chaincode

BIP 125

Opt-in Full Replace-by-Fee Signaling (Harding & Todd, 2015)

Motivation for Replace-by-Fee

Block and transaction rejection is build upon the **first-seen** rule

- Miners build on top of the first block they receive and reject others
- Nodes reject transaction spending an output that's already spend in the mempool

Yet we sometimes want to be able to replace an unconfirmed transaction…

- To bumping the fee
- To compress multiple transactions into one
- ...

History and alternative proposals

Inputs have a nSequence (Nakamoto, 2008)

Idea: Replace unconfirmed transactions with transactions having a higher nSequence

Problem: No miner incentive to include the replacements and no rate limiting on broadcasting

→ Feature removed by Nakamoto in Bitcoin v0.3.12

We want replaceable transactions, but not break existing first-seen merchants

Idea: RBF First-Seen-Safe (Todd, 2015): Pay at least equal or more to all outputs

Problem: Requires an extra input:

- Unable to use it without spare inputs
- Possible loss of privacy (change clearer and extra input included)
- Wasteful increase in transaction size

Specification I

Transactions can either **explicitly signal** or **inherit** RBF replaceability

- **Explicit signaling:**
  When at least one of its inputs have an nSequence less than MAX_INT - 1 (`0xffffffff-1`)

- **Inherited signaling:**
  When at least one unconfirmed ancestor signals replaceability

Specification II - The five RBF Rules

Original transactions are replaced by a replacement that spends at least one of the same inputs if

1. The original transaction signals replaceability (explicitly or inherent)
2. All new outputs spend in the replacement must be confirmed
3. The replacement pays an absolute fee of at least the sum paid by the original transactions
4. The replacement pays for its own bandwidth (at least the minimum relay fee)
5. The number of original transactions replaced does not exceed 100 transactions

Proposal: *Emergency RBF* (June 2019)

Problem: It's often infeasible to RBF a large child transaction paying a high absolute fee (e.g. a commercial service sweeping up your output in a transaction with a lot of other outputs)

Idea: Add a new 6th rule to BIP-125

6. Replaceable if the original transaction is *not* in the most profitable vMB of the fee-ordered mempool and the replacement transaction is, rules 3, 4 and 5 do not apply.

⇒ Transactions can be bumped to a feerate where they will most likely confirm soon without paying for the large child transaction.

Questions and Thank you?