

This concept is coming up from [@kfichter](#)'s famous "Why EVM Plasma..." article, and then he once tweeted the name "Plasma Generator". I really like this name. ("Plasma Chamber" is also my favorite though). And [@gakonst](#) also said he also once thought about this kind of idea. And [@syuhei176](#) from Cryptoeconomics Lab also interested in this idea and almost all of ideas are fostered by him. (Because this is almost same path with generalization of trustless Plasma construction and it was his focus). And thanks [@m0t0k1ch1](#) [@leohio](#) for daily discussion.

Plasma Cashflow and related discussion seems like enabled to add "swap" functionality to Plasma Cash. This achievement may open the door to the multisig implementation or beyond.

Think about developer's convenience, we'd better to formalized the way how we could safely introduce new Tx to Plasma Cash.

1. Difficulty to add non-linear history to Plasma Cash

The "swap" Tx takes 2 inputs and 2 outputs. This complexity was the actual trigger for the several discussion such as "Do we need to use confsig?" or "How can we say force include is enough safe?" so on. This required enough amount of eyeballs of professionals. I guess more various Tx would be needed if we want truly trustless & mobile friendly 2nd layer Tx. And everytime we need to owe to pay attention to new Tx. This is somehow not smart and we might be able to make this process efficient by using technology.

2. "VM on the childchain" isn't the only way

The EVM Plasma and its Plasma esque variants are great and actually I love it. I hope it will go to mass adaption. And TxVM is also cool, I hope to see the Sharded rootchain with yankable contract. But for trustless Plasma, what we need is guarantee of the Safety and Liveness. More specifically, 1) Liveness: All UTXO must have right owner in a arbitrary timing and the UTXO must be always exitable. 2) Safety: All invalid exit must be challengeable.

Everytime we add new type of Tx, we must confirm Safety and Liveness of that modified Plasma. This needs bunch of eyeballs. So I would like to suggest to build new language/framework to generate the safety&liveness-verified Plasma Contract, and additionally Plasma childchain as well. (Client SDK and docs might be needed realistically)

I prefer the "swap"-esque bottom-up approach. Because we can comprehensively confirm the minimally modified Plasma Cash. VM-esque approach is for me, somehow looks like, top-down approach. In other words, we need to remove some opcodes, then we need to proof that modified VM's all possible state transitions are really being able to secure the Safety&Liveness property.

Shall we dig more Plasma Cash application?

Let's say, I would like to suggest now it's the time to think about how to make PlasmaCash-app easier. Any feedback is much appreciated!