Bitcoin Core version 0.14.0 is now available from:

https://bitcoin.org/bin/bitcoin-core-0.14.0/

This is a new major version release, including new features, various bugfixes and performance improvements, as well as updated translations.

Please report bugs using the issue tracker at github:

https://github.com/bitcoin/bitcoin/issues

To receive security and update notifications, please subscribe to:

https://bitcoincore.org/en/list/announcements/join/

Compatibility

Bitcoin Core is extensively tested on multiple operating systems using the Linux kernel, macOS 10.8+, and Windows Vista and later.

Microsoft ended support for Windows XP on April 8th, 2014, No attempt is made to prevent installing or running the software on Windows XP, you can still do so at your own risk but be aware that there are known instabilities and issues. Please do not report issues about Windows XP to the issue tracker.

Bitcoin Core should also work on most other Unix-like systems but is not frequently tested on them.

Notable changes

Performance Improvements

Validation speed and network propagation performance have been greatly improved, leading to much shorter sync and initial block download times.

- The script signature cache has been reimplemented as a "cuckoo cache", allowing for more signatures to be cached and faster lookups.
- Assumed-valid blocks have been introduced which allows script validation to be skipped for ancestors of known-good blocks, without changing the security model. See below for more details.
- In some cases, compact blocks are now relayed before being fully validated as per BIP152.
- P2P networking has been refactored with a focus on concurrency and throughput. Network operations are no longer bottlenecked by validation. As a result, block fetching is several times faster than previous releases in many cases.
- The UTXO cache now claims unused mempool memory. This speeds up initial block download as UTXO lookups are a major bottleneck there, and there is no use for the mempool at that stage.

Manual Pruning

Bitcoin Core has supported automatically pruning the blockchain since 0.11. Pruning the blockchain allows for significant storage space savings as the vast majority of the downloaded data can be discarded after processing so very little of it remains on the disk.

Manual block pruning can now be enabled by setting `-prune=1`. Once that is set, the RPC command `pruneblockchain` can be used to prune the blockchain up to the specified height or timestamp.

`getinfo` Deprecated

The `getinfo` RPC command has been deprecated. Each field in the RPC call has been moved to another command's output with that command also giving additional information that `getinfo` did not provide. The following table shows where each field has been moved to:

`getinfo` field	Moved to
`"version"`	`getnetworkinfo()["version"]`
`"protocolversion"`	`getnetworkinfo()["protocolversion"]`
`"walletversion"`	`getwalletinfo()["walletversion"]`
`"balance"`	`getwalletinfo()["balance"]`
`"blocks"`	`getblockchaininfo()["blocks"]`
`"timeoffset"`	`getnetworkinfo()["timeoffset"]`
`"connections"`	`getnetworkinfo()["connections"]`
`"proxy"`	`getnetworkinfo()["networks"][0]["proxy"]`
`"difficulty"`	`getblockchaininfo()["difficulty"]`
`"testnet"`	`getblockchaininfo()["chain"] == "test"`
`"keypoololdest"`	`getwalletinfo()["keypoololdest"]`
`"keypoolsize"`	`getwalletinfo()["keypoolsize"]`
`"unlocked_until"`	`getwalletinfo()["unlocked_until"]`
`"paytxfee"`	`getwalletinfo()["paytxfee"]`
`"relayfee"`	`getnetworkinfo()["relayfee"]`
`"errors"`	`getnetworkinfo()["warnings"]`

ZMQ On Windows

Previously the ZeroMQ notification system was unavailable on Windows due to various issues with ZMQ. These have been fixed upstream and now ZMQ can be used on Windows. Please see this document for help with using ZMQ in general.

Nested RPC Commands in Debug Console

The ability to nest RPC commands has been added to the debug console. This allows users to have the output of a command become the input to another command without running the commands separately.

The nested RPC commands use bracket syntax (i.e. `getwalletinfo()`) and can be nested (i.e. `getblock(getblockhash(1))`). Simple queries can be done with square brackets where object values are accessed with either an array index or a non-quoted string (i.e. `listunspent()[0][txid]`). Both commas and spaces can be used to separate parameters in both the bracket syntax and normal RPC command syntax.

Network Activity Toggle

A RPC command and GUI toggle have been added to enable or disable all p2p network activity. The network status icon in the bottom right hand corner is now the GUI toggle. Clicking the icon will either enable or disable all p2p network activity. If network activity is disabled, the icon will be grayed out with an X on top of it.

Additionally the `setnetworkactive` RPC command has been added which does the same thing as the GUI icon. The command takes one boolean parameter, `true` enables networking and `false` disables it.

Out-of-sync Modal Info Layer

When Bitcoin Core is out-of-sync on startup, a semi-transparent information layer will be shown over top of the normal display. This layer contains details about the current sync progress and estimates the amount of time remaining to finish syncing. This layer can also be hidden and subsequently unhidden by clicking on the progress bar at the bottom of the window.

Support for JSON-RPC Named Arguments

Commands sent over the JSON-RPC interface and through the `bitcoin-cli` binary can now use named arguments. This follows the [JSON-RPC specification]() for passing parameters by-name with an object.

`bitcoin-cli` has been updated to support this by parsing `name=value` arguments when the `-named` option is given.

Some examples:

```
src/bitcoin-cli -named help command="help"
src/bitcoin-cli -named getblockhash height=0
src/bitcoin-cli -named getblock
blockhash=000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f
src/bitcoin-cli -named sendtoaddress address="(snip)" amount="1.0"
subtractfeefromamount=true
```

The order of arguments doesn't matter in this case. Named arguments are also useful to leave out arguments that should stay at their default value. The rarely-used arguments `comment` and `comment_to` to `sendtoaddress`, for example, can be left out. However, this is not yet implemented for many RPC calls, this is expected to land in a later release.

The RPC server remains fully backwards compatible with positional arguments.

Opt into RBF When Sending

A new startup option, `-walletrbf`, has been added to allow users to have all transactions sent opt into RBF support. The default value for this option is currently `false`, so transactions will not opt into RBF by default. The new `bumpfee` RPC can be used to replace transactions that opt into RBF.

Sensitive Data Is No Longer Stored In Debug Console History

The debug console maintains a history of previously entered commands that can be accessed by pressing the Up-arrow key so that users can easily reuse previously entered commands. Commands which have sensitive information such as passphrases and private keys will now have a `(...)` in place of the parameters when accessed through the history.

