to on-chain computing which delays transactions, has less computing speed and can be performance-limiting.

Now the thing is, you need something that gives you tools, just like the cloud system in Web 2, to navigate Web 3.

And that’s where Fluence comes in.



**Fluence**

[Fluence](http://fluence.network) is a development stack, an array of digital tools, that bypasses the need for a centralized system in building the applications and protocols that work in Web3's decentralized ecosystem.

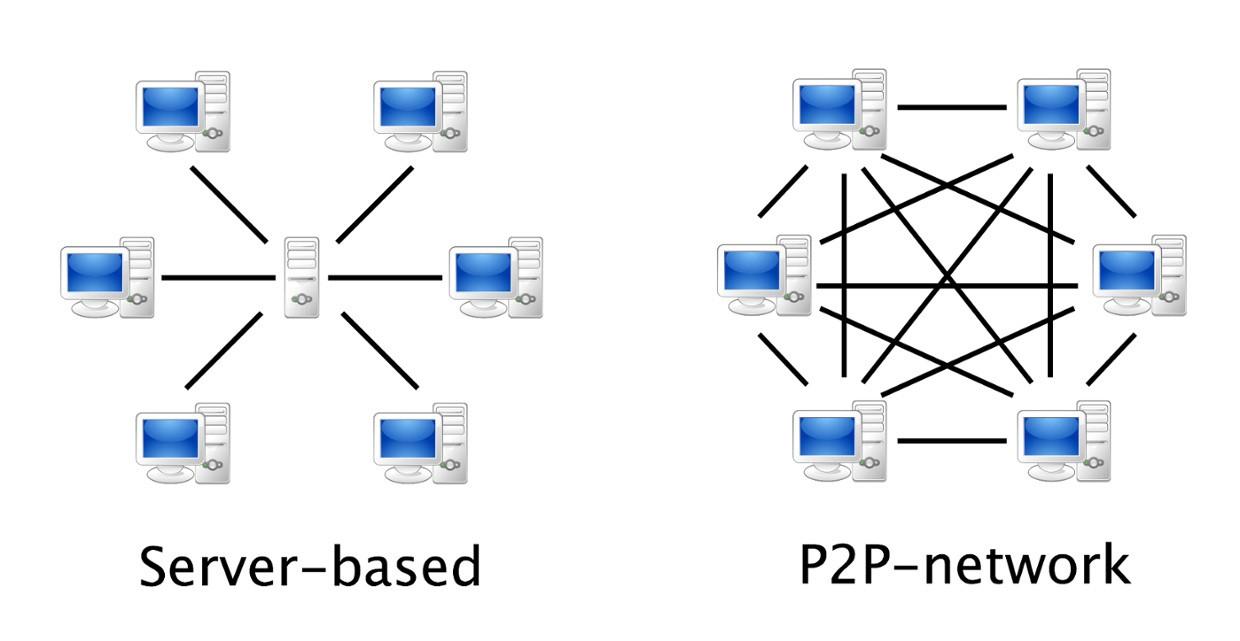
Think of Fluence as the company that made a motorable airway floating above the congested Los Angeles traffic.

They achieve this by implementing one of their core principles, and that's making it a strictly peer-to-peer controlled system. So if you don't run the Los Angeles traffic to anywhere, then you have no business being there. And this road makes sure that those who shouldn't be there, arent.

[Fluence’s](https://fluence.network/) goal is to bring all of what you need to build this alternative route from congested traffic, internally and externally, then lets you tailor it, so the information can be shared, the way it's meant to be, to who it's meant for… but among users; peer-to-peer networking.

To accomplish this incredible feat, Aquamarine was created.

Aqua, a programming language that was designed exclusively for peer-to-peer networks to develop processes and scenarios that are computable on the system while, Marine is the WebAssembly runtime environment that Aqua is hosted on.



It's like a lock and key. The lock being Marine, and Aqua being the key to be inserted in Marine to work.

