<table>
<thead>
<tr>
<th></th>
<th>Bazaar Release Notes</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>bzr 2.8b1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>bzr 2.7.1</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>bzr 2.7.0</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>bzr 2.6.0</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>bzr 2.5.2</td>
<td>13</td>
</tr>
<tr>
<td>6</td>
<td>bzr 2.4.3</td>
<td>31</td>
</tr>
<tr>
<td>7</td>
<td>bzr 2.4.2</td>
<td>32</td>
</tr>
<tr>
<td>8</td>
<td>bzr 2.4.1</td>
<td>33</td>
</tr>
<tr>
<td>9</td>
<td>bzr 2.4.0</td>
<td>34</td>
</tr>
<tr>
<td>10</td>
<td>bzr 2.4b5</td>
<td>35</td>
</tr>
<tr>
<td>11</td>
<td>bzr 2.4b4</td>
<td>37</td>
</tr>
<tr>
<td>12</td>
<td>bzr 2.4b3</td>
<td>39</td>
</tr>
<tr>
<td>13</td>
<td>bzr 2.4b2</td>
<td>41</td>
</tr>
<tr>
<td>14</td>
<td>bzr 2.4b1</td>
<td>44</td>
</tr>
<tr>
<td>15</td>
<td>bzr 2.3.5</td>
<td>49</td>
</tr>
<tr>
<td>16</td>
<td>bzr 2.3.4</td>
<td>49</td>
</tr>
<tr>
<td>17</td>
<td>bzr 2.3.3</td>
<td>50</td>
</tr>
<tr>
<td>18</td>
<td>bzr 2.3.2</td>
<td>51</td>
</tr>
<tr>
<td>19</td>
<td>bzr 2.3.1</td>
<td>52</td>
</tr>
<tr>
<td>Version</td>
<td>Release Notes</td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>---------------</td>
<td></td>
</tr>
<tr>
<td>24.1</td>
<td>bzr 1.4</td>
<td></td>
</tr>
<tr>
<td>24.2</td>
<td>bzr 1.4rc2</td>
<td></td>
</tr>
<tr>
<td>24.3</td>
<td>bzr 1.4rc1</td>
<td></td>
</tr>
<tr>
<td>25.1</td>
<td>bzr 1.3.1</td>
<td></td>
</tr>
<tr>
<td>25.2</td>
<td>bzr 1.3.1rc1</td>
<td></td>
</tr>
<tr>
<td>25.3</td>
<td>bzr 1.3</td>
<td></td>
</tr>
<tr>
<td>25.4</td>
<td>bzr 1.3rc1</td>
<td></td>
</tr>
<tr>
<td>26.1</td>
<td>bzr 1.2</td>
<td></td>
</tr>
<tr>
<td>26.2</td>
<td>bzr 1.2rc1</td>
<td></td>
</tr>
<tr>
<td>27.1</td>
<td>bzr 1.1</td>
<td></td>
</tr>
<tr>
<td>27.2</td>
<td>bzr 1.1rc1</td>
<td></td>
</tr>
<tr>
<td>28.1</td>
<td>bzr 1.0</td>
<td></td>
</tr>
<tr>
<td>28.2</td>
<td>bzr 1.0rc3</td>
<td></td>
</tr>
<tr>
<td>28.3</td>
<td>bzr 1.0rc2</td>
<td></td>
</tr>
<tr>
<td>28.4</td>
<td>bzr 1.0rc1</td>
<td></td>
</tr>
<tr>
<td>29.1</td>
<td>bzr 0.92</td>
<td></td>
</tr>
<tr>
<td>29.2</td>
<td>bzr 0.92rc1</td>
<td></td>
</tr>
<tr>
<td>30.1</td>
<td>bzr 0.91</td>
<td></td>
</tr>
<tr>
<td>30.2</td>
<td>bzr 0.91rc2</td>
<td></td>
</tr>
<tr>
<td>30.3</td>
<td>bzr 0.91rc1</td>
<td></td>
</tr>
<tr>
<td>31.1</td>
<td>bzr 0.90</td>
<td></td>
</tr>
<tr>
<td>31.2</td>
<td>bzr 0.90rc1</td>
<td></td>
</tr>
<tr>
<td>32.1</td>
<td>bzr 0.18</td>
<td></td>
</tr>
<tr>
<td>32.2</td>
<td>bzr 0.18rc1</td>
<td></td>
</tr>
<tr>
<td>33.1</td>
<td>bzr 0.17</td>
<td></td>
</tr>
<tr>
<td>33.2</td>
<td>bzr 0.17rc1</td>
<td></td>
</tr>
<tr>
<td>34.1</td>
<td>bzr 0.16</td>
<td></td>
</tr>
<tr>
<td>34.2</td>
<td>bzr 0.16rc2</td>
<td></td>
</tr>
<tr>
<td>34.3</td>
<td>bzr 0.16rc1</td>
<td></td>
</tr>
<tr>
<td>35.1</td>
<td>bzr 0.15</td>
<td></td>
</tr>
<tr>
<td>35.2</td>
<td>bzr 0.15rc3</td>
<td></td>
</tr>
<tr>
<td>35.3</td>
<td>bzr 0.15rc2</td>
<td></td>
</tr>
<tr>
<td>35.4</td>
<td>bzr 0.15rc1</td>
<td></td>
</tr>
</tbody>
</table>

iv
<table>
<thead>
<tr>
<th>Bazaar Release Notes</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>36.1 bzr 0.14</td>
<td>239</td>
</tr>
<tr>
<td>36.2 bzr 0.14rc1</td>
<td>239</td>
</tr>
<tr>
<td>37.1 bzr 0.13</td>
<td>243</td>
</tr>
<tr>
<td>37.2 bzr 0.13rc1</td>
<td>243</td>
</tr>
<tr>
<td>38.1 bzr 0.12</td>
<td>245</td>
</tr>
<tr>
<td>38.2 bzr 0.12rc1</td>
<td>245</td>
</tr>
<tr>
<td>39.1 bzr 0.11</td>
<td>249</td>
</tr>
<tr>
<td>39.2 bzr 0.11rc2</td>
<td>249</td>
</tr>
<tr>
<td>39.3 bzr 0.11rc1</td>
<td>249</td>
</tr>
<tr>
<td>40.1 bzr 0.10</td>
<td>253</td>
</tr>
<tr>
<td>41.1 bzr 0.9.0</td>
<td>255</td>
</tr>
<tr>
<td>42.1 bzr 0.8.2</td>
<td>259</td>
</tr>
<tr>
<td>42.2 bzr 0.8.1</td>
<td>259</td>
</tr>
<tr>
<td>42.3 bzr 0.8</td>
<td>260</td>
</tr>
<tr>
<td>43.1 bzr 0.7</td>
<td>265</td>
</tr>
<tr>
<td>44.1 bzr 0.6</td>
<td>269</td>
</tr>
<tr>
<td>45.1 bzr 0.1.1</td>
<td>273</td>
</tr>
<tr>
<td>45.2 bzr 0.1</td>
<td>273</td>
</tr>
<tr>
<td>45.3 bzr 0.0.9</td>
<td>275</td>
</tr>
<tr>
<td>45.4 bzr 0.0.8</td>
<td>275</td>
</tr>
<tr>
<td>45.5 bzr-0.0.7</td>
<td>276</td>
</tr>
<tr>
<td>45.6 bzr-0.0.6</td>
<td>277</td>
</tr>
<tr>
<td>45.7 bzr-0.0.5</td>
<td>278</td>
</tr>
<tr>
<td>45.8 bzr-0.0.4</td>
<td>280</td>
</tr>
<tr>
<td>45.9 bzr-0.0.3</td>
<td>281</td>
</tr>
<tr>
<td>45.10 bzr-0.0.2.1</td>
<td>282</td>
</tr>
<tr>
<td>45.11 bzr-0.0.2</td>
<td>282</td>
</tr>
<tr>
<td>45.12 bzr-0.0.1</td>
<td>282</td>
</tr>
<tr>
<td>45.13 bzr-0.0.0.69</td>
<td>283</td>
</tr>
</tbody>
</table>
BAZAAR RELEASE NOTES

1.1 bzr 2.8b1

2.8b1 NOT RELEASED YET

1.1.1 External Compatibility Breaks

1.1.2 New Features

1.1.3 Improvements

1.1.4 Bug Fixes

1.1.5 Documentation

1.1.6 API Changes

1.1.7 Internals

1.1.8 Testing

• POSIX mandates diff to exit with status 1 (was 2 previously), bzr didn’t rely on that feature but one of the tests did. (Vincent Ladeuil, #1622039)
2.1 bzs 2.7.1

2.7.1 NOT RELEASED YET

2.1.1 External Compatibility Breaks
None.

2.1.2 New Features
None.

2.1.3 Improvements
None.

2.1.4 Bug Fixes

• Complete monkey-patching of re.finditer or LazyRegexps leak. (Vincent Ladeuil, #1644003, #1657238)
• Cope with paramiko making argument to SFTPFile.prefetch() mandatory. (Jelmer Vernooij, #1579093)

2.1.5 Documentation

• Clarify pypi uploads for releases. (Vincent Ladeuil, #1542903)

2.1.6 API Changes
None.

2.1.7 Internals
None.
2.1.8 Changed Behaviour

None.

2.1.9 Testing

- Recent versions of pycurl started being stricter about the proxy url, using 'http+pycurl:' is not allowed anymore and should just be 'http:' (Vincent Ladeuil, #1645017).

2.2 bzr 2.7.0

2.7.0 2016-02-01

2.2.1 External Compatibility Breaks

None.

2.2.2 New Features

None.

2.2.3 Improvements

- bzrlib.patches.parse_patches can optionally return a list of ‘dirty’ patch headers (prefixed with ‘===’). (Kit Randel, #1400567)

2.2.4 Bug Fixes

- ‘acceptable_keys’ from ‘bazaar.conf’ is now properly handled. (Vincent Ladeuil, #1249732)
- Option names are now checked to be valid identifiers (including embedded dots or hyphens). Also ignore invalid references (i.e. using invalid option names) while expanding option values. (Vincent Ladeuil, #1235099)
- Fix pyrex version checking to be more robust. (Andrew Starr-Bochicchio, #1030521)
- Forbid more operations for ReadonlyTransportDecorator so no more write methods can be used my mistake. (Vincent Ladeuil, #150196)

2.2.5 API Changes

None.
2.2.6 Internals

• Make all transport put_bytes() raises TypeError instead of AssertionError or UnicodeEncodeError when given unicode strings rather than bytes. (Vincent Ladeuil, #106898)

• Use ssl.match_hostname from the python ssl module and stop carrying a specific version that has become obsolete. (Vincent Ladeuil, #1538480)

2.2.7 Changed Behaviour

• Also honor $XDG_CONFIG_HOME specification on Mac OS X platform. (Fabien Meghazi)

2.2.8 Testing

• Fix gpgme test failure starting on wily. (Vincent Ladeuil)

• Fix racy http tests (TestBadStatusServer is so simple, it exposes a race in python 2.7.9. This happens only when both the http server and client are run in the same process.). Only tests are affected. (Vincent Ladeuil, #1451448)

• Fix warnings on stderr caused by the atexit handler triggering for the wrong reason: the ‘config’ command should explicitly save the changes when modifying or removing an option and not rely on the atexit handler. (Vincent Ladeuil, #1331999)

• Handle (minor) incompatible change in python 2.7.6 leading to test failures. Only tests are affected. (Vincent Ladeuil, #1303879)

• Remove wrong assumption about how TCP server and client interact when run inside the same process. (Vincent Ladeuil, #1269886).

• Rename assertWarns in bt.test_config so it doesn’t clash with the assertWarns introduced in recent python (Vincent Ladeuil, #1514210)

• Restrict access to ‘.netrc’ in tests or recent python (2.7.5-8) will complain. (Vincent Ladeuil, #1233413)

• Skip windows-only tests that start failing with python 2.7.9, there is no way to fix them without testing on windows itself. (Vincent Ladeuil, #1451448)

• Take python 2.7.6 late (better than never) bugfix in ntpath.py into account. Only tests are affected (Vincent Ladeuil, #1303879).
3.1 bzr 2.6.0

Codename  Surprise
2.6.0  2013-07-27

This release marks the start of a new long-term-stable series. From here, we will only make bugfix releases on the 2.6 series (2.6.1, etc), while 2.7 will become our new development series.

This is a bugfix and polish release over the 2.5 series, with a large number of bugs fixed (~50 for the 2.6 series alone). All known fixed bugs are included here.

Users are encouraged to upgrade from the other stable series.

3.1.1 External Compatibility Breaks

- *bzr lp-propose --approve* now also adds an “Approve” vote to the merge proposal. (Jonathan Lange)

3.1.2 New Features

- *bzr switch --store* now stores uncommitted changes in the branch, and restores them when switching back to the branch. (Aaron Bentley)

- New option ‘--context’ for ‘bzr diff’ command, to configure the amount of context (i.e. showing lines that have not changed). Also available as the named parameter ‘context_lines’ to *bzrlib.diff.internal_diff()* (Paul Nixon)

- The ‘grep’ plugin is now shipped with bzr. (Jelmer Vernooij)

- The ‘ping’ plugin is now shipped with bzr. (Jelmer Vernooij)

3.1.3 Improvements

- *bzr help env-variables* now points users at *bzr help configuration* which has much more detailed information on the same stuff. (Robert Collins)

- *bzr lp-find-proposal* now only cares about the revision-id that is specified, not the branch you use. This was enabled by a new API call in Launchpad’s web service. (Aaron Bentley)

- Implement authentication.conf password obfuscation, the password_encoding option can now be set to base64. (Florian Dorn)
3.1.4 Bug Fixes

• Add a progress_bar configuration option defaulting to BZR_PROGRESS_BAR. This can be set in bazaar.conf or specified from the command line with -Oprogress_bar=text. (Vincent Ladeuil, #388275)

• Authentication.Config now always returns unicode user names and passwords. (Vincent Ladeuil, #514301)

• Fixed a bug where the entire contents of /etc/mailname is read in. We only want to read in the first line so that comments could be added and would be ignored. (Haw Loeung, #932515)

• Fixed a syntax error in doc/ja/user-guide/http_smart_server.txt that causes breakage with docutils 0.9.1. (Vincent Ladeuil, Jelmer Vernooij, #1066307)

• Support utf8 characters in platform names even without looking inside the box (Fedora’s Schrödinger’s Cat). (Toshio Kuratomi, Vincent Ladeuil, #1195783)

• Warn when --show-base is used for pull in a treeless branch instead of failing. It’s useless but harmless. (Vincent Ladeuil, #1022160)

• When a missing referenced chk root error is encountered, bzr now suggests running bzr reconcile --canonicalize-chks. (Jelmer Vernooij)

• When pushing a specific revision, create the new working tree at that revision. (#484516, Neil Martinsen-Burrell)

3.1.5 Documentation

• Clarify bzr config usages especially around other option references expansion. (Vincent Ladeuil, #996401)

3.1.6 API Changes

• None.

3.1.7 Internals

• Add __iter__ to http ResponseFile. (Jelmer Vernooij, #1184021)

• The launchpad plugin now requires API 1.6.0 or later. This version shipped with Ubuntu 9.10. (Aaron Bentley)

• Better align with upstream gzip.py in tuned_gzip.py. We may lose a bit of performance but that’s for knit and weave formats and already partly deprecated, better keep compatibility than failing fast ;)) (Vincent Ladeuil, #1116079)

3.1.8 Testing

• Fix test failures occurring only with LC_ALL=C, when prompting a user, stderr encoding should be preferred over terminal encoding if available. (Vincent Ladeuil, #1086209)
3.2  bzr 2.6b2

2.6b2  2012-09-10

This is the second beta for the 2.6 series, leading up to a 2.6.0 release in August 2012.
This release includes minor bug fixes.
This release includes all bugs fixed in previous series known at the time of this release.
Beta releases are suitable for everyday use but may cause some incompatibilities with plugins. Some plugins may need small updates to work with 2.6b2.

3.2.1 External Compatibility Breaks

None.

3.2.2 New Features

• New option --overwrite-tags for bzr pull and bzr push. (Jelmer Vernooij, #681792)

3.2.3 Improvements

• Colocated branches can now be addressed using the ‘co:NAME’ rather than the more complex ‘file://.,branch=NAME’. (Jelmer Vernooij, #833665)

3.2.4 Bug Fixes

• “bzr missing” now shows tag names when displaying revision information. (#559072, Neil Martinsen-Burrell)
• Fix branch.conf saving when unlocking the branch for BranchFormat4. (Vincent Ladeuil, #1020007)
• Implement ResponseFile.readline and ResponseFile.tell, fixing some clones over HTTP. (Jelmer Vernooij, #963769)
• Option values set on locked branches should be saved only when the branch is finally unlocked. (Vincent Ladeuil, #948339)

3.2.5 Documentation

• Document “bzr lp-propose”, “bzr register-branch” and the other Launchpad plugin commands in bzr(1). (Jelmer Vernooij, #843801, #163995)
• Force format registration to avoid generate_docs.py traceback when the registry is empty. (Vincent Ladeuil, #956860)
• Generate ENVIRONMENT section in bzr(1) from known environment variable list rather than hardcoding. (Jelmer Vernooij, #197618)
### 3.2.6 API Changes

- `register_filter_stack_map` and `lazy_register_filter_stack_map` are now deprecated. Instead, use `filter_stacks_registry.register` and `filter_stacks_registry.register_laz`. (Jelmer Vernooij)
- Remove deprecated `Branch.import_last_revision()`. (Jelmer Vernooij)
- Remove deprecated `RepositoryFormat.register_format()`. (Jelmer Vernooij)
- Remove deprecated `Repository.get_ancestry()`. (Jelmer Vernooij)
- Remove deprecated `Repository.iter_reverse_revision_history()`. (Jelmer Vernooij)
- The previously deprecated `bzrlib.annotate.annotate_file` function has been removed. (Jelmer Vernooij)

### 3.2.7 Internals

None.

### 3.2.8 Testing

- Fix test failures by removing a remaining reference to `features.sphinx` which isn’t needed anymore since we don’t test the texinfo sphinx builder anymore either. (Vincent Ladeuil)

### 3.3 bzr 2.6b1

**2.6b1** 2012-03-15

This is the first beta for the 2.6 series, leading up to a 2.6.0 release in August 2012.

This release includes ssl certificates verification from the urllib-based http implementation turned on by default, fixes some UI issues around colocated branches, documentation fixes and more.

This release includes all bugs fixed in previous series known at the time of this release.

Beta releases are suitable for everyday use but may cause some incompatibilities with plugins. Some plugins may need small updates to work with 2.6b1.

#### 3.3.1 External Compatibility Breaks

None.

#### 3.3.2 Improvements

- Access to HTTPS URLs now uses the urllib implementation by default. For the old pycurl-based implementation, specify `https+pycurl://` as the URL scheme when accessing a HTTPS location. (Jelmer Vernooij, #125055)
- Add short option alias `--N` for `--no-recurse`. (Jelmer Vernooij, #945904)
- Avoid ‘Invalid range access’ errors when whole files are retrieved with `transport.http.get()`. (Vincent Ladeuil, #924746)
• **bzr rmbranch** now supports removing colocated branches. (Jelmer Vernooij, #920653)

• **bzr rmbranch** no longer removes active branches unless --force is specified. (Jelmer Vernooij, #922953)

• **bzr verify-signatures** now shows a progress bar. (Jelmer Vernooij)

• Two new command hooks, pre_command and post_command, provide notification before and after a command has been run. (Brian de Alwis, Jelmer Vernooij)

### 3.3.3 Bug Fixes

• Fix **bzr config** display for RegistryOption values. (Vincent Ladeuil, #930182)

### 3.3.4 Documentation

• Prevent lines of command descriptions starting with a dot to accidentally be interpreted as a roff macro in bzr(1). (Jelmer Vernooij, #711079)

• Properly format apostrophes in manual page. (Jelmer Vernooij, #234771)

### 3.3.5 API Changes

• **GPGStrategy.do_verifications** has been deprecated. (Jelmer Vernooij)

• File ids in the Tree API can now be bytestring as previously, or tuples of bytestrings. (Jelmer Vernooij)

• **mail_client** now accepts a configuration stack object rather than an old style Config object. (Jelmer Vernooij)

• New method **Repository.verify_revision_signatures**. (Jelmer Vernooij)

• New configuration option class **RegistryOption** which is backed onto a registry. (Jelmer Vernooij)

• New convenience API method **WorkingTree.get_config_stack**. (Jelmer Vernooij)

• Remove **branch.PullResult.__int__** deprecated in 2.3.0, **branch.PushResult.__int__** deprecated in 2.3.0, **branch.BranchFormat.get_default_format** deprecated in 2.4.0, **branch.BranchFormat.get_formats** deprecated in 2.4.0, **branch.BranchFormat.set_default_format** deprecated in 2.4.0, **branch.BranchFormat.register_format** deprecated in 2.4.0, **branch.BranchFormat.unregister_format** deprecated in 2.4.0, **bzrdir.BzrProber.register_bzrdir_format** deprecated in 2.4.0, **bzrdir.BzrProber.unregister_bzrdir_format** deprecated in 2.4.0, **config.Config.get_editor** deprecated in 2.4.0, **hooks.Hooks.create_hook** deprecated in 2.4.0, **inventory.Inventory.__contains__** deprecated in 2.4.0, **merge.Merge3Merger.scalar_three_way** deprecated in 2.2.0, **merge.Merge3Merger.fix_root** deprecated in 2.4.0, **transform.TreeTransformBase.has_named_child** deprecated in 2.3.0, **transform.get_backup_name** deprecated in 2.3.0, **worktree.WorkingTreeFormat.get_default_format** deprecated in 2.4.0, **worktree.WorkingTreeFormat.register_format** deprecated in 2.4.0, **worktree.WorkingTreeFormat.register_extra_format** deprecated in 2.4.0, **worktree.WorkingTreeFormat.unregister_extra_format** deprecated in 2.4.0, **worktree.WorkingTreeFormat.get_formats** deprecated in 2.4.0.
2.4.0, workingtree.WorkingTreeFormat.set_default_format deprecated in 2.4.0, workingtree.WorkingTreeFormat.unregister_format deprecated in 2.4.0, (Vincent Ladeuil)

- Remove deprecated Branch.set_revision_history and Branch.revision_history methods and the set_rh hook on Branch. (Jelmer Vernooij)

### 3.3.6 Internals

- Tree.path2id now once again accepts a list of path elements in addition to a path. (Jelmer Vernooij)

- Turn config option expansion on by default. The only options for which this should be disabled are templates which should already have used conf.get(option, expand=False) or conf.get_user_option(option, expand=False). (Vincent Ladeuil)
CHAPTER
FOUR

BAZAAR RELEASE NOTES

4.1 bzr 2.5.2

2.5.2 NOT RELEASED YET

4.1.1 External Compatibility Breaks

4.1.2 New Features

4.1.3 Improvements

4.1.4 Bug Fixes

  • `bzr config` properly handles aliases and references in the `--directory` parameter (Vincent Ladeuil, Wouter van Heyst, #947049)

  • Empty arguments in EDITOR are now properly preserved. (Ross Lagerwall, #1089792)

  • Fix a traceback when trying to checkout a tree that also has an entry with file-id `TREE_ROOT` somewhere other than at the root directory. (John Arbash Meinel, #830947)

  • Lightweight checkouts of remote repositories had a bug with how they extracted texts from the repository. (Just an ordering constraint on how they consumed the stream.) (John Arbash Meinel, #1046284)

  • `osutils.send_all` now detects if we get a series of zero bytes sent, and fails with an ECONNRESET. It seems if paramiko gets disconnected, it will get into a state where it returns 0 bytes sent, but doesn’t raise an error. This change allows us to get a couple hiccups of no content sent, but if it is consistent, we will consider it to be a failure. (John Arbash Meinel, #1047309)

  • Revert use of --no-tty when gpg signing commits. (Jelmer Vernooij, #1014570)

  • Some filesystems give `EOPNOTSUPP` when trying to call `fdatasync`. This shouldn’t be treated as a fatal error. (John Arbash Meinel, #1075108)

  • Some small bug fixes wrt lightweight checkouts and remote repositories. A test permutation was added that runs all working tree tests against a lightweight checkout. (John Arbash Meinel, #1046697)
4.2 bzr 2.5.1

2.5.1 2012-05-22

This is a bugfix release. Most of the bugs dealt with https and colocated branches glitches. Upgrading is recommended for all users of earlier 2.5 releases.

4.2.1 External Compatibility Breaks

None.

4.2.2 New Features

None.

4.2.3 Improvements

- `bzr rmbranch` now supports removing colocated branches. (Jelmer Vernooij, #920653)
- `bzr rmbranch` no longer removes active branches unless `--force` is specified. (Jelmer Vernooij, #922953)

4.2.4 Bug Fixes

- Connecting with HTTPS via HTTP now correctly uses the host name of the destination rather than the proxy when checking certificates. (Martin Packman, #944696)
- Fixed merge tool availability checking and invocation to search the Windows App Path registry in addition to the PATH. (Gordon Tyler, #939605)
- Fixed problem with getting errors about failing to open /dev/tty when using Bazaar Explorer to sign commits. (Mark Grandi, #847388)
- Fix UnicodeEncodeError when translated progress task messages contain non-ascii text. (Martin Packman, #966934)
- Make sure configuration options can provide their own help topic. (Jelmer Vernooij, #941672)

4.2.5 Documentation

- The alpha-quality texinfo sphinx builder has been deprecated. Sphinx >= 1.1.2 now provides a better one. Most of the documentation can now be generated to the texinfo format with `make texinfo-sphinx`. This will generate both the .texi files and the .info ones. (Vincent Ladeuil, #940164)
4.2.6 API Changes
None.

4.2.7 Testing
• Add support for pyftpdlib >= 0.7.0 and drop support for previous pyftpdlib versions. (Vincent Ladeuil, #956027)
• Run smoketest for setup.py isolated in a tempdir. (Martin Packman, #140874)

4.3 bzr 2.5.0

Codename  Phillip
2.5.0 2012-02-24
This release marks the start of a new long-term-stable series. From here, we will only make bugfix releases on the 2.5 series (2.5.1, etc, and support it until April 2017), while 2.6 will become our new development series.

This is a bugfix and polish release over the 2.4 series, with a large number of bugs fixed (~170 for the 2.5 series alone). The 2.5 series provides a faster smart protocol implementation for many operations, basic support for colocated branches. We have started translating bzr with the 2.5 series: https://translations.launchpad.net/bzr, more than 20 languages have already been registered but these are the early days, contributions welcome.

Only a few bugfixes have been included since 2.5b6 so all known fixed bugs are included here.

Users are encouraged to upgrade from the other stable series.

4.3.1 External Compatibility Breaks
None.

4.3.2 New Features
None.

4.3.3 Improvements
• The names of colocated branches are used as branch nicks if no nick is specified. (Aaron Bentley)

4.3.4 Bug Fixes
• Show locks in `bzr info` on control directories without a repository. (Jelmer Vernooij, #936767)
• Disable ssl certificate verification on osx and windows until a native access to the the root certificates is provided there. (Vincent Ladeuil, #929179)

4.3.5 Testing
• Stop depending on the particular CPython ordering of dictionary keys when testing the result of BzrDir.get_branches. (Wouter van Heyst)
4.4  bzr 2.5b6

2.5b6  2012-02-02

This is the sixth (and last (really)) beta of the 2.5 series, leading to a 2.5.0 release in March 2012. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This introduces the support for colocated branches into the ‘2a’ format in a backward compatible way, fix more glitches in the colocated UI, verify https certificates for the urllib https client implementation, fix some more unicode issues and more.

All bugs fixed in previous series known at the time of this release are included.

4.4.1 External Compatibility Breaks

None.

4.4.2 New Features

• Support for colocated branches is now available in the default format (“2a”). (Jelmer Vernooij)

4.4.3 Improvements

• `bzr switch -b` in a standalone tree will now create a colocated branch. (Jelmer Vernooij, #918197)
• `bzr info` now reports when there are present (but unused) colocated branches. (Jelmer Vernooij, #891646)
• Checkouts can now be into target directories that already have a control directory (but no branch or working tree). (Jelmer Vernooij, #913980)
• Colocated branches can now have names including forward slashes, to allow for namespaces. (Jelmer Vernooij, #907980)
• New HPSS call for `BzrDir.get_branches`. (Jelmer Vernooij, #894460)
• Checkouts of colocated branches are now always lightweight. (Jelmer Vernooij, #918828)

4.4.4 Bug Fixes

• `bzr branch` now fetches revisions when branching into an empty control directory. (Jelmer Vernooij, #905594)
• A sane default is provided for `ssl.ca_certs` which should points to the Certificate Authority bundle for supported platforms. (Vincent Ladeuil, #920455)
• `bzr branch` generates correct target branch locations again if not specified. (Jelmer Vernooij, #919218)
• `bzr send` works on treeless branches again. (Jelmer Vernooij, #921591)
• `bzr version` no longer throws a UnicodeDecodeError if the .bzr.log path contains non-ascii characters. (Martin Packman, #312841)
• Support scripts that don’t call bzrlib.initialize() but still call run_bzr(). (Vincent Ladeuil, #917733)
• Test for equality instead of object identity where ROOT_PARENT is concerned. (Wouter van Heyst, #881142)
• urllib-based HTTPS client connections now verify the server certificate validity as well as the hostname. (Jelmer Vernooij, Vincent Ladeuil, #651161)

4.4.5 API Changes

• config.config_dir and related functions now always return paths as unicode. (Martin Packman, #825826)
• ControlDir now has a new method set_branch_reference which can be used for setting branch references. (Jelmer Vernooij)
• ControlDir.destroy_branch now raises NotBranchError rather than NoSuchFile if the branch didn’t exist. (Jelmer Vernooij, #921693)

4.4.6 Internals

• A new matcher RevisionHistoryMatches has been added. (Jelmer Vernooij)
• Add new module bzrlib.url_policy_open. (Jelmer Vernooij, #850843)
• MutableTree has two new hooks pre_transform and post_transform that are called for tree transform operations. (Jelmer Vernooij, #912084)

4.4.7 Testing

• Be more careful about closing open files for pypy interoperability. (Wouter van Heyst)

4.5 bzr 2.5b5

2.5b5 2012-01-12

This is the fifth (and last) beta of the 2.5 series, leading to a 2.5.0 release in February 2012. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This release includes many improvements in the smart server, UI polish for the colocated branches, enhancements to the config framework and more internal uses, bug fixes related to unicode and locale support and more.

All bug fixed in previous series known at the time of this release are included.

4.5.1 External Compatibility Breaks

• The ‘.bzr/branch/email’ file is no longer read to determine the users’ identity. Instead, the ‘email’ setting in ‘.bzr/branch/branch.conf’ should be used. (Jelmer Vernooij, #903894)

4.5.2 New Features

• “bzr mkdir” now includes -p (--parents) option for recursively adding parent directories. (Jared Hance, Jelmer Vernooij, #253529)
• config.Option can now declare override_from_env, a list of environment variables which, when set, that takes precedence over values defined in configuration files. (Vincent Ladeuil, #907279)
4.5.3 Improvements

- New HPSS call for `Repository.reconcile`. (Jelmer Vernooij, #894455)
- Merge now has two new hooks `pre_merge` and `post_merge` that are called before and after a merge and can make additional modifications to the trees involved. (Jelmer Vernooij, #906877)
- Override the value returned by `sys.getfilesystemencoding()` for the `bzr` script to utf-8 when it would otherwise be ascii on a posix system. This will mean `bzr` works with non-ascii files when no locale or an incorrect locale is set. (Martin Packman, #794353)
- `bzr branches` now indicates the active colocated branch. (Jelmer Vernooij, #891667)
- `bzr push` now suggests using :parent if there is a parent location set. (Jelmer Vernooij)
- `bzr send` now only opens a single connection, rather than two, to the target branch. (Jelmer Vernooij)

4.5.4 Bug Fixes

- Allow configuration option default value to be a python callable at registration. (Vincent Ladeuil, #832064)
- `bzr config` will now display the section `[DEFAULT]` used in `bazaar.conf`. (Vincent Ladeuil, #907268)
- Configuration stores can now provides a specific quoting mechanism. This is required to workaround `configobj` conflating quoting and list values automatic conversion. (Vincent Ladeuil, #906897)
- Create obsolete_packs directory when repacking if it does not exist. (Jonathan Riddell, Jelmer Vernooij, #314314)
- Fallback to the slower `bzr log` implementation when displaying a range of revisions whose ancestry is not obviously on the same development line. (Vincent Ladeuil, #904744)
- Make lazy imports resilient when resolved concurrently from multiple threads. Now the stand-in object will behave as a proxy for the real object after the initial access, rather than throwing. Assigning the object to multiple names should still be avoided. (Martin von Gagern, #396819)
- Not setting `gpg_signing_key` or setting it to `default` will use the user email (obtained from the `email` configuration option or its default value). (Vincent Ladeuil, Jelmer Vernooij, #904550)
- Prevent spurious `InconsistentDelta` error when committing a move of a non-ascii directory with contents. (Rory Yorke, #185211)
- Properly ignore ‘n’ in an option reference since this cannot be part of a config option identifier. (Vincent Ladeuil, #902125)
- Make sure that the `bzr` probers are always registered when `bzrlib.workingtree` is imported. (Jelmer Vernooij, #905218)
- Report mistake trying to move a removed file with a non-ascii name without `UnicodeEncodeError` being raised. (Martin Packman, #898541)
- Safely unquote configuration values in weird edge cases (a section seen as a dictionary which is not a supported use case for the configuration stacks). (Vincent Ladeuil, #908050)
- Stop altering `sys.platform` on OSX when initialising the locale. (Martin Packman, #570495)
- Uncommit no longer removes tags if they are part of the working trees pending merges. (Jelmer Vernooij, #905462)
4.5.5 API Changes

- `Config.signature_needed`, `Config.signing_policy`, `Config.gpg_signing_key`, `Config.gpg_signing_command`, `Config.checking_policy` and `Config.post_commit` are now deprecated. (Jelmer Vernooij)

- `Repository.get_commit_builder` now takes a `config_stack` rather than a `config` argument. (Jelmer Vernooij)

- Scripts using bzrlib should now ensure `setlocale` is called on posix platforms if they need a non-ascii user encoding. (Martin Packman)

- Send formats now accept a new optional argument `submit_branch`, which can be `None` or a `Branch` object for the submit branch location. (Jelmer Vernooij)

- `VersionedFileRepository.add_revision` no longer takes a `config` argument. (Jelmer Vernooij)

4.5.6 Internals

- Add HPSS call for `Branch.get_checkout_format`. (Jelmer Vernooij, #894459)

- Add HPSS call for `Repository.pack`. (Jelmer Vernooij, #894461)

- Add HPSS calls for `Repository.iter_files_bytes`, speeding up several commands including `bzc export` and `bzc co --lightweight`. (Jelmer Vernooij, #608640)

- All bzr control directories, branch formats, repository formats and working tree formats now support feature flags, which are serialized in their respective format files. See `doc/developers/feature-flags.txt` for details. (Jelmer Vernooij)

- `bzrlib.urlutils` now includes `quote` and `unquote` functions, rather than importing them from `urllib`. This prevents loading of the `socket`, `ssl` and `urllib` modules for local bzr operations. (Jelmer Vernooij)

- Configuration options can be SI units by using `int_SI_from_unicode` as their `convert_from_unicode` helper. (Vincent Ladeuil)

- Configuration stacks can now use `StartingPathMatcher` to select the sections matching a location while respecting the order chosen by the user in the configuration file: from generic sections to specific sections. (Vincent Ladeuil, #832046).