Retaining the Mempool Across Restarts

The mempool will be saved to the data directory prior to shutdown to a `mempool.dat` file. This file preserves the mempool so that when the node restarts the mempool can be filled with transactions without waiting for new transactions to be created. This will also preserve any changes made to a transaction through commands such as `prioritisetransaction` so that those changes will not be lost.

Final Alert

The Alert System was disabled and deprecated in Bitcoin Core 0.12.1 and removed in 0.13.0. The Alert System was retired with a maximum sequence final alert which causes any nodes supporting the Alert System to display a static hard-coded "Alert Key Compromised" message which also prevents any other alerts from overriding it. This final alert is hard-coded into this release so that all old nodes receive the final alert.

GUI Changes

- After resetting the options by clicking the `Reset Options` button in the options dialog or with the `-resetguioptions` startup option, the user will be prompted to choose the data directory again. This is to ensure that custom data directories will be kept after the option reset which clears the custom data directory set via the choose datadir dialog.

- Multiple peers can now be selected in the list of peers in the debug window. This allows for users to ban or disconnect multiple peers simultaneously instead of banning them one at a time.

- An indicator has been added to the bottom right hand corner of the main window to indicate whether the wallet being used is a HD wallet. This icon will be grayed out with an X on top of it if the wallet is not a HD wallet.

Low-level RPC changes

- `importprunedfunds` only accepts two required arguments. Some versions accept an optional third arg, which was always ignored. Make sure to never pass more than two arguments.

- The first boolean argument to `getaddednodeinfo` has been removed. This is an incompatible change.

- RPC command `getmininginfo` loses the "testnet" field in favor of the more generic "chain" (which has been present for years).

- A new RPC command `preciousblock` has been added which marks a block as precious. A precious block will be treated as if it were received earlier than a competing block.

- A new RPC command `importmulti` has been added which receives an array of JSON objects representing the intention of importing a public key, a private key, an address and script/p2sh

- Use of `getrawtransaction` for retrieving confirmed transactions with unspent outputs has been deprecated. For now this will still work, but in the future it may change to only be able to retrieve information about transactions in the mempool or if `txindex` is enabled.

- A new RPC command `getmemoryinfo` has been added which will return information about the memory usage of Bitcoin Core. This was added in conjunction with optimizations to memory management. See Pull #8753 for more information.

- A new RPC command `bumpfee` has been added which allows replacing an unconfirmed wallet transaction that signaled RBF (see the `-walletrbf` startup option above) with a new transaction that pays a higher fee, and should be more likely to get confirmed quickly.

HTTP REST Changes

- UTXO set query (`GET /rest/getutxos/<checkmempool>/<txid>-<n>/<txid>-<n>/.../<txid>-<n>.<bin|hex|json>`) responses were changed to return status code `HTTP_BAD_REQUEST` (400) instead of `HTTP_INTERNAL_SERVER_ERROR` (500) when requests contain invalid parameters.

Minimum Fee Rate Policies

Since the changes in 0.12 to automatically limit the size of the mempool and improve the performance of block creation in mining code it has not been important for relay nodes or miners to set `-minrelaytxfee`. With this release the following concepts that were tied to this option have been separated out:

- incremental relay fee used for calculating BIP 125 replacement and mempool limiting. (1000 satoshis/kB)
- calculation of threshold for a dust output. (effectively 3 * 1000 satoshis/kB)
- minimum fee rate of a package of transactions to be included in a block created by the mining code. If miners wish to set this minimum they can use the new `-blockmintxfee` option. (defaults to 1000 satoshis/kB)

The `-minrelaytxfee` option continues to exist but is recommended to be left unset.

Fee Estimation Changes

- Since 0.13.2 fee estimation for a confirmation target of 1 block has been disabled. The fee slider will no longer be able to choose a target of 1 block. This is only a minor behavior change as there was often insufficient data for this target anyway. `estimatefee 1` will now always return -1 and `estimatesmartfee 1` will start searching at a target of 2.

- The default target for fee estimation is changed to 6 blocks in both the GUI (previously 25) and for RPC calls (previously 2).

Removal of Priority Estimation

- Estimation of "priority" needed for a transaction to be included within a target number of blocks has been removed. The RPC calls are deprecated and will either return -1 or 1e24 appropriately. The format for `fee_estimates.dat` has also changed to no longer save these priority estimates. It will automatically be converted to the new format which is not readable by prior versions of the software.

- Support for "priority" (coin age) transaction sorting for mining is considered deprecated in Core and will be removed in the next major version. This is not to be confused with the `prioritisetransaction` RPC which will remain supported by Core for adding fee deltas to transactions.

P2P connection management

- Peers manually added through the `-addnode` option or `addnode` RPC now have their own limit of eight connections which does not compete with other inbound or outbound connection usage and is not subject to the limitation imposed by the `-maxconnections` option.

- New connections to manually added peers are performed more quickly.

Introduction of assumed-valid blocks

- A significant portion of the initial block download time is spent verifying scripts/signatures. Although the verification must pass to ensure the security of the system, no other result from this verification is needed: If the node knew the history of a given block were valid it could skip checking scripts for its ancestors.

- A new configuration option 'assumevalid' is provided to express this knowledge to the software. Unlike the 'checkpoints' in the past this setting does not force the use of a particular chain: chains that are consistent with it are processed quicker, but other chains are still accepted if they'd otherwise be chosen as best. Also unlike 'checkpoints' the user can configure which block history is assumed true, this means that even outdated software can sync more quickly if the setting is updated by the user.

- Because the validity of a chain history is a simple objective fact it is much easier to review this setting. As a result the software ships with a default value adjusted to match the current chain shortly before release. The use of this default value can be disabled by setting -assumevalid=0

Fundrawtransaction change address reuse

- Before 0.14, `fundrawtransaction` was by default wallet stateless. In almost all cases `fundrawtransaction` does add a change-output to the outputs of the funded transaction. Before 0.14, the used keypool key was never marked as change-address key and directly returned to the keypool (leading to address reuse). Before 0.14, calling `getnewaddress` directly after `fundrawtransaction` did generate the same address as the change-output address.

- Since 0.14, fundrawtransaction does reserve the change-output-key from the keypool by default (optional by setting `reserveChangeKey`, default = `true`)

- Users should also consider using `getrawchangeaddress()` in conjunction with `fundrawtransaction`'s `changeAddress` option.

