

Bitcoin version 0.7.0 is now available for download at: <http://sourceforge.net/projects/bitcoin/files/Bitcoin/bitcoin-0.7.0/>

We recommend that everybody running prior versions of bitcoind/Bitcoin-Qt upgrade to this release, except for users running Mac OSX 10.5.

Please report bugs using the issue tracker at github: <https://github.com/bitcoin/bitcoin/issues>

Project source code is hosted at github; you can get source-only tarballs/zipballs directly from there: <https://github.com/bitcoin/bitcoin/tarball/v0.7.0> # .tar.gz
<https://github.com/bitcoin/bitcoin/zipball/v0.7.0> # .zip

Ubuntu Linux users can use the “Personal Package Archive” (PPA) maintained by Matt Corallo to automatically keep bitcoin up-to-date. Just type `sudo apt-add-repository ppa:bitcoin/bitcoin` `sudo apt-get update` in your terminal, then install the bitcoin-qt package: `sudo apt-get install bitcoin-qt`

How to Upgrade

If you are running an older version, shut it down. Wait until it has completely shut down (which might take a few minutes for older versions), then run the installer (on Windows) or just copy over Code: /Applications/Bitcoin-Qt (on Mac) or Code: bitcoind/bitcoin-qt (on Linux).

If you were running on Linux with a version that might have been compiled with a different version of Berkeley DB (for example, if you were using the PPA and are switching to the binary release), then run the old version again with the `-detachdb` argument and shut it down; if you do not, then the new version will not be able to read the database files and will exit with an error.

Incompatible Changes

- Replaced the ‘getmemorypool’ RPC command with ‘getblocktemplate/submitblock’ and ‘getrawmempool’ commands.
- Remove deprecated RPC ‘getblocknumber’

Bitcoin Improvement Proposals implemented

BIP 22 - ‘getblocktemplate’, ‘submitblock’ RPCs BIP 34 - block version 2, height in coinbase BIP 35 - ‘mempool’ message, extended ‘getdata’ message behavior

Core bitcoin handling and blockchain database

- Reduced CPU usage, by eliminating some redundant hash calculations
- Cache signature verifications, to eliminate redundant signature checks
- Transactions with zero-value outputs are considered non-standard
- Mining: when creating new blocks, sort ‘paid’ area by fee-per-kb
- Database: better validation of on-disk stored data
- Database: minor optimizations and reliability improvements
- `-loadblock=FILE` will import an external block file
- Additional DoS (denial-of-service) prevention measures
- New blockchain checkpoint at block 193,000

JSON-RPC API

- Internal HTTP server is now thread-per-connection, rather than a single-threaded queue that would stall on network I/O.
- Internal HTTP server supports HTTP/1.1, pipelined requests and connection keep-alive.
- Support JSON-RPC 2.0 batches, to encapsulate multiple JSON-RPC requests within a single HTTP request.
- IPv6 support
- Added raw transaction API. See <https://gist.github.com/2839617>
- Added 'getrawmempool', to list contents of TX memory pool
- Added 'getpeerinfo', to list data about each connected network peer
- Added 'listaddressgroupings' for better coin control
- Rework getblock call.
- Remove deprecated RPC 'getblocknumber'
- Remove superceded RPC 'getmemorypool' (see BIP 22, above)
- listtransactions output now displays "smart" times for transactions, and 'blocktime' and 'timereceived' fields were added

P2P networking

- IPv6 support
- Tor hidden service support (see doc/Tor.txt)
- Attempts to fix "stuck blockchain download" problems
- Replace BDB database "addr.dat" with internally-managed "peers.dat" file containing peer address data.
- Lower default send buffer from 10MB to 1MB
- proxy: SOCKS5 by default
- Support connecting by hostnames passed to proxy
- Add -seednode connections, and use this instead of DNS seeds when proxied
- Added -externalip and -discover
- Add -onlynet to connect only to a given network (IPv4, IPv6, or Tor)
- Separate listening sockets, -bind=

Qt GUI

- Add UI RPC console / debug window
- Re-Enable URI handling on Windows, add safety checks and tray-notifications
- Harmonize the use of ellipsis ("...") to be used in menus, but not on buttons
- Add 2 labels to the overviewpage that display Wallet and Transaction status (obsolete or current)
- Extend the optionsdialog (e.g. language selection) and re-work it to a tabbed UI
- Merge sign/verify message into a single window with tabbed UI
- Ensure a changed bitcoin unit immediately updates all GUI elements that use units

- Update QR Code dialog
- Improve error reporting at startup
- Fine-grained UI updates for a much smoother UI during block downloads
- Remove autocorrection of 0/i in addresses in UI
- Reorganize tray icon menu into more logical order
- Persistently poll for balance change when number of blocks changed
- Much better translations
- Override progress bar design on platforms with segmented progress bars to assist with readability
- Added ‘immature balance’ display on the overview page
- (Windows only): enable ASLR and DEP for bitcoin-qt.exe
- (Windows only): add meta-data to bitcoin-qt.exe (e.g. description)

Internal codebase

- Additional unit tests
- Compile warning fixes

Miscellaneous

- Reopen debug.log upon SIGHUP
- Bash programmable completion for bitcoind(1)
- On supported OS’s, each thread is given a useful name

Thanks to everybody who contributed to this release:

Chris Moore Christian von Roques David Joel Schwartz Douglas Huff Fordy Gavin Andresen Giel van Schijndel Gregory Maxwell Jeff Garzik Luke Dashjr Matt Corallo Michael Ford Michael Hendricks Peter Todd Philip Kaufmann Pieter Wuille R E Broadley Ricardo M. Correia Rune K. Svendsen Scott Ellis Stephane Glondou Wladimir J. van der Laan cardpuncher coderrr fanquake grimd34th sje397 xanatos

Thanks to Sergio Lerner for reporting denial-of-service vulnerabilities fixed in this release.