- Configuration stores can now save incremental changes by using `save_changes()` instead of `save()`. This reduces the number of required input/outputs and allows stores to be shared between stacks. (Vincent Ladeuil)

- `ControlDir` now has a `get_branches` method that returns a dictionary whose keys are the names of the branches and whose values are the branches themselves. The active branch uses the key `None`. (Neil Martinsen-Burrell)

- Helper `osutils.path_from_environ` added for extracting a unicode path from an environment variable. (Martin Packman, #832028)

- Helper `win32utils.get_environ_unicode` added for avoiding encoding problems with `os.environ.get use`. (Martin Packman, #262874)

- Lazy imports can now only be absolute. (Jelmer Vernooij)

- `Merge3Mergers` now have an optional `other_branch` argument which contains the branch from which the `other_tree` was obtained, if any. (Jelmer Vernooij)

- `MutableTree` now has a hook `post_build_tree` which is called after a new mutable tree has been created. (Jelmer Vernooij, #912765)

- New HPSS call `BzrDir.checkout_metadir`. (Jelmer Vernooij, #894459)
• New HPSS call `VersionedFileRepository.get_inventories`, speeding up various commands including `bzr export`, `bzr checkout` and `bzr cat`. (Jelmer Vernooij, #608640)

• The `ConfigCommandLineStore` is now supported by `bzr config` and is seen as single no-name section of configuration options. (Vincent Ladeuil)

### 4.5.7 Testing

• New matcher `ContainsNoVfsCalls` which filters a list of HPSS calls for VFS requests. (Jelmer Vernooij)

• New `MemoryStack` class allows for diskless tests and locally injected configuration stacks. Lower level tests for predefined set of options can be written without setting up configuration files. (Vincent Ladeuil)

### 4.6 `bzr 2.5b4`

2.5b4 2011-12-08

This is the fourth beta of the 2.5 series, leading to a 2.5.0 release in February 2012. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This release includes many improvements in the smart server, UI polish for the colocated branches, optimizations for revision specifiers to avoid history sized operations, enhancements to the config framework, bug fixes related to unicode paths and more.

All bug fixed in previous series known at the time of this release are included.

#### 4.6.1 External Compatibility Breaks

None.

#### 4.6.2 New Features

• Provides a `po_merge` plugin to automatically merge `.po` files with `msgmerge`. See `bzr help po_merge` for details. (Vincent Ladeuil, #884270)

#### 4.6.3 Improvements

• `bzr branch --stacked` now only makes a single connection to the remote server rather than three. (Jelmer Vernooij, #444293)

• `bzr export --uncommitted` will export the uncommitted tree. (Jelmer Vernooij, #555613)

• `bzr rmbranch` can now remove colocated branches. (Jelmer Vernooij, #831464)

• `bzr status` no longer shows shelves if files are specified. (Francis Devereux)

• `bzr switch` now accepts colocated branch names to switch to. (Jelmer Vernooij, #826814)

• Plugins can now register additional “location aliases”. (Jelmer Vernooij)

• Revision specifiers will now only browse as much history as they need to, rather than grabbing the whole history unnecessarily in some cases. (Jelmer Vernooij)

• When using `bzr switch` to switch to a sibling of the current branch, the relative branch name should no longer be url-encoded. (Jelmer Vernooij)
4.6.4 Bug Fixes

- A new section local option `basename` is available to help support some `bzr-pipeline` workflows and more generally help mapping local paths to remote ones. See `bzr help configuration` for more details. (Vincent Ladeuil, #843211)

- Add HPSS call for looking up revision numbers from revision ids on remote repositories. (Jelmer Vernooij, #640253)

- Add HPSS call for retrieving file contents from remote repositories. Should improve performance for lightweight checkouts and exports of from remote repositories. (Jelmer Vernooij, #368717, #762330, #608640)

- Allow lazy compiled patterns from `bzrlib.lazy_regex` to be pickled. (Jelmer Vernooij, #893149)

- `bzc info` no longer shows empty output if only a control directory is present. (Jelmer Vernooij, #159098)

- Cope with missing revision ids being specified to `Repository.gather_stats` HPSS call. (Jelmer Vernooij, #411290)

- Fix test failures on windows related to locations.conf handling. (Vincent Ladeuil, #892992)

- Fixed parsing of the timestamp given to `commit --commit-time`. Now prohibits several invalid strings, reads the correct number of seconds, and gives a better error message if the time zone offset is not given. (Matt Giuca, #892657)

- Give meaningful file/line references when reporting deprecation warnings for `_COMPATIBILITYThunkFeature` based test features. (Vincent Ladeuil, #897718)

- Make reporting of mistakes involving unversioned files with non-ascii filenames work again without ‘Unprintable exception’ being shown. (Martin Packman, #898408)

- Provide names for lazily registered hooks. (Neil Martinsen-Burrell, #894609)

- Raise `BadIndexKey` exception in `btree_index` when a key is too large, fixing an infinite recursion issue. (Shannon Weyrick, #720853)

- Resolve regression from colocated branch path handling, by ensuring that unreserved characters are unquoted in URLs. (Martin Packman, #842223)

- Split segments from URLs for colocated branches without assuming the combined form is valid. (Martin Packman, #842233)

- Support looking up revision numbers by revision id in empty branches. (Jelmer Vernooij, #535031)

- Support verifying signatures on remote repositories. (Jelmer Vernooij, #889694)

- Teach the bzr client how to reconnect if we get `ConnectionReset` while making an RPC request. This doesn’t handle all possible network disconnects, but it should at least handle when the server is asked to shutdown gracefully. (John Arbash Meinel, #819604)

- When a remote format is unknown, bzr will now print a single-line error message rather than a backtrace. (Jelmer Vernooij, #687226)

4.6.5 API Changes

- `BzrDir.open_branch` and `BranchFormat.open` now take an optional `possible_transports` argument. (Jelmer Vernooij)

- New method `Transport.set_segment_parameter`. (Jelmer Vernooij)

- `Repository.verify_revision` has been renamed to `Repository.verify_revision_signature`. (Jelmer Vernooij)
• RevisionSpec.wants_revision_history now defaults to False and is deprecated. The revs argument of RevisionInfo.from_revision_id is now deprecated. (Jelmer Vernooij)

• Tree.get_file_by_path is now deprecated. Use Tree.get_file instead. (Jelmer Vernooij, #666897)

• Some global options for use with commands have been removed, construct an Option with the name instead. (Martin Packman)

• The unused exception HistoryMissing has been removed. (Jelmer Vernooij)

4.6.6 Internals

• Add HPSS call for Repository.pack. (Jelmer Vernooij, #894461)

• bzr config uses the new configuration implementation. (Vincent Ladeuil)

• Custom HPSS error handlers can now be installed in the smart server client using the error_translators and no_context_error_translators registries. (Jelmer Vernooij)

• New HPSS calls
  Repository.has_signature_for_revision_id,
  Repository.make_working_trees,  BzrDir.destroy_repository,
  BzrDir.has_workingtree,  Repository.get_physical_lock_status,
  Branch.get_physical_lock_status,  Branch.put_config_file,
  Branch.break_lock,  BzrDir.destroy_branch,  Repository.break_lock,
  VersionedFileRepository.get_serializer_format,  Repository.all_revision_ids,
  Repository.start_write_group,  Repository.commit_write_group,
  Repository.abort_write_group,  Repository.check_write_group,
  Repository.iter_revisions,  Repository.add_signature_revision_text and
  Repository.get_revision_signature_text. (Jelmer Vernooij)

• RemoteBranch.get_config_stack and RemoteBzrDir.get_config_stack will now use HPSS calls where possible. (Jelmer Vernooij)

• The registry of merge types has been moved to merge from option but merge.get_merge_type_registry remains as an accessor. (Martin Packman)

4.6.7 Testing

• Avoid failures in test_transform when OS error messages are localised. (Martin Packman, #891582)

• Tests are now subject to a time limit: by default 300s, and 120s when run from ‘make check’, controlled by the selftest.timeout configuration option. This is currently not supported on Windows. (Martin Pool)

4.7 bzr 2.5b3

2.5b3 2011-11-10

This is the third beta of the 2.5 series, leading to a 2.5.0 release in February 2012. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This release includes log options for push and pull, more UI polish for colocated branches, a better and more coherent implementation for UI dialogs, enhancements to the config framework and more.

This release includes all bug fixed in previous series known at the time of this release.
4.7.1 External Compatibility Breaks

None

4.7.2 New Features

• The `log_format` configuration can be used with `-Olog_format=line` to change the format push and pull use to display the revisions. I.e.: `bzr pull -v -Olog_format=short` will use the short format instead of the default long one. (Vincent Ladeuil, #861472)

• The new config scheme allows an alternative syntax for the ‘appendpath’ policy relying on option expansion and defining a new ‘relpath’ option local to a section. Instead of using `<option>:policy=appendpath`, the option value can be defined as ‘option=xxxx/{relpath}’. (Vincent Ladeuil, #832013)

4.7.3 Improvements

• `bzr info -v` now shows the number of colocated branches for control directories that support them. (Jelmer Vernooij, #863285)

• `bzr version-info` now takes a `--revision` argument. (Jelmer Vernooij, #238705)

• `bzr revno` now takes a `--revision` argument. (Jelmer Vernooij, #870649)

• `bzr serve` now can serve from URLs rather than just from the file system. I.e.: `bzr serve -d lp.bzr` or `bzr serve -d file:///data/bzr` (Jelmer Vernooij)

• all input prompts are now char-based when possible, and can be forced to line-based mode by setting the `BZR_TEXTUI_INPUT` environment variable to ‘line-based’. This replaces the previous shelf UI only patch using `INSIDE_EMACS`. (Benoît Pierre)

4.7.4 Bug Fixes

• `bzr info` now shows the master branch location too for treeless local branches. (Jelmer Vernooij, #258355)

• `bzr mkdir --quiet` now does not print a line for every created directory. (Martin von Gagern, #869915)

• `bzr mv` does not crash when attempting to move the root of a branch. (Jonathan Riddell, #809728)

• `bzr shelve` now use `UIFactory.choose` for input handling, making it usable when creating a custom `UIFactory` implementation. (Benoît Pierre)

• `bzr clean-tree` now use `UIFactory.get_boolean` for confirmation prompt, making it usable when using a custom `UIFactory` implementation. (Benoît Pierre)

• If sending a crash through Apport fails report the Apport failure to `bzr.log` rather than stderr. (Jonathan Riddell, #766735)

• `bzr upgrade` no longer treats ‘already up-to-date’ exceptions as errors. (Benoît Pierre, #716560).

• `bzr version-info` no longer populates the clean state for custom templates unless `[clean]` is explicitly asked for. (Lawrence Mitchell, #882541)

• Fix finding the CPU count when using Python >= 2.6 on BSD-based systems. (Jelmer Vernooij, #887151)

• `WorkingTree.clone()` now supports its `revision_id` being set to the null revision. (Jelmer Vernooij, #876423)

• `WorkingTree.pull` can now pull `NULL_REVISION`. (Jelmer Vernooij, #887556)
4.7.5 API Changes

- `Branch.revision_history` is now deprecated. (Jelmer Vernooij, #799519)

- Methods `add` and `items` of `LRUCache` and `LRUSizeCache` are deprecated. Use normal dict-style access instead. (Martin Packman)

- New flag `RepositoryFormat.supports_unreferenced_revisions` which indicates whether revisions can be present in a repository without being referenced from e.g. a branch history at the same time. (Jelmer Vernooij)

- `UIFactory.choose` has been added: prompt the user for a list of choices. (Benoît Pierre)

4.7.6 Internals

- `ControlDirFormat` now has a new method `supports_transport` which format implementations can use whether or not they can access a control dir over a particular transport. (Jelmer Vernooij)

- `BranchBuilder.build_commit` now takes `parent_ids` and `allow_leftmost_as_ghost` arguments. (Jelmer Vernooij)

4.7.7 Testing

- Ensure `TestCase` instances are deallocated immediately after running where possible. This greatly reduces the peak resource needs of a full test suite run. The new `-Euncollected_cases` selftest flag will add failures if any case which persists pasts its expected lifetime. (Martin Packman, #613247)

- Report exceptions from child processes during fork instead of swallowing the error and reporting that everything went okay. (Martin Packman, #804130)

4.8 bzr 2.5b2

This is the second beta of the 2.5 series, leading to a 2.5.0 release in February 2012. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This release includes more filtering options for `bzr log`, idle connections handling for `bzr serve`, a development-colo experimental format to flesh out the colocated branches UI, better support for foreign formats, enhancements to the config framework and more.

This release includes all bug fixed in previous series known at the time of this release.

**2.5b2 2011-10-06**

4.8.1 External Compatibility Breaks

None

4.8.2 New Features

- A new `-O` standard option (common to all commands) have been added. It provides a value for a config option in the `-Oname=value` form that takes precedence over all definitions found in config files. It can be used multiple times to override different options. (Vincent Ladeuil, #491196)
• `bzr log` now has an option called `--omit-merges` to omit those commits that merged branches, i.e. those having more than one parent. In order to avoid confusion, the previous command line option `--include-merges` has been renamed to `--include-merged`. The old name of the command line option will still be accepted. The name change also affects `bzr missing`. (Martin von Gagern)

• `bzr serve` will now disconnect clients if they have not issued an RPC request after 5 minutes. On POSIX platforms, this will also happen for `bzr serve --inet`. This can be overridden with the configuration variable `serve.client_timeout` or in the command line parameter `bzr serve --client-timeout=X`. Further, it is possible to request `bzr serve [--inet]` to shutdown gracefully by sending SIGHUP. It will finish the current request, and then close the connection. (John Arbash Meinel, #824797, #795025)

• The new experimental format `development-colo` supports colocated branches. This format will eventually be merged back into the 2a format when it has stabilized and there is adequate UI support for colocated branches. (Jelmer Vernooij, #831481)

### 4.8.3 Improvements

• Fixed a bug where `bzr tags -r x..y` loaded the branch history once for every revision in the range; it’s now much faster. (Vincent Ladeuil, #857335)

• `bzr info -v` can now be run against branches that don’t support `last_revision_info`, in which case the branch information will simply not be displayed. (Jelmer Vernooij)

### 4.8.4 Bug Fixes

• `bzr shelve` can now be used in emacs shells as the input handling is turned into a line-based one when `INSIDE_EMACS` is set (which is the case for all recent emacs versions). (Vincent Ladeuil, #856261)

• `bzr tags` can now be used against remote repositories that do not provide access to the revision graph. (Jelmer Vernooij, #858942)

• `bzr update PATH` will stop if you seem to be asking it to update anything less than a whole tree, because that’s not supported by `bzr`’s concept that the whole tree has a single basis revision. Previously, it would go ahead and update the whole tree, which was surprising. (Martin Pool, #557886)

• Don’t crash if `bzrlib.initialize()` has not been called while accessing configs. (Vincent Ladeuil, #863401)

• Redirects between http and https no longer discard path information in some cases. (Jelmer Vernooij, #853765)

• The `--overwrite` argument to `bzr push` and `bzr pull` no longer reports all tags as changed. (Jelmer Vernooij, #845396)

• `WorkingTree.get_file_mtime` now raises `NoSuchId` if a file id is specified that is unknown. (Jelmer Vernooij, #847435)

### 4.8.5 API Changes

• `Branch.get_revision_delta` has been deprecated. Use `Repository.get_revision_delta` instead. (Jelmer Vernooij, #859712)

• Plugins that implement custom protocols for `bzr serve` should now also take an argument `timeout`. This is used by the `bzr` protocol to close a connection if a client has been idle for more than X seconds. (Default 5 minutes). (John Arbash Meinel)
• Repository.fileids_altered_by_revision_ids has been moved to VersionedFileRepository and is no longer part of the standard Repository interface. (Jelmer Vernooij)

• The argument include_merges to missing.find_unmerged has been renamed to include_merged. The old name is still supported for now but will cause a deprecation warning. (Martin von Gagern)

• The new method ControlDirFormat.is_initializable() returns a boolean indicating whether or not it is possible to use any of the initialization methods of that format to create a new control dir. (Jelmer Vernooij)

4.8.6 Internals

• Branch objects can now use a config stack with the newly introduced get_config_stack(). Both get_config and get_config_stack can be used for the same branch but it’s recommended to stick to one for a given option.

4.8.7 Testing

• Test scripts can now use bzr shelve and provide their input as complete lines. (Vincent Ladeuil, #856261)

• Really corrupt the pack file without depending on a special length or value. (Vincent Ladeuil, #807032)

4.9 bzr 2.5b1

2.5b1 2011-09-15

This is the first beta of the 2.5 series, leading up to a 2.5.0 release in February 2012.

This release includes better support for gpg signing, better support for i18n (mostly command help and error messages), more options to filter bzr log output, more support for colocated branches (“location,branch=XXX” syntax), better feedback on updated tags for various commands, faster branching into an empty repository, enhancements to the config framework and more.

Beta releases are suitable for everyday use but may cause some incompatibilities with plugins. Some plugins may need small updates to work with 2.5b1.

4.9.1 External Compatibility Breaks

None

4.9.2 New Features

• A from_unicode parameter can be specified when registering a config option. This implements boolean, integer and list config options when the provided bool_from_store, int_from_store and list_from_store are used for this parameter. (Vincent Ladeuil)

• Accessing a packaging branch on Launchpad (eg, lp:ubuntu/bzr) now checks to see if the most recent published source package version for that project is present in the branch tags. This should help developers trust whether the packaging branch is up-to-date and can be used for new changes. The level of verbosity is controlled by the config item launchpad.packagingverbosity. It can be set to one of...
off  disable all checks

minimal  only display if the branch is out-of-date

short  also display single-line up-to-date and missing,

all  (default) display multi-line content for all states

(John Arbash Meinel, #609187, #812928)

• Add a config option gpg_signing_key for setting which GPG key should be used to sign commits. Also default
to using the gpg user identity which matches user_email() as set by whoami. (Jonathan Riddell, #68501)

• An invalid parameter can be specified when registering a config option to decide what should be done when
invalid values are encountered. ‘warning’ and ‘error’ will respectively emit a warning and ignore the value or
errors out. (Vincent Ladeuil)

• bzr add now skips large files in recursive mode. The default “large” size is 20MB, and is configurable via the
add.maximum_file_size option. A value of 0 disables skipping. Named items passed to add are never skipped.
(Shannon Weyrick, #54624)

• bzr help configuration/<option> display the help for

• config.Option can now declare default_from_env, a list of environment variables to get a default
value from. (Vincent Ladeuil)

• config.NameMatcher can be used to implement config stores and stacks that need to provide specific option
values for arbitrary unique IDs (svn repository UUIDs, etc). (Vincent Ladeuil, #843638)

• New builtin bzr branches command, which lists all colocated branches in a directory. (Jelmer Vernooij,
#826820)

• Relative local paths can now be specified in URL syntax by using the “file:” prefix. (Jelmer Vernooij)

• Report commits signed with expired keys in verify-signatures. (Jonathan Riddell, #804254)

• Translations are now enabled for command help, errors and globally for any message using gettext given on
output. (Jonathan Riddell, INADA Naoki, #83941)

4.9.3 Improvements

• bzr add will now warn about nested subtrees that are skipped. (Jelmer Vernooij, #187342)

• bzr commit -m ” can now be used to force an empty commit message. Entering an empty commit message
in the message editor still triggers an error. (Jelmer Vernooij)

• bzr pull will now mention how many tags it has updated. (Jelmer Vernooij, #164450)

• bzr tag no longer errors if a tag already exists but refers to the same revision, and will mention when a tag
has been updated rather than created. (Jelmer Vernooij, #381203)

• bzr uncommit will now remove tags that refer to removed revisions. The --keep-tags option can be
used to prevent this behaviour. (Jelmer Vernooij, #605814)

• Do not run i18n initialisation twice. (Jonathan Riddell)

• Install translation .mo files. (Jonathan Riddell)

• Locations printed by bzr upgrade are now formatted before display. (Jelmer Vernooij)
• Repository.get_parent_map now estimates the size of the returned content more accurately. This means that we get closer to the desired 64kB/request. For repositories converted from svn, this can be an improvement of approx 5:1 in round trips to discover the whole history. (John Arbash Meinel)

• Support a bugtracker option which is used by bzr commit --fixes if no bug tracker was specified on the command line. (Jelmer Vernooij, #334860)

• Use gettext.NullTranslations in i18n to allow use of i18n even when translations are not turned on. (Jonathan Riddell)

### 4.9.4 Bug Fixes

• bzr commit now correctly reports missing files as “removed”, not “modified”. (Jelmer Vernooij, #553955)

• bzr reconfigure will now allow multiple non-conflicting requests in a single invocation, e.g. --branch and --use-shared. (Martin von Gagern, #842993)

• A call to CHKInventory’s filter-method will not result in a DuplicateFileId error, if you move a subfolder and change a file in that subfolder. (Bastian Bowe, #809901)

• Branching from a stacked branch no longer does a get_parent_map request for each revisions that is in the stacked-on repository while determining what revisions need to be fetched. This mostly impacts branching initially into an empty shared repository when the source is not the development focus. (John Arbash Meinel, #388269)

• Decode BZR_HOME with fs encoding on posix platforms to avoid unicode errors. (Vincent Ladeuil, #822571)

• Fix fallout from URL handling changes in 2.5 that caused an IndexError to be raised whenever a transport at the drive root was opened on windows. (Martin [gz], #841322)

• Fixed loading of external merge tools from config to properly decode command-lines which contain embedded quotes. (Gordon Tyler, #828803)

• Rather than an error being raised, a warning is now printed when the current user does not have permission to read a configuration file. (Jelmer Vernooij, #837324)

• The pull command will now always use separate connections for the case where the destination is a heavyweight checkout of some remote branch on the same host as the source branch. (Martin von Gagern, #483661)

• TreeTransformBase.fixup_new_roots no longer forces trees to have a root, so operations that use it, like merge, can now create trees without a root. (Aaron Bentley)

### 4.9.5 Documentation

• Release instructions refreshed. (Vincent Ladeuil)

### 4.9.6 API Changes

• BranchFormat.initialize now takes a append_revisions_only argument. (Jelmer Vernooij)

• Branch._get_checkout_format now takes a lightweight argument which indicates if the format should be for a lightweight or a heavyweight checkout. (Jelmer Vernooij)

• ControlDir.create_branch now takes a append_revisions_only argument. (Jelmer Vernooij)

• New class URL in bzrlib.utils for managing parsed URLs. (Jelmer Vernooij)

• New method Config.get_user_option_as_int_from_SI added for expanding a value in SI format (i.e. “20MB”, “1GB”) into its integer equivalent. (Shannon Weyrick)
• New method `InterTree.file_content_matches` which checks that two files in different trees have the same contents. (Jelmer Vernooij)

• New method `Tree.get_file_verifier` which allows tree implementations to return non-sha1 checksums to verify files. (Jelmer Vernooij, #720831)

• New methods `get_transport_from_path` and `get_transport_from_url` have been added that only support opening from a path or a URL, unlike `get_transport`. (Jelmer Vernooij)

• New registry `OptionRegistry` specialized for configuration options. (Vincent Ladeuil)

• Remove `AtomicFile.closed` which has been deprecated in bzr 0.10. (Vincent Ladeuil)

• Remove `commands._builtin_commands`, `commands.shlex_split_unicode`, `Command._maybe_expand_globs` and `Command.run_direct` deprecated in 2.10 and 2.2.0. (Vincent Ladeuil)

• Remove `diff.get_trees_and_branches_to_diff` deprecated in 2.2.0.

• Remove `log.calculate_view_revisions`, `log._filter_revision_range`, `log.get_view_revisions` which have been deprecated in bzr 2.1.0. Also remove `log.show_one_log` which was never properly deprecated but wasn’t used and is easy to inline if needed. (Vincent Ladeuil)

• Remove `trace.info`, `trace.error` and `trace.show_log_error` deprecated in 2.1.0. (Vincent Ladeuil)

• Remove `TransportListRegistry.set_default_transport`, as the concept of a default transport is currently unused. (Jelmer Vernooij)

• Remove `UIFactory.warn_cross_format_fetch` and `UIFactory.warn_experimental_format_fetch` in favor of `UIFactory.show_user_warning`. (Jelmer Vernooij)

• `Tags` containers can now declare whether they support versioned tags and whether tags can refer to ghost tags. (Jelmer Vernooij)

• `Tags.merge_to` now returns a dictionary with the updated tags and a set of conflicts, rather than just conflicts. (Jelmer Vernooij)

• There is a new class `ContentFilterTree` that provides a facade for content filtering. The `filtered` parameter to `export` is deprecated in favor of passing a filtered tree, and the specific exporter plugins no longer support it. (Martin Pool)

• `Transport` now has a `_parsed_url` attribute instead of separate `_user`, `_password`, `_port`, `_scheme`, `_host` and `_path` attributes. Proxies are provided for the moment but may be removed in the future. (Jelmer Vernooij)

4.9.7 Internals

• A new debug flag `hpss_client_no_vfs` will now cause the HPSS client to raise a `HpssVfsRequestNotAllowed` exception when a VFS request is attempted. (Jelmer Vernooij)

• New method `ControlDir._get_selected_branch` which returns the colocated branch selected using path segment parameters. (Jelmer Vernooij, #380871)

4.9.8 Testing

• Blackbox tests (including test scripts) can be debugged interactively (see `bzrlib.debug.BzrPdb` for details). (Vincent Ladeuil)
• BranchBuilder.build_snapshot now supports a “flush” action. This cleanly and reliably allows tests using BranchBuilder to construct branches that e.g. rename files out of a directory and unversion that directory in the same revision. Previously some changes were impossible due to the order that build_snapshot performs its actions. (Andrew Bennetts)

• Don’t require os.fdatasync to be defined on all supported OSes (BSD-based OSes don’t define it). (Vincent Ladeuil, #822649)

• Fix compatibility with testtools 0.9.12. (Jelmer Vernooij, #815423)

• LockDir can now be run when the local hostname is localhost. (Jelmer Vernooij, #825994)

• ModuleAvailableFeature won’t try to import already imported modules, allowing it to be used for modules with side-effects. (Vincent Ladeuil, #712474)

• Output time stamps while running make check to get better timings from pqm. (Vincent Ladeuil, #837926)

• TestCaseWithMemoryTransport is faster now: _check_safety_net now just compares the bytes in the dirstate file to its pristine state, rather than opening the WorkingTree and calling last_revision(). This reduces the overall test suite time by about 10% on my laptop. (Andrew Bennetts)

• Update TestCase.knownFailure to the testtools way of handling expected failures to resolve Python 2.7 incompatibility. (Martin [gz], #607400)
BAZAAR RELEASE NOTES

5.1  *bzr 2.4.3*

2.4.3 NOT RELEASED YET

5.1.1 External Compatibility Breaks

5.1.2 New Features

5.1.3 Improvements

5.1.4 Bug Fixes

- Cope with Unix filesystems, such as smbfs, where chmod gives ‘permission denied’. (Martin Pool, #606537)

- During merges, when two entries end up using the same path for two different file-ids (the same file being ‘bzr added’ in two different branches), ‘duplicate’ conflicts are created instead of ‘content’ ones. This was previously leading to a ‘Malformed transform’ exception. (Vincent Ladeuil, #880701)

- Fix a traceback when trying to checkout a tree that also has an entry with file-id *TREE_ROOT* somewhere other than at the root directory. (John Arbash Meinel, #830947)

- Handle when an uncommit/update tries to add a file to an otherwise empty directory. The *Dirstate.update_basis_by_delta* would end up incorrectly flagging the delta as invalid. (Brian de Alwis, #855155)

- ‘Malformed transform’ exceptions are now recognized as internal errors instead of user errors and report a traceback. This will reduce user confusion as there is generally nothing users can do about them. (Vincent Ladeuil, #880701)

- Prevent a traceback being printed to stderr when logging has problems and accept utf-8 byte string without breaking. (Martin Packman, #714449)

- Some filesystems give *EOPNOTSUPP* when trying to call *fdatasync*. This shouldn’t be treated as a fatal error. (John Arbash Meinel, #1075108)

- Teach the *bzr* client how to reconnect if we get *ConnectionReset* while making an RPC request. This doesn’t handle all possible network disconnects, but it should at least handle when the server is asked to shutdown gracefully. This is a backport of the functionality in *bzr-2.5*. (John Arbash Meinel, #819604)

- Use *encoding_type=’exact’* for *bzr testament* so that on Windows the sha hash of the long testament matches the sha hash in the short form. (John Arbash Meinel, #1010339)
• When the limbo or pending-deletion directories exist, typically because of an interrupted tree update, but are empty, bzr no longer errors out, because there is nothing for the user to clean up. Also, errors in creation of these directories are no longer squelched. (Martin Pool, #427773)

• _Win32Stat object provides members st_uid and st_gid, those are present in Python’s os.stat object. These members required for external tools like bzr-git and dulwich. (Alexander Belchenko, #967060)

### 5.1.5 Documentation

### 5.1.6 API Changes

### 5.1.7 Internals

### 5.1.8 Testing

• Account for slightly improved compression with newer versions of zlib in `bt.test_btree_index` tests. (Martin Packman, #940453)

### 5.2 bzr 2.4.2

#### 2.4.2 2011-10-27

This is a bugfix release. Most of the bugs dealt with portability issues. Upgrading is recommended for all users of earlier 2.4 releases.

### 5.2.1 External Compatibility Breaks

None.

### 5.2.2 New Features

None.

### 5.2.3 Improvements

• Fixed a bug where `bzr tags -r x..y` loaded the branch history once for every revision in the range; it’s now much faster. (Vincent Ladeuil, #857335)

### 5.2.4 Bug Fixes

• Fixed an infinite loop when creating a repo at the root of the filesystem, i.e. “/”, due to `posixpath.normpath()` not collapsing 2 leading slashes into one, thus respecting the POSIX standard, but making `relpath()` loop infinitely. (Florian Vichot, #861008)

• Fixed loading of external merge tools from config to properly decode command-lines which contain embedded quotes. (Gordon Tyler, #828803)

• Include declaration of ‘changed’ to avoid an UnboundLocalError in dirstate pyrex code with new Cython versions. (Denys Duchier, #837221)
• Prevent several kinds of OverflowError and other fallout from failing to fit stat fields into four bytes in dirstate pack_stat implementations. (Martin Packman, #683191 #706957)
• Return early from create_delta_index_from_delta given tiny inputs. This avoids raising a spurious MemoryError on certain platforms such as AIX. (John Arbash Meinel, #856731)

5.2.5 Documentation

• Corrected documentation for bzr serve in the Admin Guide. (Morten Bøgeskov, Martin Pool, #832576)

5.2.6 API Changes

None.

5.2.7 Internals

No changes.

5.2.8 Testing

• Accept both old and new style testtools output in selftest tests. (Jelmer Vernooij, Martin Packman, #815423)
• Fix the race for TestingThreadingTCPServer in test_server_crash_while_responding. (Vincent Ladeuil, #869366)
• Really corrupt the pack file without depending on a special length or value. (Vincent Ladeuil, #807032)

5.3 bzr 2.4.1

2.4.1 2011-09-08

This is a bugfix release. Upgrading is recommended for all users of earlier 2.4 releases. It includes fixes from previous stable releases and address some issues with the test suite.

5.3.1 External Compatibility Breaks

5.3.2 New Features

5.3.3 Improvements

5.3.4 Bug Fixes

• config.LocationMatcher properly excludes unrelated sections. (Vincent Ladeuil, #829237)
• dirstate.fdatasync and repository.fdatasync can now properly be disabled. (Vincent Ladeuil, #824513)
• Disable os.fsync and os.fdatasync by default when running bzr selftest. You can use --sync to re-enable them. (John Arbash Meinel, #837293)
• Fix i18n use when no environment variables are set. (Jelmer Vernooij, #810701)
• Avoid UnicodeDecode error when reporting EINVAL from transports. (IWATA Hidetaka, #829237)

5.3.5 Documentation

• Corrected documentation for BZR_PROGRESS_BAR. (Dennis Benzinger, #735417)

5.3.6 API Changes

5.3.7 Internals

5.3.8 Testing

• The test suite should now be able to run under weird environments where /etc/passwd doesn’t contain the uid for the user running selftest or where fakeroot is used but /root is inaccessible. (Vincent Ladeuil, #825027)

5.4 bzr 2.4.0

2.4.0 2011-08-11

This release marks the start of a new long-term-stable series. From here, we will only make bugfix releases on the 2.4 series (2.4.1, etc, and support it until February 2013), while 2.5 will become our new development series.

This is a bugfix and polish release over the 2.3 series, with a large number of bugs fixed (>150 for the 2.4 series alone), and some performance improvements. Support for python 2.4 and 2.5 has been dropped, many large working tree operations have been optimized as well as some stacked branches operations.

Only bugfixes from other stables series have been included since 2.4b5 so all known fixed bugs are included here.

Users are encouraged to upgrade from the other stable series.

5.4.1 External Compatibility Breaks

5.4.2 New Features

5.4.3 Improvements

5.4.4 Bug Fixes

• A call to CHKInventory’s filter-method will not result in a DuplicateFileId error, if you move a subfolder and change a file in that subfolder. (Bastian Bowe, #809901)

• Accessing a packaging branch on Launchpad (eg, lp:ubuntu/bzr) now checks to see if the most recent published source package version for that project is present in the branch tags. This should help developers trust whether the packaging branch is up-to-date and can be used for new changes. The level of verbosity is controlled by the config item launchpad.packaging_verbosity. It can be set to one of

  off  disable all checks

  minimal  only display if the branch is out-of-date
short also display single-line up-to-date and missing.

all (default) display multi-line content for all states

(John Arbash Meinel, #609187, #812928)

• Cope with not all Python versions having a clear method on testCase._type_equality_funcs.
  (Martin [gz], Jelmer Vernooij, #809048)

• Fetching tags when fetching the tip revision of a branch is now controlled by the config setting
  branch.fetch_tags. The behavior has been reverted to 2.3’s not-fetching tagged revisions by default.
  (John Arbash Meinel, #771184)

• The fix for bug #513709 caused us to open a new connection when switching a lightweight checkout that was
  pointing at a bound branch. This isn’t necessary because we know the master URL without opening it, avoiding
  an extra SSH connection, etc. (John Arbash Meinel, #812285)

5.4.5 Documentation

5.4.6 API Changes

5.4.7 Internals

5.4.8 Testing

• BranchBuilder.build_snapshot now supports a “flush” action. This cleanly and reliably allows tests using
  BranchBuilder to construct branches that e.g. rename files out of a directory and unversion that directory in
  the same revision. Previously some changes were impossible due to the order that build_snapshot performs its
  actions. (Andrew Bennetts)

• TestCaseWithMemoryTransport is faster now: _check_safety_net now just compares the bytes in the dirstate file
  to its pristine state, rather than opening the WorkingTree and calling last_revision(). This reduces the
  overall test suite time by about 10% on my laptop. (Andrew Bennetts)

5.5 bzr 2.4b5

2.4b5 2011-07-07

This is the fifth (and last) beta of the 2.4 series leading to 2.4.0 release in August 2011. Beta releases are suitable for
everyday use but may cause some incompatibilities with plugins.