Unused mempool memory used by coincache

- Before 0.14, memory reserved for mempool (using the `-maxmempool` option) went unused during initial block download, or IBD. In 0.14, the UTXO DB cache (controlled with the `-dbcache` option) borrows memory from the mempool when there is extra memory available. This may result in an increase in memory usage during IBD for those previously relying on only the `-dbcache` option to limit memory during that time.

0.14.0 Change log

Detailed release notes follow. This overview includes changes that affect behavior, not code moves, minor refactors and string updates. For convenience in locating the code changes and accompanying discussion, both the pull request and git merge commit are mentioned.

RPC and other APIs

- #8421 `b77bb95` httpserver: drop boost dependency (theuni)
- #8638 `f061415` rest.cpp: change `HTTP_INTERNAL_SERVER_ERROR` to `HTTP_BAD_REQUEST` (djpnewton)
- #8272 `91990ee` Make the dummy argument to getaddednodeinfo optional (sipa)
- #8722 `bb843ad` bitcoin-cli: More detailed error reporting (laanwj)
- #6996 `7f71a3c` Add preciousblock RPC (sipa)
- #8788 `97c7f73` Give RPC commands more information about the RPC request (jonasschnelli)
- #7948 `5d2c8e5` Augment getblockchaininfo bip9_softforks data (mruddy)
- #8980 `0e22855` importmulti: Avoid using boost::variant::operator!=, which is only in newer boost versions (luke-jr)
- #9025 `4d8558a` Getrawtransaction should take a bool for verbose (jnewbery)
- #8811 `5754e03` Add support for JSON-RPC named arguments (laanwj)
- #9520 `2456a83` Deprecate non-txindex getrawtransaction and better warning (sipa)
- #9518 `a65ced1` Return height of last block pruned by pruneblockchain RPC (ryanofsky)
- #9222 `7cb024e` Add 'subtractFeeFromAmount' option to 'fundrawtransaction' (dooglus)
- #8456 `2ef52d3` Simplified `bumpfee` command (mrbandrews)
- #9516 `727a798` Bug-fix: listsinceblock: use fork point as reference for blocks in reorg'd chains (kallewoof)
- #9640 `7bfb770` Bumpfee: bugfixes for error handling and feerate calculation (sdaftuar)
- #9673 `8d6447e` Set correct metadata on bumpfee wallet transactions (ryanofsky)
- #9650 `40f7e27` Better handle invalid parameters to signrawtransaction (TheBlueMatt)
- #9682 `edc9e63` Require timestamps for importmulti keys (ryanofsky)
- #9108 `d8e8b06` Use importmulti timestamp when importing watch only keys (on top of #9682) (ryanofsky)
- #9756 `7a93af8` Return error when importmulti called with invalid address (ryanofsky)
- #9778 `ad168ef` Add two hour buffer to manual pruning (morcos)
- #9761 `9828f9a` Use 2 hour grace period for key timestamps in importmulti rescans (ryanofsky)
- #9474 `48d7e0d` Mark the minconf parameter to move as ignored (sipa)
- #9619 `861cb0c` Bugfix: RPC/Mining: GBT should return 1 MB sizelimit before segwit activates (luke-jr)
- #9773 `9072395` Return errors from importmulti if complete rescans are not successful (ryanofsky)

Block and transaction handling

- #8391 `37d83bb` Consensus: Remove ISM (NicolasDorier)
- #8365 `618c9dd` Treat high-sigop transactions as larger rather than rejecting them (sipa)
- #8814 `14b7b3f` wallet, policy: ParameterInteraction: Don't allow 0 fee (MarcoFalke)
- #8515 `9bdf526` A few mempool removal optimizations (sipa)
- #8448 `101c642` Store mempool and prioritization data to disk (sipa)
- #7730 `3c03dc2` Remove priority estimation (morcos)
- #9111 `fb15610` Remove unused variable `UNLIKELY_PCT` from fees.h (fanquake)
- #9133 `434e683` Unset fImporting for loading mempool (morcos)
- #9179 `b9a87b4` Set `DEFAULT_LIMITFREERELAY` = 0 kB/minute (MarcoFalke)
- #9239 `3fbf079` Disable fee estimates for 1-block target (morcos)
- #7562 `1eef038` Bump transaction version default to 2 (btcdrak)
- #9313,#9367 If we don't allow free txs, always send a fee filter (morcos)
- #9346 `b99a093` Batch construct batches (sipa)

- #9262 `5a70572` Prefer coins that have fewer ancestors, sanity check txn before ATMP (instagibbs)
- #9288 `1ce7ede` Fix a bug if the min fee is 0 for FeeFilterRounder (morcos)
- #9395 `0fc1c31` Add test for `-walletrejectlongchains` (morcos)
- #9107 `7dac1e5` Safer modify new coins (morcos)
- #9312 `a72f76c` Increase mempool expiry time to 2 weeks (morcos)
- #8610 `c252685` Share unused mempool memory with coincache (sipa)
- #9138 `f646275` Improve fee estimation (morcos)
- #9408 `46b249e` Allow shutdown during LoadMempool, dump only when necessary (jonasschnelli)
- #9310 `8c87f17` Assert FRESH validity in CCoinsViewCache::BatchWrite (ryanofsky)
- #7871 `e2e624d` Manual block file pruning (mrbandrews)
- #9507 `0595042` Fix use-after-free in CTxMemPool::removeConflicts() (sdaftuar)
- #9380 `dd98f04` Separate different uses of minimum fees (morcos)
- #9596 `71148b8` bugfix save feeDelta instead of priorityDelta in DumpMempool (morcos)
- #9371 `4a1dc35` Notify on removal (morcos)
- #9519 `9b4d267` Exclude RBF replacement txs from fee estimation (morcos)
- #8606 `e2a1a1e` Fix some locks (sipa)
- #8681 `6898213` Performance Regression Fix: Pre-Allocate txChanged vector (JeremyRubin)
- #8223 `744d265` c++11: Use std::unique_ptr for block creation (domob1812)
- #9125 `7490ae8` Make CBlock a vector of shared_ptr of CTransactions (sipa)
- #8930 `93566e0` Move orphan processing to ActivateBestChain (TheBlueMatt)
- #8580 `46904ee` Make CTransaction actually immutable (sipa)
- #9240 `a1dcf2e` Remove txConflicted (morcos)
- #8589 `e8cfe1e` Inline CTxInWitness inside CTxIn (sipa)
- #9349 `2db4cbc` Make CScript (and prevector) c++11 movable (sipa)
- #9252 `ce5c1f4` Release cs_main before calling ProcessNewBlock, or processing headers (cmpctblock handling) (sdaftuar)
- #9283 `869781c` A few more CTransactionRef optimizations (sipa)
- #9499 `9c9af5a` Use recent-rejects, orphans, and recently-replaced txn for compact-block-reconstruction (TheBlueMatt)
- #9813 `3972a8e` Read/write mempool.dat as a binary (paveljanik)

