

Demystifying the road to Ethereum 2.0 ;)

Serenity Q&A session at ETHMagicians

Participants: Danny Ryan (EF), Vitalik Buterin (EF), Afri Schoedon (Parity), Paul Hauner (Sigma Prime - Lighthouse [TODO LINK](#)), Greg Colvin (ETHMagicians), Jacek Sieka (Status - Nimbus <https://our.status.im/nimbus-receives-ethereum-foundation-grant/> & <https://github.com/status-im/nimbus>), Bryant Eisenbach (SecurETH), Casey Detrio (EF/Ewasm), Lane Rettig (EF/Ewasm).

Facilitated by Lane Rettig, Boris Mann, Jamie Pitts, María Paula Fernández, Chelsea Palmer.

Note: the dialogue has been modified and condensed for easier readability - this is meant as a high-level guide, for now, but we're open to your ideas.

READ BELOW FOR CONTRIBUTIONS

During Day 1 of the Prague Council of the Fellowship of Ethereum Magicians, we had the chance to have a Q&A session in a fishbowl format with client implementers, the Ethereum Foundation Ethereum 2.0 team, and the audience. These transcripts below aim to inform the community about the future of Ethereum, most specifically on sharding, proof of stake, eWasm and more components of Ethereum 2.0 (now called [Serenity](#)).



1. About the new specs and the Beacon Chain (TODO LINK):

Danny - The basis of the spec for 2.0 is there, however, there are still a lot of components down the line. The 2.0 roadmap is based in an effort to create this new blockchain in parallel to the Proof of Work existing chain. We are creating a side universe, that allows us to rapidly develop and iterate.

There are a number of teams working on the beacon chain, a parallel ecosystem, where the validators “live”, where the shards are secured and coordinated. It is the core PoS chain and when the shard chains exist, they can be added there. The next layer is to add execution and cross-shard communication, and later on, snarks.

[More on the Beacon chain in the Two-Point-OH Nimbus blogpost series.](#)

2. What do we do with the existing POW workchain and how do we begin to migrate the community?

Paul - It's very difficult to plan but we are hoping that March/April next year we are gonna be able to have clients and some interoperability. Depends on the coordination of the clients.

Danny - 1st half of next year for the clients to start is gonna give us enough pressure to work effectively.

3. Challenges & Clients' perspective

Paul - There are not many challenges known. Any unknown challenges that might bring significant problems to the spec, seem unlikely at this stage.

Afri - Parity is excited. We have PoC implementations, upgraded on top of Parity Substrate. We joined later, but we have Shasper implemented now. We are looking forward to a new testnet, cross-client, we need a new non PoW testnet to advance.

Jacek- From Status' (Nimbus) perspective, ETH2.0 is not just the beacon chain. we started out as a sharding client, we are interested in the potential sharding has to offer and the ways PoS gives access to a lot of devices. In regards to the beacon chain, we are sticking to the March timeline. On top of that, there is a lot of execution work on sharding, which will take longer. I see March as a good indicator to put something out there, a target to work against, and it gives the community something to play around with and be able to validate.

3. Where are the EIPs or other specs? How does the ETH2 group decide on roadmap/features/coordination? What is the “Ethereum roadmap”, exactly?

Vitalik - The 2.0 roadmap has existed in many places over time. At the highest level, there has always been this impression that we're supposed to first of all, just keep on upgrading and improving the protocol, getting the proof of stake switch finished, achieving much higher scalability through sharding and from there just keep on adding technical improvements.

Maybe before 2017, this was much less clear, as there was no clear idea on how proof of stake would look like and what sharding would look like. Then we came up with [Casper FFG](#). Time passed, and we realized that it would make more sense to do the Proof of Stake and the sharding together in this beacon chain architecture.

4. Where can we find more information?

Vitalik - These ideas are scattered around some sites:

[ETHresear.ch](#): protocol discussions have been happening there.

Usually, the abstract ideas live on Ethresearch.

