Suppose you want to have a Plasma chain with multiple parent chains, where the Plasma chain contains assets that come from all of these chains (this can be useful for applications like running a decentralized exchange between these assets, without requiring either chain to contain a light client of the other). At first this seems simple to implement: just put a Plasma root contract on each chain, and then have each contract recognize only the assets on the Plasma chain that corresponds to assets on that particular root chain.

But this leads to a problem: how do you ensure consistency between the root chains? For example, suppose that there is a Plasma chain with the ETH and ETC chains as roots, and ETH and ETC as the two assets. Suppose block X of the Plasma chain contains a transaction sending ETH from A to B and sending ETC from B to A (this is a decentralized exchange swap). A malicious operator colluding with B can attack by creating block X, then publishing the root into the ETH chain but not

into the ETC chain (or publishing some alternative root X' into the ETC chain that does not contain that transaction). This causes the ETH side to be fulfilled but not the ETC side.

We can solve this problem with Lamport's 99% fault tolerant consensus. Suppose the Plasma chain has its own native currency, and deposits in the native currency form a proof of stake system, with bonded validators. We add the following rule. In order for one of the root chains to accept a block header, it must be signed by a validator; the index of the validator used is used as a randomness seed to pick out a random other 40 validators. The inclusion transaction must also specify an inclusion timestamp, T, and it must be submitted between time T and time T+D (eg. D = 1 hour). The same inclusion transaction can then be published into the other root chains within the same time window, or, if k validators from the random subset co-sign, it can be published into the other root chains before time T + k * D. If the block is included into one root chain, any honest validator can thus add their own signature and cause it to be submitted into the other root chains within the additional time window of D seconds by which their signature extends the deadline.

This does require a trust assumption of Plasma chain validators, but it is a very limited one, requiring only ~1-10% of validators to be honest depending on the risk tolerance level and the validator set size (it can be less or more than 40 as desired).