P2P protocol and network code

- #8128 `1030fa7` Turn net structures into dumb storage classes (theuni)
- #8282 `026c6ed` Feeler connections to increase online addrs in the tried table (EthanHeilman)
- #8462 `53f8f22` Move AdvertiseLocal debug output to net category (Mirobit)
- #8612 `84decb5` Check for compatibility with download in FindNextBlocksToDownload (sipa)
- #8594 `5b2ea29` Do not add random inbound peers to addrman (gmaxwell)
- #8085 `6423116` Begin encapsulation (theuni)
- #8715 `881d7ea` only delete CConnman if it's been created (theuni)
- #8707 `f07424a` Fix maxuploadtarget setting (theuni)
- #8661 `d2e4655` Do not set an addr time penalty when a peer advertises itself (gmaxwell)
- #8822 `9bc6a6b` Consistent checksum handling (laanwj)
- #8936 `1230890` Report NodeId in misbehaving debug (rebroad)
- #8968 `3cf496d` Don't hold cs_main when calling ProcessNewBlock from a cmpctblock (TheBlueMatt)
- #9002 `e1d1f57` Make connect=0 disable automatic outbound connections (gmaxwell)
- #9050 `fcf61b8` Make a few values immutable, and use deterministic randomness for the localnonce (theuni)

- #8969 `3665483` Decouple peer-processing-logic from block-connection-logic (#2) (TheBlueMatt)
- #8708 `c8c572f` have CConnman handle message sending (theuni)
- #8709 `1e50d22` Allow filterclear messages for enabling TX relay only (rebroad)
- #9045 `9f554e0` Hash P2P messages as they are received instead of at process-time (TheBlueMatt)
- #9026 `dc6b940` Fix handling of invalid compact blocks (sdaftuar)
- #8996 `ab914a6` Network activity toggle (luke-jr)
- #9131 `62af164` fNetworkActive is not protected by a lock, use an atomic (jonasschnelli)
- #8872 `0c577f2` Remove block-request logic from INV message processing (TheBlueMatt)
- #8690 `791b58d` Do not fully sort all nodes for addr relay (sipa)
- #9128 `76fec09` Decouple CConnman and message serialization (theuni)
- #9226 `3bf06e9` Remove fNetworkNode and pnodeLocalHost (gmaxwell)
- #9352 `a7f7651` Attempt reconstruction from all compact block announcements (sdaftuar)
- #9319 `a55716a` Break addnode out from the outbound connection limits (gmaxwell)
- #9261 `2742568` Add unstored orphans with rejected parents to recentRejects (morcos)
- #9441 `8b66bf7` Massive speedup. Net locks overhaul (theuni)
- #9375 `3908fc4` Relay compact block messages prior to full block connection (TheBlueMatt)
- #9400 `8a445c5` Set peers as HB peers upon full block validation (instagibbs)
- #9561 `6696b46` Wake message handling thread when we receive a new block (TheBlueMatt)
- #9535 `82274c0` Split CNode::cs_vSend: message processing and message sending (TheBlueMatt)
- #9606 `3f9f962` Consistently use GetTimeMicros() for inactivity checks (sdaftuar)
- #9594 `fd70211` Send final alert message to older peers after connecting (gmaxwell)
- #9626 `36966a1` Clean up a few CConnman cs_vNodes/CNode things (TheBlueMatt)
- #9609 `4966917` Fix remaining net assertions (theuni)
- #9671 `7821db3` Fix super-unlikely race introduced in 236618061a445d2cb11e72 (TheBlueMatt)
- #9730 `33f3b21` Remove bitseed.xf2.org form the dns seed list (jonasschnelli)
- #9698 `2447c10` Fix socket close race (theuni)
- #9708 `a06ede9` Clean up all known races/platform-specific UB at the time PR was opened (TheBlueMatt)
- #9715 `b08656e` Disconnect peers which we do not receive VERACKs from within 60 sec (TheBlueMatt)
- #9720 `e87ce95` Fix banning and disallow sending messages before receiving verack (theuni)
- #9268 `09c4fd1` Fix rounding privacy leak introduced in #9260 (TheBlueMatt)
- #9075 `9346f84` Decouple peer-processing-logic from block-connection-logic (#3) (TheBlueMatt)
- #8688 `047ded0` Move static global randomizer seeds into CConnman (sipa)
- #9289 `d9ae1ce` net: drop boost::thread_group (theuni)

Validation

- #9014 `d04aeba` Fix block-connection performance regression (TheBlueMatt)
- #9299 `d52ce89` Remove no longer needed check for premature v2 txs (morcos)
- #9273 `b68685a` Remove unused `CDiskBlockPos*` argument from ProcessNewBlock (TheBlueMatt)
- #8895 `b83264d` Better SigCache Implementation (JeremyRubin)
- #9490 `e126d0c` Replace FindLatestBefore used by importmulti with FindEarliestAtLeast (gmaxwell)
- #9484 `812714f` Introduce assumevalid setting to skip validation presumed valid scripts (gmaxwell)
- #9511 `7884956` Don't overwrite validation state with corruption check (morcos)
- #9765 `1e92e04` Harden against mistakes handling invalid blocks (sdaftuar)
- #9779 `3c02b95` Update nMinimumChainWork and defaultAssumeValid (gmaxwell)
- #8524 `19b0f33` Precompute sighashes (sipa)
- #9791 `1825a03` Avoid VLA in hash.h (sipa)