In terms of very fundamental features of the protocol, Ethresearch has become the defacto place for example for: Casper FFG, proof-of-custody, cross-shard transactions. If an idea is more complex then it can be turned into a white paper.

Github

The initial specs on [HackMD](#) moved to [Github](#): the specs have just been written, and it's being edited and becoming more incremental.

Proposed changes now live in the issues and pull requests. As well as more fine-grained discussions.

<https://github.com/ethereum/eth2.0-pm> – like AllCoreDevs, but for Serenity

5. [Ewasm](#) and how it fits into the timeline:

Vitalik - there is rough consensus that Ewasm will be the execution engine. I favor the point we should retire the EVM soon. We should have a contract on Ewasm with an EVM interpreter.

Bryant - Ewasm is a whole new development and tooling that needs to be built. We have solid tools for EVM, but it's not quite clear to me how we can get ready to go with Ewasm tooling.

Greg - somehow Ewasm and EVM have been in an argument. We have a toolset growing on EVM, and the longer it takes to get EVM2 out, the longer it will take to get tooling out. We don't know how long it's gonna take for Ewasm (to be ready) if we try to deprecate the EVM1 we are going to create a huge amount of complexity. The contracts now work on Solidity and there are lots out there, and now we are putting companies in a position where they can lose the Solidity code. That doesn't mean Ewasm shouldn't be the fundamental layer but we can't just abandon what we are without creating much more complexity. We have to keep evolving what we have and talk to other development teams to see what problems they find on EVM1. Who's gonna write the contracts? they are not DOS hardened.

Vitalik- Ewasm in general, is meant to run untrusted code.

Casey - one of the goals of proposing new compiles, is that the clients have to run a Wasm engine to execute them. So it's not forcing clients to run a Wasm engine, but it gives them the option to use a Wasm engine in the future as a fallback. This is just one proposal to improve the EVM. The question is how do we add the precompile without adding complexity.

Vitalik - the Solidity compilers have been adding Ewasm. We have been removing overcomplicated things. (On account abstraction) How much we can abstract? I feel we are on par with the simple versions of abstraction but the more complex ones have many tradeoffs we need to look after.

Danny- in the roadmap as i see it the EVM exists till we fork it into a contract and a shard. Depending on the company, they won't have to migrate to wasm. As long as people put some work on the EVM, given that we have this parallel scaling roadmap and the EVM remains untouched, we do have time to coordinate on these forks in the next 2 years. I think that it's probably a good idea to get the EVM to a good place. The general consensus is that we should have regular forks. Right now we should be considering what is going into the next fork.

Casey - I think there is a parallel in the EVM vs Ewasm, and making the switch. One reason to replace (the EVM) is because it's easier to switch to Ewasm because there is tooling already available. The question (here) is about 2.0 launching in parallel and leaving the EVM alone. This adds more complexity to the community, ideally, this should be seamless, the mainchain should upgrade.

Greg - I can see Ewasm can be used to implement a fair amount of EVM 2.0, you can put a canonical EVM1 to Ewasm compiler in the blockchain. You don't have to port your EVM code, just have to use a compiler for the new client for speed. The arguments over performance can be solved thanks to modern compilers. Once you get something solid and a lot of people are using it the tendency is to keep it.

6. On EIP processes for Ethereum 2.0

Danny - In my opinion, we are a little bit early to go down the EIP process and formalizing it. The community working on the specifications is ready to go into the real launch and then write an ETH2 genesis EIP. Later on, formalize how to upgrade and build standards. The implementers nowadays are more focused on rapid communication and building, working towards client testnets, rather than the formalization of the next EIPs.

7. What kind of tangible obstacles could delay PoS? eg. the source of randomness for the beacon chain, has it been solved?

Vitalik- The fundamental problems are figured out- I'm more worried about small things. Yes there is a massive lake (*in regards of the complete migration to 2.0*) and yes the lake is miles away, but the way the protocol has been designed it does not depend on randomness to achieve security. The attacker will not be able to take on a committee because of the size, and not able to revert a single block. There are defensive components and randomness is not a prerequisite for anything.

