

BIP 0031 - Pong message

[return to the Bitcoin Improvement Proposals](#)

Table 1: Data sheet for Pong message

Title	Pong message
Layer	Peer Services
Author	Mike Hearn
Comments-Summary	No comments yet.
Comments-URI	https://github.com/bitcoin/bips/wiki/Comments:BIP-0031
Status	Final
Type	Standards Track
Created	2012-04-11
Post History	
Description	https://github.com/bitcoin/bips/blob/master/bip-0031.mediawiki

Note: The following is an excerpt from the official [Bitcoin](#) site. It is provided here as a convenience and is not authoritative. Refer to the original document(s) as the authoritative reference.

Abstract

This document describes a trivial protocol extension that makes it easier for [clients](#) to detect dead peer connections.

Motivation

Today there are a few network related problems that can degrade the Bitcoin user experience:

- 1. Some Bitcoin clients run on platforms that can go to sleep and essentially stop running at any time without warning. Notably, this is very common on both mobiles and laptops (shut the lid). When the system comes back, TCP connections that existed before the sleep still exist but may no longer function correctly, eg, because the IP address has changed, or because the remote peer went away or the connection was timed out by some other system. Currently it can often take a while to notice this has happened.*
- 2. The reference Satoshi client is largely single threaded and when placed under heavy load (e.g., because it is downloading the block chain) becomes very slow to respond to network messages. There's no easy way to detect this has occurred, especially if you are just passively waiting for broadcasts from that peer. A way to detect overloaded remote peers and avoid them would both help balance load and provide a better, more responsive system.*
- 3. When downloading large data structures like the block chain it is efficient to choose a peer that is near to you network-wise, in order to reduce load on often congested trans-national links and*

ensure lower latency. Currently it is difficult to measure the [latency](#) to a remote peer so clients don't bother, and instead just select a random peer to download from.

All of these can be solved by a backwards compatible protocol modification.

From:

<https://www.omgwiki.org/dido/> - **DIDO Wiki**

Permanent link:

https://www.omgwiki.org/dido/doku.php?id=dido:public:ra:xapend:xapend.b_stds:defact:bitcoin:bips:bip_0031

Last update: **2021/08/09 13:22**