This release includes all bug fixed in previous series known at the time of this release.

5.5.1 External Compatibility Breaks

None.

5.5.2 New Features

• New command verify-signatures to check if all commits or specified commits have digital signatures
  from trusted keys. Requires python-gpgme to be installed.

• New option --signatures for bzr log to display digital signature verification results for each commit.
Bazaar Release Notes, Release 2.8.0dev1

• Config option acceptable_keys to list which GPG keys are verified as trusted.
• Config option validate_signatures_in_log to always show signatures in `bzr log`.

5.5.3 Improvements

• `Branch.open` is now about 3x faster (about 2ms instead of 6.5ms). (Andrew Bennetts).
• Pack, dirstate, and index files are synced to persistent storage if possible when writing finishes, to reduce the risk of problems caused by a machine crash or similar problem. This can be turned off through the `dirstate.fdatasync` and `repository.fdatasync` options, which can be set in `locations.conf` or `bazaar.conf`. (Martin Pool, #343427)

5.5.4 Bug Fixes

• Display a proper error message when a config file content cannot be decoded as UTF-8 or when it cannot be parsed. (Vincent Ladeuil, #502060, #688677, #797246)
• Generate a single conflict (instead of two) when merging a branch modifying and renaming a file in a branch that deleted it (or vice-versa). (Vincent Ladeuil, #688101)
• Give a more helpful message when the `bzr` executable doesn’t match the library. (This typically happens because of a misconfigured PYTHONPATH or half-installed `bzr`). (Martin Pool, #804553)
• Properly load utf8-encoded config files. (Vincent Ladeuil, #799212)
• `GraphThunkIdsToKeys.merge_sort` now properly returns keys rather than ids. (Jelmer Vernooij, #799677)
• `TreeTransformBase.fixup_new_roots` can now check that a tree root is present. (Jelmer Vernooij, #801257)

5.5.5 API Changes

• New attributes `WorkingTreeFormat.supports_versioned_directories` and `RepositoryFormat.supports_versioned_directories`. (Jelmer Vernooij, #765815)
• The “revno” field type when using the python version-info format is now a string (to handle dotted revnos) (Benoît Pierre, #796259)

5.5.6 Internals

• Start implementing localization, starting with command help text (but not the command options themselves). This will allow bootstrapping the `bzr` internationalization process. (Inada Naoki)

5.5.7 Testing

• Fix test failures when running as a homeless user (debian buildd). Tests leaking into `~/.bzr.log` should be detected properly now. (Vincent Ladeuil, #798698)
5.6 bzr 2.4b4

2.4b4  2011-06-16

This is the fourth beta of the 2.4 series, leading to a 2.4.0 release in August 2011. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This release includes all bug fixed in previous series known at the time of this release.

### 5.6.1 External Compatibility Breaks

- Do not treat configuration option ‘check_signatures = require’ as if it were ‘create_signatures = always’ (Jonathan Riddell)

### 5.6.2 New Features

- Hooks have been added for config stacks: get, set and remove are called when an option is respectively read, modified or deleted. Also added load and save hooks for config stores, called when the stores are loaded or saved. (Vincent Ladeuil)
- New hook server_exception in bzrlib.smart.server to catch any exception caused while running bzr serve. (Jonathan Riddell, #274578)
- New hook set_commit_message in bzrlib.msgeditor to set a commit message and revision properties. (Jonathan Riddell, #274578)
- Support -S as an alias for --short for the log and missing commands. (Martin von Gagern, #38655)

### 5.6.3 Improvements

- `bzr annotate` can be run without setting whoami data first. (Jonathan Riddell, #667408)

### 5.6.4 Bug Fixes

- Bazaar can now detect when a lock file is held by a dead process originating from the same machine, and steal the lock after printing a message to the user. This is off by default, for safety, but can be turned on by setting the configuration variable `locks.steal_dead` to True. (Martin Pool, #220464)
- `bzr version-info` now works when the tree is on a dotted revno. (Benoît Pierre, #796259)
- Credentials in the log output produced by –Dhttp are masked so users can more freely post them in bug reports. (Vincent Ladeuil, #723074)
- Fix a race condition for server_started hooks leading to a spurious test failure. (Vincent Ladeuil, #789167)
- Fix exporting subdirectory with --per-file-timestamps. (Szilveszter Farkas, #795557)
- Handle files that get created but don’t get used during TreeTransform. open() can create a file, and still raise an exception before it returns. So anything we might have created, make sure we destroy during finalize(). (Martin [gz], #597868)
- `pack_repo` now uses Transport.move instead of Transport.rename, deleting any existing targets even on SFTP. (Martin von Gagern, #421776)
• Pass the build_mo command to the rest of the setup() calls in setup.py. The bdist_wininst and py2exe code paths were failing because build_mo became a required step that they didn’t know about. (John Arbash Meinel, #787122)

• Preserve existing root-id when merging an unrelated branch. (Aaron Bentley, #806356)

• Properly avoid re-adding a file after it changes case on CICP filesystems. (John Arbash Meinel, #798130)

• Reports the original error when an InvalidHttpResponse exception is encountered to facilitate debug. (Vincent Ladeuil, #788530)

• Reports a non-existent file error when trying to merge in a file that does not exist. (Jonathan Riddell, #330063)

• UIFactory.prompt, UIFactory.get_username, UIFactory.get_password and UIFactory.get_boolean now require a unicode prompt to be passed in. (Jelmer Vernooij, #592083)

• Support merging into the empty tree. (Aaron Bentley, #595328)

5.6.5 Documentation

• Improve documentation of bzr merge --force. (Neil Martinsen-Burrell, #767307)

• Make docs for configuration options for digital signatures match reality. (Jonathan Riddell)

• Add user-guide page on GPG signatures. (Jonathan Riddell)

5.6.6 API Changes

• Checking for a file id in a Tree or Inventory using in is now deprecated. Instead, use has_id. (Martin Pool)

• Exporters are now all exposed as generators, rather than as single-call functions, so that calling code can take stream the output. (Xaav, Martin Pool)

• Information about held lockdir locks returned from eg LockDir.peek is now represented as a LockHeldInfo object, rather than a plain Python dict. (Martin Pool)

• Remove file_status function. (Martin Pool)

• Repository.iter_reverse_revision_history is now deprecated. Use Graph.iter_lefthand_ancestry instead. (Jelmer Vernooij, #739481)

• Repository.get_ancestry has been deprecated. Use Graph.iter_ancestry instead. (Jelmer Vernooij, #784511)

5.6.7 Internals

• tools/check-newbugs.py accepts a --browser option to open corresponding launchpad pages in a browser. (Vincent Ladeuil)

5.6.8 Testing

• A ImportTariffTestCase base class has been added in bzrlib.tests.test_import_tariff, which can be used for import tariff tests in plugins. (Jelmer Vernooij, #793465)

• Fix deadlock in TestImportTariffs.test_simple_serve when stderr gets more output than fits in the default buffer. This was happening on the Windows buildslave, and could easily happen in other circumstances where the default OS buffer size for pipes is small or the python -v output is large. (Andrew Bennetts, #784802)
• Fix spurious test failure on OSX for WorkingTreeFormat2. (Vincent Ladeuil, #787942)
• Re-target `bb.test_merge.TestMerge.test_merge_reversed_revision_range` and rewrite it as a parametrized test to avoid unrelated failures. (Vincent Ladeuil, #795456)
• Show log file contents from subprocesses started by `start_bzr_subprocess` in test failure details. This may help diagnose strange hangs and failures involving subprocesses. (Andrew Bennetts)
• Skip `utextwrap` tests when `sphinx` breaks `textwrap` by an hostile monkey-patch to `textwrap.TextWrapper.wordsep_re`. (Vincent Ladeuil, #785098)
• Multiple `selftest --exclude` options are now combined instead of overriding each other. (Vincent Ladeuil, #746991)
• Restore some `FTPTransport` test coverage by allowing `pyftpdlib 0.6.0` to be used. Also restore `medusa` support while leaving it disabled to make it easier to use if/when we can in the future. (Vincent Ladeuil, #781140)
• `TestImportTariffs` no longer uses the real `$HOME`. This prevents it from polluting `$HOME/.bzr.log` or being accidentally influenced by user configuration such as aliases. It still runs with all the user’s plugins enabled, as intended. (Vincent Ladeuil, Andrew Bennetts, #789505)

5.7  `bzr 2.4b3`

`2.4b3`  2011-05-26

This is the third beta of the 2.4 series, leading to a 2.4.0 release in August 2011. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This release includes all bug fixed in previous series known at the time of this release.

5.7.1  External Compatibility Breaks

• `bzr-2.4` has officially dropped support for python2.4 and python2.5. We will continue to maintain `bzr-2.3` for people who still need to use those versions of python. (John Arbash Meinel)

5.7.2  New Features

• The text compressor used for 2a repositories now has a tweakable parameter that can be set in `bazaar.conf. bzr.groupcompress.max_entries_per_source` default of 65536. When doing compression, we build up an index of locations to match against. Setting this higher will result in slightly better compression, at a cost of more memory. Note that a value of 65k represents fully sampling a 1MB file. So this only has an effect when compressing texts larger than N*16 bytes. (John Arbash Meinel, #602614)

5.7.3  Improvements

• `bzr branch --stacked` from a smart server uses the network a little more efficiently. For a simple branch it reduces the number of round-trips by about 20%. (Andrew Bennetts)
• `bzr log --line` scales the width of the author field with the size of the line. This means that the full author name is shown when the environment variable `BZR_COLUMNS=0`. (Neil Martinsen-Burrell)
• `bzr pull` now properly triggers the fast `CHKInventory.iter_changes` rather than the slow generic `inter-Inventory changes`. It used to use a `DirStateRevisionTree` as one of the source trees, which is faster when we have to read the whole inventory anyway, but much slower when we can get just the delta out of the repository. On a 70k record tree, this changes `bzr pull` from 28s down to 17s. (John Arbash Meinel, #780677)

• Slightly reduced memory consumption when fetching into a 2a repository by reusing existing caching a little better. (Andrew Bennetts)

• Speed up `bzr status` by a little bit when there are a couple of modified files. We now track how many files we have seen that need updating, and only rewrite the dirstate file if enough of them have changed. The default is 10, and can be overridden by setting the branch option “`bzr.workingtree.worth_saving_limit`”. (Ian Clatworthy, John Arbash Meinel, #380202)

• Speed up `bzr uncommit`. Instead of resetting the dirstate from scratch, use `update_basis_by_delta`, computing the delta from the repository. (John Arbash Meinel, #780544)

5.7.4 Bug Fixes

• All Tree types can now be exported as `tar.*`, `zip` or directories. (Aaron Bentley)

• `bzr merge --no-remember location` never sets `submit_branch`. (Vincent Ladeuil, #782169)

• `bzr pull --no-remember location` never sets `parent_location`. `bzr push --no-remember location` never sets `push_location`. `bzr send --no-remember submit_location public_location` never sets `submit_branch` nor `public_branch`. (Vincent Ladeuil)

• Conflicts involving non-ascii filenames are now properly reported rather than failing with a UnicodeEncodeError. (Martin [GZ], #686161)

• Correct parent is now set when using ‘switch -b’ with bound branches. (A. S. Budden, #513709)

• Fix `bzr plugins` regression in bzr 2.4 which resulted in a traceback from writelines on cj.k terminals. (Martin [GZ], #754082)

• `WT.inventory` and `WT.iter_entries_by_dir()` was not correctly reporting subdirectories that were tree references (in formats that supported them). (John Arbash Meinel, #764677)

• Merging into empty branches now gives an error as this is currently not supported. (Jonathan Riddell, #242175)

• Do not show exception to user on pointless commit error. (Jonathan Riddell #317357)

• `WT.update_basis_by_delta` no longer requires that the deltas match the current WT state. This allows `update_basis_by_delta` to be used by more commands than just commit. Updating with a delta allows us to not load the whole inventory, which can take 10+s with large trees. (Jonathan Riddell, John Arbash Meinel, #781168)

• `bzr mv --after old_name new_name` now works if “new_name” is newly added. (Benoît Pierre)

5.7.5 Documentation

• Restore the workaround for option names including dots (~1.14) which was disabled when we stopped listing –1.9 as a format. (Vincent Ladeuil, #782289)
5.7.6 API Changes

- `annotate_file` has been deprecated in favor of `annotate_file_revision_tree`. (Jelmer Vernooij, #775598)
- `Branch.fetch` now takes an optional `limit` argument. (Andrew Bennetts, Jelmer Vernooij, #750175)
- `Inter.get` now raises `NoCompatibleInter` if there are no compatible optimisers rather than an instance of the class it is called on. (Jelmer Vernooij)
- `Branch.push` now takes a `lossy` argument. `Branch.lossy_push` has been removed. (Jelmer Vernooij)
- New method `Repository.get_file_graph` which can return the per-file revision graph. (Jelmer Vernooij, #775578)
- The default implementation of `Branch` is now oriented to storing the branch tip. Branch implementations which store the full history should now subclass `FullHistoryBzrBranch`. `Branch._last_revision_info` has been renamed to `Branch._read_last_revision_info`. (Jelmer Vernooij)
- `Tree.__iter__` has been deprecated; use `Tree.all_file_ids` instead. (Jelmer Vernooij)
- `Tree.get_symlink_target` now takes an optional `path` argument. (Jelmer Vernooij)

5.7.7 Internals

- `MutableTree.smart_add` now uses inventory deltas. (Jelmer Vernooij, #146165)
- Removed `bzrlib.branch._run_with_write_locked_target` as `bzrlib.cleanup` provides the same functionality in a more general way. (Andrew Bennetts)

5.7.8 Testing

- A test that was expected to fail but passes instead now counts as a failure catching up with new testtools and subunit handling. (Martin [GZ], #654474)
- Make it easier for plugins to reuse the per_workingtree scenarios by restoring the `wt_scenarios` helper that was accidentally deleted. (Vincent Ladeuil, #783472)
- Removed `test_breakin` tests that were excessively prone to hanging, did not work on Wine, and partly already disabled. (Martin Pool, #408814, #746985)
- Windows locations are different and should be tested accordingly. (Vincent Ladeuil, #788131)

5.8 bzr 2.4b2

2.4b2 2011-04-28

This is the second beta of the 2.4 series, leading to a 2.4.0 release in August 2011. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

This release includes all bug fixed in previous series known at the time of this release.
5.8.1 External Compatibility Breaks

- Two command synonyms for `bzr branch` have been deprecated, to avoid confusion and to allow the names to later be reused. The removed names are: `get` and `clone`. (Martin Pool, #506265)

5.8.2 New Features

- `bzr commit` now supports a `--lossy` argument that can be used to discard any data that can not be natively represented when committing to a foreign VCS. (Jelmer Vernooij, #587721)

5.8.3 Improvements

- `bzr merge` in large trees is now significantly faster. On a 70k entry tree, the time went from ~3min down to 30s. This also effects `bzr pull` and `bzr update` since they use the same merge logic to update the WorkingTree. (John Arbash Meinel, #759091)

- `bzr revert` now properly uses `bzr status`'s optimized `iter_changes`. This can be a significant performance difference (33s to 5s on large trees). (John Arbash Meinel, #759096)

- Resolve `lp:FOO` urls locally rather than doing an XMLRPC request if the user has done `bzr launchpad-login`. The `bzr+ssh` URLs were already being handed off to the remote server anyway (xmlrpc has been mapping `lp:bzr to `bzr+ssh://bazaar.launchpad.net/+branch/bzr, rather than `bzr+ssh://bazaar.launchpad.net/~bzr-pqm/bzr/bzr.dev for a few months now.) By doing it ourselves, we can cut out substantial startup time. From Netherlands to London it was taking 368ms to do the XMLRPC call as much as 2s from Sydney. You can test the local logic by using `-Dlaunchpad`. (John Arbash Meinel, #397739)

- When building a new WorkingTree (such as during `bzr co` or `bzr branch`) we now properly store the stat and hash of files that are old enough. This saves a fair amount of time on the first `bzr status` (on a 500MB tree, it saves about 30+s). (John Arbash Meinel, #740932)

5.8.4 Bug Fixes

- Arguments that can't be decoded to unicode in the current posix locale give a clearer error message without a traceback. (Martin [gz], #745712)

- `bzrlib.log._DEFAULT_REQUEST_PARAMS` is no longer accidentally mutated by `bzrlib.log._apply_log_request_defaults`. In practice these default values aren't relied on very often so this probably wasn't causing any trouble. (Andrew Bennetts)

- `bzr log` now works on revisions which are not in the current branch. (Matt Giuca, #241998)

- Don’t rewrite the dirstate file when non-interesting changes have occurred. This can significantly improve ‘bzr status’ times when there are only small changes to a large tree. (Ian Clatworthy, John Arbash Meinel, #380202)

- Lazy hooks are now reset between test runs. (Jelmer Vernooij, #745566)

- `bzrlib.merge.Merge` now calls `iter_changes` without `include_unversioned=True`. This makes it significantly faster in many cases, because it only looks at modified files, rather than building information about all files. This can cause failures in other TreeTransform code, because it had been expecting to know the names of things which had not changed (such as parent directories). All cases we know about so far have been fixed, but there may be fallout for edge cases that we are missing. (John Arbash Meinel, #759091)

- `SFTPTransport` is more pro-active about closing file-handles. This reduces the chance of having threads fail from async requests while running the test suite. (John Arbash Meinel, #656170)
• Standalone bZR.exe installation on Windows: user can put additional python libraries into site-packages subdirectory of the installation directory, this might be required for “installing” extra dependencies for some plugins. (Alexander Belchenko, #743256)

• transform.revert() has been updated to use wt.iter_changes(basis_tree) rather than basis_tree.iter_changes(wt). This allows the optimized code path to kick in, improving bZR revert times significantly (33s to 4s on large trees, 0.7s to 0.3s on small trees.) (John Arbash Meinel, #759096)

• TreeTransform.create_file/new_file can now take an optional sha1 parameter. If supplied, when the transform is applied, it will then call self._tree._observed_sha1 for those files. This lets us update the hash-cache for content that we create, preventing us from re-reading the content in the next bZR status. (John Arbash Meinel, #740932)

### 5.8.5 Documentation

• Added a section about using a shared SSH account on a server for bZR+ssh access. (Russell Smith)

• The documentation now recommends using SSH rather than SFTP in the tutorials and the examples, because that will generally be much faster and better in cases where it can be used. SFTP is still available and mentioned as an alternative. (Martin Pool, #636712)

### 5.8.6 API Changes

• Branch.update_revisions has been made private and should no longer be used by external users. Use Branch.pull or Branch.push instead. (Jelmer Vernooij, #771765)

• Commands now have an invoked_as attribute, showing the name under which they were called before alias expansion. (Martin Pool)

• Hooks.create_hook is now deprecated in favour of Hooks.add_hook. (Jelmer Vernooij)

• If you call bzrlib.initialize but forget to enter the resulting object as a context manager, bzrlib will now be initialized anyhow. (Previously simple programs calling bzrlib might find the library was mysteriously silent.) (Martin Pool)

• Inventory-specific functionality has been split out of Tree into a new InventoryTree class. Tree instances no longer necessarily provide an inventory attribute. (Jelmer Vernooij)

• Inventory-specific functionality has been split out of RevisionTree into a new InventoryRevisionTree class. RevisionTree instances no longer necessarily provide an inventory attribute. (Jelmer Vernooij)

• New method Hooks.uninstall_named_hook. (Jelmer Vernooij, #301472)

• revision_graph_can_have_wrong_parents is now an attribute on RepositoryFormat rather than a method on Repository. (Jelmer Vernooij)

• Testament now takes a tree rather than an inventory. (Jelmer Vernooij, #762608)

• TestCase.failUnlessExists and failIfExists are deprecated in favour of assertPathExists and assertPathDoesNotExist respectively. (Martin Pool)

• The revno parameter of log.LogRevision may now be None, representing a revision which is not in the current branch. (Matt Giuca, #241998)

• The various knit pack repository format classes have been moved from bzrlib.repofmt.pack_repo to bzrlib.repofmt.knitpack_repo. (Jelmer Vernooij)

• RevisionTree now has a new method get_file_revision. (Jelmer Vernooij)
• WorkingTree no longer provides an inventory. Instead, all inventory-related functionality is now on the subclass InventoryWorkingTree that all native Bazaar working tree implementations derive from. (Jelmer Vernooij)

5.8.7 Internals

• Added osutils.lstat and osutils.fstat. These are just the os functions on Linux, but they are wrapped on Windows so that fstat matches lstat results across all python versions. (John Arbash Meinel)

• WorkingTree._observed_sha1 also updates the ‘size’ column. It happened to be updated as a side-effect of commit, but if we start using the function elsewhere we might as well do it directly. (John Arbash Meinel)

5.8.8 Testing

• Stop using failIf, failUnless, failIfEqual, etc, that give PendingDeprecationWarnings on Python2.7. (Martin Pool, #760435)

5.9 bzr 2.4b1

2.4b1 2011-03-17

This is the first beta of the 2.4 series, leading up to a 2.4.0 release in August 2011. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins. Some plugins may need small updates to work with 2.4b1.

5.9.1 External Compatibility Breaks

(none)

5.9.2 New Features

• Added changelog_merge plugin for merging changes to Changelog files in GNU format. See bzr help changelog_merge for details. (Andrew Bennetts)

• Configuration options can now use references to other options in the same file by enclosing them with curly brackets ({{other_opt}}). This makes it possible to use, for example, push_location=lp:~vila/bzr/config-{{nickname}} in branch.conf when using a loom. During the beta period, the default behaviour is to disable this feature. It can be activated by declaring bzr.config.expand = True in bazaar.conf. (Vincent Ladeuil)

• External merge tools can now be configured in bazaar.conf. See bzr help configuration for more information. (Gordon Tyler, #489915)

• The lp: directory service now supports Launchpad’s QA staging. (Jelmer Vernooij, #667483)

5.9.3 Improvements

• A new hidden command bzr repair-workingtree. This is a way to force the dirstate file to be rebuilt, rather than using a bzr checkout workaround. (John Arbash Meinel)
• Added a \texttt{Branch\_heads\_to\_fetch} RPC to the smart server protocol. This allows formats from plugins (such as looms) to efficiently tell the client which revisions need to be fetched. (Andrew Bennetts)

• Branching, merging and pulling a branch now copies revisions named in tags, not just the tag metadata. (Andrew Bennetts, \#309682)

• \texttt{bzr cat-revision} no longer requires a working tree. (Jelmer Vernooij, \#704405)

• \texttt{bzr export \--per-file-timestamps} for .tar.gz files will now override the mtime for trees exported on Python 2.7 and later, which expose the ‘mtime’ field in gzip files. This makes the output of \texttt{bzr export \--per-file-timestamps} for a particular tree deterministic. (Jelmer Vernooij, \#711226)

• \texttt{bzr export \--format=zip} can now export to standard output, like the other exporters can. (Jelmer Vernooij, \#513752)

• \texttt{bzr export} can now create .tar.xz and .tar.lzma files. (Jelmer Vernooij, \#551714)

• Getting all entries from \texttt{CHKInventory.iter\_entries\_by\_dir()} has been sped up dramatically for large trees. Iterating by dir is not the best way to load data from a CHK inventory, so it preloads all the items in the correct order. (With the gcc-tree, this changes it (re)reading 8GB of CHK data, down to just 150MB.) This has noticeable affects for things like building checkouts, etc. (John Arbash Meinel, \#737234)

5.9.4 Bug Fixes

• A MemoryError thrown on the server during a remote operation will now be usefully reported, and other unexpected errors will include the class name. (Martin [gz], \#722416)

• \texttt{bzr annotate \--r-1 file} will now properly annotate a deleted file. (Andrew King, \#537442)

• \texttt{bzr export to zip files} will now set a mode on directories. (Jelmer Vernooij, \#207253)

• \texttt{bzr export to tgz files} will only write out the basename of the tarfile to the gzip file. (Jelmer Vernooij, \#102234)

• \texttt{bzr push \--overwrite} with an older revision specified will now correctly roll back the target branch. (Jelmer Vernooij, \#386576)

• \texttt{bzr lp\_propose} can now propose merges against packaging branches on Launchpad without requiring the target branch to be specified. (Jelmer Vernooij, \#704647)

• \texttt{bzr lp\_propose} no longer requires a reviewer to be specified. It will instead leave setting the reviewer up to Launchpad if it was not specified. (Jelmer Vernooij, \#583772)

• \texttt{bzr pull} will now exit with exit code 1 if there were tag conflicts. (Jelmer Vernooij, \#213185)

• \texttt{bzr mv} user errors no longer throw UnicodeEncodeError with non-ascii paths, however they may still print junk if not on a UTF-8 terminal. (Martin [gz], \#707954)

• \texttt{bzr reconfigure \--unstacked} now copies revisions (and their ancestors) named in tags into the unstacked repository, not just the ancestry of the branch’s tip. (Andrew Bennetts, \#401646)

• \texttt{bzr serve} no longer crashes when a server\_started hook is installed and IPv6 support is available on the system. (Jelmer Vernooij, \#293697)

• \texttt{bzr status} will not rewrite the dirstate file if it only has ‘trivial’ changes. (Currently limited to dir updates and newly-added files changing state.) This saves a bit of time for regular operations. eg. \texttt{bzr status} in a 100k tree takes 1.4s to compute the status, but 1s to re-save the dirstate file. (John Arbash Meinel, \#765881)

• \texttt{bzr tags} will no longer choke on branches with ghost revisions in their mainline and tags on revisions not in the branch ancestry. (Jelmer Vernooij, \#397556)

• \texttt{bzr whoami} will now display an error if both a new identity and \texttt{\--email} were specified. (Jelmer Vernooij, \#680449)
• **launchpadlib** doesn’t provide the **uris** module in some old versions. (Vincent Ladeuil, #706835)

• Empty entries in the **NO_PROXY** variable are no longer treated as matching every host. (Martin Pool, #586341)

• Plugins incompatible with the current version of bzr no longer produce a warning on every command invocation. Instead, a message is shown by **bzr plugins** and in crash reports. (#704195, Martin Pool)

• The “pretty” version of **needs_read_lock** and **needs_write_lock** now preserves the identity of default parameter values. (Andrew Bennetts, #718569)

• **bzr dump-btree --raw** no longer tracebacks on a B-Tree file containing no rows. (Eric Siegerman, #715508)

• Fix **bzr lp-mirror** to work on command line branch URLs and branches without an explicit public location. (Max Bowsher)

• On Python 2.6 and higher, use **multiprocessing.cpu_count()** to retrieve the number of available processors. (Jelmer Vernooij, #693140)

## 5.9.5 API Changes

• Added **Branch.heads_to_fetch** method. Implementations of the Branch API must now inherit or implement this method. (Andrew Bennetts, #721328)

• Added **bzrlib.mergetools** module with helper functions for working with the list of external merge tools. (Gordon Tyler, #489915)

• All methods and arguments that were deprecated before 2.0 have been removed. (Jelmer Vernooij)

• Branch formats should now be registered on the format registry (**bzrlib.branch.format_registry**) rather than using the class methods on **BranchFormat**. (Jelmer Vernooij, #714729)

• **Branch.set_revision_history** is now deprecated. (Jelmer Vernooij)

• **BranchFormat.supports_leaving_lock()** and **RepositoryFormat.supports_leaving_lock** flags have been added. (Jelmer Vernooij)

• **Branch.fetch** implementations must now accept an optional **fetch_tags** keyword argument. (Andrew Bennetts)

• **Branch.import_last_revision_info** is deprecated. Use the **import_last_revision_info_and_tags** method instead. (Andrew Bennetts)

• Because it was too specific to BzrDir implementations, **ControlDir.sprout** no longer has a default implementation; it now raises **NotImplementedError**. (Jelmer Vernooij, #717937)

• **bzrlib.deprecated_graph** has been removed. **bzrlib.graph** scales better tree and should be used instead. (Jelmer Vernooij, #733612)

• **ControlDirFormat.register_format** has been removed. Instead, Prober implementations should now implement a **known_formats** method. (Jelmer Vernooij)

• **ControlDirFormats** can now provide a **check_status** method and raise a custom exception or warning when an unsupported or deprecated format is being opened. (Jelmer Vernooij, #731311)

• **bzrlib.revisionspec.dwim_revspecs** is deprecated. Use **bzrlib.revisionspec.RevisionSpec_dwim.append_possible_revspec** and **bzrlib.revisionspec.RevisionSpec_dwim.append_possible_lazy_revspec** instead. (Jelmer Vernooij, #721971)

• **BzrDirFormat** has a new attribute **fixed_components** that indicates whether the components of the bzrdir can be upgraded independent of the **BzrDir**. (Jelmer Vernooij)
• BzrProber.register_format and BzrProber.unregister_format are now deprecated in favour of the BzrProber.formats format registry. (Jelmer Vernooij)

• ControlDir implementations no longer have to provide the get_branch_transport, get_workingtree_transport and get_repository_transport methods. (Jelmer Vernooij, #730325)

• Converter has been moved from bzrlib.bzrdir to bzrlib.controldir. (Jelmer Vernooij)

• Repository formats can now provide _get_extra_interrepo_test_combinations in the same module to provide extra test combinations for bzrlib.tests.per_repository. (Jelmer Vernooij)

• Repository formats should now be registered on the format registry (bzrlib.repository.format_registry) rather than using the class methods on RepositoryFormat. (Jelmer Vernooij)

• Repository formats can now indicate they do not support the full VersionedFiles API by setting the supports_full_versioned_files attribute to False. A subset of the VersionedFiles API (signatures and text graphs) still needs to be supported. (Jelmer Vernooij)

• Repository formats have a new method is_deprecated that implementations can override to return True to trigger a deprecation warning. (Jelmer Vernooij)

• The revision_id parameter of Repository.search_missing_revision_ids and InterRepository.search_missing_revision_ids is deprecated. It is replaced by the revision_ids parameter. (Andrew Bennetts)

• Working tree formats should now be registered on the format registry (bzrlib.working_tree.format_registry) rather than using the class methods on WorkingTreeFormat. (Jelmer Vernooij, #714730)

• Exporting may now be done with a generator (bzrlib.export.get_export_generator) (Geoff/xaav, #791005)

5.9.6 Internals

• CatchingExceptionThread (formerly ThreadWithException) has been moved out of the bzrlib.tests hierarchy to make it clearer that it can be used outside of tests. This class makes it easier to track exceptions in threads by catching them so they can be re-raised in the controlling thread. It’s available in the bzrlib.cethread module. (Vincent Ladeuil)

• Correctly propagate malloc failures from diff-delta.c code as MemoryError so OOM conditions during group-compress are clearly reported. This entailed a change to several function signatures. (Martin [gz], #633336)

• HookPoint.lazy_hook and Hooks.install_named_lazy_hook can install hooks for which the callable is loaded lazily. (Jelmer Vernooij)

5.9.7 Testing

• The Range parsing for HTTP requests will correctly parse incomplete ranges. (Vincent Ladeuil, #731240)
BAZAAR RELEASE NOTES

6.1 bzr 2.3.5

2.3.5 NOT RELEASED YET

6.1.1 External Compatibility Breaks

6.1.2 New Features

6.1.3 Improvements

6.1.4 Bug Fixes

- Cope cleanly with buggy HTTP proxies that close the socket in the middle of a multipart response. (Martin Pool, #198646).
- cStringIO is now unconditionally imported in bzrlib.config. (Jelmer Vernooij, #905361)
- Fix “Unprintable exception” error when a RetryWithNewPacks error is displayed. (Andrew Bennetts)

6.1.5 Documentation

6.1.6 API Changes

6.1.7 Internals

6.1.8 Testing

6.2 bzr 2.3.4

Codename One and counting

2.3.4 2011-07-14

This is a bugfix release. Upgrading is recommended for all users of earlier 2.3 releases.
This mainly fixes bug #786980 which blocked the SRU process for Ubuntu Natty.
6.2.1 External Compatibility Breaks

6.2.2 New Features

6.2.3 Improvements

- Tweak an RPC implementation for `Repository.get_parent_map`, it was doing an inefficient `small_set.difference_update(large_set)` when we can do `small_set = small_set.difference(large_set)`. This speeds up discovery time by about 10%. (John Arbash Meinel)

6.2.4 Bug Fixes

- Accept some differences for `bound_location` from the config files that were leading to a ‘ReadOnlyError: A write attempt was made in a read only transaction’ error. (Vincent Ladeuil, #786980)
- Don’t fail with traceback if `bzr serve` is running as a service on Windows, and there is no `USERNAME`, nor `BZR_EMAIL` or other whoami-related environment variables set. (Alexander Belchenko, Bug #660174)

6.2.5 Documentation

- Updated the “Using stacked branches” section of the user guide to describe committing to stacked branches and expanded its discussion of pushing a stacked branch. (Andrew Bennetts)

6.2.6 API Changes

6.2.7 Internals

6.2.8 Testing

- Remove the deprecation decorators for `failUnlessExists` and `failIfExists`. The deprecation “will” occur in 2.4, not before. Providing the wrappers is enough as far as 2.3 is concerned. (Vincent Ladeuil #794960)

6.3 bzr 2.3.3

2.3.3 2011-05-13

This is a bugfix release. Upgrading is recommended for all users of earlier 2.3 releases.

This fixed a bug in the test suite triggered by python-2.7 deprecating some tests helpers.

6.3.1 Testing

- Stop using `failIf`, `failUnless`, `failIfEqual`, etc, that give `PendingDeprecationWarnings` on Python2.7. (Martin Pool, #760435)
6.4  bzr 2.3.2

2.3.2  2011-05-12

This is a bugfix release. Upgrading is recommended for all users of earlier 2.3 releases.

This was never released due to bug #760435 interrupting the release process by breaking the test suite under python-2.7 on natty.

6.4.1 External Compatibility Breaks

None

6.4.2 New Features

None

6.4.3 Improvements

• Getting all entries from CHKInventory.iter_entries_by_dir() has been sped up dramatically for large trees. Iterating by dir is not the best way to load data from a CHK inventory, so it preloads all the items in the correct order. (With the gcc-tree, this changes it (re)reading 8GB of CHK data, down to just 150MB.) This has noticeable affects for things like building checkouts, etc. (John Arbash Meinel, #737234)

6.4.4 Bug Fixes

• Bazaar now infers the default user email address on Unix from the local account name plus the contents of /etc/mailname if that file exists. In particular, this means that committing as root through etckeeper will normally not require running bzr whoami first. (Martin Pool, #616878)

• bzr merge --preview --pull should respect the --preview option first, and not actually change the branch tip revision. (John Arbash Meinel, Dennis Duchier, #760152)

• bzr push into a repository (that doesn’t have a branch), will no longer copy all revisions in the repository. Only the ones in the ancestry of the source branch, like it does in all other cases. (John Arbash Meinel, #465517)

• Fix UnboundLocalError: local variable ‘lock_url’ in wait_lock error, especially while trying to save configuration from QBzr. (Martin Pool, #733136)

• Fix “Unable to obtain lock” error when pushing to a bound branch if tags had changed. Bazaar was attempting to open and lock the master branch twice in this case. (Andrew Bennetts, #733350)

• Standalone bzr.exe installation on Windows: user can put additional python libraries into site-packages subdirectory of the installation directory, this might be required for “installing” extra dependencies for some plugins. (Alexander Belchenko, #743256)

• When reporting a crash without apport, don’t print the full list of plugins because it’s often too long. (Martin Pool, #716389)

6.4.5 API Changes

None.
6.4.6 Testing

- FreeBSD8 has switched to python-2.7 which revealed a re-occurrence of a test failure in the launchpad plugin. `xmlrpclib.py` on natty carries a patch that is not in python-2.7 upstream and masked the issue. An additional fix has been added in the interim (<http://psf.upfronthosting.co.za/roundup/tracker/issue8194> should be fixed in python > 2.7.1). (Vincent Ladeuil, #654733)

6.5 bzr 2.3.1

2.3.1 2011-03-10

This is a bugfix release. Upgrading is recommended for all users of earlier 2.3 releases.