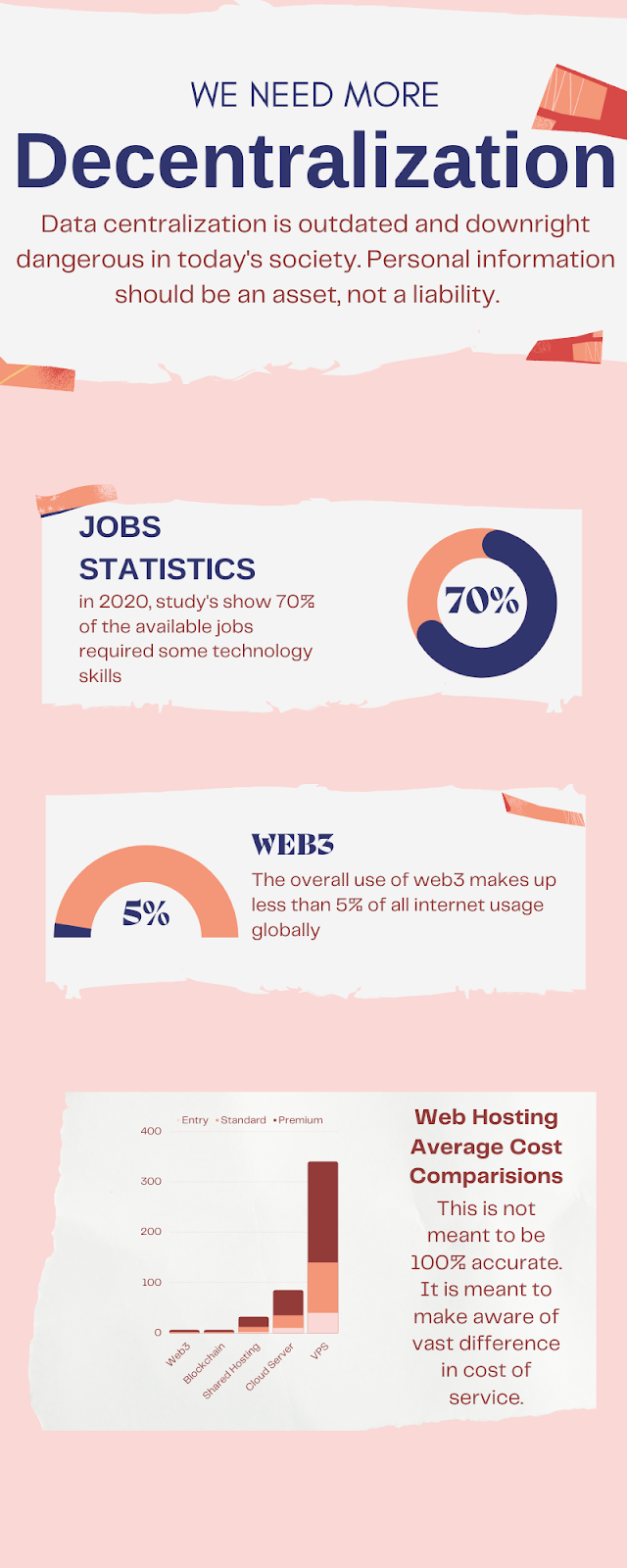
Marine also allows other hosted WASM services to work on this system while making use of Fluence's nodes. In turn, rather than subscribing to a whole server, fluence allows you to pay with crypto for each node, or peer, your code accesses.

Still not making sense?

Think of it this way also; In a game, you need to work on the settings and your characters before the actual game starts, to make sure that your character and game play performs optimally to your specifications. But instead of interacting with an unseen developer to get the right things you need, you’re interacting with fellow gamers on the network (peer-to-peer), and this allows you the power to share specific information with specific people, and specific need cases. So fluence is the game application, it provides a network that allows for other game players or peers to interact with you and your code I order for each person tobaccos what they need as they build the off-chain protocols, apps, etc.

This makes your process faster, more efficient, and your strategies and information more secure.

So when you show up on the game play, no one really knows what’s gone on behind the scenes, all they see is the finished product and how you're using it.



**Why Fluence?**

So why [Fluence](https://twitter.com/fluence_project)? Fluence has accumulated tools that would enable you bypass this node-to-node protocol that the blockchain runs on. It allows you to mutate, compute, and specify, what exactly you need from other peers on the network. So when it runs on the network, it executes the tasks you need, reaches out to other systems under the rules that you’ve set, and then gets sent back to you, as the originator of this data. When that’s done, it reflects the changes you’ve made in the code on the blockchain, maintaining its integrity, but runs on a faster and more secure system.

