

 **omigroup / omigroup** Public[Code](#) [Issues 38](#) [Pull requests](#) [Discussions](#) [Projects 2](#) [...](#)

Constellation (formally Stellar Mesh) #73

technobaboo started this conversation in **Ideas**



technobaboo on Jul 28, 2021 Collaborator

edited ▾

I propose that OMI are the ones to create the Stellar Mesh, in protocols and possibly a reference implementation.

First, the Stellar Mesh is a new architecture to encompass web and native apps, sites, etc. that reimagines how they work in relation to our devices, so we're not stuck on the old design paradigms. See <https://twitter.com/simulacracid/status/1299079649784287232>, Stella explains what it is. As well, see <https://github.com/TrevorFSmith/stellar-mesh/wiki>. I do think some modifications would help, including having a Stellar Mesh browser/manager to launch runnables from URLs. But overall, I think this is great.

So why not use the old method of a web browser/electron and native apps? For one, browser engines are way too bloated. It takes teams of hundreds of engineers years to make an engine, which explains why only 3 major ones exist. 1 engine is trying to do everything on the web, from HTML to CSS to JS to WebGL to WASM to WebXR to WebGPU to web payments to even wallets now. No small team could ever hope to maintain this, so the Stellar Mesh splits this up into engines that can handle the things a runnable needs to without all the extras. Another big issue is the web vs native vs bytecode wars. They're all good at different things, and no 1 is the best.

Another issue is that talking between web and native apps is a massive hassle by putting it through a browser and piping it out that way. It's a massive hassle with no 2-way communication, and any deeper communication requires altering web standards on a massive scale or compromising security. Since Stellar Mesh puts web and native and bytecode apps on the same level, they can talk to each other like siblings instead of being at different distances to each other.

If we want to bridge the Metaverse, having a bunch of apps with no clear communication channels between them is a massive pain that will only grow as everyone has a new protocol or engine they want to try. With Stellar Mesh, they can make an engine and a runnable, and it'll work amongst others. Many of OMI's avatar and item and portal proposals are a nightmare hopping between platforms while being unable to share any data app to app directly, but with Stellar Mesh you can just directly send that data over. Portaling between experiences is much easier when you can have a 2-way conversation as siblings to make a smooth handoff instead of the web/native divide we have now. Even Kim's IPSME idea is well executed in the Stellar Mesh.

Unlike many proposals on making OSes that only do web or integrated metaverse wannabes that have carefully curated content, Stellar Mesh does not attempt to replace the OS. You can always run programs outside of it if you want to, but they won't have first-class abilities to connect to other apps and there's no guarantee of sandboxing. But in Stellar Mesh, engines can be as wild as they want with new features and ideas and architectures, using native or bytecode or web or whatever while still being interoperable. Instead of having to install the dotnet runtime, mono runtime, JVM, web engine, WASM runtime, C++ libraries, etc. all to the system causing instability, engines group all the needed dependencies in a nice package without slowdown or inflexibility.

This will be a fair bit of work, but there is already precedent for this system. Flatpak on Linux already has a strikingly similar structure with its apps being Stellar Mesh's runnables, backends being engines, having containerization, and portals being capabilities. Even Stellar Mesh's advertisements are perfectly summed up in Linux's D-Bus object-oriented cross-app messaging system (though in Stardust the UX of advertisements will be different). With Stellar Mesh's architecture and OMI protocols, I think this is the best way to organize the metaverse. Problem is, it's a lot of work and nobody's doing it.

Advertisements would be where OMI shines, from possible crypto wallet info being sent through there to federated identity to world URIs/URLs to assets. A user can always cut off an advertisement at any time with good UX, so nobody is forced to use crypto or have their identity sent across apps or such.

From a user perspective, they'd be able to go on their universal search and search up a runnable to use. Then, if they like it they can "install" it which would move it from the cache to more permanent storage. They could always cut off internet access to it so it's offline and private, control what data goes into it, etc. This means they can try it out temporarily and then decide they like it. Paid apps can be done through internet verification or a cryptographic way of unlocking the engine when a certain code is downloaded for that exact copy or such, maybe even an NFT. Updates for runnables/engines would go through feeds similar to PWAs/package repositories or directly, but the manager would ensure nobody is forced to update if they don't want to. Stellar Mesh works with the package manager/app store model, PWA model, instant app model, and even webpage model. But it would never be locked into any one of those.

Stellar Mesh deserves its own organization partnering with OMI, but since it isn't even a prototype yet I think OMI should be the ones to kickstart it. I don't have time to make it myself, Stella and Trevor don't, etc. But I think together everyone at OMI does, and it'd be worth it. It combines the best ideas of existing and new technology to make a system that can't be locked down easily yet is easier to manage and understand.

↑ 2

2 comments · 1 reply

Oldest

Newest

Top



mrmetaverse on Jul 28, 2021

Maintainer

Stellar mesh seems like a wonderful pursuit, and potentially powerful foundation for an open metaverse. Thank you for using the format of discussions for this. This will require a much larger conversation. I'd be curious to see an implementation.

Keep in mind "not having time for a this" is a challenge many members will be facing here. What does baseline look like, and why should we focus on that baseline?

Another thing that comes to mind, OMI is meant to form protocols and isn't in the business of creating businesses. An effort of this magnitude would depend on a product partner, especially given limited bandwidth. I think the best chance to see this move forward, is to break down some of the work and consider where to begin.

If we can put our finger on a starting point as a group, there may be more willingness to get it started.

↑ 1

1 reply



technobaboo on Jul 28, 2021

Collaborator

Author

Baseline would probably be 2 engines and runnables talking between each other using a common protocol and a few capabilities using a Flatpak fork and D-Bus on a Linux system. It's the bare minimum in functionality and since there's already software to modify it's achievable through a few work sessions (I think).

Why would this need to be a business?



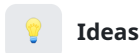
Toasterson on Dec 20, 2021

I had the same Idea, seeing what we need to run our apps and what current browsers are it starts to make sense. I would like to propose this with something like wasmer and webgpu for a WASM demo. Or are works already progressing towards a simple demo?

↑ 1

0 replies

Category



Ideas

Labels

None yet

3 participants