6.5.1 Bug Fixes

- Correctly resolve text conflicts for files in subdirs. (Vincent Ladeuil, #715058)
- Fix “AssertionError: repository.user_url ... does not match URL from server response” when reusing a smart transport. (Andrew Bennetts, #726584)
- Restore proper logging of bytes transferred. We accidentally reset the counter when commands finished before we logged the total transferred. (John Arbash Meinel, #713258)

6.6 bzr 2.3.0

2.3.0 2011-02-03

This release marks the start of another long-term-stable series. From here, we will only make bugfix releases on the 2.3 series (2.3.1, etc, and support it until August 2012), while 2.4 will become our new development series. The 2.1 and 2.2 series will also continue to get bugfixes. (Currently 2.0 is planned to be EOLed circa September 2011 and will receive only critical bugfixes.)

This is a bugfix and polish release over the 2.2 series, with a large number of bugs fixed (>130), and some performance improvements. Some features have been enhanced including commits on stacked branches, upgrades of related branches, shortcut URL schemes for ubuntu and debian on launchpad and better conflict resolution.

Only bugfixes from other stables series have been included since 2.3b5 so all known fixed bugs are included here. Users are encouraged to upgrade from the other stable series.

6.7 bzr 2.3b5

2.3b5 2011-01-13

This is the fifth and last beta of the 2.3 series, leading up to a 2.3.0 release in February. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

2.3b5 includes bug fixes for committing to stacked branches, smoother upgrades of multiple branches, compatibility with python-2.7, full test suite passing on Ubuntu Natty and windows, less round-trips for several smart server operations, better support text conflicts resolve actions and some more.

All known fixed bugs in other series (2.0, 2.1, 2.2) are also included here.
6.7.1 External Compatibility Breaks

(none)

6.7.2 Improvements

- A redundant parent inventories calculation was removed from fetch.py, as Repository.insert_stream already reports any missing inventories. This removes at least one network roundtrip when pushing to a stacked branch. (Andrew Bennetts)

- ControlDir.sprout no longer opens the target repository more than once. This avoids some unnecessary IO, and removes a network roundtrip when doing bzr branch to a smart server URL. (Andrew Bennetts)

- bzr modified now read-locks the working tree (and branch and repository) just once. (Andrew Bennetts)

- bzr resolve now accepts --take-this and --take-other actions for text conflicts. This replace the whole file with the content designated by the action. This will ignore all differences that would have been merge cleanly otherwise. (Vincent Ladeuil, #638451)

- bzr tags's “sort” argument now allows registering custom sort methods using the bzrlib.tag.tag_sort_methods registry. (Jelmer Vernooij, #701244)

- bt.test_http was breaking os.environ by erasing the values saved by TestCase leading to bt.test_import_tariff failures. (Vincent Ladeuil, #690563)

- upgrade now upgrades dependent branches when a shared repository is specified. It also supports new options: --dry-run for showing what will happen and --clean to remove the backup directory on successful completion. (Ian Clatworthy, Matthew Fuller, #89830, #374734, #422450)

6.7.3 Bug Fixes

- Avoid leaking SSH subprocess communication socket into unrelated child processes, which could cause bzr to hang on exit. (Max Bowshier, #696285)

- bzr break-lock on a corrupted lock file works correctly, rather than raising a PermissionDenied error. We were accidentally holding open the file we were trying to delete. (John Arbash Meinel, #659978)

- bzr update in a checkout of a readonly branch works again, without trying to set the tags in the master branch. This had been broken by the bug fix for bug #603395. (John Arbash Meinel, #701212)

- Per-transport tests now prefer to use Transport.get_bytes() rather than Transport.get().read(). The SFTP code uses an async message to close the file handle if you let the handle die from refcounting, while it uses a synchronous message if you close it directly. This should help prevent random test suite failures from race conditions. (John Arbash Meinel, #681047)

- Stop using bzrlib.tuned_gzip.GzipFile. It is incompatible with python-2.7 and was only used for Knit format repositories, which haven’t been recommended since 2007. The file itself will be removed in the next release. (John Arbash Meinel)

- The BZR_COLUMNS environment variable can be set to 0 to indicate no limitation on the width of the terminal. (Neil Martinsen-Burrell, #675652)

- Treat WSAECONNABORTED the same as WSAECONNRESET for the purposes of considering a smart data stream as being interrupted. This fixes a failure in the windows test suite, that was trying to ensure we cleanly handled a server disconnect. (John Arbash Meinel, #581311, #686587)

- Unshelving changes that occur in a now-unversioned directory now restore the directory properly rather than crashing. (John Arbash Meinel, #389674)
• You are now able to commit directly to a stacked branch. Any needed parent inventories will be filled in as part of the commit process. (John Arbash Meinel, #375013)

6.7.4 Documentation

• Better document the rules to update the bzr freshmeat page when doing a release. (Vincent Ladeuil, #690515)

6.7.5 API Changes

• Branch.sprout, BranchFormat.initialize and ControlDir.create_branch now take an optional repository keyword argument, and BranchFormat.open now takes an optional found_repository keyword argument. These provide the repository object for new branch object to use (for cases when the caller has already opened that repository). Implementations of these APIs will need to be updated to accept these arguments. (Andrew Bennetts)

• bzrlib.tuned_gzip.GzipFile is now deprecated and will be removed in the bzr-2.4 series. Code that was using it can just use the python stdlib gzip.GzipFile. (John Arbash Meinel)

6.7.6 Testing

• bzrlib.tests defines isolated_environ with the definitions of all the environment variables the tests should care about. It also defines override_os_environ and restore_os_environ to properly implement isolation from os.environ for tests. bzrlib.tests now defines a DocTestSuite class using this facility for all bzrlib doctests. (Vincent Ladeuil, #321320)

• Catch exceptions related to bug #637821 during test cleanup to avoid spurious failures. (Vincent Ladeuil, #686008).

• Check sphinx compatibility for tests requiring older sphinx versions. (Vincent Ladeuil, #686008).

• test_onto_transport in the Launchpad plugin can now run with Python 2.7. (Vincent Ladeuil, #654733)

• TestCase._captureVar and TestCase._old_env have been deleted due to bug #690563. Test writers are encouraged to use TestCase.overrideEnv instead. (Vincent Ladeuil)

• TestDebuntuExpansions was escaping the test isolation by calling the wrong base class setUp. (Vincent Ladeuil, #684662)

6.8 bzr 2.3b4

2.3.b4 2010-12-03
This is the fourth beta of the 2.3 series, leading up to a 2.3.0 release in February. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins.

2.3b4 includes bug fixes for the config command and conflict resolution. More changes were made for the test scripts handling to make it easier to add reproducing recipes to bugs.

It also includes bug fixes from the 2.2.2 release as well as the bug fixes in the upcoming 2.0.7, 2.1.4 and 2.2.3 releases. This means that all known fixed bugs at the time of this release are included.
6.8.1 External Compatibility Breaks

(none)

6.8.2 Improvements

- Bazaar now caches a branch’s tags while that branch is read-locked. This removes 1 network roundtrip from most interactions with a remote branch. (Andrew Bennetts)
- `bzr config <option>` will now display only the value itself so scripts can use it to query the currently active configuration. Displaying several options matching a given regular expression is now controlled via the `--all` option. (Vincent Ladeuil, bug #670251)
- `bzr resolve` now reports the number of conflicts resolved and the number of remaining conflicts. This provides a better feedback about the whole resolution process. (Vincent Ladeuil)
- Read configuration files in $XDG_CONFIG_HOME/bazaar on Unix if there is already a directory there. (Neil Martinsen-Burrell, #195397)

6.8.3 Bug Fixes

- Better message if there is an error while setting ownership of `.bazaar` directory. (Parth Malwankar, #657553)
- `bzr config` properly displays list values. (Vincent Ladeuil, #672382)
- `bzr config` will now respect option policies when displaying the value and display the definition sections when appropriate. (Vincent Ladeuil, #671050)
- Don’t create commit message files in the current directory to avoid nasty interactions with emacs (which tries to establish the status of the file during the commit which breaks on windows). (Vincent Ladeuil, #673637)
- `bzr resolve --take-other <file>` will not crash anymore if `<file>` is involved in a text conflict (but the conflict is still not resolved). (Vincent Ladeuil, #646961)
- Merge will now correctly locate a lca where there is a criss-cross merge of a new root. (Gary van der Merwe, #588698)
- Report error if non-ASCII command option given. (Rory Yorke, #140563)
- `tools/check-newsbug.py` is now based on `lp:hydrazine` and no longer crashes when encountering private bugs (they are just displayed as such). (Vincent Ladeuil, #354985)

6.8.4 Internals

- `BranchBuilder.build_snapshot` now accepts `parent_ids == []`. This can be used to create a new root in the graph. (Gary van der Merwe)
- Old repository development formats `RepositoryFormatCHK1` and `RepositoryFormatCHK2` have been removed, and so have the corresponding metadir format options `development-rich-root`, `development6-rich-root`, and `development7-rich-root`.

6.8.5 Testing

- Add a `null_output_matchesAnything` keyword argument with default False to `bzr-lib-tests.script.ScriptRunner.run_script` to specify that the command output should not be checked (as opposed to expecting an empty output). (Neil Martinsen-Burrell, #662509)
• Blank output section in scriptrunner tests no longer match any output. Instead, use ‘...’ as a wildcard if you
don’t care about the output. (Martin Pool, #637830)
• Bump minimum testtools version required to run `b` from 0.9.2 to 0.9.5 which will allow tests
that need the fixed unicode handling to be written. (Martin [gz])
• Introduce an `overrideEnv()` helper for tests that needs to change the environment variables while respecting
the isolation rules. Get rid of `TestCase._restoreEnvironment` which is now useless. (Vincent Ladeuil, #690563)
• Printing selftest results to a non-UTF-8 console will now escape characters that can’t be encoded rather than
aborting the test run with an exception. (Martin [gz], #633216)

6.9  `bzr 2.3b3`

2.3b3  2010-11-05

6.9.1  External Compatibility Breaks

(none)

6.9.2  New Features

• Add –no-tree option to ‘bzr push’ and ‘bzr init’ for creating a new or mirrored branch without working trees.
  (Matthew Gordon, #506730)
• `bzr config` is a new command that displays the configuration options for a given directory. It accepts a glob
to match against multiple options at once. It can also be used to set or delete a configuration option in any
configuration file. (Vincent Ladeuil)
• New shortcut URL schemes `ubuntu:` and `debianlp:` access source branches on Launchpad. E.g. `bzr branch ubuntu:foo` gives you the source branch for project `foo` in the current distrosseries for Ubuntu
while `bzr branch debianlp:lenny/foo` gives you the source branch (on Launchpad) for project `foo`
in Debian Lenny. (Barry Warsaw, #609186)
• Provide a configuration option “default_format” that controls the default format for new branches created with
`bzr init`. (Neil Martinsen-Burrell, #484101)

6.9.3  Bug Fixes

• Always set `PATH` in `start_bzr.bat` on Windows. (Matthäus G. Chajdas, #470264)
• `bzr status -r X..Y` was failing because `RevisionTree` didn’t implement `get_shelf_manager`. (John
Arbash Meinel, #662053)
• Correctly add directory contents when the name was previously added as a normal file, rather than throwing
`AttributeError: children` during `smart_add`. (Martin [gz], #251864)
• Correctly handle the --directory option for all code paths of `resolve` and `shelve`, this was previously
ignored when paths were provided as parameters. When both are provided, --directory becomes the base
directory for the other paths. (Vincent Ladeuil, #670851)
• Correctly set the Content-Type header when HTTP POSTing to comply with stricter web frameworks. (Vincent
Ladeuil, #665100)
• Don’t force `openssh` to use `protocol=2`, since that is now the default. (Neil Martinsen-Burrell, #561061)
• Fix KeyError: ‘port’ when getting the stored password for an HTTP URL. (Martin Pool, #654684)
• Make `bzr tag --quiet` really quiet. (Neil Martinsen-Burrell, #239523)
• Missing files (files `bzr add`ed and then OS deleted) are now shown in `bzr status` output. (Rory Yorke, #134168)
• `NotBranchError` no longer allows errors from calling `bzrdir.open_repository()` to propagate. This is unhelpful at best, and at worst can trigger infinite loops in callers. (Andrew Bennetts)
• The `branch.tags.merge_to(target_branch)` API used by plugins such as `bzr-builddeb` now propagates changes to the master branch of the target branch (if there is one). This makes it consistent with the other tag APIs. (Andrew Bennetts, #603395)
• Windows installers no longer requires the Microsoft vcredist to be installed. (Martin [gz], Gary van der Merwe, #632465)

6.9.4 Documentation
• Add documentation of the ability to edit hunks when shelving. (Neil Martinsen-Burrell, #517660)
• Be more specific about the meaning of revision ranges for `bzr diff`. (Neil Martinsen-Burrell, #247282)
• Document the comment character in the `.bzrignore` file, including a workaround for ignore patterns that begin with #. (Neil Martinsen-Burrell, #631515)

6.9.5 API Changes
• Add `bzrlib.pyutils` module with helper functions for some Python tasks such as resolving a dotted name to a Python object (`get_named_object`). (Andrew Bennetts)
• `bzrlib.tests.ForwardingResult` no longer exists. Use `testtools.ExtendedToOriginalDecorator` instead. (Andrew Bennetts)
• `known_hooks_key_to_parent_and_attribute` in `bzrlib.hooks` has been deprecated in favour of `known_hooks.key_to_parent_and_attribute` in the same module. (Andrew Bennetts)

6.9.6 Internals
• `tools/fixed-in.py` find a bug in NEWS from its number or a regexp matching the news entry and display the corresponding release, date, fix authors and the news entry itself. (Vincent Ladeuil)

6.9.7 Testing
• Blank output section in scriptrunner tests no longer match any output. Instead, use ‘...’ as a wildcard if you don’t care about the output. (Martin Pool, #637830)
• `bzr test-script script` is a new command that runs a shell-like script from an the `script` file. (Vincent Ladeuil)
• Fix spurious test failures on babune related to the http pipe cleanup and get rid of some ‘bytes left on the HTTP socket’ useless log messages. (Vincent Ladeuil, #655557)
• `bzrlib.tests.per_workingtree.TestCaseWithWorkingTree.make_branch_builder` respects its `relpath` parameter. (Vincent Ladeuil)
6.10 bzr 2.3b2

2.3b2 2010-10-08

6.10.1 External Compatibility Breaks

- The `bzr tags` command sorts tag names using a natural sort by default (so tag2 sorts before tag10). The previous default was strictly “asciibetical”. That behavior is still available as `bzr tags --sort=alpha`. (Neil Martinsen-Burrell, #640760)
- `BzrDir.generate_backup_name` has been deprecated and replaced by a private method. `osutils.available_backup_name` provides an extensible replacement. This allowed the deprecation of `bzrlib.transform.get_backup_name`, `bzrlib.transform._get_backup_name` and `bzrlib.transform.TreeTransformBase.has_named_child`. (Vincent Ladeuil)

6.10.2 New Features

- Add `mainline` revision specifier, which selects the revision that merged a specified revision into the mainline. (Aaron Bentley)
- Add `annotate` revision specifier, which selects the revision that introduced a specified line of a file. (Aaron Bentley)
- Add `--Dmem_dump` debug flag, which uses meliae to dump memory to a file upon an out of memory error. (Karl Bielefeldt, #551391)
- `bzr status` now displays a summary of existing shelves after the other status information. This is done using a `post_status` hook. (Parth Malwankar, #403687)
- GNU lsh is now a supported lsh client; just set BZR_SSH to ‘lsh’. Also, bzr will recognize if the ‘ssh’ comand is a symlink to lsh. (Matthew Gordon, #374700)
- The `pull` and `update` commands now take a `--show-base` option that, in the case of conflicts, shows the base revision text. (Rory Yorke, #202374)

6.10.3 Improvements

- `bzr break-lock --force` breaks the lock without prompting. (Before using this, make sure the process holding the lock really is dead.) (Martin Pool, #397315)
- `bzr remove` now takes a `--no-backup` option for when you don’t want it to backup anything, just delete it. This option used to be called `--force` which is now deprecated. (Marius Kruger, #400554)
- When using the pycurl client module, Bazaar shows some of the text from HTTP server error messages. (Martin Pool, #656667)

6.10.4 Bug Fixes

- Don’t force openssh to use protocol=2, since that is now the default. (Neil Martinsen-Burrell, #561061)
- Fix signature of RemoteBzrDir.create_workingtree to match that of its superclass. (Neil Martinsen-Burrell, Martin Pool, #524627)
- Fix traceback with python-2.7’s xmlrpclib (Toshio Kuratomi, #612096)
• Print junk rather than throwing a UnicodeDecodeError from `bzr version-info` when using the rio format (with non-ascii information) on a non-utf-8 terminal. (Andrej A Antonov, #518609)

• Treat all IO, OS, and socket errors consistently when establishing SSH/SFTP connections via a subprocess. (Andrew Bennetts)

• `unshelve --preview` now can show diff in a non-ascii encoding. (Andrej A Antonov, #518916)

• Deleting a versioned directory can leave orphans: unversioned files that were present don’t have a parent anymore. The `bzr.transform.orphan_policy` configuration option controls the `bzr` behaviour: `conflict` (the default) leave the orphans in place and create a conflict for the directory, `move` create orphans named `<file>..#~` in a `bzr-orphans` directory at the root of the working tree. (Vincent Ladeuil, #323111)

### 6.10.5 Improvements

### 6.10.6 Documentation

• Correct the documentation for setting up the smart server with Apache. (Neil Martinsen-Burrell, #311518)

• Fix rst typos in `bzrlib/help_topics/en/conflict-types.txt`. (Vincent Ladeuil, #660943)

• Provide more detailed help on the Launchpad plugin at “bzr help plugins/launchpad”. (Neil Martinsen-Burrell, #589379)

• Suggest `bzr revert` for restoring locally deleted files in help text for `bzr update`. (John C Barstow, #191466)

### 6.10.7 API Changes

• `WorkingTree` methods `pull`, `update`, and `__update_tree` now have an optional argument, `show_base`, which is by default False. This is flag is ultimately passed to `merge.merge_inner` in each case. (Rory Yorke, #202374)

### 6.10.8 Internals

• Small change to `GroupCompressBlock` to work more in terms of ‘chunks’ rather than ‘content’ for its compressed storage. (John Arbash Meinel)

• When running `bzr selftest --subunit` the subunit stream will no longer include the “log” information for tests which are considered to be ‘successes’ (success, xfail, skip, etc) (John Arbash Meinel)

### 6.10.9 Testing

• Add a new simpler way to generate multiple test variations, by setting the `scenarios` attribute of a test class to a list of scenarios descriptions, then using `load_tests_apply_scenarios`. (See the testing guide and `bzr-lib.tests.scenarios`.) Simplify `test_http` using this. (Martin Pool, #597791)

• Add `tests/ssl_certs/ca.crt` to the required test files list. Test involving the pycurl https test server fail otherwise when running selftest from an installed version. (Vincent Ladeuil, #651706)

• Fix tests that failed when run under `LANG=C`. (Andrew Bennetts, #632387)

• Skip tests that needs a `bzr` source tree when there isn’t one. This is needed to successfully run the test suite for installed versions. (Vincent Ladeuil, #644855).

• Skip the tests that requires respecting the chmod bits when running as root. (Vincent Ladeuil, #646133)
• Suppress the “maximum recursion depth exceeded in __subclasscheck__” warning on stderr emitted during test_dict_deepnested in bzrlib/tests/test__bencode.py. (Andrew Bennetts)
• Use tests.TestCaseInTempDir for tests that requires disk resources. (Vincent Ladeuil, #650001)

6.11 bzc 2.3b1

2.3b1  2010-09-20

This is the first beta of the 2.3 series, leading up to a 2.3.0 release in January or February. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins. Some plugins may need small updates to work with 2.3b1.

2.3b1 includes some performance improvements in both speed and memory consumption, some preliminary support for generating a texinfo version of the doc and better support for launchpad. Many changes were made to make our test suite more robust as well as numerous documentation fixes. It improves the common infrastructure for dealing with colocated named branches and foreign branches. We plan to continue with these themes through the 2.3 series.

It also includes bug fixes for 2.0.6, 2.1.3 and 2.2.1 and over 40 fixes of its own.

6.11.1 External Compatibility Breaks

(none)

6.11.2 New Features

• Added pre_status and post_status hooks. This allows plugins to register custom handlers which will be invoked before/after the standard status output is displayed. (Parth Malwankar)
• bzr break-lock --config [location] can now break config files locks. (Vincent Ladeuil, #525571)
• bzrlib.config.LockableConfig is a base class for config files that needs to be protected against multiple writers. All methods that change a configuration variable value must be decorated with @needs_write_lock (set_option() for example). (Vincent Ladeuil, #525571)
• The lp: prefix will now use your known username (from bzr launchpad-login) to expand ~ to your username. For example: bzr launchpad-login user && bzr push lp:~project/branch will now push to lp:~user/project/branch. (John Arbash Meinel)
• New development format development8~subtree which is similar to the 2a format and adds subtree support. (Jelmer Vernooij)

6.11.3 Improvements

• Allow using both --using and --diff-options. (Matthäus G. Chajdas, #234708)
• Allow using non-integer bug ID with generic bug trackers. (Alexandre Garnier, #440472)
• bzr remove now just backs up changed files instead of exiting, forcing you to choose to either keep or delete them. Bazaar will now delete the files if they can easily be recovered using revert, otherwise they will be backed up (adding an extension of the form .~#~). (Marius Kruger, #400554)
• bzr revert and bzr status are up to 15% faster on large trees with many changes by not repeatedly building a list of all file-ids. (Andrew Bennetts)
• Decrease memory consumption when many chk index pages are loaded. (Such as during `bzr co` or `bzr ls -R` of a large tree.) Often we need to read many chk pages because the individual chk map nodes will be spread randomly. Peak memory for `bzr ls -R` on a large tree dropped from 396MB down to 247MB, expect even more significant savings on 64-bit platforms. (John Arbash Meinel)

• Decrease peak memory during `bzr send`. The old code was caching all text content and all inventory strings for all revisions before computing the diffs. Now we only cache as long as there is a child that will need them. Sending 2000 bzr revisions drops from 1.2GB peak to 256MB peak. (John Arbash Meinel, #614576)

• `DirState` internals use a little bit less memory. For bzr.dev it drops the memory from 1MB down to about 800kB. And replaces a few thousand tuples and sets with StaticTuple. (John Arbash Meinel)

• Inventory entries now consume less memory (on 32-bit Ubuntu file entries have dropped from 68 bytes to 40, and directory entries from 120 bytes to 48). (Andrew Bennett)

• Reduce peak memory by one copy of compressed text when writing to pack repositories. (John Arbash Meinel, #566940)

• When building new working trees, default to reading from the repository rather than the source tree unless explicitly requested. (via `--files-from` and `--hardlink` for `bzr branch` and `bzr checkout`. Generally, 2a format repositories extract content faster than seeking and reading content from another tree, especially in cold-cache situations. (John Arbash Meinel, #607298)

• Add `__pycache__` to the default ignores file. Future releases of Python will use this directory to store bytecodes. (Andrea Corbellini, #626687)

### 6.11.4 Bug Fixes

• Additional merges after an unrelated branch has been merged with its history no longer crash when deleted files are involved. (Vincent Ladeuil, John Arbash Meinel, #375898)

• `bzr add SYMLINK/FILE` now works properly when the symlink points to a previously-unversioned directory within the tree: the directory is marked versioned too. (Martin Pool, #192859)

• `bzr clean-tree` issues a warning if it is unable to delete a file due to `errno.EACCES` instead of exiting with an error on Windows. (Parth Malwankar, #430785)

• `bzr commit SYMLINK` now works, rather than trying to commit the target of the symlink. (Martin Pool, John Arbash Meinel, #128562)

• `bzr ignore PATTERNS` exits with error if a bad pattern is supplied. `InvalidPattern` exception error message now shows faulting regular expression. (Parth Malwankar #300062)

• `bzr upgrade` now creates the `backup.bzr` directory with the same permissions as `.bzr` directory on a POSIX OS. (Parth Malwankar, #262450)

• CommitBuilder now uses the committer instead of `_config.username` to generate the revision-id. (Aaron Bentley, #614404)

• Configuration files in `$BZR_HOME` are now written in an atomic way which should help avoid problems with concurrent writers. (Vincent Ladeuil, #525571)

• CommitBuilder now uses the committer instead of `_config.username` to generate the revision-id. (Aaron Bentley, #614404)

• Configuration files in `$BZR_HOME` are now protected against concurrent writers by using a lock. (Vincent Ladeuil, #525571)

• Cope with Microsoft FTP Server and VSFTPD that return reply ‘250 Directory created’ when mkdir succeeds. (Martin Pool, #224373)
• **decode** parameter to get() method in FtpTransport and GioTransport classes is deprecated. (Alexander Belchenko)

• Don’t print internal object name when print an invalid revision spec error. (Neil Martinsen-Burrell, #598701)

• Don’t traceback when a lockdir’s **held/info** file is corrupt (e.g. contains only NUL bytes). Instead warn the user, and allow **bzr break-lock** to remove it. (Andrew Bennetts, #619872)

• EPIPE can be raised during test server shutdown. This happened on gentoo only so far. (Vincent Ladeuil, #627277)

• Errors occurring during HTTP(S) test server starts should now be handled cleanly. (Vincent Ladeuil, #392402)

• Fix **AttributeError** on **parent.children** when adding a file under a directory that was a symlink in the previous commit. (Martin Pool, #192859)

• Fix **AttributeError: ‘NoneType’ object has no attribute ‘close’** in **_close_ssh_proc** when using **bzr+ssh://**. This was causing connections to pre-1.6 bzr+ssh servers to fail, and causing warnings on stderr in some other circumstances. (Andrew Bennetts, #633745)

• Fix a problem in bzr’s **Makefile** that meant syntax errors in some parts of bzr’s source code could cause **make check-nodocs** to incorrectly return an exit code of 0. (Vincent Ladeuil, #626667)

• **HTTP/1.1** test servers now set a **Content-Length** header to comply with pedantic **HTTP/1.1** clients. (Vincent Ladeuil, #568421)

• Most of the leaked threads during selftest are now fixed, allowing the full test suite to pass on gentoo. (Vincent Ladeuil, #392127)

• Only call **setlocale** in the bzr startup script on posix systems. This avoids an issue with the newer windows C runtimes used by Python 2.6 and later which can mangle bytestrings printed to the console. (Martin [gz], #631350)

• **PathNotChild** should not give a traceback. (Martin Pool, #98735)

• Prevent **CHKMap.apply_delta** from generating non-canonical CHK maps, which can result in “missing referenced chk root keys” errors when fetching from repositories with affected revisions. (Andrew Bennetts, #522637)

• Raise **ValueError** instead of a string exception. (John Arbash Meinel, #586926)

• Repositories accessed via a smart server now reject being stacked on a repository in an incompatible format, as is the case when accessing them via other methods. This was causing fetches from those repositories via a smart server (e.g. using **bzr branch**) to receive invalid data. (Andrew Bennetts, #562380)

• Selftest with versions of subunit that support **stopTestRun** will no longer error. This error was caused by 2.0 not being updated when upstream python merged the end of run patch, which chose **stopTestRun** rather than **done**. (Robert Collins, #571437)

• Stop **AttributeError: ‘module’ object has no attribute ‘ElementTree’** being thrown from **xml_serializer** on certain cElementTree setups. (Martin [gz], #254278)

• strace test-helper tests cope with the new Ubuntu policy of not allowing users to attach to their own processes by default. (Martin Pool, #626679)
• Test classes like `TestCase`, `TestLoader`, and `TestSuite` should be available from `bzrlib.tests.*`. They used to be, but were accidentally removed. (John Arbash Meinel, #627438)

• `Transport.stat` on a symlink, including a transport pointing directly to a symlink, now returns information about the symlink. (Martin Pool)

• Upgrading or fetching from a non-rich-root repository to a rich-root repository (e.g. from pack-0.92 to 2a) no longer fails with ‘`Interland2Helper` object has no attribute ‘source_repos’. This was a regression from Bazaar 2.1. (Andrew Bennetts, #636930)

### 6.11.5 Documentation

• Added a builder/writer sphinx extension that can generate texinfo files. The generated files are syntactically correct but the info navigation nodes needs more work. (Vincent Ladeuil, #219334)

• First tests defined for sphinx, including a new `bzrlib.tests.features.sphinx` to make the tests conditional. (Vincent Ladeuil)

• Fix a lot of references in the docs to the old [http://bazaar-vcs.org](http://bazaar-vcs.org) to the new [http://bazaar.canonical.com](http://bazaar.canonical.com) or [http://wiki.bazaar.canonical.com](http://wiki.bazaar.canonical.com) (John Arbash Meinel, #617503)

### 6.11.6 API Changes

• When passing a file to `UTF8DirReader` make sure to close the current directory file handle after the chdir fails. Otherwise when passing many filenames into a command line `bzz status` we would leak descriptors. (John Arbash Meinel, #583486)

• `ControlDirFormat` and `ControlDir` have been split out of `BzrDirFormat` and `BzrDir`, respectively. `ControlDirFormat` and `ControlDir` should be used as the base classes for new non-.bzr implementations.

  `BzrDirFormat.register_control_format` has been renamed to `ControlDirFormat.register_format`.

  `BzrDirFormat.register_server_control_format` has been removed.

  Probing for control directories is now done by separate objects derived from `bzrlib.controldir.Prober` and registered using `bzrlib.controldir.ControlDirFormat.register_prober` or `bzrlib.controldir.ControlDirFormat.register_server_prober`. `BzrDirFormat.probe_transport` has been moved onto `Prober`.

  `BzrDirFormat.register_format` has been renamed to `BzrProber.register_bzrdir_format`.

  `bzrlib.bzdir.network_format_registry` has been moved to `bzrlib.controldir`.

  (Jelmer Vernooij)

• `bzrlib.transform.TreeTransformBase.final_kind`, `bzrlib.transform.TreeTransform.tree_kind` and `bzrlib.transform.TransformPreview.tree_kind` now return `None` instead of raising `NoSuchFile`. (Vincent Ladeuil)

• `BzrError` subclasses no longer support the name “message” to be used as an argument for `__init__` or in `_fmt` format specification as this breaks in some Python versions. errors.LockError.__init__ argument is now named “msg” instead of earlier “message”. (Parth Malwankar, #603461)

• Configuration files should now use the `from_string` constructor rather than the `file` parameter of the `__get_parser` method. The later has been deprecated. `from_string` also accept a `save=True` parameter to have the configuration file immediately written to disk. (Vincent Ladeuil)

• Deprecate treating a `PushResult` and `PullResult` as an integer for the relative change in revno. (Martin Pool)

• `FileInWrongBranch` is deprecated in favour of `PathNotChild` and no longer raised. (Martin Pool)
• IniBaseConfig objects should now use the from_string constructor rather than the file parameter of the _get_parser method. The later has been deprecated. (Vincent Ladeuil)

• InventoryEntry instances now raise AttributeError if you try to assign to attributes that are irrelevant to that kind of entry. e.g. setting symlink_target on an InventoryFile will fail. It is still okay to read those attributes on any kind of InventoryEntry. The complete list of affected attributes is: executable, text_id, text_sha1, text_size (only valid for kind == file); symlink_target (only valid for kind == link); and reference_revision (only valid for kind == tree-reference). (Andrew Bennetts)

• InventoryEntry objects no longer have _put_in_tar or _put_on_disk methods. (Andrew Bennetts)

• The get_filename parameter in the config.IniBaseConfig constructor has been deprecated, use the file_name parameter instead. (Vincent Ladeuil)

• tree_files and internal_tree_files are now deprecated in favor of WorkingTree.open_containing_paths. (Martin Pool)

6.11.7 Internals

• Remove used and broken code path in BranchInitHookParams.__repr__. (Andrew Bennetts)

6.11.8 Testing

• Avoid spurious failures in ssh tests: wait for the SSH server to actually finish, rather than just waiting for it to negotiate the key exchange. (John Arbash Meinel, #626876)

• build_tree_contents can create symlinks. (Martin Pool, John Arbash Meinel)

• Catch socket errors to avoid bt.test_sftp_transport.SSHVendorBadConnection.test_bad_connection_ssh random failures. (Vincent Ladeuil, #601804)

• HTTP test servers will leak less threads (and sockets) and will not hang on AIX anymore. (Vincent Ladeuil, #405745)

• On platforms that don’t support forking give a nice error message saying so when bzr selftest --parallel=fork is used. (Martin [gz], #528730)

• Rearrange thread leak detection code to eliminate global state and make it possible to extend the reporting. (Martin [gz], #633462)

• The old bzr selftest --benchmark option has been removed. <https://launchpad.net/bzr-usertest> is an actively-maintained macrobenchmark suite. (Martin Pool)

• The test suite now simply holds log files in memory, rather than writing them out to disk and then reading them back in and deleting them. (Andrew Bennetts)

• The way bzr selftest --parallel generates N partitions of tests to run in parallel has changed. Instead of splitting the list of tests at N-1 points, it distributes the tests one-by-one into the partitions in a round robin fashion. This reduces the total time to run the tests in parallel because a series of slow tests in the test suite will be distributed evenly among the parallel test suites, rather than slowing down just one suite. (Andrew Bennetts)

• Tracebacks from a parameterized test are no longer reported against every parameterization of that test. This was done by adding a hack to bzrlib.tests.clone_test so that it no longer causes testtools.TestCase instances to share a details dict. (Andrew Bennetts, #625574)
BAZAAR RELEASE NOTES

7.1 bzr 2.2.6

2.2.6 NOT RELEASED YET

7.1.1 Compatibility Breaks
7.1.2 New Features
7.1.3 Bug Fixes
7.1.4 Improvements
7.1.5 Documentation
7.1.6 API Changes
7.1.7 Internals
7.1.8 Testing

7.2 bzr 2.2.5

2.2.5 2011-09-01
This is a bugfix release. One regression introduced in 2.2b1 has been fixed for some rare conflict resolutions. Also a warning is now emitted when branching an out-of-date ubuntu packaging branch. Upgrading is recommended for all users on earlier 2.2 releases.

7.2.1 Compatibility Breaks
None.

7.2.2 New Features
None.
7.2.3 Bug Fixes

- Correctly handle `bzr log` and `get_known_graph_ancestry` on a doubly-stacked branch. (James Westby, Martin Pool, #715000)
- Don’t crash while merging and encountering obscure path conflicts involving different root-ids. (Vincent Ladeuil, #805809)

7.2.4 Internals

- Fixed bug in the bundled copy of ConfigObj with quoting of triple quotes in the value string. Fix suggested by ConfigObj’s author Michael Foord. (Alexander Belchenko, #710410)

7.3 bzr 2.2.4

2.2.4 2011-02-04

This is a bugfix release. Only one bug has been fixed, a regression from 2.2.3 involving only certain operations with launchpad. Upgrading is recommended for all users on earlier 2.2 releases.

