

[

image

1386×799 138 KB

](https://ethresear.ch/uploads/default/original/2X/f/fe51381218e40a2d6b38ee4de91e94ef280976ec.png)

You can read this post on [here](#) too.

## Intro

The Plasma World Map

is designed to guide anyone traveling on a long journey to plasma safely to their desired destination without getting lost on the way.

It allows new plasma travelers to draw large pictures of numerous studies and provides existing plasma researchers with a good cheat sheet for doing one's research efficiently.

Note : The Plasma World Map is not perfect.

Please let me know if you find something wrong in this article and the map itself.

## Classification

Two major categories of the World Map are Simple Transfer

and General States & Computation

. Each plasma model was classified to each category, considering the purpose it is trying to achieve. You are always welcome if you have an idea of better classification.

## Index

### 1.Simple Transfer

This category includes plasma models which implement simple transfer feature like Bitcoin.

#### UTXO based

- Plasma MVP
- With ZKP
- Plasma Cash

#### Account based

- General
- With ZKP

### 2.General States & Computation

This category includes plasma models which implement general states & computation features like smart contract in Ethereum.

#### UTXO based

#### Account based

## 1. Simple Transfer

### 1) UTXO based : Plasma MVP

This category is for a plasma model which implements the Simple Transfer feature based on UTXO data structures. It is

based on the Minimal Viable Plasma(Plasma MVP).

### **Plasma MVP**

Authors : Vitalik Buterin

Plasma MVP is the first plasma model presented after the plasma white paper was released. It used binary Merkle tree based on UTXO for its data structures. In addition, a challenge system was set up to prevent an invalid exit, and Confirmation and Exit Priority were set up to resolve the data unavailability issue.

Although there was the issue of poor UX caused by the Confirmation phase which requires 2 steps of signature in sending transaction, it was a good starting point for presenting various future researches.

### **Plasma with client-side validation**

Authors : Dan Robinson

In this model, the client-side validation method was applied to Plasma MVP, and the structure of UTXO was transformed to include only signatures and destination. Clients in Plasma chain can verify the UTXO which they received. It also suggested a way of tumbling transaction through pruning.

### **Simple Fast Withdrawals**

Authors : Kelvin Fichter

This article designed a method of implementing Fast Withdrawal which is presented in the white paper. The key idea in here is we can build it only by smart contract on the root chain.

### **More Viable Plasma**

Authors : Ben Jones, Kelvin Fichter

This research proved that we can build the existing valid exit game schema excluding the Confirmation step by changing the exit priority rule from the age of output of UTXO to age of the youngest input.

### **Enabling Fast Withdrawals for Faulty Plasma Chains**

Authors : Kelvin Fichter

This research suggested enabling the Simple Fast Withdrawal even on faulty plasma chains by making the operator deposit on the user's Fast Withdrawal in return for some fees.

### **Optimistic cheap multi-exit for Plasma (Cash or MVP)**

Authors : Vitalik Buterin

This article discussed about enabling single exit for multiple UTXOs or coins by submitting not a Merkle proof for each UTXO or coin but the block number and indices of each UTXO or coin.

### **Basic Mass Exits for Plasma MVP**

Authors : Kelvin Fichter

It showed how mass exit can be executed through the Sum Merkle Tree, which represents the sum of UTXO aggregation and the aggregated value of each UTXO.

### **Debt & Liquidity for L2 Scalability UX**

Authors : Eva Beylin

This is a discussion of ways to liquidate for collateral like deposits for Exit in several Layer2 solutions such as Plasma.

### **RFC : Low Latency Transaction Confirmations Proposal**

Authors : Bing Yang

This article presents a way to reduce transaction-to-transaction latency. It is making transactions contained in blocks but not committed to the root chain available as input to other transactions.

## 2) UTXO based : With ZKP

This category is for a Plasma model that adds verification methods with ZK Proof to the simple transfer feature based on the UTXO data structure.

### [Plasma is Plasma](#)

Authors : josojo

This article is about how to configure a kind of light-client on a Plasma chain by introducing a verification method using zk-snark in the model of Plasma MVP, More Viable Plasma.

## 3) UTXO based : Plasma Cash

This category is for a Plasma model which implements the Simple Transfer feature based on UTXO data structures. It is based on the research of Plasma Cash.