Danny - [RanDAO](#) is the base of what we are going to use but won't degrade the base layers. With PoS we don't have a source of entropy, we need a source of entropy to allow us to shuffle validators, etc.

[RanDAO](#) is a mechanism where all the validators participate and when there is a new block on the beacon, RanDAO provides the new source of entropy. It is a distributed way to come up with random numbers, and the DAO is formed by the validators, and a solid source of entropy. Justin Drake has been working on improving this source of randomness.

8. Would it not be good to have a better evolutionary approach and more hard forks?

Danny - We would like to have more frequent releases and not roll so many things at once. You'll see that this is demonstrated soon. how that process goes will inform how we go in the future. One of the design goals for Ethereum 2.0 is to be ready for World War 3.

9. Should be governance a part of an Ethereum standard (governance formalized as a part of Ethereum standards)? How do you coordinate consensus among clients to determine which forks?

Danny - being PoS and scalable is the goal. the coordination is now in open dialogue, and from there we can formalize but it's too early still. In terms of research, we have seen more teams involved on research so what clients are hopefully doing is understanding the research, and moving from there. This is a coordination problem, not a governance one. Right now it's still pretty early. Research is still primarily EF, but we're seeing more from Parity, Status, Pegasys, etc.

10. How do current ERC20 contracts fit and migrate to 2.0?

Vitalik- If you have a contract for simple, short-lived interactions, just copy/paste onto Ewasm and phase one out. Could be via centralized airdrop or a script that burns coins on one side and mints them on another side

Non-financial dApps can just move really quickly. Solidity in many cases will compile to Ewasm. People could stay on 1.0 chain until it eventually folds into 2.0 chain, in which case it will require some client-side work. We could also make a light client for the 2.0 chain that fit into the 1.0 chain. Ultimately, this research is something we'd be interested in giving research grants for.

11. Is the cryptocurrency of Ethereum 2.0 ether?

Vitalik - There is the intention to make deposits on the beacon chain. There are intentions to make ether in the PoW chain transfer without becoming validators. There is the idea to have convertibility. I don't think that we are fully decided at this point.

Paul - ethereum 2.0 eth has to be the same eth, should be moved across. I agree on moving it from 2.0.

Jacek- it gives applications a lot of freedom about when they want to make the move. That's up to every dApp to decide. which is an interesting set up in general.

[For more info, check this great post by Bruno of the Nimbus team.](#)

12. Do you think that Ethereum is developing slowly?

Danny- Research is fuzzier and takes more time. I don't know if this could have happened any sooner, but now that we are in the implementation phase, I want it to happen as soon as possible. Real-world things are gonna hold us down, like testing. Its hard to have a standard on whether the research took too long. Maybe in a year, we can assess that. There are a lot of ideas to create and debate cross-shard communication.

Vitalik - on the research side for 2.0 there is a point to improve, a state execution layer.

13 . How do we do proper POS testnets, because there won't be any real value at stake?

Vitalik - we can launch the beacon chain relatively early, and just launch it with real deposits that will give us the technical testnet.

These are the questions and notes we gathered during the session. We would like to continue to improve this community resource, as it affects all of us. If you have notes, slides, videos, anything please drop them [here](#) or as a comment on this doc, and we will roll them into this blogpost!

We're trying to keep it high level, but we take ALL contributions.

As the Magicians is an organization that's aiming to take pragmatic actions to contribute to Ethereum's improvement, we have decided the Fellowship will take the following action items:

- Build a roadmap opened to community contributions as a parallel resource to EF's efforts.
- Follow up on all core devs calls and summarize within such roadmap for the community
- Continue to host these events and ask questions.

Please join us, we would love to hear from you and have you helped out. Onward!

Note: even though this is a 2.0 - related piece, I would like to link the following about Ethereum 1.x, a proposal that's being developed to improve the state of the current chain:

- [1.x meeting notes](#)
- [Casey de Trio's 1.x explainer post](#)
- [Ewasm proposal for 1.x](#)
- [Alexey Akhunov on Ethereum State Rent](#)