7.3.1 Bug Fixes

- Fix communications with the Launchpad web service when using launchpadlib >= 1.5.5. This was a latent bug in bzr’s communication with Launchpad’s production instance, which only became a problem when the default instance was switched from edge to production in bzr 2.2.3. (Max Bowsher, #707075)

7.4 bzr 2.2.3

2.2.3 2011-01-20

This is a bugfix release. Upgrading is recommended for all users on earlier 2.2 releases.

7.4.1 Compatibility Breaks

- Launchpad has announced that the `edge.launchpad.net` instance is deprecated and may be shut down in the future [http://blog.launchpad.net/general/edge-is-deprecated](http://blog.launchpad.net/general/edge-is-deprecated). Bazaar has therefore been updated in this release to talk to the main (`launchpad.net`) servers, rather than the `edge` ones. (Vincent Ladeuil, #583667)

7.4.2 Bug Fixes

- Avoid `UnicodeDecodeError` in `bzr add` with multiple files under a non-ascii path on windows from symlink support addition. (Martin [gz], #686611)
- Correctly resolve content (and path) conflicts for files in subdirs. (Vincent Ladeuil, #660935)
- Don’t probe for a repository from within `NotBranchError.__repr__`, because this can cause knock-on errors at awkward times. (Andrew Bennetts, #687653)
- Fix a crash during `RepositoryPackCollection.pack` caused by a concurrent repository pack operation. This was particularly affecting `bzr-svn` users. (Andrew Bennetts, #701940)
• **https** access works again with recent versions of python2.7. (Vincent Ladeuil, #693880)
• RevisionTree.is_executable no longer returns None for directories and symlinks. Instead, it returns False, like other Trees and methods. (Aaron Bentley, #681885)

## 7.5 bzr 2.2.2

2.2.2 2010-11-25

This is a bugfix release. None of these bugfixes are critical, but upgrading is recommended for all users on earlier 2.2 releases.

### 7.5.1 Bug Fixes

• `bzr resolve --take-other <file>` will not crash anymore if `<file>` is involved in a text conflict (but the conflict is still not resolved). (Vincent Ladeuil, #649661)
• Commit in a bound branch or heavyweight checkout now propagates tags (e.g. from a merge) to the master branch (and informs the user if there is a conflict). (Andrew Bennetts, #603395)
• Correctly set the Content-Type header when HTTP POSTing to comply with stricter web frameworks. (Vincent Ladeuil, #665100)
• **NotBranchError** no longer allows errors from calling `bzrdir.open_repository()` to propagate. This is unhelpful at best, and at worst can trigger infinite loops in callers. (Andrew Bennetts)
• Skip tests that needs a bzr source tree when there isn’t one. This is needed to succesfully run the test suite for installed versions. (Vincent Ladeuil, #644855).
• Skip the tests that requires respecting the chmod bits when running as root. Including the one that wasn’t present in 2.1. (Vincent Ladeuil, #646133)
• Using bzr with `lp`: URLs behind an HTTP proxy should work. (Robert Collins, #558343)
• Windows installers no longer requires the Microsoft vcredist to be installed. (Martin [gz], Gary van der Merwe, #632465)
• Close leaked socket to SSH subprocesses, which caused dput sftp uploads to hang. (Max Bowsher, #659590)

### 7.5.2 Testing

• Add tests/ssl_certs/ca.crt to the required test files list. Test involving the pycurl https test server fail otherwise when running selftest from an installed version. (Vincent Ladeuil, #651706)
• Fix tests that failed when run under `LANG=C`. (Andrew Bennetts, #632387)

## 7.6 bzr 2.2.1

2.2.1 2010-09-17

This is a bugfix release which also includes bugfixes from 2.0.6 and 2.1.3. None are critical, but upgrading is recommended for all users on earlier 2.2 releases.
7.6.1 Bug Fixes

- Additional merges after an unrelated branch has been merged with its history no longer crash when deleted files are involved. (Vincent Ladeuil, John Arbash Meinel, #375898)

- `bzr add SYMLINK/FILE` now works properly when the symlink points to a previously-unversioned directory within the tree: the directory is marked versioned too. (Martin Pool, #192859)

- `bzr commit SYMLINK` now works, rather than trying to commit the target of the symlink. (Martin Pool, John Arbash Meinel, #128562)

- `bzr upgrade` now creates the `backup.bzr` directory with the same permissions as `.bzr` directory on a POSIX OS. (Parth Malwankar, #262450)

- `CommitBuilder` now uses the committer instead of `_config.username` to generate the revision-id. (Aaron Bentley, #614404)

- Configuration files in `$BZR_HOME` are now written in an atomic way which should help avoid problems with concurrent writers. (Vincent Ladeuil, #525571)

- Cope with Microsoft FTP server that returns reply ‘250 Directory created’ when mkdir succeeds. (Martin Pool, #224373)

- Don’t traceback trying to unversion children files of an already unversioned directory. (Vincent Ladeuil, #494221)

- Don’t traceback when a lockdir’s `held/info` file is corrupt (e.g. contains only NUL bytes). Instead warn the user, and allow `bzr break-lock` to remove it. (Andrew Bennetts, #619872)

- Fix `AttributeError` on `parent.children` when adding a file under a directory that was a symlink in the previous commit. (Martin Pool, #192859)

- Fix `AttributeError: 'NoneType' object has no attribute 'close'` in `_close_ssh_proc` when using `bzr+ssh://`. This was causing connections to pre-1.6 `bzr+ssh` servers to fail, and causing warnings on stderr in some other circumstances. (Andrew Bennetts, #633745)

- Only call `setlocale` in the `bzr` startup script on posix systems. This avoids an issue with the newer windows C runtimes used by Python 2.6 and later which can mangle bytestrings printed to the console. (Martin [gz], #631350)

- Prevent `CHKMap.apply_delta` from generating non-canonical CHK maps, which can result in “missing referenced chk root keys” errors when fetching from repositories with affected revisions. (Andrew Bennetts, #522637)

- Raise `ValueError` instead of a string exception. (John Arbash Meinel, #586926)

- Reduce peak memory by one copy of compressed text. (John Arbash Meinel, #566940)

- Repositories accessed via a smart server now reject being stacked on a repository in an incompatible format, as is the case when accessing them via other methods. This was causing fetches from those repositories via a smart server (e.g. using `bzr branch`) to receive invalid data. (Andrew Bennetts, #562380)

- Selftest with versions of subunit that support `stopTestRun` will no longer error. This error was caused by 2.0 not being updated when upstream python merged the end of run patch, which chose `stopTestRun` rather than `done`. (Robert Collins, #571437)

- Stop `AttributeError: 'module' object has no attribute 'ElementTree'` being thrown from `xml_serializer` on certain `cElementTree` setups. (Martin [gz], #254278)

- Upgrading or fetching from a non-rich-root repository to a rich-root repository (e.g. from pack-0.92 to 2a) no longer fails with ‘`Interland2Helper` object has no attribute ‘source_repo’’. This was a regression from Bazaar 2.1. (Andrew Bennetts, #636930)
• When passing a file to `UTF8DirReader` make sure to close the current directory file handle after the chdir fails. Otherwise when passing many filenames into a command line `bzd status` we would leak descriptors. (John Arbash Meinel, #583486)

7.6.2 Documentation

• Fix a lot of references in the docs to the old `http://bazaar-vcs.org` to the new `http://bazaar.canonical.com` or `http://wiki.bazaar.canonical.com` (John Arbash Meinel, #617503)

7.6.3 Internals

• Remove used and broken code path in `BranchInitHookParams.__repr__`. (Andrew Bennetts)

7.6.4 Testing

• `build_tree_contents` can create symlinks. (Martin Pool, John Arbash Meinel)

• Tracebacks from a parameterized test are no longer reported against every parameterization of that test. This was done by adding a hack to `bzrlib.tests.clone_test` so that it no longer causes testtools.TestCase instances to share a details dict. (Andrew Bennetts, #625574)

7.7 bzr 2.2

**Codename** La Hulpe

2.2 2010-08-06

This release marks the start of another long-term-stable series. From here, we will only make bugfix releases on the 2.2 series (2.2.1, etc), while 2.3 will become our new development series. The 2.0 and 2.1 series will also continue to get bugfixes. (Currently 2.0 is planned to be supported for another 6 months.)

This is primarily a bugfix and polish release over the 2.1 series, with a large number of bugs fixed (>120), and some performance improvements.

There are some compatibility changes in this release. For users of bzrlib as a library, we now request that they call `bzrlib.initialize` and use the returned context manager appropriately. For commandline users we no longer guess user identity for `bzd commit`, users must specify their identity using `bzr whoami` (you don’t need to specify your identity for readonly operations).

Users are encouraged to upgrade from the other stable series.

7.7.1 Compatibility Breaks

• BzrError subclasses no longer support the name “message” to be used as an argument for `__init__` or in `_fmt` format specification as this breaks in some Python versions. `errors.LockError.__init__` argument is now named “msg” instead of earlier “message”. (Parth Malwankar, #603461)

• The old `bzd selftest --benchmark` option has been removed. [<https://launchpad.net/bzd-usertest>](https://launchpad.net/bzd-usertest) is an actively-maintained macrobenchmark suite. (Martin Pool)
7.7.2 Bug Fixes

- `bzr ignore PATTERNS` exits with error if a bad pattern is supplied. `InvalidPattern` exception error message now shows faulting regular expression. (Parth Malwankar #300062)
- Configuration files in `$BZR_HOME` are now written in an atomic way which should help avoid problems with concurrent writers. (Vincent Ladeuil, #525571)
- Don’t traceback trying to unversion children files of an already unversioned directory. (Vincent Ladeuil, #494221)
- HTTP/1.1 test servers now set a `Content-Length` header to comply with pedantic HTTP/1.1 clients. (Vincent Ladeuil, #568421)
- Progress bars prefer to truncate the text message rather than the counters. The spinner is shown between the network transfer indicator and the progress message. Progress bars are correctly cleared off when they finish. (Martin Pool, #611127)
- Recursive binding for checkouts is now detected by `bzr`. A clear error message is shown to the user. (Parth Malwankar, #405192)

7.7.3 Improvements

- Add `bzrlib.merge.MergeIntoMerger`, which can merge part or all of a tree, and works with unrelated branches. (Andrew Bennett)
- Add py2exe windows target `bzrw.exe`. This allow for starting a Bazaar GUI with out have a console open in the background. (Gary van der Merwe, #433781)

7.7.4 Documentation

- `bzr help patterns` now explains case insensitive patterns and points to Python regular expression documentation. (Parth Malwankar, #594386)

7.7.5 API Changes

- Delete `ProgressTask.note`, which was deprecated in 2.1.

7.7.6 Testing

- Unit test added to ensure that “message” is not uses as a format variable name in BzrError subclasses as this conflicts with some Python versions. (Parth Malwankar, #603461)

7.8 `bzr 2.2b4`

**Codename**  Monkey Magic

**2.2b4**  2010-07-10

This fourth and final beta in the 2.2 series now stabilizes the internal APIs. Plugin authors are recommended to ensure their releases are compatible, so that 2.2rc1 can be a true release candidate, containing stable and compatible plugin versions.
For users of bzrlib as a library, one of the primary changes is to request that they call `bzrlib.initialize` and use the returned context manager appropriately.

Better interaction with `bzr-loom` to make sure branching from a loom even over a smart server still yields a local loom. Not to mention lots of bugfixes over 2.2b3.

### 7.8.1 Compatibility Breaks

- bzrlib library users now need to call `__enter__` and `__exit__` on the result of `bzrlib.initialize`. This change was made when fixing the bad habit recent bzr versions have had of leaving progress bars behind on the screen. That required calling another function before exiting the program, and it made sense to provide a full context manager at the same time. (Robert Collins)

- The `bzr` front end now requires a `bzrlib.ui.ui_factory` which is a context manager in the Python 2.5 and above sense. The bzrlib base class is such a manager, but third party UI factories which do not derive from `bzrlib.ui.UIFactory` will be incompatible with the command line front end.

- URLs like `foo:bar/baz` are now always parsed as a URL with scheme “foo” and path “bar/baz”, even if `bzr` does not recognize “foo” as a known URL scheme. Previously these URLs would be treated as local paths. (Gordon Tyler)

### 7.8.2 New Features

- Support `--directory` option for a number of additional commands: conflicts, merge-directive, missing, resolve, shelve, switch, unshelve, whoami. (Martin von Gagern, #527878)

### 7.8.3 Bug Fixes

- `bzr branch` to a new repository with a default stacking policy no longer transfers the full history unnecessarily. (Andrew Bennetts, #597942)

- `bzr init` does not recursively scan directory contents anymore leading to faster init for directories with existing content. (Martin [gz], Parth Malwankar, #501307)

- `bzr log --exclude-common-ancestry` is now taken into account for linear ancentries. (Vincent Ladeuil, #575631)

- `bzr log -r branch:REMOTE` can now properly log the remote branch, rather than trying to fetch the data locally and failing because of a readonly error. (Martin von Gagern, #149270)

- `bzr pull` now works when a lp: URL is explicitly defined as the parent or pull location in locations.conf or branch.conf. (Gordon Tyler, #534787)

- `bzr reconfigure --unstacked` now works with branches accessed via a smart server. (Andrew Bennetts, #551525)

- `BzrDir.find_branches` should ignore branches with missing repositories. (Marius Kruger, Robert Collins)

- `BzrDir.find_bzrdirs` should ignore dirs that raise `PermissionDenied`. (Marius Kruger, Robert Collins)

- Ensure that wrong path specifications in `BZR_PLUGINS_AT` display proper error messages. (Vincent Ladeuil, #591215)

- Explicitly removing `--profile-imports` option from parsed command-line arguments on Windows, because `bzr` script does the same. (Alexander Belchenko, #588277)
• Fetching was slightly confused about the best code to use and was using a new code path for all branches, resulting in more lookups than necessary on old branches. (Robert Collins, #593515)

• Final fix for ‘no help for command’ issue. We now show a clean message when a command has no help, document how to set help more clearly, and test that all commands available to the test suite have help. (Robert Collins, #177500)

• Invalid patterns supplied to Globster or lazy_regex now raise InvalidPattern exception showing clear error message to the user. (Parth Malwankar #300062)

• Progress output is cleaned up when exiting. (Aaron Bentley)

• Raise ValueError instead of a string exception. (John Arbash Meinel, #586926)

• Relative imports in plugins are now handled correctly when using BZR_PLUGINS_AT. (Vincent Ladeuil, #588959)

• ScriptRunner now strips off leading indentation from test scripts, which previously caused “SyntaxError: No command for line”. (Martin Pool)

• Show unicode filenames in diff headers using terminal encoding. (Alexander Belchenko, Bug #382699) NOTE for Windows users: If user need to save diff to file then user need to change encoding of the terminal to ANSI encoding with command chcp XXX (e.g. chcp 1251 for Russian Windows).

• URL displayed for use with break-lock when smart server sees lock contention are now valid. Default timeout for lock contention retry is now 30 seconds instead of 300 seconds. (Parth Malwankar, #250451)

• walkdirs now raises a useful message when the filenames are not using the filesystem encoding. (Eric Moritz, #488519)

• Enable debugging of bzr on windows with pdb and other tools. This was broken because we call GetCommandLineW on windows. The fix adjusts the command line we get to be the same length as sys.argv. (Jason Spashett, Alexander Belchenko, #587868)

7.8.4 Improvements

• Bazaar now reads data from SSH connections more efficiently on platforms that provide the socketpair function, and when using paramiko. (Andrew Bennetts, #590637)

• Branch.copy_content_into is now a convenience method dispatching to a InterBranch multi-method. This permits bzr-loom and other plugins to intercept this even when a RemoteBranch proxy is in use. (Robert Collins, #201613)

• Branch formats can now be loaded lazily by registering a MetaDirBranchFormatFactory rather than an actual format. This will cause the named format class to be loaded only when an enumeration of formats is needed or when the format string for the object is encountered. (Robert Collins, Jelmer Vernooij)

• The encoding that bzr uses to output things other than file content can now be overridden via the output_encoding configuration option. (Martin Pool, #340394)

• Use lazy imports in bzrlib/merge.py so that plugins like news_merge do not cause modules to be loaded unnecessarily just because the plugin registers a merge hook. This improves bzr rocks time by about 25% in a default installation (with just the core plugins). (Andrew Bennetts)

7.8.5 Documentation

• Added regression tag to our tags list. (Robert Collins)

• Improved our release checklist to have a bit less churn and leave things ready-to-go for the next action (including other people doing development). (Robert Collins)
• Remove obsolete discussion of PQM in documentation about how to contribute to Bazaar. (Martin Pool, #588444)

7.8.6 API Changes

• `bzrlib.branch.InterBranch._get_branch_formats_to_test` now returns an iterable of format pairs, rather than just a single pair, permitting InterBranch objects that work with multiple permutations to be comprehensively tested. (Robert Collins)

• `bzrlib.lsprof.profile` will no longer silently generate bad threaded profiles when concurrent profile requests are made. Instead the profile requests will be serialised. Reentrant requests will now deadlock. (Robert Collins)

• `bzrlib.knit.KnitSequenceMatcher`, which has been deprecated since 2007, has been deleted. Use `PatienceSequenceMatcher` from `bzrlib.patiencediff` instead. (Andrew Bennetts)

• `bzrlib.re_compile_checked` is now deprecated. Caller should handle `bzrlib.errors.InvalidPattern` exception thrown by `re.match` in case the default error message not suitable for the use case. (Parth Malwankar)

• `bzrlib.tests.blackbox.ExternalBase` is deprecated. It provided only one method `check_output`, and we now recommend checking command output using `run_script`. (Martin Pool)

• `bzrlib.transport.ssh.SSHVendor.connect_ssh` now returns an object that implements the interface of `bzrlib.transport.ssh.SSHConnection`. Third-party implementations of `SSHVendor` may need to be updated accordingly. Similarly, any code using `SSHConnection` directly will need to be updated. (Andrew Bennetts)

• The constructor of `bzrlib.smart.medium.SmartSSHClientMedium` has changed to take an `SSHParams` instance (replacing many individual values). (Andrew Bennetts)

7.8.7 Internals

• `bzrlib.osutils.get_terminal_encoding` will now only mutter its selection when explicitly requested; this avoids many duplicate calls being logged when helpers, wrappers and older code that manually calls it are executed it is now logged deliberately by the UI setup code. (Robert Collins)

• Improved `bzrlib.urlutils` to handle lp:foo/bar URLs. (Gordon Tyler)

• `bzrlib._c_static_tuple.StaticTuple` now implements `__sizeof__`, so that `sys.getsizeof` and other memory analysis tools will report more accurate results. (Andrew Bennetts)

• The `symbol_versioning` module can now cleanup after itself - `suppress_deprecation_warnings` now returns a cleanup function. (Robert Collins)

7.8.8 Testing

• Add `bzrlib.tests.fixtures` to hold code for setting up objects to test. (Martin Pool)

• `test_import_tariff` now respects `BZR_PLUGINS_AT` and `BZR_PLUGINS_DISABLE`. (Vincent Ladeuil, #595587)
7.9  bzr 2.2b3

2.2b3  2010-05-28

This third beta in the 2.2 series brings with it all the goodness of 2.1.2 and 2.0.6 (though it preceeds 2.0.6 slightly). Of particular note for users are compatibility fixes with bzr 1.5 and below servers, a hopeful end to the EINTR errors caused by SIGWINCH interactions, a shiny new bash completion script and bzr will no longer guess at identity details - it was too unreliable in reality. Use `bzr whoami` on every new install. For developers we have some API changes which may impact plugins as well as a bunch of our regular improvements to internal clarity and test support.

7.9.1  Compatibility Breaks

- An API break has been made to the lock_write method of Branch and Repository objects; they now return `branch.BranchWriteLockResult` and `repository.RepositoryWriteLockResult` objects. This makes changing the API in future easier and permits some cleaner calling code. The lock_read method has also changed from having no defined return value to returning `LogicalLockResult` objects. (Robert Collins)

- `bzr` does not try to guess the username as `username@hostname` and requires it to be explicitly set. This can be set using `bzr whoami`. (Parth Malwankar, #549310)

- `bzrlib.commands.Command` will now raise `ValueError` during construction if there is no `__doc__` set. (Note, this will be reverted in 2.2b4) (Robert Collins)

- The source tree no longer contains a contrib/zsh/_bzr completion script. The new file contrib/zsh/README suggests alternatives. (Martin von Gagern, #560030)

7.9.2  New Features

- `bzr commit` accepts `-p` (for “patch”) as a shorter name for `--show-diff`. (Parth Malwankar, #571467)

- `bzr ignore` now supports a `--default-rules` option that displays the default ignore rules used by `bzr`. The flag `--old-default-rules` is no longer supported by `ignore`. (Parth Malwankar, #538703)

- `bzr pack` now supports a `--clean-obsolete-packs` option that can save disk space by deleting obsolete pack files created during the pack operation. (Parth Malwankar, #304320)

- New command line option `--authors` to `bzr log` allows users to select which of the apparent authors and committer should be included in the log. Defaults depend on format. (Martin von Gagern, #513322)

- Support `--directory` option for a number of additional commands: added, annotate, bind, cat, cat-revision, clean-tree, deleted, export, ignore, ignored, lookup-revision, ls, modified, nick, re-sign, unbind, unknowns. (Martin von Gagern, #527878)

- The bash_completion plugin from the bzr-bash-completion project has been merged into the tree. It provides a bash-completion command and replaces the outdated `contrib/bash/bzr` script with a version using the plugin. (Martin von Gagern, #560030)

- A new transport based on GIO (the Gnome I/O library) provides access to Samba shares, WebDAV using gio+smb and gio+dav. It is also possible to use gio for some already existing transport methods as gio+file, gio+sftp, gio+ftp. (Mattias Eriksson)

7.9.3  Bug Fixes

- Alias information shown by `bzr help` is now accurate. This was showing an internal object name for some plugin aliases. (Parth Malwankar, #584650)
• .bazaar, .bazaar/bazaar.conf and .bzr.log inherit user and group ownership from the containing directory. This allow bzr to work better with sudo. (Martin <gzlist@googlemail.com>, Parth Malwankar, #376388)

•bzr clean-tree should not delete nested bzrdirs. Required for proper support of bzr-externals and scmproj plugins. (Alexander Belchenko, bug #572098)

•bzr ignore will no longer add duplicate patterns to .bzrignore. (Gordon Tyler, #572092)

•bzr log --exclude-common-ancestry -r X..Y displays the revisions that are part of Y ancestry but not part of X ancestry (aka the graph difference). (Vincent Ladeuil, #320119)

•bzr lp-propose which was switched to use production Launchpad API servers a few commits ago has been reverted to use edge: there is a problem with using production which isn’t trivially obvious, so we’ve filed a bug to track it, and until thats fixed will be using edge. (Robert Collins, #583667)

•bzr rm should not refuse to delete directories which contained a file which has been moved elsewhere in the tree after the previous commit. (Marius Kruger, Daniel Watkins, #129880)

•bzr selftest --parallel=fork wait for its children avoiding zombies. (Vincent Ladeuil, #566670)

•bzr selftest should not use ui.note() since it’s not unicode safe. (Vincent Ladeuil, #563997)

•CommitBuilder refuses to create revisions whose trees have no root. (Aaron Bentley)

•Do not register a SIGWINCH signal handler, instead just poll for the terminal width as needed. This avoids the “Interrupted System Call” problems that occur on POSIX with all currently released versions of Python. (Andrew Bennetts, #583941)

•Don’t mention –no-strict when we just issue the warning about unclean trees. (Vincent Ladeuil, #401599)

•Fixed AssertionError when accessing smart servers running Bazaar versions before 1.6. (Andrew Bennetts, #528041)

•Improved progress bar for fetch (2a format only). Bazaar now shows an estimate of the number of records to be fetched vs actually fetched. (Parth Malwankar, #374740, #538868)

•Reduce peak memory by one copy of compressed text. (John Arbash Meinel, #566940)

•RemoteBranch.lock_write raises ReadOnlyError if called during a read lock, rather than causing an AttributeError. (Andrew Bennetts, Danilo Segan, #582781)

•Selftest was failing with testtools 0.9.3, which caused an AssertionError raised from a cleanUp to be reported as a Failure, not an Error, breaking on of our test hygiene tests. (Robert Collins, Vincent Ladeuil).

•set_user_option with a dict on remote branches no longer fails with an AttributeError. There is a new Branch.set_config_option_dict RPC to support this efficiently. (Andrew Bennetts, #430382)

•Show the filenames when a file rename fails so that the error will be more comprehensible. (Martin Pool, #491763)

•Support Pyrex 0.9.9, required changing how we handle exceptions in Pyrex. (John Arbash Meinel, #582656)

•Unicode characters in aliases are now handled correctly and do not cause UnicodeEncodeError exception. (Parth Malwankar, #529930)

•Unicode commit messages that are the same as a file name no longer cause UnicodeEncodeError. ui.text.show_warning now handles unicode messages. (Parth Malwankar, #563646)

•When passing a file to UTF8DirReader make sure to close the current directory file handle after the chdir fails. Otherwise when passing many filenames into a command line bzr status we would leak descriptors. (John Arbash Meinel, #583486)
7.9.4 Improvements

- `append_revisions_only` will now be interpreted as a boolean and a warning emitted if illegal values are used. Note that for projects that needs to maintain compatibility with previous bzr versions, only ‘True’ and ‘False’ strings must be used (previous versions of bzr will interpret all strings differing from ‘True’ (case-sensitive) as false. (Brian de Alwis, Vincent Ladeuil)

- `bzr ls` now supports short options for existing long options. -k/--kind, -i/--ignored, -u/--unknown and -0/--null. (Parth Malwankar, #181124)

- `Config.get_user_option_as_bool` will now warn if a value cannot be interpreted as a boolean. (Vincent Ladeuil)

- The all-in-one Windows installer will now be built with docstrings stripped from the library zip, reducing the size and slightly improving cold startup time. Bundled plugins are unchanged for the moment, but if adding other new plugins to an all-in-one installation, ensure they are compiled and installed with -O1 or help may not work. (Martin [gz])

7.9.5 API Changes

- Added `bzrlib.merge.PerFileMerger`, a more convenient way to write some kinds of `merge_file_content` hook functions. (Andrew Bennetts)

- `BzrDir`, `Branch`, `Repository` and `WorkingTree` now all support `user_url`, `user_transport`, `control_url` and `control_transport` members pointing respectively to the directory containing the `.bzr` control directory, and to the directory within `.bzr` used for the particular component. All of them inherit from `ControlComponent` which provides default implementations. (Martin Pool)

- Lock methods on `Tree`, `Branch` and `Repository` are now expected to return an object which can be used to unlock them. This reduces duplicate code when using cleanups. The previous ‘tokens’ returned by `Branch.lock_write` and `Repository.lock_write` are now attributes on the result of the lock_write. `repository.RepositoryWriteLockResult` and `branch.BranchWriteLockResult` document this. (Robert Collins)

- `Repository.refresh_data` may now be called in a write group on pack-based repositories. Older repositories will still raise an error in this case. Subclasses of `Repository` can still override `Repository._refresh_data`, but are now responsible for raising `bzrlib.repository.IsInWriteGroupError` if they do not support `refresh_data` during a write group. (Andrew Bennetts, #574236)

7.9.6 Internals

- `chk_map._bytes_to_text_key` is now an optimized function to extract the (file-id, revision-id) key from a CHKInventory entry. This can potentially shave 5-10% time off during a large fetch. Related to bug #562666. (John Arbash Meinel)

- `log._get_info_for_log_files` now takes an add_cleanup callable. (Robert Collins)

- `_remember_remote_is_before` no longer raises AssertionError when suboptimal network behaviour is noticed; instead it just mutters to the log file (and warns the user if they have set the `hpss` debug flag). This was causing unnecessary aborts for performance bugs that are minor at worst. (Andrew Bennetts, #528041)

- Permit `bzr` to run under `python -O0` which reduces the size of bytecode files loaded from disk. To ensure docstrings needed for help are never stripped, the prefix `__doc__` = should now be used. (Martin <gzlist@googlemail.com>)
• No longer require zlib headers to build extensions, and remove the need for separate copy of zlib library on windows. (John Arbash Meinel, Martin <gzlist@googlemail.com>, #566923)

7.9.7 Testing

• Added `bzrlib.tests.matchers` as a place to put matchers, along with our first in-tree matcher. See the module docstring for details. (Robert Collins)
• `bzr selftest --parallel=subprocess now works correctly on win32`. (Gordon Tyler, #551332)
• Workaround `Crypto.Random` check leading to spurious test failures on Lucid, FreeBSD and gentoo. (Vincent Ladeuil, #528436)
• New class `ExecutableFeature` for checking the availability of executables on the PATH. Migrated from bash_completion plugin. (Martin von Gagern)

7.10 bzr 2.2b2

2.2b2 2010-04-16
This is a somewhat early second beta of the 2.2 series, to fix a python2.4 incompatibility in the 2.2b1 release. It also includes a swag of performance, usability and correctness improvements: test feedback on all of these would be welcome.

7.10.1 New Features

• `bzr diff` now supports a `--format` option, which can be used to select alternative diff formats. (Jelmer Vernooij, #555994)

7.10.2 Bug Fixes

• `bzr dpush`, `bzr push` and `bzr send` will now issue a warning instead of failing when dirty trees are involved. The corresponding `dpush_strict`, `push_strict` and `send_strict` should be set to True explicitly to get the previous behaviour. (Vincent Ladeuil, #519319)
• `bzr export` to tar file does not fail if any parent directory contains unicode characters. This works around upstream Python bug http://bugs.python.org/issue8396. (Parth Malwankar, #413406)
• `bzr switch` does not die if a ConfigurableFileMerger is used. (Aaron Bentley, #559436)
• `bzr update` when a pending merge in the working tree has been merged into the master branch will no longer claim that old commits have become pending merges. (Robert Collins, #562079)
• `bzrlib.mutabletree.MutableTree.commit` will now support a passed in config as in previous versions of bzrlib. (Robert Collins)
• Fix glitch in the warning about unclean trees display. (Vincent Ladeuil, #562665)
• Fixed Python2.4 incompatibilities in the bzr2.2b1 source tarball. (Martin Pool)
• Help messages generated by `RegistryOption.from_kwargs` list the switches in alphabetical order, rather than in an undefined order. (Martin von Gagern, #559409)
• Make sure the `ExecutablePath` and `InterpreterPath` are set in Apport crash reports, to avoid “This problem report applies to a program which is not installed any more” error. (Martin Pool, James Westby, #528114)
• Reset `siginterrupt` flag to False every time we handle a signal installed with `set_signal_handler(..., restart_syscall=True)` (from `bzrlib.osutils`). Reduces the likelihood of "Interrupted System Call" errors compared to registering `signal.signal` directly. (Andrew Bennetts)

• When invoked with a range revision, `bzr log` doesn’t show revisions that are not part of the Y revisions ancestry anymore when invoked with `-rX..Y`. (Vincent Ladeuil, #474807)

• Properly handle `param_name` attribute for `ListOption`. (Martin von Gagern, #387117)

7.10.3 Improvements

• `bzr commit` will prompt before using a commit message that was generated by a template and not edited by the user. (Robert Collins, #530265)

• `bzr diff` read-locks the trees and branches only once, saving about 10-20ms on `bzr diff` in a bzr.dev tree. (Andrew Bennetts)

• `bzr missing` read-locks the branches only once. (Andrew Bennetts)

• `bzr pull` locks the branches and tree only once. (Andrew Bennetts)

• Index lookups in pack repositories search recently hit pack files first. In repositories with many pack files this can greatly reduce the number of files accessed, the number of bytes read, and the number of read calls. An incremental pull via plain HTTP takes half the time and bytes for a moderately large repository. (Andrew Bennetts)

• Index lookups only re-order the indexes when the hit files aren’t already first. Reduces the cost of reordering (John Arbash Meinel, #562429)

• Less code is loaded at startup. (Cold-cache start time is about 10-20% less.) (Martin Pool, #553017)

7.10.4 API Changes

• `bzrlib.diff.get_trees_and_branches_to_diff` is deprecated. Use `get_trees_and_branches_to_diff_locked` instead. (Andrew Bennetts)

• `TreeTransform.commit` supports the full set of commit parameters, and auto-determines branch nick if not supplied. (Aaron Bentley)

7.10.5 Internals

• `bzrlib.commands.Command.run_direct` is no longer needed - the pre 2.1 method of calling `run()` to perform testing or direct use via the API is now possible again. As part of this, the `_operation` attribute on `Command` is now transient and only exists for the duration of `run()`. (Robert Collins)

7.11 `bzr 2.2b1`

2.2b1 2010-04-01

This is the first beta of the 2.2 series, leading up to a 2.2.0 release in July or August. Beta releases are suitable for everyday use but may cause some incompatibilities with plugins. Some plugins may need small updates to work with 2.2b1.
2.2b1 includes some changes to make merge conflicts easier to understand and resolve. It also removes some old unnecessary code, and loads somewhat less code at startup. It starts adding a common infrastructure for dealing with colocated named branches, which can be implemented in various ways in either bzr native or foreign formats. On Ubuntu and other platforms with the apport bug-reporting library, there’s an easier path to report problems with bzr. We plan to continue with these themes through the 2.2 series.

Over thirty bugs have been fixed, including in the log command, exporting to tarballs, restarting interrupted system calls, portability of compiled extensions, making backups during upgrade, and locking on FTP.