That’s essentially what off-chain computing is about. Getting the load off the blockchain you need to work on, and then using the final result on-chain after its off-chain development.

Fluence has numerous applications/use cases across diverse fields, some of which include:

* **Peer-to-peer applications**: Because it is censor-proof and decentralized, Fluence provides the perfect framework for the development of streaming services, social network apps, decentralized messenger platforms, and other communications channels that cannot be censored in the event of a political crisis.
* **Cloud-native computing**: Applications that are based on microservices and built to work in cloud-native applications and in cloud environments can easily use Fluence to manage their microservice server-side without having to rely on a centralized server. Also, seamless backward compatibility makes sure that application upgrades can easily be implemented without the app overloading and crashing- an experience that users are used to whenever an app rolls out an update.
* **Multi-party computations**: Fluence could serve as a cryptographic tool that allows numerous servers to make calculations independently but using their combined data, while still maintaining the individual input.
* **NFT platforms**: Fluence issues decentralized authentic certificates for Non-Fungible Tokens as well as captures mints and transfers data for NFTs or ERC20 tokens while making use of Aqua.
* Building Blockchain infrastructure like Blockchain oracle, wallet apps amongst others.
* Building Decentralized protocols.
* Building Dapps.
* Building Community run platforms.
* DeFi Governance.

Other highly relevant characteristics of Fluence includes:

* **High Scalability**

In the global fluence network, applications will be able to create as many sub-networks as they need, which will make the global capacity in billions of peers. Peer-to-peer transactions have higher connectivity between each other with low priority for outside connections.

* **Pluggable Data**

Any external data source can be plugged into applications on Fluence.

* **Permissionless Protocol**

Fluence doesn't hold anyone back from connecting to the network, running a node, deploying services and applications or paying for them. It does all this without any permission of a third party.

* **Computation Verification**

The security model of Fluence allows applications to execute only on nodes they know and trust.

**Fluence Vs Other Protocols**

The structure and organization of Fluence is really unique when compared to other protocols in the tech space. Compared to closest rivals like iExec, Elastic Cloud Compute(EC2), and Lambda, which offer decentralized cloud computing services, Fluence is unique in its offering of a decentralized all-in-one development stack; it is open source, and engages permissionless protocols.

For example, EC2, Elastic Cloud Compute (EC2) is a product of Amazon and an on-demand cloud computing service that lets users easily scale up or down their computing resources depending on their needs. As the name implies, it is very flexible in usage, but it comes at a fee! Users also have to worry about management and configuration of the cloud resources.

**iExec**: iExec is a blockchain-based decentralized cloud computing marketplace where users can connect with resource providers. Data Sets, computing power and applications are some of the resources available on the iExec marketplace.

**Lambda**: Lambda offers ‘serverless’ cloud computing by abstracting away the complexity associated with EC2 using code/scripts. Lambda is however not as flexible as EC2.

So if you're looking for something that gives you the perfect balance of all these networking cloud services to use in building your Web3 projects, I bet you're thinking of checking Fluence out.

**Conclusion**

You won’t find a Web3 Project offering more potential, security, room for expansion, and ease of use than the Fluence Network. Using Fluence’s peer to peer development stack, essentially, cuts out the middleman (or in other words, proprietary cloud providers, for example Amazon with AWS, Google, Microsoft with Azure, or centralized APIs). Developers can create and program peer-to-peer applications, workflows, and compose services with decentralized assurance in the privacy and security regarding a user’s personal data. The best part is Fluence is open source, and as a consequence, completely free for anybody to use.

When you come across a platform offering tons of features, security, and decentralization while at the same time offering ease of employment, cost effectiveness, and a higher range of compatibility amongst a variety of networks; wouldn't you make a move? Fluence is getting ahead of the curve, and you can too.

Interested in learning more, but don't know how to? Check out these extra resources -

* [Discord](https://discord.com/invite/5qSnPZKh7u)
* [Github](https://github.com/fluencelabs)
* [YouTube](https://www.youtube.com/channel/UC3b5eFyKRFlEMwSJ1BTjpbw)
* [Calendly](https://calendly.com/fluencehack)
* [Twitter](https://twitter.com/fluence_project)