Build system

- #8238 `6caf3ee` ZeroMQ 4.1.5 && ZMQ on Windows (fanquake)
- #8520 `b40e19c` Remove check for `openssl/ec.h` (laanwj)
- #8617 `de07fdc` Include instructions to extract Mac OS X SDK on Linux using 7zip and SleuthKit (luke-jr)
- #8566 `7b98895` Easy to use gitian building script (achow101)
- #8604 `f256843` build,doc: Update for 0.13.0+ and OpenBSD 5.9 (laanwj)
- #8640 `2663e51` depends: Remove Qt46 package (fanquake)
- #8645 `8ea4440` Remove unused Qt 4.6 patch (droark)
- #8608 `7e9ab95` Install manpages via make install, also add some autogenerated manpages (nomnombtc)
- #8781 `ca69ef4` contrib: delete `qt_translations.py` (MarcoFalke)
- #8783 `64dc645` share: remove qt/protobuf.pri (MarcoFalke)
- #8423 `3166dff` depends: expat 2.2.0, ccache 3.3.1, fontconfig 2.12.1 (fanquake)
- #8791 `b694b0d` travis: cross-mac: explicitly enable gui (MarcoFalke)
- #8820 `dc64141` depends: Fix Qt compilation with Xcode 8 (fanquake)
- #8730 `489a6ab` depends: Add libevent compatibility patch for windows (laanwj)
- #8819 `c841816` depends: Boost 1.61.0 (fanquake)
- #8826 `f560d95` Do not include `env_win.cc` on non-Windows systems (paveljanik)
- #8948 `e077e00` Reorder Windows gitian build order to match Linux (Michagogo)
- #8568 `078900d` new var `DIST_CONTRIB` adds useful things for packagers from contrib (nomnombtc)
- #9114 `21e6c6b` depends: Set `OSX_MIN_VERSION` to 10.8 (fanquake)
- #9140 `018a4eb` Bugfix: Correctly replace generated headers and fail cleanly (luke-jr)
- #9156 `a8b2a82` Add compile and link options echo to configure (jonasschnelli)
- #9393 `03d85f6` Include cuckoocache header in Makefile (MarcoFalke)
- #9420 `bebe369` Fix linker error when configured with --enable-lcov (droark)
- #9412 `53442af` Fix 'make deploy' for OSX (jonasschnelli)
- #9475 `7014506` Let autoconf detect presence of `EVP_MD_CTX_new` (luke-jr)
- #9513 `bbf193f` Fix qt distdir builds (theuni)
- #9471 `ca615e6` depends: libevent 2.1.7rc (fanquake)
- #9468 `f9117f2` depends: Dependency updates for 0.14.0 (fanquake)
- #9469 `01c4576` depends: Qt 5.7.1 (fanquake)
- #9574 `5ac6687` depends: Fix QT build on OSX (fanquake)
- #9646 `720b579` depends: Fix cross build for qt5.7 (theuni)
- #9705 `6a55515` Add options to override BDB cflags/libs (laanwj)
- #8249 `4e1567a` Enable (and check for) 64-bit ASLR on Windows (laanwj)
- #9758 `476cc47` Selectively suppress deprecation warnings (jonasschnelli)
- #9783 `6d61a2b` release: bump gitian descriptors for a new 0.14 package cache (theuni)
- #9789 `749fe95` build: add --enable-werror and warn on vla's (theuni)
- #9831 `99fd85c` build: force a c++ standard to be specified (theuni)

GUI

- #8192 `c503863` Remove URLs from About dialog translations (fanquake)
- #8540 `36404ae` Fix random segfault when closing "Choose data directory" dialog (laanwj)
- #8517 `2468292` Show wallet HD state in statusbar (jonasschnelli)
- #8463 `62a5a8a` Remove Priority from coincontrol dialog (MarcoFalke)
- #7579 `0606f95` Show network/chain errors in the GUI (jonasschnelli)
- #8583 `c19f8a4` Show XTHIN in GUI (rebroad)

- #7783 `4335d5a` RPC-Console: support nested commands and simple value queries (jonasschnelli)
- #8672 `6052d50` Show transaction size in transaction details window (Cocosoft)
- #8777 `fec6af7` WalletModel: Expose disablewallet (MarcoFalke)
- #8371 `24f72e9` Add out-of-sync modal info layer (jonasschnelli)
- #8885 `b2fec4e` Fix ban from qt console (theuni)
- #8821 `bf8e68a` sync-overlay: Don't block during reindex (MarcoFalke)
- #8906 `088d1f4` sync-overlay: Don't show progress twice (MarcoFalke)
- #8918 `47ace42` Add "Copy URI" to payment request context menu (luke-jr)
- #8925 `f628d9a` Display minimum ping in debug window (rebroad)
- #8774 `3e942a7` Qt refactors to better abstract wallet access (luke-jr)
- #8985 `7b1bfa3` Use pindexBestHeader instead of setBlockIndexCandidates for NotifyHeaderTip() (jonasschnelli)
- #8989 `d2143dc` Overhaul smart-fee slider, adjust default confirmation target (jonasschnelli)
- #9043 `273bde3` Return useful error message on ATMP failure (MarcoFalke)
- #9088 `4e57824` Reduce ambiguity of warning message (rebroad)
- #8874 `e984730` Multiple Selection for peer and ban tables (achow101)
- #9145 `924745d` Make network disabled icon 50% opaque (MarcoFalke)
- #9130 `ac489b2` Mention the new network toggle functionality in the tooltip (paveljanik)
- #9218 `4d955fc` Show progress overlay when clicking spinner icon (laanwj)
- #9280 `e15660c` Show ModalOverlay by pressing the progress bar, allow hiding (jonasschnelli)
- #9296 `fde7d99` Fix missed change to WalletTx structure (morcos)
- #9266 `2044e37` Bugfix: Qt/RPCConsole: Put column enum in the right places (luke-jr)
- #9255 `9851a84` layoutAboutToChange signal is called layoutAboutToBeChanged (laanwj)
- #9330 `47e6a19` Console: add security warning (jonasschnelli)
- #9329 `db45ad8` Console: allow empty arguments (jonasschnelli)
- #8877 `6dc4c43` Qt RPC console: history sensitive-data filter, and saving input line when browsing history (luke-jr)
- #9462 `649cf5f` Do not translate tilde character (MarcoFalke)
- #9457 `123ea73` Select more files for translation (MarcoFalke)
- #9413 `fd7d8c7` CoinControl: Allow non-wallet owned change addresses (jonasschnelli)
- #9461 `b250686` Improve progress display during headers-sync and peer-finding (jonasschnelli)
- #9588 `5086452` Use nPowTargetSpacing constant (MarcoFalke)
- #9637 `d9e4d1d` Fix transaction details output-index to reflect vout index (jonasschnelli)
- #9718 `36f9d3a` Qt/Intro: Various fixes (luke-jr)
- #9735 `ec66d06` devtools: Handle Qt formatting characters edge-case in update-translations.py (laanwj)
- #9755 `a441db0` Bugfix: Qt/Options: Restore persistent "restart required" notice (luke-jr)
- #9817 `7d75a5a` Fix segfault crash when shutdown the GUI in disablewallet mode (jonasschnelli)