### 7.11.1 Compatibility Breaks

- BTreeGraphIndex can now take an offset to indicate that the data starts somewhere other than then beginning of the file. (John Arbash Meinel)
- Deleted very old hidden commands versionedfile-list, weave-plan-merge, weave-merge-text. (Martin Pool)
- Repository.get_inventory_sha1() and Repository.get_revision_xml() have been removed. (Jelmer Vernooij)
- Repository.get_revision_inventory() has been removed in favor of Repository.get_inventory(). (Jelmer Vernooij)
- All test servers have been moved out of the bzrlib.transport hierarchy to bzrlib.tests.test_server except for MemoryServer, ChrootServer and PathFilteringServer. bzrlib users may encounter test failures that can be fixed by updating the related imports from bzrlib.transport.xxx to bzrlib.tests.test_server. (Vincent Ladeuil)
- BranchReferenceFormat.initialize() now takes an optional name argument as its second parameter, for consistency with the initialize() method of other formats. (Jelmer Vernooij)

### 7.11.2 New Features

- Added bzr remove-branch command that can remove a local or remote branch. (Jelmer Vernooij, #276295)
- bzr export now takes an optional argument --per-file-timestamps to set file mtimes to the last timestamp of the last revision in which they were changed rather than the current time. (Jelmer Vernooij)
- If the Apport crash-reporting tool is available, bzr crashes are now stored into the /var/crash apport spool directory, and the user is invited to report them to the developers from there, either automatically or by running apport-bug. No information is sent without specific permission from the user. (Martin Pool, #515052)
- Parsing of command lines, for example in diff --using, no longer treats backslash as an escape character on Windows. (Gordon Tyler, #392428)
- Plugins can be disabled by defining BZR_DISABLE_PLUGINS as a list of plugin names separated by ‘;’ (‘;’ on windows). (Vincent Ladeuil, #411413)
- Plugins can be loaded from arbitrary locations by defining BZR_PLUGINS_AT as a list of name@path separated by ‘;’ (‘;’ on windows). This takes precedence over BZR_PLUGIN_PATH for the specified plugins. This is targeted at plugin developers for punctual needs and not intended to replace BZR_PLUGIN_PATH. (Vincent Ladeuil, #82693)
- Tag names can now be determined automatically by automatic_tag_name hooks on Branch if they are not specified on the command line. (Jelmer Vernooij)
• Tree-shape conflicts can be resolved by providing `--take-this` and `--take-other` to the `bzr resolve` command. Just marking the conflict as resolved is still accessible via the `--done` default action. (Vincent Ladeuil)

• Merges can be proposed on Launchpad with the new `lp-propose-merge` command. (Aaron Bentley, Jonathan Lange)

### 7.11.3 Bug Fixes

• Added docstring for `Tree.iter_changes` (John Arbash Meinel, #304182)

• Allow additional arguments to `RemoteRepository.add_inventory_by_delta()`. (Jelmer Vernooij, #532631)

• Allow exporting a single file using `bzr export`. (Michal Junák, #511987)

• Allow syscalls to automatically restart when `TextUIFactory`'s SIGWINCH handler is invoked, avoiding `EINTR` errors during blocking IO, which are often poorly handled by Python’s libraries and parts of `bzrlib`. (Andrew Bennetts, #496813)

• Avoid infinite recursion when probing for apport. (Vincent Ladeuil, #516934)

• Avoid `malloc(0)` in `patiencediff`, which is non-portable. (Martin Pool, #331095)

• Avoid truncating `svn` URLs. (Martin Pool, Martin von Gagern, #545185)

• `bzr add` will not add conflict related files unless explicitly required. (Vincent Ladeuil, #322767, #414589)

• `bzr dump-btree` now works on `*.cix` and `*.six` files. Those indices do not have reference lists, so `dump-btree` will simply show `None` instead. (Andrew Bennetts, #488607)

• `bzr help` will no longer trigger the `get_missing_command` hook when doing a topic lookup. This avoids prompting (like ‘no command plugins/loom, did you mean log?’) when getting help. In future we may trigger the hook deliberately when no help topics match from any help index. (Robert Collins, #396261)

• `bzr log -n0 -r..A.B.C` should not crash but just consider the None revspec as representing the first revision of the branch. (Vincent Ladeuil, #519862)

• `bzr remove-tree` can now remove multiple working trees. (Jared Hance, Andrew Bennetts, #253137)

• `bzr resolve --take-this` and `--take-other` now correctly renames the kept file on content conflicts where one side deleted the file. (Vincent Ladeuil, #529968)

• `bzr upgrade` now creates the `backup.bzr` directory with the same permissions as `.bzr` directory on a POSIX OS. (Parth Malwankar, #262450)

• `bzr upgrade` now names backup directory as `backup.bzr` instead of `backup.bzr`. This directory is ignored by bzr commands such as add. (Parth Malwankar, #335033, #300001)

• Cope with non-utf8 characters inside `.bzrignore`. (Jason Spashett, #183504)

• Correctly interpret “451 Rename/move failure: Directory not empty” from FTP servers while trying to take a lock. (Martin Pool, #528722)

• `DirStateRevisionTree.kind()` was returning wrong result when ‘kind’ changes occured between the workingtree and one of its parents. (Vincent Ladeuil, #535547)

• Fix `log` to better check ancestors even if merged revisions are involved. (Vincent Ladeuil, #476293)

• Loading a plugin from a given path with `BZR_PLUGINS_AT` doesn’t depend on `os.listdir()` order and is now reliable. (Vincent Ladeuil, #552922).
Many IO operations that returned EINTR were retried even if it wasn’t safe to do so via careless use of until_no_eintr. Bazaar now only retries operations that are safe to retry, and in some cases has switched to operations that can be retried (e.g. sock.send rather than sock.sendall). (Andrew Bennetts, Martin <gzlist@googlemail.com>, #496813)

Path conflicts now support –take-this and –take-other even when a deletion is involved. (Vincent Ladeuil, #531967)

Network transfer amounts and rates are now displayed in SI units according to the Ubuntu Units Policy <https://wiki.ubuntu.com/UnitsPolicy>. (Gordon Tyler, #514399)

Support kind markers for socket and fifo filesystem objects. This prevents bzr status --short from crashing when those files are present. (John Arbash Meinel, #303275)

bzr mkdir DIR will not create DIR unless DIR’s parent is a versioned directory. (Parth Malwankar, #138600)

SSH child processes will now ignore SIGQUIT on nix systems so breaking into the debugger won’t kill the session. (Martin <gzlist@googlemail.com>, #162502)

Tolerate patches with leading noise in bzr-handle-patch. (Toshio Kuratomi, Martin Pool, #502076)

update –r now supports updating to revisions that are not on mainline (i.e. it supports dotted revisions). (Parth Malwankar, #517800)

Use first apparent author not committer in GNU Changelog format. (Martin von Gagern, #513322)

7.11.4 API Changes

bzrlib.merge_directive._BaseMergeDirective has been renamed to bzrlib.merge_directive.BaseMergeDirective and is now public. (Jelmer Vernooij)

BranchFormat.initialize now takes an optional name of the colocated branch to create. (Jelmer Vernooij)

BzrDir.get_branch_transport now takes an optional name of the colocated branch to open. (Jelmer Vernooij)

Added bzrlib.osutils.set_signal_handler, a convenience function that can set a signal handler and call signal.siginterrupt(signum, False) for it, if the platform and Python version supports it. (Andrew Bennetts, #496813)

New bzrlib.initialize is recommended for programs using bzrlib to run when starting up; it sets up several things that previously needed to be done separately. (Martin Pool, #507710)

Exporters now support a per_file_timestamps argument to write out the timestamp of the commit in which a file revision was introduced. (Jelmer Vernooij)

New method BzrDir.list_branches() that returns a sequence of branches present in a control directory. (Jelmer Vernooij)

New method Repository.get_known_graph_ancestry(). (Jelmer Vernooij, #495502)

New transport methods readlink, symlink and hardlink. (Neil Santos)

Remove unused CommandFailed exception. (Martin Pool)

7.11.5 Internals

bzrlib.branchbuilder.BranchBuilder.build_snapshot now accepts a message_callback in the same way that commit does. (Robert Collins)
• `bzrlib.builtins.Commit.run` raises `bzrlib.errors.BoundBranchOutOfDate` rather than `bzrlib.errors.BzrCommandError` when the bound branch is out of date. (Gary van der Merwe)

• `bzrlib.commands.run_bzr` is more extensible: callers can supply the functions to load or disable plugins if they wish to use a different plugin mechanism; the `--help`, `--version` and no-command name code paths now use the generic pluggable command lookup infrastructure. (Robert Collins)

• `bzrlib.errors.BoundBranchOutOfDate` has a new field `extra_help` which can be set to add extra help to the error. (Gary van der Merwe)

• New method `Branch.automatic_tag_name` that can be used to find the tag name for a particular revision automatically. (Jelmer Vernooij)

• The methods `BzrDir.create_branch()`, `BzrDir.destroy_branch()` and `BzrDir.open_branch()` now take an optional `name` argument. (Jelmer Vernooij)

### 7.11.6 Testing

• `bzr` now has a `.testr.conf` file in its source tree configured appropriately for running tests with Testrepository ([https://launchpad.net/testrepository](https://launchpad.net/testrepository)). (Robert Collins)

• Documentation about testing with `subunit` has been tweaked. (Robert Collins)

• Known failures has been added for `resolve --take-other` on `ParentLoop` conflicts. This reflects bug #537956 without fixing it. (Vincent Ladeuil)

• New `bzrlib.tests.test_import_tariff` can make assertions about what Python modules are loaded, to guard against startup time or library dependency regressions. (Martin Pool)

• PQM will now run with subunit output. To analyze a PQM error use `tribunal`, or `cat log | subunit-filter | subunit2pyunit`. (Robert Collins)

• Stop sending `apport` crash files to `.cache` in the directory from which `bzr selftest` was run. (Martin Pool, #422350)

• Tests no longer fail if “`close()` called during concurrent operation on the same file object” occurs when closing the log file (which can happen if a thread tries to write to the log file at the wrong moment). An warning will be written to `stderr` when this happens, and another warning will be written if the log file could not be closed after retrying 100 times. (Andrew Bennetts, #531746)
CHAPTER EIGHT

BAZAAR RELEASE NOTES

8.1 bzr 2.1.5

2.1.5 NOT RELEASED YET

8.1.1 Compatibility Breaks

8.1.2 New Features

8.1.3 Bug Fixes

- Accessing a packaging branch on Launchpad (eg, lp:ubuntu/bzr) now checks to see if the most recent published source package version for that project is present in the branch tags. This should help developers trust whether the packaging branch is up-to-date and can be used for new changes. The level of verbosity is controlled by the config item launchpad.packaging_verbosity. It can be set to one of
  off disable all checks
  minimal only display if the branch is out-of-date
  short also display single-line up-to-date and missing,
  all (default) display multi-line content for all states

(John Arbash Meinel, #609187, #812928)

8.1.4 Improvements

8.1.5 Documentation

8.1.6 API Changes

8.1.7 Internals

8.1.8 Testing

8.2 bzr 2.1.4

2.1.4 2011-05-16
The fourth release in our 2.1 series addresses some user-inconvenience bugs. None are critical, but upgrading is recommended for all users on earlier 2.1 releases.

### 8.2.1 Compatibility Breaks

- Launchpad has announced that the edge.launchpad.net instance is deprecated and may be shut down in the future [http://blog.launchpad.net/general/edge-is-deprecated](http://blog.launchpad.net/general/edge-is-deprecated). Bazaar has therefore been updated in this release to talk to the main (launchpad.net) servers, rather than the edge ones. (Vincent Ladeuil, #583667)

### 8.2.2 New Features

None.

### 8.2.3 Bug Fixes

- Avoid UnicodeDecodeError in bzr add with multiple files under a non-ascii path on windows from symlink support addition. (Martin [gz], #686611)
- Skip tests that needs a bzr source tree when there isn’t one. This is needed to succesfully run the test suite for installed versions. (Vincent Ladeuil, #644855).
- Skip the tests that requires respecting the chmod bits when running as root. (Vincent Ladeuil, #646133)
- Using bzr with *lp*: URLs behind an HTTP proxy should work. (Robert Collins, #558343)

### 8.2.4 Improvements

### 8.2.5 Documentation

### 8.2.6 API Changes

### 8.2.7 Internals

### 8.2.8 Testing

## 8.3 bzr 2.1.3

**Codename** Do run run

2.1.3 2010-09-17

The third release in our 2.1 series addresses several user-inconvenience bugs (and includes the fixes done in 2.0.6). None are critical, but upgrading is recommended for all users on earlier 2.1 releases.

### 8.3.1 Bug Fixes

- Additional merges after an unrelated branch has been merged with its history no longer crash when deleted files are involved. (Vincent Ladeuil, John Arbash Meinel, #375898)
- bzr add SYMLINK/FILE now works properly when the symlink points to a previously-unversioned directory within the tree: the directory is marked versioned too. (Martin Pool, #192859)
• **bzr commit SYMLINK** now works, rather than trying to commit the target of the symlink. (Martin Pool, John Arbash Meinel, #128562)

• **bzr upgrade** now creates the *backup.bzr* directory with the same permissions as .bzr directory on a POSIX OS. (Parth Malwankar, #262450)

• Configuration files in ${BZR_HOME} are now written in an atomic way which should help avoid problems with concurrent writers. (Vincent Ladeuil, #525571)

• Don’t traceback trying to unversion children files of an already unversioned directory. (Vincent Ladeuil, #494221)

• Don’t traceback when a lockdir’s held/info file is corrupt (e.g. contains only NUL bytes). Instead warn the user, and allow *bzr break-lock* to remove it. (Andrew Bennetts, #619872)

• Fix **AttributeError** on parent.children when adding a file under a directory that was a symlink in the previous commit. (Martin Pool, #192859)

• Prevent **CHKMap.apply_delta** from generating non-canonical CHK maps, which can result in “missing referenced chk root keys” errors when fetching from repositories with affected revisions. (Andrew Bennetts, #522637)

• Raise **ValueError** instead of a string exception. (John Arbash Meinel, #586926)

• Reduce peak memory by one copy of compressed text. (John Arbash Meinel, #566940)

• Repositories accessed via a smart server now reject being stacked on a repository in an incompatible format, as is the case when accessing them via other methods. This was causing fetches from those repositories via a smart server (e.g. using *bzr branch*) to receive invalid data. (Andrew Bennetts, #562380)

• Selftest with versions of subunit that support stopTestRun will no longer error. This error was caused by 2.0 not being updated when upstream python merged the end of run patch, which chose stopTestRun rather than done. (Robert Collins, #571437)

• Stop **AttributeError**: 'module' object has no attribute 'ElementTree' being thrown from xml_serializer on certain cElementTree setups. (Martin [gz], #254278)

• When passing a file to **UTF8DirReader** make sure to close the current directory file handle after the chdir fails. Otherwise when passing many filenames into a command line *bzr status* we would leak descriptors. (John Arbash Meinel, #583486)

### 8.3.2 Testing

• **build_tree_contents** can create symlinks. (Martin Pool, John Arbash Meinel)

### 8.4 bzr 2.1.2

#### 2.1.2 2010-05-28

This release fixes two critical networking issues with older servers and with interrupted system call errors when pushing or pulling. We recommend upgrading to anyone running a 2.1.x version of bzr.

### 8.4.1 Bug Fixes

• **bzr clean-tree** should not delete nested bzrdirs. Required for proper support of bzr-externals and scmproj plugins. (Alexander Belchenko, bug #572098)

• **bzr switch** does not die if a ConfigurableFileMerger is used. (Aaron Bentley, #559436)
• Do not register a SIGWINCH signal handler, instead just poll for the terminal width as needed. This avoids the “Interrupted System Call” problems that occur on POSIX with all currently released versions of Python. (Andrew Bennetts, #583941)

• Fixed AssertionError when accessing smart servers running Bazaar versions before 1.6. (Andrew Bennetts, #528041)

• Reset siginterrupt flag to False every time we handle a signal installed with set_signal_handler(..., restart_syscall=True) (from bzrlib.osutils. Reduces the likelihood of “Interrupted System Call” errors compared to registering signal.signal directly. (Andrew Bennetts)

• Reduce peak memory by one copy of compressed text. (John Arbash Meinel, #566940)

8.4.2 Internals

• _remember_remote_is_before no longer raises AssertionError when suboptimal network behaviour is noticed; instead it just mutters to the log file (and warns the user if they have set the hpss debug flag). This was causing unnecessary aborts for performance bugs that are minor at worst. (Andrew Bennetts, #528041)

8.5 bzr 2.1.1

2.1.1 2010-03-24

This is a small bugfix release. Upgrading is recommended for anyone running 2.1.0 or earlier.

8.5.1 Bug Fixes

• Allow syscalls to automatically restart when TextUIFactory’s SIGWINCH handler is invoked, avoiding EINTR errors during blocking IO, which are often poorly handled by Python’s libraries and parts of bzrlib. (Andrew Bennetts, #496813)

• Avoid malloc(0) in patiencediff, which is non-portable. (Martin Pool, #331095)

• Fix plugin packaging on Windows. (Ian Clatworthy, #524162)

• Fix stub SFTP test server to call os.getcwdu(). (Vincent Ladeuil, #526221, #526353)

• Fixed CHM generation by moving the NEWS section template into a separate file. (Ian Clatworthy, #524184)

• Merge correctly when this_tree is not a WorkingTree. (Aaron Bentley)

• Register SIGWINCH handler only when creating a TextUIFactory; avoids problems importing bzrlib from a non-main thread. (Elliot Murphy, #521989)

• Repositories accessed via a smart server now reject being stacked on a repository in an incompatible format, as is the case when accessing them via other methods. This was causing fetches from those repositories via a smart server (e.g. using bzr branch) to receive invalid data. (Andrew Bennetts, #562380)

• Standardize the error handling when creating a new StaticTuple (problems will raise TypeError). (Matt Nordhoff, #457979)
• Warn if pyrex is too old to compile the new `SimpleSet` and `StaticTuple` extensions, rather than having the build fail randomly. (John Arbash Meinel, #449776)

### 8.5.2 Documentation

• Added a link to the Desktop Guide. (Ian Clatworthy)
• Added What’s New in Bazaar 2.1 document. (Ian Clatworthy)
• Drop Google Analytics from the core docs as they caused problems in the CHM files. (Ian Clatworthy, #502010)

### 8.5.3 API Changes

• Added `bzrlib.osutils.set_signal_handler`, a convenience function that can set a signal handler and call `signal.siginterrupt(signum, False)` for it, if the platform and Python version supports it. (Andrew Bennetts, #496813)

### 8.6 bzr 2.1.0

**Codename** Strasbourg

**2.1.0** 2010-02-11

This release marks our second long-term-stable series. The Bazaar team has decided that we will continue to make bugfix-only 2.0.x and 2.1.x releases, along with 2.2 development releases.

This is a fairly incremental update, focusing on polish and bugfixing. There are no changes for supported disk formats. Key updates include reduced memory consumption for many operations, a new per-file merge hook, ignore patterns can now include ‘!’ to exclude files, globbing support for all commands on Windows, and support for addressing home directories via `bzr+ssh://host/~` syntax.

Users are encouraged to upgrade from the 2.0 stable series.

### 8.6.1 Bug Fixes

• Don’t require testtools to use SFTP. (Vincent Ladeuil, #516183)
• Fix "AttributeError in Interland2Helper" during fetch. (Martin Pool, #513432)
• `bzr update` performs the two merges in a more logical order and will stop when it encounters conflicts. (Gerard Krol, #113809)
• Give a better error message when doing `bzr bind` in an already bound branch. (Neil Martinsen-Burrell, #513063)
• Ignore `KeyError` from `remove_index` during `_abort_write_group` in a pack repository, which can happen harmlessly if the abort occurs during finishing the write group. Also use `bzrlib.cleanup` so that any other errors that occur while aborting the individual packs won’t be hidden by secondary failures when removing the corresponding indices. (Andrew Bennetts, #423015)
• Set the mtime of files exported to a directory by `bzr export` all to the same value to avoid confusing `make` and other date-based build systems. (Robert Collins, #515631)
8.6.2 Improvements

- Fetching into experimental formats will now print a warning. (Jelmer Vernooij)

8.6.3 API Changes

- `Repository.deserialise_inventory` has been renamed to `Repository._deserialise_inventory` to indicate it is private. (Jelmer Vernooij)
- `Repository.get_inventory_xml` has been renamed to `Repository._get_inventory_xml` to indicate it is private. (Jelmer Vernooij)
- `Repository.serialise_inventory` has been renamed to `Repository._serialise_inventory` to indicate it is private.
- Using the `bzrlib.chk_map` module from within multiple threads at the same time was broken due to race conditions with a module level page cache. This shows up as a `KeyError` in the `bzrlib.lru_cache` code with `bzrlib.chk_map` in the backtrace, and can be triggered without using the same high level objects such as `bzrlib.repository.Repository` from different threads. `chk_map` now uses a thread local cache which may increase memory pressure on processes using threads. (Robert Collins, John Arbash Meinel, #514090)
- The new `merge_file_content` should now be ok with tests to avoid regressions. (Vincent Ladeuil, #515597)

8.6.4 Internals

- Use `bzrlib.cleanup` rather than less robust `try/finally` blocks in several places in `bzrlib.merge`. This avoids masking prior errors when errors like `ImmortalPendingDeletion` occur during cleanup in `do_merge`. (Andrew Bennetts, #517275)

8.6.5 API Changes

- The `remove_index` method of `bzrlib.repofmt.pack_repo.AggregateIndex` no longer takes a `pack` argument. This argument was always ignored. (Andrew Bennetts, #423015)

8.7 bzr 2.1.0rc2

**Codename** after the bubbles

2.1.0rc2 2010-01-29

This is a quick-turn-around to update a small issue with our new per-file merge hook. We expect no major changes from this to the final 2.1.0.

8.7.1 API Changes

- The new `merge_file_content` hook point has been altered to provide a better API where state for extensions can be stored rather than the too-simple function based approach. This fixes a performance regression where branch configuration would be parsed per-file during merge. As part of this the included `news_merger` has been refactored into a base helper class `bzrlib.merge.ConfigurableFileMerger`. (Robert Collins, John Arbash Meinel, #513822)
This is the first stable release candidate for Bazaar's 2.1 series. From this point onwards, the 2.1 series will be considered stable (as the 2.0 series) and only bugfixes are expected to be incorporated. The dozen or so bugfixes in the 2.0.4 release are also included in this release (along with more than 15 more bugfixes). Some of the interesting features are support for per-file merge hooks, `bzr unshelve --preview`, support for using ! in ignore files to exclude files from being ignored, a small memory leak was squashed, and many `ObjectNotLocked` errors were fixed. This looks to be a very good start for a new stable series.

**8.8.1 New Features**

- Add bug information to log output when available. (Neil Martinsen-Burrell, Guillermo Gonzalez, #251729)
- Added `merge_file_content` hook point to `Merger`, allowing plugins to register custom merge logic, e.g. to provide smarter merging for particular files.
- Bazaar now includes the `news_merge` plugin. It is disabled by default, to enable it add a `news_merge_files` option to your configuration. Consult `bzr help news_merge` for more information. (Andrew Bennetts)
- `bzr branch` now takes a `--bind` option. This lets you branch and bind all in one command. (Ian Clatworthy)
- `bzr switch` now takes a `--revision` option, to allow switching to a specific revision of a branch. (Daniel Watkins, #183559)
- `bzr unshelve --preview` can now be used to show how a patch on the shelf would be applied to the working tree. (Guilherme Salgado, #308122)
- `bzr update` now takes a `--revision` argument. This lets you change the revision of the working tree to any revision in the ancestry of the current or master branch. (Matthieu Moy, Mark Hammond, Martin Pool, #45719)
- `-Dbytes` can now be used to display the total number of bytes transferred for the current command. This information is always logged to `.bzr.log` for later inspection. (John Arbash Meinel)
- New ignore patterns. Patterns prefixed with ‘!’ are exceptions to ignore patterns and take precedence over regular ignores. Such exceptions are used to specify files that should be versioned which would otherwise be ignored. Patterns prefixed with ‘!!’ act as regular ignore patterns, but have highest precedence, even over the ‘!’ exception patterns. (John Whitley, #428031)
- The `supress_warnings` configuration option has been introduced to disable various warnings (it currently only supports the `format_deprecation` warning). The new option can be set in any of the following locations: `bazaar.conf`, `locations.conf` and/or `branch.conf`. (Ted Gould, Matthew Fuller, Vincent Ladeuil)

**8.8.2 Bug Fixes**

- Always show a message if an OS error occurs while trying to run a user-specified commit message editor. (Martin Pool, #504842)
- `bzr diff` will now use the epoch when it is unable to determine the timestamp of a file, if the revision it was introduced in is a ghost. (Jelmer Vernooij, #295611)
• `bzr switch -b` can now create branches that are located using directory services such as lp:, even when the branch name doesn’t contain a ‘/’. (Neil Martinsen-Burrell, #495263)

• `bzr unshelve` has improved messages about what it is doing. (Neil Martinsen-Burrell, #496917)

• Concurrent autopacking is more resilient to already-renamed pack files. If we find that a file we are about to obsolete is already obsoleted, we do not try to rename it, and we leave the file in `obsolete Packs`. The code is also fault tolerant if a file goes missing, assuming that another process already removed the file. (John Arbash Meinel, Gareth White, #507557)

• Fix “Too many concurrent requests” in reconcile when network connection fails. (Andrew Bennetts, #503878)

• Fixed a side effect mutation of `RemoteBzrDirFormat._network_name` that caused some tests to fail when run in a non-default order. Probably no user impact. (Martin Pool, #504102)

• Fixed `ObjectNotLocked` error in `bzr cat -rbranch:../foo FILE`. (Andrew Bennetts, #506274)

• FTP transports support Unicode paths by encoding/decoding them as utf8. (Vincent Ladeuil, #472161)

• Listen to the `SIGWINCH` signal to update the terminal width. (Vincent Ladeuil, #316357)

• Progress bars are now hidden when `--quiet` is given. (Martin Pool, #320035)

• `SilentUIFactory` now supports `make_output_stream` and discards whatever is written to it. This un-breaks some plugin tests that depended on this behaviour. (Martin Pool, #499757)

• When operations update the working tree, all affected files should end up with the same mtime. (eg. when versioning a generated file, if you update the source and the generated file together, the generated file should appear up-to-date.) (John Arbash Meinel, Martin <gzlist>, #488724)

### 8.8.3 Improvements

• Added `add_cleanup` and `cleanup_now` to `bzrlib.command.Command`. All the built-in commands now use `add_cleanup` rather than `try/finally` blocks where applicable as it is simpler and more robust. (Andrew Bennetts)

• All except a small number of storage formats are now hidden, making the help for numerous commands far more digestible. (Ian Clatworthy)

• Attempts to open a shared repository as a branch (e.g. `bzr branch path/to/repo`) will now include “location is a repository” as a hint in the error message. (Brian de Alwis, Andrew Bennetts, #440952)

• Push will now inform the user when they are trying to push to a foreign VCS for which roundtripping is not supported, and will suggest them to use dpush. (Jelmer Vernooij)

• The version of `bzr` being run is now written to the log file. (__monty__, #257170)

• Transport network activity indicator is shown more of the time when Bazaar is doing network IO. (Martin Pool)

### 8.8.4 Documentation

• Add documentation on creating merges with more than one parent. (Neil Martinsen-Burrell, #481526)

• Better explain the `--uncommitted` option of `merge`. (Neil Martinsen-Burrell, #505088)

• Improve discussion of pending merges in the documentation for `revert`. (Neil Martinsen-Burrell, #505093)

• Improved help for `bzr send`. (Martin Pool, Bojan Nikolic)

• There is a System Administrator’s Guide in `doc/en/admin-guide`, including discussions of installation, relevant plugins, security and backup. (Neil Martinsen-Burrell)
• The conflicts help topic has been renamed to conflict-types. (Ian Clatworthy)

• The User Reference is now presented as a series of topics. Many of the included topics have link and format tweaks applied. (Ian Clatworthy)

### 8.8.5 API Changes

• Added cachedproperty decorator to bzrlib.decorators. (Andrew Bennetts)

• Many test features were renamed from FooFeature to foo_feature to be consistent with instances being lower case and classes being CamelCase. For the features that were more likely to be used, we added a deprecation thunk, but not all. (John Arbash Meinel)

• Merger classes (such as Merge3Merger) now expect a this_branch parameter in their constructors, and provide this_branch as an attribute. (Andrew Bennetts)

• The Branch hooks pre_change_branch_tip no longer masks exceptions raised by plugins - the original exceptions are now preserved. (Robert Collins)

• The Transport Server.tearDown method is now renamed to stop_server and setUp to start_server for consistency with our normal naming pattern, and to avoid confusion with Python’s TestCase.tearDown. (Martin Pool)

• WorkingTree.update implementations must now accept a revision parameter.

### 8.8.6 Internals

• Added BzrDir.open_branchV3 smart server request, which can receive a string of details (such as “location is a repository”) as part of a nobranch response. (Andrew Bennetts, #440952)

• New helper osutils.UnicodeOrBytesToBytesWriter which encodes unicode objects but passes str objects straight through. This is used for selftest but may be useful for diff and other operations that generate mixed output. (Robert Collins)

• New exception NoRoundtrippingSupport, for use by foreign branch plugins. (Jelmer Vernooij)

### 8.8.7 Testing

• bzrlib.tests.permute_for_extension is a helper that simplifies running all tests in the current module, once against a pure python implementation, and once against an extension (pyrex/C) implementation. It can be used to dramatically simplify the implementation of load_tests. (John Arbash Meinel)

• bzrlib.tests.TestCase now subclasses testtools.testcase.TestCase. This permits features in testtools such as getUniqueInteger and getUniqueString to be used. Because of this, testtools version 0.9.2 or newer is now a dependency to run bzr selftest. Running with versions of testtools less than 0.9.2 will cause bzr to error while loading the test suite. (Robert Collins)

• Shell-like tests now support the command “mv” for moving files. The syntax for mv file1 file2, mv dir1 dir2 and mv file dir is supported. (Neil Martinsen-Burrell)

• The test progress bar no longer distinguishes tests that ‘errored’ from tests that ‘failed’ - they’re all just failures. (Martin Pool)
8.9  `bzr 2.1.0b4`

**Codename**  san francisco airport

**2.1.0b4**  2009-12-14

The fourth beta release in the 2.1 series brings with it a significant number of bugfixes (~20). The test suite is once again (finally) “green” on Windows, and should remain that way for future releases. There are a few performance related updates (faster upgrade and log), and several UI tweaks. There has also been a significant number of tweaks to the runtime documentation. 2.1.0b4 include everything from the 2.0.3 release.

8.9.1 Compatibility Breaks

- The BZR_SSH environmental variable may now be set to the path of a secure shell client. If currently set to the value ssh it will now guess the vendor of the program with that name, to restore the old behaviour that indicated the SSH Corporation client use sshcorp instead as the magic string. (Martin <gzlist@googlemail.com>, #176292)

8.9.2 New Features

- `bzr commit` now has a `--commit-time` option. (Alexander Sack, #459276)
- `-Dhpss` now increases logging done when run on the bzr server, similarly to how it works on the client. (John Arbash Meinel)
- New option `bzr unshelve --keep` applies the changes and leaves them on the shelf. (Martin Pool, Oscar Fuentes, #492091)
- The BZR_COLUMNS environment variable can be set to force bzr to respect a given terminal width. This can be useful when output is redirected or in obscure cases where the default value is not appropriate. Pagers can use it to get a better control of the line lengths. (Vincent Ladeuil)
- The new command `bzr lp-mirror` will request that Launchpad update its mirror of a local branch. This command will only function if launchpadlib is installed. (Jonathan Lange)

8.9.3 Bug Fixes

- After renaming a file, the dirstate could accidentally reference `source\\path` rather than `source/path` on Windows. This might be a source of some dirstate-related failures. (John Arbash Meinel)
- `bzr commit` now detects commit messages that looks like file names and issues a warning. (Gioele Barabucci, #73073)
- `bzr ignore /` no longer causes an IndexError. (Gorden Tyler, #456036)
- `bzr log -n0 -rN` should not return revisions beyond its merged revisions. (#325618, #484109, Marius Kruger)
- `bzr merge --weave` and `--lca` will now create .BASE files for files with conflicts (similar to `--merge3`). The contents of the file is a synthesis of all bases used for the merge. (John Arbash Meinel, #40412)
- `bzr mv --quiet` really is quiet now. (Gordon Tyler, #271790)
- `bzr serve` is more clear about the risk of supplying `--allow-writes`. (Robert Collins, #84659)
- `bzr serve --quiet` really is quiet now. (Gordon Tyler, #252834)
• Fix bug with redirected URLs over authenticated HTTP. (Glen Mailer, Neil Martinsen-Burrell, Vincent Ladeuil, #395714)

• Interactive merge doesn’t leave branch locks behind. (Aaron Bentley)

• Lots of bug fixes for the test suite on Windows. We should once again have a test suite with no failures on Windows. (John Arbash Meinel)

• `osutils.terminal_width()` obeys the `BZR_COLUMNS` environment variable but returns None if the terminal is not a tty (when output is redirected for example). Also fixes its usage under OSes that doesn’t provide `termios.TIOCGWINSZ`. Make sure the corresponding tests runs on windows too. (Joke de Buhr, Vincent Ladeuil, #353370, #62539) (John Arbash Meinel, Vincent Ladeuil, #492561)

• Terminate SSH subprocesses when no references to them remain, fixing subprocess and file descriptor leaks. (Andrew Bennett, #426662)

• The `--hardlink` option of `bzr branch` and `bzr checkout` now works for 2a format trees. Only files unaffected by content filters will be hardlinked. (Andrew Bennett, #408193)

• The new glob expansion on Windows would replace all `\` characters with `/` even if it there wasn’t a glob to expand, the arg was quoted, etc. Now only change slashes if there is something being glob expanded. (John Arbash Meinel, #485771)

• Use our faster `KnownGraph.heads()` functionality when computing the new rich-root heads. This can cut a conversion time in half (mysql from 13.5h => 6.2h) (John Arbash Meinel, #487632)

• When launching an external diff tool via `bzr diff` -using, temporary files are no longer created, rather, the path to the file in the working tree is passed to the external diff tool. This allows the file to be edited if the diff tool provides for this. (Gary van der Merwe, #490738)

• The launchpad-open command can now be used from a subdirectory of a branch, not just from the root of the branch. (Neil Martinsen-Burrell, #489102)

8.9.4 Improvements

• `bzr log` is now faster. (Ian Clatworthy)

• `bzr update` provides feedback on which branch it is up to date with. (Neil Martinsen-Burrell)

• `bzr upgrade` from pre-2a to 2a can be significantly faster (4x). For details see the xml8 patch and heads() improvements. (John Arbash Meinel)

• `bzrлib.urlutils.local_path_from_url` now accepts ‘file://localhost/’ as well as ‘file:///’ URLs on POSIX. (Michael Hudson)

• The progress bar now shows only a spinner and per-operation counts, not an overall progress bar. The previous bar was often not correlated with real overall operation progress, either because the operations take nonlinear time, or because at the start of the operation Bazaar couldn’t estimate how much work there was to do. (Martin Pool)

8.9.5 Documentation

• Lots of documentation tweaks for inline help topics and command help information.

8.9.6 API Changes

• `bzrlib.textui` (vestigial module) removed. (Martin Pool)
• The Launchpad plugin now has a function `login` which will log in to Launchpad with launchpadlib, and `load_branch` which will return the Launchpad Branch object corresponding to a given Bazaar Branch object. (Jonathan Lange)

8.9.7 Internals

• New test Feature: `ModuleAvailableFeature`. It is designed to make it easier to handle what tests you want to run based on what modules can be imported. (Rather than lots of custom-implemented features that were basically copy-and-pasted.) (John Arbash Meinel)

• `osutils.timer_func()` can be used to get either `time.time()` or `time.clock()` when you want to do performance timing. `time.time()` is limited to 15ms resolution on Windows, but `time.clock()` gives CPU and not wall-clock time on other platforms. (John Arbash Meinel)

• Several code paths that were calling `Transport.get().read()` have been changed to the equivalent `Transport.get_bytes()`. The main difference is that the latter will explicitly call `file.close()`, rather than expecting the garbage collector to handle it. This helps with some race conditions on Windows during the test suite and SFTP tests. (John Arbash Meinel)

8.9.8 Testing

• `TestCaseWithMemoryTransport` no longer sets `$HOME` and `$BZR_HOME` to unicode strings. (Michael Hudson, #464174)

8.10 bzr 2.1.0b3

**Codename** after sprint recovery

**2.1.0b3** 2009-11-16

This release was pushed up from its normal release cycle due to a regression in python 2.4 compatibility in 2.1.0b2. Since this regression was caught before 2.1.0b2 was officially announced, the full changelog includes both 2.1.0b3 and 2.1.0b2 changes.