(Plasma Cash is based on the Plasma MVP, but is classified separately because many researches were conducted based on Plasma Cash.)

### [Plasma Cash](#)

Authors : Vitalik Buterin

Plasma Cash is a newly proposed model to improve the problems of Plasma MVP. Each coin was given a unique ID and denomination in this model. It used the Sparse Merkle Tree(SMT) to place the index of each leaf node as a coin ID. And the value contains relevant transaction information when the coin is used. If the coin is not used in that block, the value is null because there is no transaction information for that coin as well. This means that the proof of inclusion and non-inclusion in SMT allows clear verification of the use history of coins.

However, it was pointed out that it is hard to pay partially due to non-fungibility, and the verification of history for each coin could be a very heavy work.

### [PoS Plasma Cash with Sharded Validation](#)

Authors : Karl Floersch

To improve the shortcomings of the single operator model in [Plasma Cash Simple Spec](#), it suggested PoS consensus to resolve the Transaction Censorship and the issue that all users should exit when the data is withheld.

### [Plasma XT: Plasma Cash with much less per-user data checking](#)

Authors : Kelvin Fichter

As the title suggests, the burden of verification of the coin history in Plasma Cash was alleviated through secure checkpointing with [Cryptoeconomic signature aggregation](#) in Plasma XT.

### [Plasma Cash with Sparse Merkle Trees, Bloom filters, and Probabilistic Transfers](#)

Authors : Sourabh Niyogi

In this model, Bloom filters were introduced for efficient non-spent verification, and probabilistic transfers were introduced for partial payment.

### [Loom Network: Plasma Cash for ERC721 Tokens](#)

Authors : Georgios Konstantopoulos

Plasma Cash Implementation specialized for use of ERC721.

### [Plasma Debit: Arbitrary-denomination payments in Plasma Cash](#)

Authors : Dan Robinson

Plasma Debit is a model that improves the problem of non-fungibility, which was pointed out as a disadvantage of Plasma Cash. To do so, it made each coin to be used as a form of payment channel between users and operators or other users.

### [Plasma Cash Minimal Atomic Swap](#)

Authors : Vitalik Buterin

Atomic Swap protocol for Plasma Cash. In this protocol, most of the actions required for the Atomic swap are executed on the plasma chain. Only the sharing keys is executed on the main chain so that the Atomic swap could be carried out efficiently.

### [Plasma Cash Defragmentation](#)

Authors : Vitalik Buterin

Plasma Cash Defragmentation is about the Defragmentation method to improve the indivisibility of the coin in Plasma Cash. The key idea is to allow the users to put those coins scattered on single Merkle branch through kind of Atomic swap when they send coins.

### [Plasma Cash defragmentation, take 2](#)

Authors : Vitalik Buterin

This research suggested how the operator can execute defragmentation by sending a permutation transaction. This special transaction consists of swaps of each coin to defragment them.

### [Plasma Cash defragmentation, take 3](#)

Authors : Vitalik Buterin

It showed how defragmentation can be performed by minimizing the generation of fragment when users send transaction each other.

### [Plasma Cashflow](#)

Authors : Hayden Adams

Plasma Cashflow is a Plasma Cash Implementation that is based on the ideas of Plasma Debit and Plasma Defragmentation.

### [RSA Accumulators for Plasma Cash history reduction](#)

Authors : Vitalik Buterin

This research showed how to reduce the verification burden for the history of each coin in Plasma Cash using RSA Accumulators. This idea was the basis for the Plasma Prime.

### [Log\(coins\)-sized proofs of inclusion and exclusion for RSA accumulators](#)

Authors : Vitalik Buterin

This research proved that we can execute the inclusion and non-inclusion proof of the coin in Plasma Cash with  $O(\log n)$  using RSA accumulators.

### [Plasma Prime\(spec by vitalik\)](#)

Authors : Sourabh Niyogi

This is a summary of what Vitalik Buterin explained about the specific spec of Plasma Prime in the Plasma Implementers Call.

### [Plasma Prime design proposal\(spec by Bankex\)](#)

Authors : Bankex team

This is the Plasma Prime Implementation designed by the Bankex team. It is based on Plasma Cashflow.

## **4) Account based : General**

This category is for a plasma model which implements the simple transfer feature based on Account-balance data structures.