Wallet

- #8152 `b9c1cd8` Remove `CWalletDB*` parameter from CWallet::AddToWallet (pstratem)
- #8432 `c7e05b3` Make CWallet::fFileBacked private (pstratem)
- #8445 `f916700` Move CWallet::setKeyPool to private section of CWallet (pstratem)
- #8564 `0168019` Remove unused code/conditions in ReadAtCursor (jonasschnelli)
- #8601 `37ac678` Add option to opt into full-RBF when sending funds (rebase, original by petertodd) (laanwj)
- #8494 `a5b20ed` init, wallet: ParameterInteraction() iff wallet enabled (MarcoFalke)
- #8760 `02ac669` init: Get rid of some `ENABLE_WALLET` (MarcoFalke)

- #8696 `a1f8d3e` Wallet: Remove last external reference to CWalletDB (pstratem)
- #8768 `886e8c9` init: Get rid of fDisableWallet (MarcoFalke)
- #8486 `ab0b411` Add high transaction fee warnings (MarcoFalke)
- #8851 `940748b` Move key derivation logic from GenerateNewKey to DeriveNewChildKey (pstratem)
- #8287 `e10af96` Set fLimitFree = true (MarcoFalke)
- #8928 `c587577` Fix init segfault where InitLoadWallet() calls ATMP before genesis (TheBlueMatt)
- #7551 `f2d7056` Add importmulti RPC call (pedrobranco)
- #9016 `0dcb888` Return useful error message on ATMP failure (instagibbs)
- #8753 `f8723d2` Locked memory manager (laanwj)
- #8828 `a4fd8df` Move CWalletDB::ReorderTransactions to CWallet (pstratem)
- #8977 `6a1343f` Refactor wallet/init interaction (Reaccept wtx, flush thread) (jonasschnelli)
- #9036 `ed0cc50` Change default confirm target from 2 to 6 (laanwj)
- #9071 `d1871da` Declare wallet.h functions inline (sipa)
- #9132 `f54e460` Make strWalletFile const (jonasschnelli)
- #9141 `5ea5e04` Remove unnecessary calls to CheckFinalTx (jonasschnelli)
- #9165 `c01f16a` SendMoney: use already-calculated balance (instagibbs)
- #9311 `a336d13` Flush wallet after abandontransaction (morcos)
- #8717 `38e4887` Addition of ImmatureCreditCached to MarkDirty() (spencerlievens)
- #9446 `510c0d9` SetMerkleBranch: remove unused code, remove cs_main lock requirement (jonasschnelli)
- #8776 `2a524b8` Wallet refactoring leading up to multiwallet (luke-jr)
- #9465 `a7d55c9` Do not perform ECDSA signing in the fee calculation inner loop (gmaxwell)
- #9404 `12e3112` Smarter coordination of change and fee in CreateTransaction (morcos)
- #9377 `fb75cd0` fundrawtransaction: Keep change-output keys by default, make it optional (jonasschnelli)
- #9578 `923dc44` Add missing mempool lock for CalculateMemPoolAncestors (TheBlueMatt)
- #9227 `02464da` Make nWalletDBUpdated atomic to avoid a potential race (pstratem)
- #9764 `f8af89a` Prevent "overrides a member function but is not marked 'override'" warnings (laanwj)
- #9771 `e43a585` Add missing cs_wallet lock that triggers new lock held assertion (ryanofsky)
- #9316 `3097ea4` Disable free transactions when relay is disabled (MarcoFalke)
- #9615 `d2c9e4d` Wallet incremental fee (morcos)
- #9760 `40c754c` Remove importmulti always-true check (ryanofsky)

Tests and QA

- #8270 `6e5e5ab` Tests: Use portable #! in python scripts (/usr/bin/env) (ChoHag)
- #8534,#8504 Remove java comparison tool (laanwj,MarcoFalke)
- #8482 `740cff5` Use single cache dir for chains (MarcoFalke)
- #8450 `21857d2` Replace `rpc_wallet_tests.cpp` with python RPC unit tests (pstratem)
- #8671 `ddc3080` Minimal fix to slow prevector tests as stopgap measure (JeremyRubin)
- #8680 `666eaf0` Address Travis spurious failures (theuni)
- #8789 `e31a43c` pull-tester: Only print output when failed (MarcoFalke)
- #8810 `14e8f99` tests: Add exception error message for JSONRPCException (laanwj)
- #8830 `ef0801b` test: Add option to run bitcoin-util-test.py manually (jnewbery)
- #8881 `e66cc1d` Add some verbose logging to bitcoin-util-test.py (jnewbery)
- #8922 `0329511` Send segwit-encoded blocktxn messages in p2p-compactblocks (TheBlueMatt)
- #8873 `74dc388` Add microbenchmarks to profile more code paths (ryanofsky)
- #9032 `6a8be7b` test: Add format-dependent comparison to bctest (laanwj)
- #9023 `774db92` Add logging to bitcoin-util-test.py (jnewbery)
- #9065 `c9bdf9a` Merge `doc/unit-tests.md` into `src/test/README.md` (laanwj)

