InstantSend Overview

InstantSend is a feature provided by the Dash network that allows for zero-confirmation transactions to be safely accepted by Merchants and other service providers. All InstantSend Transactions are secured for 25 blocks by the “Masternode Network” at the moment of broadcast. The transaction is mined into the next block in accordance with standard blockchain principles.

InstantSend is enabled by the Masternode Network which comprises approximately 4,800 masternode servers. These nodes are differentiated from standard nodes by having proven ownership of 1,000 Dash. One responsibility that is appointed to this special type of server is to perform "Transaction Locking", also known as InstantSend.

This concept works as an extension to network consensus. When an "InstantSend" transaction occurs the network goes through an extra validation process which examines the following two properties of the transaction:

1. **Input Maturity**: the network will require all inputs to have at least 6 confirmations.

2. **Input Composition**: the number of inputs in use dictates fee requirements.

    a. 1 - 4 inputs: per-kB fee of 0.00001 DASH using Automatic InstantSend.

    b. 5+ inputs: per-input fee of 0.0001 DASH is required.

Assuming the **Input Maturity** and **Input Composition** requirements are met, the network will "lock" the inputs related to this transaction for 25 blocks. Transactions carrying 4 or fewer inputs are referred to as a “simple transaction” and carry no extra fee.

Automatic InstantSend

Dash Core v0.13.x introduces the process of Automatic InstantSend. Any transaction which is classified as a “simple transaction” will automatically be broadcast as an InstantSend transaction when using standard transaction broadcast endpoints.

InstantSend vs. Standard Transactions

The term “InstantSend” is used to describe a standard transaction that has been provided additional assurances by the Masternode Network. As a result, and from an integration perspective, there is no technical difference between the two types of transactions.

The most notable difference relates to the way that confirmation policy is applied within an integrated system. The receiving system must be aware of InstantSend Status in order to safely apply transaction confirmation policies that are enabled using this technology.

Receiving InstantSend Transactions

Receiving an InstantSend Transaction introduces two requirements:

1. The ability to determine the “InstantSend Status” of a given transaction.
2. The ability to adjust “Confirmation Status” independently of block confirmation.

InstantSend Status is typically determined through direct connection with the dash daemon, ZMQ notification, or through the usage of an external wallet notification script.

Direct Connection: InstantSend Status can be identified through direct connection with the dash daemon using JSON-RPC protocol. The “instantlock” attribute of the JSON response reflects the status of the transaction and is included in the following commands:

getrawmempool, getmempoolancestors, getmempooldescendants, getmempoolentry, getrawtransaction, decoderawtransaction, gettransaction, listtransactions, listsinceblock.

Wallet Notification: The Dash Core Daemon can be configured to execute an external script whenever an InstantSend transaction relating to that wallet is observed. This is configured by adding the following line to the dash.conf file:

➔ instantsendnotify=/path/to/concurrent/safe/handler %s

This is typically used with a wallet that has been populated with “watch-only” addresses.

Broadcasting InstantSend Transactions

Automatic InstantSend introduces two requirements into the system being integrated:

1. The ability to evaluate the number of inputs in a given transaction.
2. The ability to apply an increased fee-level in cases of 5+ inputs.

In many cases an integrated system will already contain logic intended to keep transaction fees to a minimum by optimizing input usage. If this is true, and it can be assumed that only “simple transactions” are being formed, no additional development effort is required.

In cases where this is not possible or will be unknown, the integrated system should be able to calculate a fee based on the number of inputs being used to form the transaction. This per-input fee of 0.0001 DASH must be applied in order for the transaction to be successfully broadcast as an InstantSend.

In these cases, it’s important to note that the “instantsend” flag must be set as “true” when issuing the sendrawtransaction command, e.g.:

➔ sendrawtransaction "hexstring" false true

Additional Resources

The following resources provide additional information about InstantSend and are intended to help provide a more complete understanding of the underlying technologies.

★ InstantSend Whitepaper

★ InstantSend Technical Information

★ InstantSend Developer Documentation

★ Product Brief: Dash Core v0.13 Release