Highlights of 2.1.0b3 are: new globbing code for all commands on Windows, the test suite now conforms to python’s trunk enhanced semantics (skip, etc.), and `bzr info -v` will now report the correct branch and repo formats for Remote objects.

8.10.1 New Features

• Users can define a shelve editor to provide shelf functionality at a granularity finer than per-patch-hunk. (Aaron Bentley)

8.10.2 Bug Fixes

• Fix for shell completion and short options. (Benoît PIERRE)

• Fix `bzr --profile-imports` with Python 2.6. (Martin Pool)

• Hooks daughter classes should always call the base constructor. (Alexander Belchenko, Vincent Ladeuil, #389648)

• Improve “Binary files differ” hunk handling. (Aaron Bentley, #436325)
• On Windows, do glob expansion at the command-line level (as is usually done in bash, etc.) This means that all commands get glob expansion (bzr status, bzr add, bzr mv, etc). It uses a custom command line parser, which allows us to know if a given section was quoted. It means you can now do bzr ignore "*.py". (John Arbash Meinel, #425510, #426410, #194450)

• Sanitize commit messages that come in from the '-m' flag. We translate ‘rn’ => ‘n’ and a plain ‘r’ => ‘n’. The storage layer doesn’t allow those because XML store silently translate it anyway. (The parser auto-translates rn => n in ways that are hard for us to catch.)

• Show correct branch and repository format descriptions in bzr info -v on a smart server location. (Andrew Bennetts, #196080)

• The fix for bug #186920 accidentally broke compatibility with python 2.4. (Vincent Ladeuil, #475585)

• Using Repository.get_commit_builder().record_iter_changes() now correctly sets self.inv_sha1 to a sha1 string and self.new_inventory to an Inventory instance after calling self.finish_inventory(). (Previously it accidentally set both values as a tuple on self.inv_sha1. This was missed because repo.add_revision ignores the supplied inventory sha1 and recomputes the sha1 from the repo directly. (John Arbash Meinel))

• Shelve command refuse to run if there is no real terminal. (Alexander Belchenko)

• Avoid unnecessarily flushing of trace file; it’s now unbuffered at the Python level. (Martin Pool)

### 8.10.3 Documentation

• Include Japanese translations for documentation (Inada Naoki)

• New API ui_factory.make_output_stream to be used for sending bulk (rather than user-interaction) data to stdout. This automatically coordinates with progress bars or other terminal activity, and can be overridden by GUIs. (Martin Pool, 493944)

### 8.10.4 Internals

• Some of the core groupcompress functionality now releases the GIL before operation. Similar to how zlib and bz2 operate without the GIL in the core compression and decompression routines. (John Arbash Meinel)

### 8.10.5 Testing

• -Dhpssvfs will now trigger on RemoteBzrDir._ensure_real, providing more debugging of VFS access triggers. (Robert Collins)

• KnownFailure is now signalled to ExtendedTestResult using the same method that Python 2.7 uses -addExpectedFailure. (Robert Collins)

• --parallel=fork is now compatible with --subunit. (Robert Collins, Vincent Ladeuil, #419776)

• Reporting of failures shows test ids not descriptions and thus shows parameterised tests correctly. (Robert Collins)

• TestNotApplicable is now handled within the TestCase.run method rather than being looked for within ExtendedTestResult.addError. This provides better handling with other TestResult objects, degrading to success rather than error. (Robert Collins)

• The private method _testConcluded on ExtendedTestResult has been removed - it was empty and unused. (Robert Collins)
• UnavailableFeature is now handled within the TestCase.run method rather than being looked for within addError. If the Result object does not have an addNotSupported method, addSkip is attempted instead, and failing that addSuccess. (Robert Collins)

• When a TestResult does not have an addSkip method, skipped tests are now reported as successful tests, rather than as errors. This change is to make it possible to get a clean test run with a less capable TestResult. (Robert Collins)

8.11  bzc 2.1.0b2

  Codename  a load off my mind

  2.1.0b2  2009-11-02

This is our second feature-filled release since 2.0, pushing us down the path to a 2.1.0. Once again, all bugfixes in 2.0.2 are present in 2.1.0b2.

Key highlights in this release are: improved handling of failures-during-cleanup for commit, fixing a long-standing bug with bzc+http and shared repositories, all lp: URLs to be resolved behind proxies, and a new StaticTuple datatype, allowing us to reduce memory consumption (50%) and garbage collector overhead (40% faster) for many operations.

• A new --concurrency option has been added as well as an associated BZR_CONCURRENCY environment variable to specify the number of processes that can be run concurrently when running bzc selftest. The command-line option overrides the environment variable if both are specified. If none is specified, the number of processes is obtained from the OS as before. (Matt Nordhoff, Vincent Ladeuil)

8.11.1  Bug Fixes

• bzc+http servers no longer give spurious jail break errors when serving branches inside a shared repository. (Andrew Bennetts, #348308)

• Errors during commit are handled more robustly so that knock-on errors are less likely to occur, and will not obscure the original error if they do occur. This fixes some causes of TooManyConcurrentRequests and similar errors. (Andrew Bennetts, #429747, #243391)

• Launchpad URLs can now be resolved from behind proxies. (Gordon Tyler, Vincent Ladeuil, #186920)

• Reduce the strictness for StaticTuple, instead add a debug flag -Dstatic_tuple which will change apis to be strict and raise errors. This way, most users won’t see failures, but developers can improve internals. (John Arbash Meinel, #471193)

• TreeTransform.adjust_path updates the limbo paths of descendants of adjusted files. (Aaron Bentley)

• Unicode paths are now handled correctly and consistently by the smart server. (Andrew Bennetts, Michael Hudson, #458762)

8.11.2  Improvements

• When reading index files, we now use a StaticTuple rather than a plain tuple object. This generally gives a 20% decrease in peak memory, and can give a performance boost up to 40% on large projects. (John Arbash Meinel)

• Peak memory under certain operations has been reduced significantly. (eg, ‘bzc branch launchpad standalone’ is cut in half) (John Arbash Meinel)
8.11.3 Documentation

- Filtered views user documentation upgraded to refer to format 2a instead of pre-2.0 formats. (Ian Clatworthy)

8.11.4 API Changes

- Remove deprecated CLIUIFactory. (Martin Pool)
- UIFactory now has new show_error, show_message and show_warning methods, which can be hooked by non-text UIs. (Martin Pool)

8.11.5 Internals

- Added bzrlib._simple_set_pyx. This is a hybrid between a Set and a Dict (it only holds keys, but you can lookup the object located at a given key). It has significantly reduced memory consumption versus the builtin objects (1/2 the size of Set, 1/3rd the size of Dict). This is used as the interning structure for StaticTuple objects. (John Arbash Meinel)
- bzrlib._static_tuple_c.StaticTuple is now available and used by the btree index parser and the chk map parser. This class functions similarly to tuple objects. However, it can only point to a limited collection of types. (Currently StaticTuple, str, unicode, None, bool, int, long, float, but not subclasses). This allows us to remove it from the garbage collector (it cannot be in a cycle), it also allows us to intern the objects. In testing, this can reduce peak memory by 20-40%, and significantly improve performance by removing objects from being inspected by the garbage collector. (John Arbash Meinel)
- GroupCompressBlock._ensure_content() will now release the zlib.decompressobj() when the first request is for all of the content. (Previously it would only be released if you made a request for part of the content, and then all of it later.) This turns out to be a significant memory savings, as a zstream carries around approx 260kB of internal state and buffers. (For branching bzr.dev this drops peak memory from 382MB => 345MB.) (John Arbash Meinel)
- When streaming content between 2a format repositories, we now clear caches from earlier versioned files. (So ‘revisions’ is cleared when we start reading ‘inventories’, etc.) This can have a significant impact on peak memory for initial copies (~200MB). (John Arbash Meinel)

8.12 bzr 2.1.0b1

**Codename** While the cat is away

**2.1.0b1** 2009-10-14

This is the first development release in the new split “stable” and “development” series. As such, the release is a snapshot of bzr.dev without creating a release candidate first. This release includes a fair amount of internal changes, with deprecated code being removed, and several new feature developments. People looking for a stable code base with only bugfixes should focus on the 2.0.1 release. All bugfixes present in 2.0.1 are present in 2.1.0b1.

Highlights include support for bzr+ssh://host/~/homedir style URLs, finer control over the plugin search path via extended BZR_PLUGIN_PATH syntax, visible warnings when extension modules fail to load, and improved error handling during unlocking.

8.12.1 New Features

- Bazaar can now send mail through Apple OS X Mail.app. (Brian de Alwis)
• **bzr+ssh** and **bzr** paths can now be relative to home directories specified in the URL. Paths starting with a path segment of `~` are relative to the home directory of the user running the server, and paths starting with `~user` are relative to the home directory of the named user. For example, for a user “bob” with a home directory of `/home/bob`, these URLs are all equivalent:

- `bzr+ssh://bob@host/~repo`
- `bzr+ssh://bob@host/~bob/repo`
- `bzr+ssh://bob@host/home/bob/repo`

If **bzr serve** was invoked with a **--directory** argument, then no home directories outside that directory will be accessible via this method.

This is a feature of **bzr serve**, so pre-2.1 clients will automatically benefit from this feature when **bzr** on the server is upgraded. (Andrew Bennetts, #109143)

• Extensions can now be compiled if either Cython or Pyrex is available. Currently Pyrex is preferred, but that may change in the future. (Arkanes)

• Give more control on **BZR_PLUGIN_PATH** by providing a way to refer to or disable the user, site and core plugin directories. (Vincent Ladeuil, #412930, #316192, #145612)

**8.12.2 Bug Fixes**

• Bazaar’s native protocol code now correctly handles EINTR, which most noticeably occurs if you break in to the debugger while connected to a **bzr+ssh** server. You can now continue from the debugger (by typing ‘c’) and the process continues. However, note that pressing C-in the shell may still kill the SSH process, which is bug 162509, so you must sent a signal to the **bzr** process specifically, for example by typing `kill -QUIT PID` in another shell. (Martin Pool, #341535)

• **bzr add** in a tree that has files with `\r` or `\n` in the filename will issue a warning and skip over those files. (Robert Collins, #3918)

• **bzr dpush** now aborts if uncommitted changes (including pending merges) are present in the working tree. The configuration option **dpush_strict** can be used to set the default for this behavior. (Vincent Ladeuil, #438158)

• **bzr merge** and **bzr remove-tree** now requires **–force** if pending merges are present in the working tree. (Vincent Ladeuil, #426344)

• Clearer message when Bazaar runs out of memory, instead of a **MemoryError** traceback. (Martin Pool, #109115)

• Don’t give a warning on Windows when failing to import **_readdir_pyx** as it is never built. (John Arbash Meinel, #430645)

• Don’t restrict the command name used to run the test suite. (Vincent Ladeuil, #419950)

• FTP transports were built differently when the kerberos python module was present leading to obscure failures related to ASCII/BINARY modes. (Vincent Ladeuil, #443041)

• Network streams now decode adjacent records of the same type into a single stream, reducing layering churn. (Robert Collins)

• **PreviewTree** behaves correctly when **get_file_mtime** is invoked on an unmodified file. (Aaron Bentley, #251532)

• Registry objects should not use **iteritems()** when asked to use **items()**. (Vincent Ladeuil, #430510)

• Weave based repositories couldn’t be cloned when committers were using domains or user ids embedding ‘.sig’. Now they can. (Matthew Fuller, Vincent Ladeuil, #430868)
8.12.3 Improvements

- Revision specifiers can now be given in a more DWIM form, without needing explicit prefixes for specifiers like tags or revision id’s. See `bzc help revisionspec` for full details. (Matthew Fuller)

- Bazaar gives a warning before exiting, and writes into `.bzr.log`, if compiled extensions can’t be loaded. This typically indicates a packaging or installation problem. In this case Bazaar will keep running using pure-Python versions, but this may be substantially slower. The warning can be disabled by setting `ignore_missing_extensions = True` in `bazaar.conf`. See also `<https://answers.launchpad.net/bzr/+faq/703>`_. (Martin Pool, #406113, #430529)

- Secondary errors that occur during `Branch.unlock` and `Repository.unlock` no longer obscure the original error. These methods now use a new decorator, `only_raises`. This fixes many causes of `TooManyConcurrentRequests` and similar errors. (Andrew Bennetts, #429747)

8.12.4 Documentation

- Describe the new shell-like test feature. (Vincent Ladeuil)

- Help on hooks no longer says ‘Not deprecated’ for hooks that are currently supported. (Ian Clatworthy, #422415)

8.12.5 API Changes

- `bzrlib.user_encoding` has been removed; use `bzrlib.osutils.get_user_encoding` instead. (Martin Pool)

- `bzrlib.tests` now uses `stopTestRun` for its `TestResult` subclasses - the same as python’s unittest module. (Robert Collins)

- `diff._get_trees_to_diff` has been renamed to `diff.get_trees_and_branches_to_diff`. It is now a public API, and it returns the old and new branches. (Gary van der Merwe)

- `bzrlib.trace.log_error`, `error` and `info` have been deprecated. (Martin Pool)

- `MutableTree.has_changes()` does not require a tree parameter anymore. It now defaults to comparing to the basis tree. It now checks for pending merges too. `Merger.check_basis` has been deprecated and replaced by the corresponding `has_changes()` calls. `Merge.compare_basis`, `Merger.file_revisions` and `Merger.ensure_revision_trees` have also been deprecated. (Vincent Ladeuil, #440631)

- `ProgressTask.note` is deprecated. (Martin Pool)

8.12.6 Internals

- Added `-Drelock` debug flag. It will note a message every time a repository or branch object is unlocked then relocked the same way. (Andrew Bennetts)

- `BTreeLeafParser.extract_key` has been tweaked slightly to reduce mallocs while parsing the index (approx 3=>1 mallocs per key read). This results in a 10% speedup while reading an index. (John Arbash Meinel)

- The `bzrlib.lsprof` module has a new class `BzrProfiler` which makes profiling in some situations like callbacks and generators easier. (Robert Collins)
8.12.7 Testing

- Passing `--lsprof-tests -v` to `bzr selftest` will cause lprof output to be output for every test. Note that this is very verbose! (Robert Collins)

- Setting `BZR_TEST_PDB=1` when running selftest will cause a pdb post_mortem to be triggered when a test failure occurs. (Robert Collins)

- Shell-like tests can now be written. Code in `bzrlib/tests/script.py`, documentation in `developers/testing.txt` for details. (Vincent Ladeuil)

- Some tests could end up with the same id, that was dormant for a long time. (Vincent Ladeuil, #442980)

- Stop showing the number of tests due to missing features in the test progress bar. (Martin Pool)

- Test parameterisation now does a shallow copy, not a deep copy of the test to be parameterised. This is not expected to break external use of test parameterisation, and is substantially faster. (Robert Collins)

- Tests that try to open a bzr dir on an arbitrary transport will now fail unless they have explicitly permitted the transport via `self.permit_url`. The standard test factories such as `self.get_url` will permit the URLs they provide automatically, so only exceptional tests should need to do this. (Robert Collins)

- The break-in test no longer cares about clean shutdown of the child, instead it is happy if the debugger starts up. (Robert Collins)

- The full test suite is expected to pass when the C extensions are not present. (Vincent Ladeuil, #430749)
BAZAAR RELEASE NOTES

9.1  bzr 2.0.7

2.0.7  NOT RELEASED YET

9.1.1  Compatibility Breaks

- Launchpad has announced that the edge.launchpad.net instance is deprecated and may be shut down in the future <http://blog.launchpad.net/general/edge-is-deprecated>. Bazaar has therefore been updated in this release to talk to the main (launchpad.net) servers, rather than the edge ones. (Vincent Ladeuil, #583667)

9.1.2  New Features

9.1.3  Bug Fixes

- Avoid spurious ValueErrors when autopacking a subset of the repository, and seeing a revision without its related inventory (John Arbash Meinel, #437003)

- Avoid UnicodeDecodeError in bzr add with multiple files under a non-ascii path on windows from symlink support addition. (Martin [gz], #686611)

- Fix activity reporting for pycurl when using https with some implementations of curl. (Vincent Ladeuil, #614713)

9.1.4  Improvements

9.1.5  Documentation

9.1.6  API Changes

9.1.7  Internals

9.1.8  Testing

9.2  bzr 2.0.6

2.0.6  2010-09-17
The sixth release in our 2.0 series addresses several user-inconvenience bugs. None are critical, but upgrading is recommended for all users on earlier 2.0 releases.

### 9.2.1 Bug Fixes

- Additional merges after an unrelated branch has been merged with its history no longer crash when deleted files are involved. (Vincent Ladeuil, John Arbash Meinel, #375898)

- `bzr add SYMLINK/FILE` now works properly when the symlink points to a previously-unversioned directory within the tree: the directory is marked versioned too. (Martin Pool, #192859)

- `bzr commit SYMLINK` now works, rather than trying to commit the target of the symlink. (Martin Pool, John Arbash Meinel, #128562)

- `bzr revert` now only takes write lock on working tree, instead of on both working tree and branch. (Danny van Heumen, #498409)

- `bzr upgrade` now creates the `backup.bzr` directory with the same permissions as `.bzr` directory on a POSIX OS. (Parth Malwankar, #262450)

- Don’t traceback trying to unversion children files of an already unversioned directory. (Vincent Ladeuil, #494221)

- Don’t traceback when a lockdir’s `held/info` file is corrupt (e.g. contains only NUL bytes). Instead warn the user, and allow `bzr break-lock` to remove it. (Andrew Bennetts, #619872)

- Fix `AttributeError` on `parent.children` when adding a file under a directory that was a symlink in the previous commit. (Martin Pool, #192859)

- Prevent `CHKMap.apply_delta` from generating non-canonical CHK maps, which can result in “missing referenced chk root keys” errors when fetching from repositories with affected revisions. (Andrew Bennetts, #522637)

- Raise `ValueError` instead of a string exception. (John Arbash Meinel, #586926)

- Reduce peak memory by one copy of compressed text. (John Arbash Meinel, #566940)

- Repositories accessed via a smart server now reject being stacked on a repository in an incompatible format, as is the case when accessing them via other methods. This was causing fetches from those repositories via a smart server (e.g. using `bzr branch`) to receive invalid data. (Andrew Bennetts, #562380)

- Selftest with versions of subunit that support `stopTestRun` will no longer error. This error was caused by 2.0 not being updated when upstream python merged the end of run patch, which chose `stopTestRun` rather than `done`. (Robert Collins, #571437)

- When passing a file to `UTF8DirReader` make sure to close the current directory file handle after the chdir fails. Otherwise when passing many filenames into a command line `bzr status` we would leak descriptors. (John Arbash Meinel, #583486)

### 9.2.2 Testing

- `build_tree_contents` can create symlinks. (Martin Pool, John Arbash Meinel)

### 9.3 `bzr 2.0.5`

2.0.5 2010-03-23
This fifth release in our 2.0 series addresses several user-inconvenience bugs. None are critical, but upgrading is recommended for all users on earlier 2.0 releases.

### 9.3.1 Bug Fixes

- Avoid `malloc(0)` in `patiencediff`, which is non-portable. (Martin Pool, #331095)
- Concurrent autopacking is more resilient to already-renamed pack files. If we find that a file we are about to obsolete is already obsoleted, we do not try to rename it, and we leave the file in `obsolete_packs`. The code is also fault tolerant if a file goes missing, assuming that another process already removed the file. (John Arbash Meinel, Gareth White, #507557)
- Cope with the lockdir `held/info` file being empty, which seems to happen fairly often if the process is suddenly interrupted while taking a lock. (Martin Pool, #185103)
- Give the warning about potentially slow cross-format fetches much earlier on in the fetch operation. Don’t show this message during upgrades, and show the correct format indication for remote repositories. (Martin Pool, #456077, #515356, #513157)
- Handle renames correctly when there are files or directories that differ only in case. (Chris Jones, Martin Pool, #368931)
- If `bzr push --create-prefix` triggers an unexpected `NoSuchFile` error, report that error rather than failing with an unhelpful `UnboundLocalError`. (Andrew Bennetts, #423563)
- Running `bzr` command without any arguments now shows `bzr` version number along with rest of the help text. (Parth Malwankar, #369501)
- Use `osutils.O_NOINHERIT` for some files on win32 to avoid `PermissionDenied` errors. (Inada Naoki, #524560)

### 9.3.2 Documentation

- Added `location-alias` help topic. (Andrew Bennetts, #337834)
- Fixed CHM generation by moving the NEWS section template into a separate file. (Ian Clatworthy, #524184)

### 9.4 `bzr 2.0.4`

**Codename** smooth sailing

**2.0.4** 2010-01-21

The fourth bugfix-only release in the 2.0 series contains more than a dozen bugfixes relative to 2.0.3. The primary focus is on handling interruptions and concurrent operations more cleanly, there is also a fair improvement to `bzr export` when exporting a remote branch.

### 9.4.1 Bug Fixes

- `bzr annotate` on another branch with `-r branch:`... no longer fails with an `ObjectNotLocked` error. (Andrew Bennetts, #496590)
- `bzr export dir` now requests all file content as a record stream, rather than requesting the file content one file-at-a-time. This can make exporting over the network significantly faster (54min => 9min in one case). (John Arbash Meinel, #343218)
• *bzr serve* no longer slowly leaks memory. The compiled `bzrlib.bencode.Encoder()` class was using `__del__` to cleanup and free resources, and it should have been using `__dealloc__`. This will likely have an impact on any other process that is serving for an extended period of time. (John Arbash Meinel, #494046)

• Check for SIGINT (Ctrl-C) and other signals immediately if `readdir` returns `EINTR` by calling `PyErr_CheckSignals`. This affected the optional `_readdir_pyx` extension. (Andrew Bennetts, #495023)

• Concurrent autopacks will no longer lose a newly created pack file. There was a race condition, where if the reload happened at the right time, the second packer would forget the name of the newly added pack file. (John Arbash Meinel, Gareth White, #507566)

• Give a clearer message if the lockdir disappears after being apparently successfully taken. (Martin Pool, #498378)

• Give a warning when fetching between repositories (local or remote) with sufficiently different formats that the content will need to be serialized (ie `InterDifferingSerializer` or `inventory-deltas`), so the user has a clue that upgrading could make it faster. (Martin Pool, #456077)

• If we fail to open `~/.bzr.log` write a clear message to stderr rather than using `warning()`. The log file is opened before logging is set up, and it leads to very confusing: 'no handlers for “bzr”' messages for users, rather than something nicer. (John Arbash Meinel, Barry Warsaw, #503886)

• Refuse to build with any Pyrex 0.9.4 release, as they have known bugs. (Martin Pool, John Arbash Meinel, #449372)

• `setup.py bdist_rpm` now properly finds extra files needed for the build. (there is still the distutils bug `http://bugs.python.org/issue644744`) (Joe Julian, #175839)

• The 2a format wasn’t properly restarting autopacks when something changed underneath it (like another autopack). Now concurrent autopackers will properly succeed. (John Arbash Meinel, #495000)

• `TreeTransform` can now handle when a delta says that the file id for the tree root changes. Rather than trying to rename your working directory, or failing early saying that you can’t have multiple tree roots. This also fixes revert, update, and pull when the root id changes. (John Arbash Meinel, #494269, #504390)

• `_update_current_block` no longer suppresses exceptions, so `^C` at just the right time will get propagated, rather than silently failing to move the block pointer. (John Arbash Meinel, Gareth White, #495023)

### 9.4.2 Testing

• We have a new `test_source` that ensures all pyrex `cdef` functions handle exceptions somehow. (Possibly by setting `# cannot_raise` rather than an `except` clause.) This should help prevent bugs like bug #495023. (John Arbash Meinel)

### 9.5 *bzr 2.0.3*

**Codename** little italy  
**2.0.3** 2009-12-14  

The third stable release of Bazaar has a small handful of bugfixes. As expected, this has no internal or external compatibility changes versus 2.0.2 (or 2.0.0).
9.5.1 Bug Fixes

- `bzr push --use-existing-dir` no longer crashes if the directory exists but contains an invalid `.bzr` directory. (Andrew Bennetts, #423563)
- Content filters are now applied correctly after pull, merge and switch. (Ian Clatworthy, #385879)
- Fix a potential segfault in the groupcompress hash map handling code. When inserting new entries, if the final hash bucket was empty, we could end up trying to access if `(last_entry+1)->ptr == NULL`. (John Arbash Meinel, #490228)
- Improve “Binary files differ” hunk handling. (Aaron Bentley, #436325)

9.6 bzr 2.0.2

**Codename** after the scare

2.0.2 2009-11-02

The second in our “let’s keep the stable bugfixes flowing” series. As expected this has a few (~9) bugfixes relative to 2.0.1, and no major api changes or features.

9.6.1 Bug Fixes

- Avoid “NoneType has no attribute st_mode” error when files disappear from a directory while it’s being read. (Martin Pool, #446033)
- Content filters are now applied correctly after revert. (Ian Clatworthy)
- Diff parsing handles “Binary files differ” hunks. (Aaron Bentley, #436325)
- Fetching from stacked pre-2a repository via a smart server no longer fails intermittently with “second push failed to complete”. (Andrew Bennetts, #437626)
- Fix typos left after test_selftest refactoring. (Vincent Ladeuil, Matt Nordhoff, #461149)
- Fixed `ObjectNotLocked` errors during `bzr log -r NNN somefile`. (Andrew Bennetts, #445171)
- PreviewTree file names are not limited by the encoding of the temp directory’s filesystem. (Aaron Bentley, #436794)

9.6.2 Improvements

- `bzr log` now read-locks branches exactly once, so makes better use of data caches. (Andrew Bennetts)

9.6.3 Documentation

- Filtered views user documentation upgraded to refer to format 2a instead of pre-2.0 formats. (Ian Clatworthy)

9.7 bzr 2.0.1

**Codename** Stability First

2.0.1 2009-10-14
The first of our new ongoing bugfix-only stable releases has arrived. It includes a collection of 12 bugfixes applied to bzr 2.0.0, but does not include any of the feature development in the 2.1.0 series.

9.7.1 Bug Fixes

- `bzr add` in a tree that has files with \r or \n in the filename will issue a warning and skip over those files. (Robert Collins, #3918)
- bzr will attempt to authenticate with SSH servers that support keyboard-interactive auth but not password auth when using Paramiko. (Andrew Bennetts, #433846)
- Fixed fetches from a stacked branch on a smart server that were failing with some combinations of remote and local formats. This was causing “unknown object type identifier 60” errors. (Andrew Bennetts, #427736)
- Fixed `ObjectNotLocked` errors when doing some log and diff operations on branches via a smart server. (Andrew Bennetts, #389413)
- Handle things like `bzr add foo` and `bzr rm foo` when the tree is at the root of a drive. `osutils._cicp_canonical_relnpath` always assumed that `abspath()` returned a path that did not have a trailing /, but that is not true when working at the root of the filesystem. (John Arbash Meinel, Jason Spashett, #322807)
- Hide deprecation warnings for ‘final’ releases for python2.6. (John Arbash Meinel, #440062)
- Improve the time for `bzr log DIR` for 2a format repositories. We had been using the same code path as for <2a formats, which required iterating over all objects in all revisions. (John Arbash Meinel, #374730)
- Make sure that we unlock the tree if we fail to create a TreeTransform object when doing a merge, and there is limbo, or pending-deletions directory. (Gary van der Merwe, #427773)
- Occasional IndexError on renamed files have been fixed. Operations that set a full inventory in the working tree will now go via the apply_inventory_delta code path which is simpler and easier to understand than dirstates set_state_from_inventory method. This may have a small performance impact on operations built on _write_inventory, but such operations are already doing full tree scans, so no radical performance change should be observed. (Robert Collins, #403322)
- Retrieving file text or mtime from a _PreviewTree has good performance when there are many changes. (Aaron Bentley)
- The CHK index pages now use an unlimited cache size. With a limited cache and a large project, the random access of chk pages could cause us to download the entire cix file many times. (John Arbash Meinel, #402623)
- When a file kind becomes unversionable after being added, a sensible error will be shown instead of a traceback. (Robert Collins, #438569)

9.7.2 Documentation

- Improved README. (Ian Clatworthy)
- Improved upgrade documentation for Launchpad branches. (Barry Warsaw)

9.8 bzr 2.0.0

2.0.0  2009-09-22

Codename  Instant Karma
This release of Bazaar makes the 2a (previously ‘brisbane-core’) format the default when new branches or repositories are created. This format is substantially smaller and faster for many operations. Most of the work in this release focuses on bug fixes and stabilization, covering both 2a and previous formats. (See the Upgrade Guide for information on migrating existing projects.)

This release also improves the documentation content and presentation, including adding Windows HtmlHelp manuals. The Bazaar team decided that 2.0 will be a long-term supported release, with bugfix-only 2.0.x releases based on it, continuing for at least six months or until the following stable release.

### 9.8.1 Changes from 2.0.0rc2 to final

- Officially branded as 2.0.0 rather than 2.0 to clarify between things that “want to happen on the 2.0.x stable series” versus things that want to “land in 2.0.0”. (Changes how bzrlib._format_version_tuple() handles micro = 0.) (John Arbash Meinel)

### 9.9 bzr 2.0.0rc2

#### 2.0.0rc2 2009-09-10

### 9.9.1 New Features

- Added post_commit hook for mutable trees. This allows the keywords plugin to expand keywords on files changed by the commit. (Ian Clatworthy, #408841)

### 9.9.2 Bug Fixes

- Bazaar’s native protocol code now correctly handles EINTR, which most noticeably occurs if you break in to the debugger while connected to a bzr+ssh server. You can now can continue from the debugger (by typing ‘c’) and the process continues. However, note that pressing C-in the shell may still kill the SSH process, which is bug 162509, so you must send a signal to the bzr process specifically, for example by typing `kill -QUIT PID` in another shell. (Martin Pool, #341535)

- `bzr check` in pack-0.92, 1.6 and 1.9 format repositories will no longer report incorrect errors about Missing inventory (‘TREE_ROOT’, ...) (Robert Collins, #416732)

- `bzr info -v` on a 2a format still claimed that it was a “Development format” (John Arbash Meinel, #424392)

- `bzr log stacked-branch` shows the full log including revisions that are in the fallback repository. (Regression in 2.0rc1). (John Arbash Meinel, #419241)

- Clearer message when Bazaar runs out of memory, instead of a `MemoryError` traceback. (Martin Pool, #109115)

- Conversion to 2a will create a single pack for all the new revisions (as long as it ran without interruption). This improves both `bzr upgrade` and `bzr pull` or `bzr merge` from local branches in older formats. The autopack logic that occurs every 100 revisions during local conversions was not returning that pack’s identifier, which resulted in the partial packs created during the conversion not being consolidated at the end of the conversion process. (Robert Collins, #423818)

- Fetches from 2a to 2a are now again requested in ‘groupcompress’ order. Groups that are seen as ‘underutilized’ will be repacked on-the-fly. This means that when the source is fully packed, there is minimal overhead during the fetch, but if the source is poorly packed the result is a fairly well packed repository (not as good as ‘bzr pack’ but good-enough.) (Robert Collins, John Arbash Meinel, #402652)
• Fix a potential segmentation fault when doing ‘log’ of a branch that had ghosts in its mainline. (Evaluating None as a tuple is bad.) (John Arbash Meinel, #419241)

• groupcompress sort order is now more stable, rather than relying on topo_sort ordering. The implementation is now KnownGraph.gc_sort. (John Arbash Meinel)

• Local data conversion will generate correct deltas. This is a critical bugfix vs 2.0rc1, and all 2.0rc1 users should upgrade to 2.0rc2 before converting repositories. (Robert Collins, #422849)

• Network streams now decode adjacent records of the same type into a single stream, reducing layering churn. (Robert Collins)

• Prevent some kinds of incomplete data from being committed to a 2a repository, such as revisions without inventories, a missing chk_bytes record for an inventory, or a missing text referenced by an inventory. (Andrew Bennetts, #423506, #406687)

9.9.3 Documentation

• Fix assertion error about "_remember_remote_is_before" when pushing to older smart servers. (Andrew Bennetts, #418931)

• Help on hooks no longer says ‘Not deprecated’ for hooks that are currently supported. (Ian Clatworthy, #422415)

• PDF and CHM (Windows HtmlHelp) formats are now supported for the user documentation. The HTML documentation is better broken up into topics. (Ian Clatworthy)

• The developer and foreign language documents are now separated out so that searching in the HTML and CHM files produces more useful results. (Ian Clatworthy)

• The main table of contents now provides links to the new Migration Docs and Plugins Guide. (Ian Clatworthy)

9.10 bzr 2.0.0rc1

Codename no worries

2.0.0rc1 2009-08-26

9.10.1 Compatibility Breaks

• The default format for bzr is now 2a. This format brings many significant performance and size improvements. bzr can pull from any existing repository into a 2a one, but can only transfer from 2a into rich-root repositories. The Upgrade guide has more information about this change. (Robert Collins)

• On Windows auto-detection of Putty’s plink.exe is disabled. Default SSH client for Windows is paramiko. User still can force usage of plink if explicitly set environment variable BZR_SSH=plink. (#414743, Alexander Belchenko)

9.10.2 New Features

• bzr branch --switch can now switch the checkout in the current directory to the newly created branch. (Lukáš Lalinský)
9.10.3 Bug Fixes

- Further tweaks to handling of bzr add messages about ignored files. (Jason Spashett, #76616)
- Fetches were being requested in ‘groupcompress’ order, but weren’t recombing the groups. Thus they would ‘fragment’ to get the correct order, but not ‘recombine’ to actually benefit from it. Until we get recombing to work, switching to ‘unordered’ fetches avoids the fragmentation. (John Arbash Meinel, #402645)
- Fix a pycurl related test failure on karmic by recognizing an error raised by newer versions of pycurl. (Vincent Ladeuil, #306264)
- Fix a test failure on karmic by making a locale test more robust. (Vincent Ladeuil, #413514)
- Fix IndexError printing CannotBindAddress errors. (Martin Pool, #286871)
- Fix “Revision ... not present” errors when upgrading stacked branches, or when doing fetches from a stacked source to a stacked target. (Andrew Bennetts, #399140)
- bzr branch of 2a repositories over HTTP is much faster. bzr now batches together small fetches from 2a repositories, rather than fetching only a few hundred bytes at a time. (Andrew Bennetts, #402657)

9.10.4 Improvements

- A better description of the platform is shown in crash tracebacks, bzr --version and bzr selftest. (Martin Pool, #409137)
- bzr can now (again) capture crash data through the apport library, so that a single human-readable file can be attached to bug reports. This can be disabled by using -Dno_apport on the command line, or by putting no_apport into the debug_flags section of bazaar.conf. (Martin Pool, Robert Collins, #389328)
- bzr push locally on windows will no longer give a locking error with dirstate based formats. (Robert Collins)
- bzr shelve and bzr unshelve now work on windows. (Robert Collins, #305006)
- Commit of specific files no longer prevents using the iter_changes codepath. On 2a repositories, commit of specific files should now be as fast, or slightly faster, than a full commit. (Robert Collins)
- The internal core code that handles specific file operations like bzr st FILENAME or bzr commit FILENAME has been changed to include the parent directories if they have altered, and when a directory stops being a directory its children are always included. This fixes a number of causes for InconsistentDelta errors, and permits faster commit of specific paths. (Robert Collins, #347649)

9.10.5 Documentation

- New developer documentation for content filtering. (Martin Pool)

9.10.6 API Changes

- bzrlib.shelf_ui has had the from_args convenience methods of its classes changed to manage lock lifetime of the trees they open in a way consistent with reader-exclusive locks. (Robert Collins, #305006)

9.10.7 Testing
BAZAAR RELEASE NOTES

10.1 bzr 1.18.1

Codename  nein nein nein!