- #9069 `ed64bce` Clean up bctest.py and bitcoin-util-test.py (jnewbery)
- #9095 `b8f43e3` test: Fix test_random includes (MarcoFalke)
- #8894 `faec09b` Testing: Include fRelay in mininode version messages (jnewbery)
- #9097 `e536499` Rework `sync_*` and preciousblock.py (MarcoFalke)
- #9049 `71bc39e` Remove duplicatable duplicate-input check from CheckTransaction (TheBlueMatt)
- #9136 `b422913` sync_blocks cleanup (ryanofsky)
- #9151 `4333b1c` proxy_test: Calculate hardcoded port numbers (MarcoFalke)
- #9206 `e662d28` Make test constant consistent with consensus.h (btcdrak)
- #9139 `0de7fd3` Change sync_blocks to pick smarter maxheight (on top of #9196) (ryanofsky)
- #9100 `97ec6e5` tx_valid: re-order inputs to how they are encoded (dcousens)
- #9202 `e56cf67` bench: Add support for measuring CPU cycles (laanwj)
- #9223 `5412c08` unification of Bloom filter representation (s-matthew-english)
- #9257 `d7ba4a2` Dump debug logs on travis failures (sdaftuar)
- #9221 `9e4bb31` Get rid of duplicate code (MarcoFalke)
- #9274 `919db03` Use cached utxo set to fix performance regression (MarcoFalke)
- #9276 `ea33f19` Some minor testing cleanups (morcos)
- #9291 `8601784` Remove mapOrphanTransactionsByPrev from DoS_tests (sipa)
- #9309 `76fcd9d` Wallet needs to stay unlocked for whole test (morcos)
- #9172 `5bc209c` Resurrect pstratem's "Simple fuzzing framework" (laanwj)
- #9331 `c6fd923` Add test for rescan feature of wallet key import RPCs (ryanofsky)
- #9354 `b416095` Make fuzzer actually test CTxOutCompressor (sipa)
- #9390,#9416 travis: make distdir (MarcoFalke)
- #9308 `0698639` test: Add CCoinsViewCache Access/Modify/Write tests (ryanofsky)
- #9406 `0f921e6` Re-enable a blank v1 Tx JSON test (droark)
- #9435 `dbc8a8c` Removed unused variable in test, fixing warning (ryanofsky)
- #9436 `dce853e` test: Include tx data in `EXTRA_DIST` (MarcoFalke)
- #9525 `02e5308` test: Include tx data in `EXTRA_DIST` (MarcoFalke)
- #9498 `054d664` Basic CCheckQueue Benchmarks (JeremyRubin)
- #9554 `0b96abc` test: Avoid potential NULL pointer dereference in `addrman_tests.cpp` (practicalswift)
- #9628 `f895023` Increase a sync_blocks timeout in pruning.py (sdaftuar)
- #9638 `a7ea2f8` Actually test assertions in pruning.py (MarcoFalke)
- #9647 `e99f0d7` Skip RAII event tests if libevent is built without `event_set_mem_functions` (luke-jr)
- #9691 `fc67cd2` Init ECC context for `test_bitcoin_fuzzy` (gmaxwell)
- #9712 `d304fef` bench: Fix initialization order in registration (laanwj)
- #9707 `b860915` Fix RPC failure testing (jnewbery)
- #9269 `43e8150` Align struct COrphan definition (sipa)
- #9820 `599c69a` Fix pruning test broken by 2 hour manual prune window (ryanofsky)
- #9824 `260c71c` qa: Check return code when stopping nodes (MarcoFalke)
- #9875 `50953c2` tests: Fix dangling pwalletMain pointer in wallet tests (laanwj)
- #9839 `eddaa6b` [qa] Make import-rescan.py watchonly check reliable (ryanofsky)

Documentation

- #8332 `806b9e7` Clarify witness branches in transaction.h serialization (dcousens)
- #8935 `0306978` Documentation: Building on Windows with WSL (pooleja)
- #9144 `c98f6b3` Correct waitforblockheight example help text (fanquake)
- #9407 `041331e` Added missing colons in when running help command (anditto)
- #9378 `870cd2b` Add documentation for CWalletTx::fFromMe member (ryanofsky)

- #9297 `0b73807` Various RPC help outputs updated (Mirobit)
- #9613 `07421cf` Clarify getbalance help string to explain interaction with bumpfee (ryanofsky)
- #9663 `e30d928` Clarify listunspent amount description (instagibbs)
- #9396 `d65a13b` Updated listsinceblock rpc documentation (accraze)
- #8747 `ce43630` rpc: Fix transaction size comments and RPC help text (jnewbery)
- #8058 `bbd9740` Doc: Add issue template (AmirAbrams)
- #8567 `85d4e21` Add default port numbers to REST doc (djpnewton)
- #8624 `89de153` build: Mention curl (MarcoFalke)
- #8786 `9da7366` Mandatory copyright agreement (achow101)
- #8823 `7b05af6` Add privacy recommendation when running hidden service (laanwj)
- #9433 `caa2f10` Update the Windows build notes (droark)
- #8879 `f928050` Rework docs (MarcoFalke)
- #8887 `61d191f` Improve GitHub issue template (fanquake)
- #8787 `279bbad` Add missing autogen to example builds (AmirAbrams)
- #8892 `d270c30` Add build instructions for FreeBSD (laanwj)
- #8890 `c71a654` Update Doxygen configuration file (fanquake)
- #9207 `fa1f944` Move comments above bash command in build-unix (AmirAbrams)
- #9219 `c4522e7` Improve windows build instructions using Linux subsystem (laanwj)
- #8954 `932d02a` contrib: Add README for pgp keys (MarcoFalke)
- #9093 `2fae5b9` release-process: Mention GitHub release and archived release notes (MarcoFalke)
- #8743 `bae178f` Remove old manpages from contrib/debian in favour of doc/man (fanquake)
- #9550 `4105cb6` Trim down the XP notice and say more about what we support (gmaxwell)
- #9246 `9851498` Developer docs about existing subtrees (gmaxwell)
- #9401 `c2ea1e6` Make rpcauth help message clearer, add example in example .conf (instagibbs)
- #9022,#9033 Document dropping OS X 10.7 support (fanquake, MarcoFalke)
- #8771 `bc9e3ab` contributing: Mention not to open several pulls (luke-jr)
- #8852 `7b784cc` Mention Gitian building script in doc (Laudaa) (laanwj)
- #8915 `03dd707` Add copyright/patent issues to possible NACK reasons (petertodd)
- #8965 `23e03f8` Mention that PPA doesn't support Debian (anduck)
- #9115 `bfc7aad` Mention reporting security issues responsibly (paveljanik)
- #9840 `08e0690` Update sendfrom RPC help to correct coin selection misconception (ryanofsky)
- #9865 `289204f` Change bitcoin address in RPC help message (marijnfs)