### [Chronos: A Quirky Application Proposal for Plasma](#)

Authors : Paul Berg, Mark Milton

Plasma implementation that used kind of payment channel.

### [A DEX on Plasma](#)

Authors : Bharath Rao

It explained why it is difficult to build the decentralized exchange(DEX) in Plasma MVP and Plasma Cash. It suggested a different way to build the DEX in Plasma. More details can be found at Gluon Plasma.

### [Gluon Plasma](#)

Authors : Bharath Rao

Gluon Plasma suggested a plasma model that focuses on improving UX and Scaling. However, due to the lack of solutions to the Data unavailability issue, it is hard to say that it was well constructed.

### [Off-chain Plasma state validation with on-chain smart contract](#)

Authors : Jieyi Long

It suggested a way to force only the valid state transition through probabilistic verification of account balance in the plasma chain.

## **5) Account based : With ZKP**

This category is for a plasma model that adds verification methods with ZK Proof to the simple transfer feature based on the Account-balance data structures.

### [Quark-gluon Plasma — verified Plasma chain without confirmation signatures](#)

Authors : Alex Vlasov

Quark-gluon Plasma is a plasma chain based on account-balance data structures which can be verified its valid state transition with zkSNARKs. However, the solution to the data unavailability was insufficient.

### [Roll up / roll back snark side chain](#)

Authors : Barry White Hat, Alex Gluchowski, HarryR 33, Yondon Fu, Philippe Castonguay

It is research about the side chain with zkSNARKs. It suggested operator replacement auction mechanism when the operator stops the operation for a specific period. And roll-back mechanism was suggested to enable new operator to keep making the blocks.

### [Plasma Snapp](#)

Authors : josojo

Plasma Snapp is a plasma chain that can verify the state transition including deposit and withdrawal with zkSNARKs. Like the [Roll up / roll back snark side chain](#), it also allowed anyone to become an operator when the operator was absent for a long period of time. And the roll back mechanism was used to return the chain to its previous state for the new operator. And users can keep safe one's fund through higher priority in exits. Although the data is withheld by the operator, priority of user's exit is always higher than operator's if the users did not send transaction after the withholding.

## **2. General States & Computation**

### **1) UTXO based**

This category is for a plasma model which implements general states & computation features using the advantages of limited features that UTXO has.

### [Why Smart Contracts are NOT feasible on Plasma](#)

Authors : Johann Barbie

This is discussion about why it is difficult to implement Smart Contract in Plasma. It was briefly stated that it is not possible to

force a valid state transition with existing Exit game in the data unavailability. It was also the theoretical basis for Plasma leap.

### [Plasma Leap — a State-Enabled Computing Model for Plasma](#)

Authors : Johann Barbie

Plasma Leap implemented general computation feature with a small program called Spending Condition which is similar to Pay-to-Script-Hash (P2SH) in Bitcoin. And it also implemented general states feature with non-fungible storage token(NST) which is kind of ERC721 token with storage root hash. It resolved the data availability issue using exit model of More Viable Plasma.

## **2) Account based**

This category is for a plasma model which implements general states & computation features similar or equal to Ethereum.

### [Plasmabits: Viable Stateful Sidechain](#)

Authors : Esteban Ordano

Plasmabits is a plasma implementation proposal that can use EVM-based smart contract. It suggested PASITO contract (Plasma Arbitration Stepping Instruction Test Operator) borrowing the Truebit-like verification game as a way to verify valid state transition. And it also suggested query and respond system for availability of blocks data. But anyone can attack the operator by making query to the operator for all blocks. That means, every plasma block should be included on the root chain and that is not so good.

### [Plasma EVM : state-enforceable construction](#)

Authors : Carl Park , Aiden Park, Kevin Jeong

Plasma EVM is a new version of Plasma that can execute EVM in plasma chain. It used Truebit-like verification game as a way to verify valid state transition in plasma chain. It also suggested concept of Requestable Contract to resolve exit authority problem in Contract Account. And User Activated Fork model was introduced to deal with the data unavailability.

## **Comment**

Plasma World has been growing rapidly since its birth in August 2017. As many researchers keep working to expand Plasma's territory, Plasma World Map will also be updated periodically to quickly reflect the newly pioneered areas.