Miscellaneous

- #8274 `7a2d402` util: Update tinyformat (laanwj)
- #8291 `5cac8b1` util: CopyrightHolders: Check for untranslated substitution (MarcoFalke)
- #8557 `44691f3` contrib: Rework verifybinaries (MarcoFalke)
- #8621 `e8ed6eb` contrib: python: Don't use shell=True (MarcoFalke)
- #8813 `fb24d7e` bitcoind: Daemonize using daemon(3) (laanwj)
- #9004 `67728a3` Clarify `listenonion` (unsystemizer)
- #8674 `bae81b8` tools for analyzing, updating and adding copyright headers in source files (isle2983)
- #8976 `8c6218a` libconsensus: Add input validation of flags (laanwj)
- #9112 `46027e8` Avoid ugly exception in log on unknown inv type (laanwj)
- #8837 `2108911` Allow bitcoin-tx to parse partial transactions (jnewbery)
- #9204 `74ced54` Clarify CreateTransaction error messages (instagibbs)
- #9265 `31bcc66` bitcoin-cli: Make error message less confusing (laanwj)
- #9303 `72bf1b3` Update comments in ctaes (sipa)

- #9417 `c4b7d4f` Do not evaluate hidden LogPrint arguments (sipa)
- #9506 `593a00c` RFC: Improve style for if indentation (sipa)
- #8883 `d5d4ad8` Add all standard TXO types to bitcoin-tx (jnewbery)
- #9531 `23281a4` Release notes for estimation changes (morcos)
- #9486 `f62bc10` Make peer=%d log prints consistent (TheBlueMatt)
- #9552 `41cb05c` Add IPv6 support to qos.sh (jamesmacwhite)
- #9542 `e9e7993` Docs: Update CONTRIBUTING.md (jnewbery)
- #9649 `53ab12d` Remove unused clang format dev script (MarcoFalke)
- #9625 `77bd8c4` Increase minimum debug.log size to 10MB after shrink (morcos)
- #9070 `7b22e50` Lockedpool fixes (kazcw)
- #8779 `7008e28` contrib: Delete spendfrom (MarcoFalke)
- #9587,#8793,#9496,#8191,#8109,#8655,#8472,#8677,#8981,#9124 Avoid shadowing of variables (paveljanik)
- #9063 `f2a6e82` Use deprecated `MAP_ANON` if `MAP_ANONYMOUS` is not defined (paveljanik)
- #9060 `1107653` Fix bloom filter init to isEmpty = true (robmcl4)
- #8613 `613bda4` LevelDB 1.19 (sipa)
- #9225 `5488514` Fix some benign races (TheBlueMatt)
- #8736 `5fa7b07` base58: Improve DecodeBase58 performance (wjx)
- #9039 `e81df49` Various serialization simplifcations and optimizations (sipa)
- #9010 `a143b88` Split up AppInit2 into multiple phases, daemonize after datadir lock errors (laanwj)
- #9230 `c79e52a` Fix some benign races in timestamp logging (TheBlueMatt)
- #9183,#9260 Mrs Peacock in The Library with The Candlestick (killed main.{h,cpp}) (TheBlueMatt)
- #9236 `7f72568` Fix races for strMiscWarning and `fLargeWork*Found`, make QT runawayException use GetWarnings (gmaxwell)
- #9243 `7aa7004` Clean up mapArgs and mapMultiArgs Usage (TheBlueMatt)
- #9387 `cfe41d7` RAII of libevent stuff using unique ptrs with deleters (kallewoof)
- #9472 `fac0f30` Disentangle progress estimation from checkpoints and update it (sipa)
- #9512 `6012967` Fix various things -fsanitize complains about (sipa)
- #9373,#9580 Various linearization script issues (droark)
- #9674 `dd163f5` Lock debugging: Always enforce strict lock ordering (try or not) (TheBlueMatt)
- #8453,#9334 Update to latest libsecp256k1 (laanwj,sipa)
- #9656 `7c93952` Check verify-commits on pushes to master (TheBlueMatt)
- #9679 `a351162` Access WorkQueue::running only within the cs lock (TheBlueMatt)
- #9777 `8dee822` Handle unusual maxsigcachesize gracefully (jnewbery)
- #8863,#8807 univalue: Pull subtree (MarcoFalke)
- #9798 `e22c067` Fix Issue #9775 (Check returned value of fopen) (kirit93)
- #9856 `69832aa` Terminate immediately when allocation fails (theuni)

Credits

Thanks to everyone who directly contributed to this release:

- accraze
- adlawren
- Alex Morcos
- Alexey Vesnin
- Amir Abrams
- Anders Øyvind Urke-Sætre

- Anditto Heristyo
- Andrew Chow
- anduck
- Anthony Towns
- Brian Deery
- BtcDrak
- Chris Moore
- Chris Stewart
- Christian Barcenas
- Christian Decker
- Cory Fields
- crowning-
- CryptAxe
- CryptoVote
- Dagur Valberg Johannsson
- Daniel Cousens
- Daniel Kraft
- Derek Miller
- djpnewton
- Don Patterson
- Doug
- Douglas Roark
- Ethan Heilman
- fsb4000
- Gaurav Rana
- Geoffrey Tsui
- Greg Walker
- Gregory Maxwell
- Gregory Sanders
- Hampus Sjöberg
- isle2983
- Ivo van der Sangen
- James White
- Jameson Lopp
- Jeremy Rubin
- Jiaxing Wang
- jnewbery
- John Newbery
- Johnson Lau
- Jon Lund Steffensen
- Jonas Schnelli
- jonnynewbs
- Jorge Timón
- Justin Camarena
- Karl-Johan Alm
- Kaz Wesley
- kirit93
- Koki Takahashi
- Lauda
- leijurv

- lizhi
- Luke Dashjr
- maiiz
- MarcoFalke
- Marijn Stollenga
- Marty Jones
- Masahiko Hyuga
- Matt Corallo
- Matthew King
- matthias
- Micha
- Michael Ford
- Michael Rotarius
- Mitchell Cash
- mrbandrews
- mruddy
- Nicolas DORIER
- nomnombtc
- Patrick Strateman
- Pavel Janík
- Pedro Branco
- Peter Todd
- Pieter Wuille
- poole_party
- practicalswift
- R E Broadley
- randy-waterhouse
- Richard Kiss
- Robert McLaughlin
- rodasmith
- Russell Yanofsky
- S. Matthew English
- Sev
- Spencer Lievens
- Stanislas Marion
- Steven
- Suhas Daftuar
- Thomas Snider
- UdjinM6
- unsystemizer
- whythat
- Will Binns
- Wladimir J. van der Laan
- wodry
- Zak Wilcox

As well as everyone that helped translating on Transifex.