1.18.1  2009-09-09

This release fixes two small but worthwhile bugs relevant to users on Microsoft Windows: some commands that failed
on with locking errors will now work, and a bug that caused poor performance after committing a file with line-ending
conversion has now been fixed. It also fixes a bug in pushing to older servers.

10.1.1 Bug Fixes

• Fixed a problem where using content filtering and especially end-of-line conversion will commit too many
copies a file. (Martin Pool, #415508)
• Fix assertion error about _remember_remote_is_before in set_tags_bytes when pushing to older
smart servers. (Andrew Bennetts, Alexander Belchenko, #418931)

10.1.2 Improvements

• bzr push locally on Windows will no longer give a locking error with dirstate based formats. (Robert Collins)
• bzr shelve and bzr unshelve now work on Windows. (Robert Collins, #305006)

10.1.3 API Changes

• bzrlib.shelf_ui has had the from_args convenience methods of its classes changed to manage lock
lifetime of the trees they open in a way consistent with reader-exclusive locks. (Robert Collins, #305006)
• Tree.path_content_summary may return a size of None, when called on a tree with content filtering
where the size of the canonical form cannot be cheaply determined. (Martin Pool)
• When manually creating transport servers in test cases, a new helper TestCase.start_server that regis-
ters a cleanup and starts the server should be used. (Robert Collins)
10.2 bzr 1.18

10.2.1 Compatibility Breaks

- Committing directly to a stacked branch from a lightweight checkout will no longer work. In previous versions this would appear to work but would generate repositories with insufficient data to create deltas, leading to later errors when branching or reading from the repository. (Robert Collins, bug #375013)

10.2.2 New Features

10.2.3 Bug Fixes

- Fetching from 2a branches from a version-2 bzr protocol would fail to copy the internal inventory pages from the CHK store. This cannot happen in normal use as all 2a compatible clients and servers support the version-3 protocol, but it does cause test suite failures when testing downlevel protocol behaviour. (Robert Collins)
- Fix a test failure on karmic by making a locale test more robust. (Vincent Ladeuil, #413514)
- Fixed “Pack ... already exists” error when running `bzr pack` on a fully packed 2a repository. (Andrew Bennetts, #382463)
- Further tweaks to handling of `bzr add` messages about ignored files. (Jason Spashett, #76616)
- Properly handle fetching into a stacked branch while converting the data, especially when there are also ghosts. The code was filling in parent inventories incorrectly, and also not handling when one of the parents was a ghost. (John Arbash Meinel, #402778, #412198)
- `RemoteStreamSource.get_stream_for_missing_keys` will fetch CHK inventory pages when appropriate (by falling back to the vfs stream source). (Andrew Bennetts, #406686)
- StreamSource generates rich roots from non-rich root sources correctly now. (Andrew Bennetts, #368921)
- When deciding whether a repository was compatible for upgrading or fetching, we previously incorrectly checked the default repository format for the bzrdir format, rather than the format that was actually present on disk. (Martin Pool, #408824)

10.2.4 Improvements

- A better description of the platform is shown in crash tracebacks, `bzr --version` and `bzr selftest`. (Martin Pool, #409137)
- Cross-format fetches (such as between 1.9-rich-root and 2a) via the smart server are more efficient now. They send inventory deltas rather than full inventories. The smart server has two new requests, `Repository.get_stream_1.19` and `Repository.insert_stream_1.19` to support this. (Andrew Bennetts, #374738, #385826)
- Extracting the full ancestry and computing the `merge_sort` is now significantly faster. This effects things like `bzr log -n0`. (For example, `bzr log -r -10..-1 -n0 bzr.dev` is 2.5s down to 1.0s. (John Arbash Meinel)
10.2.5 Documentation

10.2.6 API Changes

10.2.7 Internals

- `-Dstrict_locks` can now be used to check that read and write locks are treated properly w.r.t. exclusivity. (We don’t try to take an OS read lock on a file that we already have an OS write lock on.) This is now set by default for all tests, if you have a test which cannot be fixed, you can use `self.thisFailsStrictLockCheck()` as a compatibility knob. (John Arbash Meinel)

- `InterDifferingSerializer` is now only used locally. Other fetches that would have used `InterDifferingSerializer` now use the more network friendly `StreamSource`, which now automatically does the same transformations as `InterDifferingSerializer`. (Andrew Bennetts)

- `KnownGraph` now has `.topo_sort` and `.merge_sort` member which are implemented in pyrex and significantly faster. This is exposed along with `CombinedGraphIndex.find_ancestry()` as `VersionedFiles.get_known_graph_ancestry(keys)`. (John Arbash Meinel)

- `RemoteBranch.open` now honours `ignore_fallbacks` correctly on `bzr-v2` protocols. (Robert Collins)

- The index code now has some specialized routines to extract the full ancestry of a key in a more efficient manner. `CombinedGraphIndex.find_ancestry()`. (Time to get ancestry for `bzr.dev` drops from 1.5s down to 300ms. For OOo from 33s => 10.5s) (John Arbash Meinel)

10.2.8 Testing

- Install the test ssl certificate and key so that installed bzr can run the https tests. (Denys Duchier, #392401)

10.3 `bzr 1.18rc1`

**Codename** little traveller

1.18 2009-08-20

1.18rc1 2009-08-10

This release of Bazaar marches on towards the 2.0 release in which the 2a `brisbane-core` format becomes generally recommended. Most of the work in this release now focusses on bug fixes and stabilization, covering both 2a and previous formats. There is a new text-mode interactive merge feature, a new guide to migration to 2a format in the user documentation, and pushing branches to a smart server is now much faster.

The Bazaar team decided that 2.0 will be a long-term supported release, with bugfix-only releases based on it continuing for at least six months or until the following stable release.

There are no changes from 1.18rc1 to 1.18.

10.3.1 New Features

- `bzr merge --interactive` applies a user-selected portion of the merge. The UI is similar to `shelve`. (Aaron Bentley)

- `bzr reconfigure` now takes options `--stacked-on URL` and `--unstacked` to change stacking of a branch. (Martin Pool, #391411)
10.3.2 Bug Fixes

- Annotating on a stacked branch will now succeed in simple scenarios. There are still some complex scenarios where it will fail (bug #399884) (John Arbash Meinel, #393366)

- A progress bar is no longer left dangling when `bzr selftest` completes, and the progress bar updates with zero latency so the displayed test name is always the one that’s actually running. (Martin Pool, #123688)

- Authenticating against an SSH server now uses `auth_none` to determine if password authentication is even supported. This fixes a bug where users would be prompted for a launchpad password, even though launchpad only supports publickey authentication. (John Arbash Meinel, #375867)

- BranchBuilder now accepts timezone to avoid test failures in countries far from GMT. (Vincent Ladeuil, #397716)

- `bzr commit` no longer saves the unversioning of missing files until the commit has completed on the branch. This means that aborting a commit that found a missing file will leave the tree unedited. (Robert Collins, #282402)

- `bzr mv` no longer takes out branch locks, which allows it to work when the branch is readonly. (Robert Collins, #216541)

- `bzr revert .` no longer generates an InconsistentDelta error when there are missing subtrees. (Robert Collins, #367632)

- `bzr send` now generates valid bundles with --2a formats. However, do to internal changes necessary to support this, older clients will fail when trying to insert them. For newer clients, the bundle can be used to apply the changes to any rich-root compatible format. (John Arbash Meinel, #393349)

- Cope with FTP servers that don’t support restart/append by falling back to reading and then rewriting the whole file, such as TahoeLAFS. (This fallback may be slow for some access patterns.) (Nils Durner, #294709)

- Encode the paths in `mbcs` encoding on Windows when spawning an external diff client. This at least allows supporting filenames that are not ascii, but are present in the current locale. Ideally we would be able to pass the Unicode path, but that would be client dependent. (John Arbash Meinel, #382709)

- Fix a compile bug on Solaris having to do with const and pointer-to-pointers. (John Arbash Meinel, #408441)

- Fixed a NameError that occurs when merging or pulling from a URL that causes a redirection loop when bzr tries to read a URL as a bundle. (Andrew Bennetts, #400847)

- Fix `AttributeError: TestUIFactory object has no attribute 'tick'` running send and similar commands on 2a formats. (Martin Pool, #408201)

- Fix crash in some invocations of `bzr status` in format 2a. (Martin Pool, #403523)

- Fixed export to existing directory: if directory is empty then export will succeed, otherwise it fails with error. (Alexander Belchenko, #406174)

- Fixed spurious “Source branch does not support stacking” warning when pushing. (Andrew Bennetts, #388908)

- Fixed spurious transport activity indicator appearing while tests are running. (Martin Pool, #343532)

- Merge now correctly handles empty right-hand revision specs. (Aaron Bentley, #333961)

- Renames to lexographically lower basenames in trees that have never been committed to will no longer corrupt the dirstate. This was caused by an bug in the dirstate update_minimal method. (Robert Collins, #395556)

- Requests for unknown methods no longer cause the smart server to log lots of backtraces about `UnknownSmartMethod, do_chunk or do_end`. (Andrew Bennetts, #338561)

- Shelve will not shelve the initial add of the tree root. (Aaron Bentley)
• Streaming from bzr servers where there is a chain of stacked branches (A stacked on B stacked on C) will now work. (Robert Collins, #406597)

• The environment variable BZR_PROGRESS_BAR set to either text or none always forces progress bars either on or off respectively. Otherwise, they’re turned on if TERM is not dumb and stderr is a terminal. bzr always uses the ‘text’ user interface when run as a command, so BZR_USE_TEXT_UI is no longer needed. (Martin Pool, #339385, #387717)

• The optional _knit_load_data_pyx C extension was never being imported. This caused significant slowdowns when reading data from repositories. (Andrew Bennetts, #405653)

• The --hardlink option to branch and checkout is not supported at the moment on workingtree formats that can do content filtering. (See <https://bugs.launchpad.net/bzr/+bug/408193>.) bzr now says so, rather than just ignoring the option. (Martin Pool)

• There was a bug in osutils.relpath that was only triggered on Windows. Essentially if you were at the root of a drive, and did something to a branch/repo on another drive, we would go into an infinite loop while trying to find a ‘relative path’. (John Arbash Meinel, #394227)

• WorkingTree4.unversion will no longer fail to unversion ids which were present in a parent tree but renamed in the working tree. (Robert Collins, #187207)

10.3.3 Improvements

• Can now rename/move files even if they have been removed from the inventory. (Marius Kruger)

• Pushing branches with tags via bzr:// and bzr+ssh:// is much faster, using a new Branch.set_tags_bytes smart server verb rather than VFS methods. For example, pushes of small branches with tags take 11 rather than 18 smart server requests. (Andrew Bennetts, #398608)

• Sending Ctrl-Break on Windows will now drop you into the debugger, in the same way that sending Ctrl-\ does on other platforms. (John Arbash Meinel)

10.3.4 Documentation

• Added Bazaar 2.0 Upgrade Guide. (Ian Clatworthy)

10.3.5 API Changes

• CLIUIFactory is deprecated; use TextUIFactory instead if you need to subclass or create a specific class, or better yet the existing make_ui_for_terminal. SilentUIFactory is clarified to do no user interaction at all, rather than trying to read from stdin but not writing any output, which would be strange if reading prompts or passwords. (Martin Pool)

• New TransformPreview.commit() allows committing without a working tree. (Aaron Bentley)

• pb parameter to TextTestResult is deprecated and ignored. (Martin Pool)

• ProgressTasks now prefer to talk direct to their ProgressView not to the UIFactory. (Martin Pool)

• WorkingTree._check now requires a references dict with keys matching those returned by WorkingTree._get_check_refs. (Robert Collins)
10.3.6 Internals

- CHKInventory.path2id uses the parent_id to basename hash to avoid reading the entries along the path, reducing work to lookup ids from paths. (Robert Collins)

- CHKMap.apply_delta now raises InconsistentDelta if a delta adds as new a key which was already mapped. (Robert Collins)

- Inventory delta application catches more cases of corruption and can prevent corrupt deltas from affecting consistency of data structures on disk. (Robert Collins)

- --subunit support now adds timestamps if the subunit version supports it. (Robert Collins)

- The Windows all-in-one installer now bundles the PyQt image format plugins, which allows previewing more images as part of ‘qdiff’. (Alexander Belchenko)

10.3.7 Testing

- Merge directive cherrypick tests must use the same root id. (Martin Pool, #409684)

- Spurious failure in check tests on rich-root formats fixed. (Martin Pool, #408199)

- The bzrlib.tests.TextTestRunner will no longer call countTestsCases on the test being run. Progress information is instead handled by having the test passed in call result.progress before running its contents. This improves the behaviour when using TextTestRunner with test suites that don’t support countTestsCases. (Robert Collins)
11.1 bzr 1.17

Codename so-late-its-brunch

1.17rc1 2009-07-13
1.17 2009-07-20

Bazaar continues to blaze a straight and shining path to the 2.0 release and the elevation of the 2a beta format to the full glory of “supported and stable”.

Highlights in this release include greatly reduced memory consumption during commits, faster ls, faster annotate, faster network operations if you’re specifying a revision number and the final destruction of those annoying progress bar artifacts.

11.1.1 Changes from 1.17rc1 to 1.17final

• Change an extension to call the python frozenset() rather than the C api PyFrozenSet_New. It turns out that python2.4 did not expose the C api. (John Arbash Meinel, #399366)

• Fixes for the Makefile and the rename of generate_docs.py to tools/generate_docs.py to allow everything to be built on Windows. (John Arbash Meinel, #399356)

• bzr serve once again applies a ChrootServer to the given directory before serving it. (Andrew Bennetts, #400535)

11.1.2 Compatibility Breaks

• bzr register-branch from the Launchpad plugin now refers to “project” instead of “product” which is the correct Launchpad terminology. The –product option is deprecated and users should switch to using –project. (Neil Martinsen-Burrell, #238764)

11.1.3 New Features

• bzr push now aborts if uncommitted changes (including pending merges) are present in the working tree (if one is present) and no revision is specified. The configuration option push_strict can be used to set the default for this behavior. (Vincent Ladeuil, #284038, #322808, #65286)

• bzr revno and bzr revision-info now have a --tree option to show revision info for the working tree instead of the branch. (Matthew Fuller, John Arbash Meinel)
• `bzr send` now aborts if uncommitted changes (including pending merges) are present in the working tree and no revision is specified. The configuration option `send_strict` can be used to set the default for this behavior. (Vincent Ladeuil, #206577)

• `bzr switch --create-branch/-b` can now be used to create and switch to a new branch. Supplying a name without a / will create the branch relative to the existing branch. (similar to how `bzr switch name` works when the branch already exists.) (John Arbash Meinel)

11.1.4 Bug Fixes

• Accept uppercase “Y/N” to prompts such as from break lock. (#335182, Tim Powell, Martin Pool)

• Add documentation about diverged branches and how to fix them in the centralized workflow with local commits. Mention `bzr help diverged-branches` when a push fails because the branches have diverged. (Neil Martinsen-Burrell, #269477)

• Annotate would sometimes ‘latch on’ to trivial lines, causing important lines to be incorrectly annotated. (John Arbash Meinel, #387952)

• Automatic format upgrades triggered by default stacking policies on a 1.16rc1 (or later) smart server work again. (Andrew Bennetts, #388675)

• Avoid progress bar artifacts being left behind on the screen. (Martin Pool, #321935)

• Better message in `bzr split` error suggesting a rich root format. (Neil Martinsen-Burrell, #220067)

• `Branch.set_append_revisions_only` now works with branches on a smart server. (Andrew Bennetts, #365865)

• By default, `bzr branch` will fail if the target directory exists, but does not already have a control directory. The flag `--use-existing-dir` will allow operation to proceed. (Alexander Belchenko, #307554)

• `bzr ls DIR --from-root` now shows only things in DIR, not everything. (Ian Clatworthy)

• Fetch between repositories does not error if they have inconsistent data that should be irrelevant to the fetch operation. (Aaron Bentley)

• Fix `AttributeError` exception when reconfiguring lightweight checkout of a remote repository. (Jelmer Vernooij, #332194)

• Fix bug in decoding v3 smart server messages when receiving multiple lots of excess bytes after an end-of-message. (Andrew Bennetts)

• Force deletion of readonly files during merge, update and other tree transforms. (Craig Hewetson, Martin Pool, #218206)

• Force socket shutdown in threaded HTTP test servers to avoid client hangs (pycurl). (Vincent Ladeuil, #383920).

• `LRUCache` will maintain the linked list pointers even if a nodes cleanup function raises an exception. (John Arbash Meinel, #396838)

• Progress bars are now suppressed again when the environment variable `BZR_PROGRESS_BAR` is set to `none`. (Martin Pool, #339385)

• Reduced memory consumption during `bzr commit` of large files. For pre 2a formats, should be down to ~3x the size of a file. For `--2a` format repositories, it is down to the size of the file content plus the size of the compressed text. Related to bug #109114. (John Arbash Meinel)

• Set hidden attribute on .bzr directory below unicode path should never fail with error. The operation should succeed even if `bzr` unable to set the attribute. (Alexander Belchenko, related to bug #335362).
• Stacking will no longer accept requests to stack on the same branch/repository. Existing branches that incorrectly reference the same repository in a stacking configuration will now raise UnstackableLocationError when the branch is opened. This can be fixed by removing the stacking location inside .bzr/branch. (Robert Collins, #376243)

• The log+ decorator, useful in debugging or profiling, could cause “AttributeError: ‘list’ object has no attribute ‘next’”. This is now fixed. The log decorator no longer shows the elapsed time or transfer rate because they’re available in the log prefixes and the transport activity display respectively. (Martin Pool, #340347)

• Unshelve works correctly when multiple zero-length files are present on the shelf. (Aaron Bentley, #363444)

• Progress bars no longer show the network transport scheme or direction. (Martin Pool)

• launchpad-login now respects the ‘verbose’ option. (Jonathan Lange, #217031)

11.1.5 Internals

• bzrlib.user_encoding is now officially deprecated. It is not possible to write a deprecation wrapper, but the variable will be removed in the near future. Use bzrlib.osutils.get_user_encoding() instead. (Alexander Belchenko)

• Command lookup has had hooks added. bzrlib.Command.hooks has three new hook points: get_command, get_missing_command and list_commands, which allow just-in-time command name provision rather than requiring all command names be known a-priori. (Robert Collins)

• get_app_path from win32utils.py now supports REG_EXPAND_SZ data type and can read path to wordpad.exe. (Alexander Belchenko, #392046)

• graph.KnownGraph has been added. This is a class that can give answers to heads() very quickly. However, it has the assumption that the whole graph has already been loaded. This is true during annotate so it is used there with good success (as much as 2x faster for files with long ancestry and ‘cherrypicked’ changes.) (John Arbash Meinel, Vincent Ladeuil)

• OS file locks are now taken out using CreateFile rather than LockFileEx on Windows. The locking remains exclusive with LockFileEx but now it also works on older versions of Windows (such as Win98). (Martin <gzlist>)

• pack <=> pack fetching is now done via a PackStreamSource rather than the Packer code. The user visible change is that we now properly fetch the minimum number of texts for non-smart fetching. (John Arbash Meinel)

• VersionedFiles._add_text is a new api that lets us insert text into the repository as a single string, rather than a list of lines. This can improve memory overhead and performance of committing large files. (Currently a private api, used only by commit). (John Arbash Meinel)

11.1.6 Improvements

• bzr annotate can now be significantly faster. The time for bzr annotate NEWS is down to 7s from 22s in 1.16. Files with long histories and lots of ‘duplicate insertions’ will be improved more than others. (John Arbash Meinel, Vincent Ladeuil)

• bzr ls is now faster. On OpenOffice.org, the time drops from 2.4 to 1.1 seconds. The improvement for bzr ls -r-1 is more substantial dropping from 54.3 to 1.1 seconds. (Ian Clatworthy)

• Improve “Path(s) are not versioned” error reporting for some commands. (Benoit PIERRE)

• Initial commit performance in --2a repositories has been improved by making it cheaper to build the initial CHKMap. (John Arbash Meinel)
• Resolving a revno to a revision id on a branch accessed via bzr:// or bzr+ssh:// is now much faster and involves no VFS operations. This speeds up commands like bzr pull -r 123. (Andrew Bennetts)

• revision-info now properly aligns the revnos/revids in the output and doesn’t traceback when given revisions not in the current branch. Performance is also significantly improved when requesting multiple revs at once. (Matthew Fuller, John Arbash Meinel)

• Tildes are no longer escaped by Transports. (Andy Kilner)

**11.1.7 Documentation**

• Avoid bad text wrapping in generated documentation. Slightly better formatting in the user reference. (Martin Pool, #249908)

• Minor clarifications to the help for End-Of-Line conversions. (Ian Clatworthy)

**11.1.8 API Changes**

• Removed overspecific error class InvalidProgressBarType. (Martin Pool)

• The method ProgressView._show_transport_activity is now show_transport_activity because it’s part of the contract between this class and the UI. (Martin Pool)
CHAPTER
TWELVE

BAZAAR RELEASE NOTES

12.1 bzr 1.16.1

Released 2009-06-26
End user testing of the 2a format revealed two serious bugs. The first, #365615, caused bzr to raise AbsentContentFactory errors when autopacking. This meant that commits or pushes to 2a-format repositories failed intermittently.

The second bug, #390563, caused the smart server to raise AbsentContentFactory when streaming 2a stacked 2a-format branches. This particularly affected branches stored on Launchpad in the 2a format.

Both of these bugs cause command failures only, neither of them cause data corruption or data loss. And, of course, both of these bugs are now fixed.

12.1.1 Bug Fixes

- We now properly request a more minimal set of file texts when fetching multiple revisions. (Robert Collins, John Arbash Meinel, #390563)

- Repositories using CHK pages (which includes the new 2a format) will no longer error during commit or push operations when an autopack operation is triggered. (Robert Collins, #365615)

- `chk_map.iter_interesting_nodes` now properly uses the intersection of referenced nodes rather than the union to determine what uninteresting pages we still need to look at. Prior to this, incrementally pushing to stacked branch would push the minimal data, but fetching everything would request extra texts. There are some unhandled cases wrt trees of different depths, but this fixes the common cases. (Robert Collins, John Arbash Meinel, #390563)

- `GroupCompress` repositories now take advantage of the pack hints parameter to permit cross-format fetching to incrementally pack the converted data. (Robert Collins)

- `Repository.commit_write_group` now returns opaque data about what was committed, for passing to the `Repository.pack`. Repositories without atomic commits will still return None. (Robert Collins)

- `Repository.pack` now takes an optional hint parameter which will support doing partial packs for repositories that can do that. (Robert Collins)

- `RepositoryFormat` has a new attribute 'pack_compresses' which is True when doing a pack operation changes the compression of content in the repository. (Robert Collins)

- `StreamSink` and `InterDifferingSerialiser` will call `Repository.pack` with the hint returned by `Repository.commit_write_group` if the formats were different and the repository can increase compression by doing a pack operation. (Robert Collins, #376748)
12.2 bzr 1.16

Codename  yesterday-in-california

1.16rc1  2009-06-11
1.16     2009-06-18

This version of Bazaar contains the beta release of the new 2a repository format, suitable for testing by fearless, advanced users. This format or an updated version of it will become the default format in Bazaar 2.0. Please read the NEWS entry before even thinking about upgrading to the new format.

Also included are speedups for many operations on huge projects, a bug fix for pushing stacked new stacked branches to smart servers and the usual bevy of bug fixes and improvements.

12.2.1 Changes from 1.16rc1 to 1.16final

- Fix the nested tree flag check so that upgrade from development formats to 2a can work correctly. (Jelmer Vernooij, #388727)
- Automatic format upgrades triggered by default stacking policies on a 1.16rc1 (or later) smart server work again. (Andrew Bennetts, #388675)

12.2.2 Compatibility Breaks

- Display prompt on stderr (instead of stdout) when querying users so that the output of commands can be safely redirected. (Vincent Ladeuil, #376582)

12.2.3 New Features

- A new repository format 2a has been added. This is a beta release of the brisbane-core (aka group-compress) project. This format now suitable for wider testing by advanced users willing to deal with some bugs. We would appreciate test reports, either positive or negative. Format 2a is substantially smaller and faster for many operations on many trees. This format or an updated version will become the default in bzr 2.0.

This is a rich-root format, so this repository format can be used with bzr-svn. Bazaar branches in previous non-rich-root formats can be converted (including by merge, push and pull) to format 2a, but not vice versa. We recommend upgrading previous development formats to 2a.

Upgrading to this format can take considerable time because it expands and more concisely repacks the full history.

If you use stacked branches, you must upgrade the stacked branches before the stacked-on branches. (See <https://bugs.launchpad.net/bugs/374735>)
- --development7-rich-root is a new dev format, similar to --dev6 but using a Revision serializer using bencode rather than XML. (Jelmer Vernooij, John Arbash Meinel)
- mail_client=claws now supports --body (and message body hooks). Also uses configured from address. (Barry Warsaw)

12.2.4 Improvements

- --development6-rich-root can now stack. (Modulo some smart-server bugs with stacking and non default formats.) (John Arbash Meinel, #373455)
• --development6-rich-root delays generating a delta index for the first object inserted into a group. This has a beneficial impact on bzr commit since each committed texts goes to its own group. For committing a 90MB file, it drops peak memory by about 200MB, and speeds up commit from 7s => 4s. (John Arbash Meinel)

• Numerous operations are now faster for huge projects, i.e. those with a large number of files and/or a large number of revisions, particularly when the latest development format is used. These operations (and improvements on OpenOffice.org) include:
  – branch in a shared repository (2X faster)
  – branch –no-tree (100X faster)
  – diff (2X faster)
  – tags (70X faster)

(Ian Clatworthy)

• Pyrex version of bencode support. This provides optimized support for both encoding and decoding, and is now found at bzrlib.bencode. bzrlib.utils.bencode is now deprecated. (Alexander Belchenko, Jelmer Vernooij, John Arbash Meinel)

12.2.5 Bug Fixes

• Bazaar can now pass attachment files to the mutt email client. (Edwin Grubbs, #384158)

• Better message in bzr add output suggesting using bzr ignored to see which files can also be added. (Jason Spashett, #76616)

• bzr pull -r 123 from a stacked branch on a smart server no longer fails. Also, the Branch.revision_history() API now works in the same situation. (Andrew Bennetts, #380314)

• bzr serve on Windows no longer displays a traceback simply because a TCP client disconnected. (Andrew Bennetts)

• Clarify the rules for locking and fallback repositories. Fix bugs in how RemoteRepository was handling fallbacks along with the _real_repository. (Andrew Bennetts, John Arbash Meinel, #375496)

• Fix a small bug with fetching revisions w/ ghosts into a new stacked branch. Not often triggered, because it required ghosts to be part of the fetched revisions, not in the stacked-on ancestry. (John Arbash Meinel)

• Fix status and commit to work with content filtered trees, addressing numerous bad bugs with line-ending support. (Ian Clatworthy, #362030)

• Fix problem of “directory not empty” when contending for a lock over SFTP. (Martin Pool, #340352)

• Fix rule handling so that eol is optional, not mandatory. (Ian Clatworthy, #379370)

• Pushing a new stacked branch to a 1.15 smart server was broken due to a bug in the BzrDirFormat.initialize_ex smart verb. This is fixed in 1.16, but required changes to the network protocol, so the BzrDirFormat.initialize_ex verb has been removed and replaced with a corrected BzrDirFormat.initialize_ex_1.16 verb. 1.15 clients will still work with a 1.16 server as they will fallback to slower (and bug-free) methods. (Jonathan Lange, Robert Collins, Andrew Bennetts, #385132)

• Reconcile can now deal with text revisions that originated in revisions that are ghosts. (Jelmer Vernooij, #336749)

• Support cloning of branches with ghosts in the left hand side history. (Jelmer Vernooij, #248540)

• The “‘bz diff” now catches OSError from osutils.rmtree and logs a helpful message to the trace file, unless the temp directory really was removed (which would be very strange). Since the diff operation has succeeded from the user’s perspective, no output is written to stderr or stdout. (Maritza Mendez, #363837)
• Translate errors received from a smart server in response to a BzrDirFormat.initialize or BzrDirFormat.initialize_ex request. This was causing tracebacks even for mundane errors like PermissionDenied. (Andrew Bennetts, #381329)

12.2.6 Documentation

• Added directory structure and started translation of docs in Russian. (Alexey Shtokalo, Alexander Ilin, Alexander Belchenko, Dmitry Vasiliev, Volodymyr Kotulskyi)

12.2.7 API Changes

• Added osutils.parent_directories(). (Ian Clatworthy)

• bzrlib.progress.ProgressBar, ChildProgress, DotsProgressBar, TTYProgressBar and child_progress are now deprecated; use ui_factory.nested_progress_bar instead. (Martin Pool)

• graph.StackedParentsProvider is now a public API, replacing graph._StackedParentsProvider. The api is now considered stable and ready for external users. (Gary van der Merwe)

• bzrlib.user_encoding is deprecated in favor of get_user_encoding. (Alexander Belchenko)

• TreeTransformBase no longer assumes that limbo is provided via disk. DiskTreeTransform now provides disk functionality. (Aaron Bentley)

12.2.8 Internals

• Remove weave.py script for accessing internals of old weave-format repositories. (Martin Pool)

12.2.9 Testing

• make check no longer repeats the test run in LANG=C. (Martin Pool, #386180)

• The number of cores is now correctly detected on OSX. (John Szakmeister)

• The number of cores is also detected on Solaris and win32. (Vincent Ladeuil)

• The number of cores is also detected on FreeBSD. (Matthew Fuller)
BAZAAR RELEASE NOTES

13.1 bzr 1.15

1.15rc1 2009-05-16
1.15 2009-05-22
1.15.1 2009-06-09

The smart server will no longer raise ‘NoSuchRevision’ when streaming content with a size mismatch in a reconstructed graph search. New command `bzr dpush`. Plugins can now define their own annotation tie-breaker when two revisions introduce the exact same line.

13.1.1 Changes from 1.15.1 to 1.15.2

• Use zdll on Windows to build `_chk_map_pyx` extension. (Alexander Belchenko)

13.1.2 Changes from 1.15final to 1.15.1

• Translate errors received from a smart server in response to a `BzrDirFormat.initialize` or `BzrDirFormat.initialize_ex` request. This was causing tracebacks even for mundane errors like `PermissionDenied`. (Andrew Bennetts, #381329)

13.1.3 Changes from 1.15rc1 to 1.15final

• No changes

13.1.4 Compatibility Breaks

• `bzr ls` is no longer recursive by default. To recurse, use the new `-R` option. The old `--non-recursive` option has been removed. If you alias `ls` to `ls -R`, you can disable recursion using `--no-recursive` instead. (Ian Clatworthy)

13.1.5 New Features

• New command `bzr dpush` that can push changes to foreign branches (svn, git) without setting custom bzr-specific metadata. (Jelmer Vernooij)
• The new development format --development6-rich-root now supports stacking. We chose not to use a new format marker, since old clients will just fail to open stacked branches, the same as if we used a new format flag. (John Arbash Meinel, #373455)

• Plugins can now define their own annotation tie-breaker when two revisions introduce the exact same line. See bzrlib.annotate._break_annotation_tie Be aware though that this is temporary, private (as indicated by the leading '_') and a first step to address the problem. (Vincent Ladeuil, #348459)

• New command bZR dpush that can push changes to foreign branches (svn, git) without setting custom bzr- specific metadata. (Jelmer Vernooij)

• bZr send will now check the child_submit_format setting in the submit branch to determine what format to use, if none was specified on the command-line. (Jelmer Vernooij)

13.1.6 Improvements

• -Dhpss output now includes the number of VFS calls made to the remote server. (Jonathan Lange)

• --coverage works for code running in threads too. (Andrew Bennets, Vincent Ladeuil)

• bZr pull now has a --local option to only make changes to the local branch, and not the bound master branch. (Gary van der Merwe, #194716)

• bZr rm * is now as fast as bZr rm * --keep. (Johan Walles, #180116)

13.1.7 Bug Fixes

• Adding now works properly when path contains a symbolic link. (Geoff Bache, #183831)

• An error is now raised for unknown eol values. (Brian de Alwis, #358199)

• bZr merge --weave will now generate a conflict if one side deletes a line, and the other side modifies the line. (John Arbash Meinel, #328171)

• bZr reconfigure --standalone no longer raises IncompatibleRepositories. (Martin von Gagern, #248932)

• bZr send works to send emails again using MAPI. (Neil Martinsen-Burrell, #346998)

• Check for missing parent inventories in StreamSink. This prevents incomplete stacked branches being created by 1.13 bZr:// and bZR+ssh:// clients (which have bug #354036). Instead, the server now causes those clients to send the missing records. (Andrew Bennetts)

• Correctly handle HTTP servers proposing multiple authentication schemes. (Vincent Ladeuil, #366107)

• End-Of-Line content filters are now loaded correctly. (Ian Clatworthy, Brian de Alwis, #355280)

• Fix a bug in the pure-python GroupCompress code when handling copies longer than 64KiB. (John Arbash Meinel, #364900)

• Fix TypeError in running bZr break-lock on some URLs. (Alexander Belchenko, Martin Pool, #365891)

• Non-recursive bZr ls now works properly when a path is specified. (Jelmer Vernooij, #357863)

• SSH usernames (defined in ~/.ssh/config) are honoured for bZR+ssh connections. (Vincent Ladeuil, #367726)

• Several bugs related to unicode symlink links have been fixed and the test suite enhanced to better catch regressions for them. (Vincent Ladeuil)

• The smart server will no longer raise ‘NoSuchRevision’ when streaming content with a size mismatch in a reconstructed graph search: it assumes that the client will make sure it is happy with what it got, and this sort of mismatch is normal for stacked environments. bZr 1.13.0/1 will stream from unstacked branches only - in
that case not getting all the content expected would be a bug. However the graph search is how we figured out what we wanted, so a mismatch is both odd and unrecoverable without starting over, and starting over will end up with the same data as if we just permitted the mismatch. If data is gc’d, doing a new search will find only the truncated data, so sending only the truncated data seems reasonable. bzr versions newer than this will stream from stacked branches and check the stream to find missing content in the stacked-on branch, and thus will handle the situation implicitly. (Robert Collins, #360791)

- Upgrading to, or fetching into a ‘rich-root’ format will now correctly set the root data the same way that reconcile does. (Robert Collins, #368921)
- Using unicode Windows API to obtain command-line arguments. (Alexander Belchenko, #375934)

### 13.1.8 Documentation

### 13.1.9 API Changes

- `InterPackRepo.fetch` and `RepoFetcher` now raise `NoSuchRevision` instead of `InstallFailed` when they detect a missing revision. `InstallFailed` itself has been deleted. (Jonathan Lange)
- Not passing arguments to `bzrlib.commands.main()` will now grab the arguments from `osutils.get_unicode_argv()` which has proper support for unicode arguments on windows. Further, the supplied arguments are now required to be unicode strings, rather than `user_encoded` strings. (Alexander Belchenko)

### 13.1.10 Internals

- `bzrlib.branch.Branch.set_parent` is now present on the base branch class and will call `_set_parent_location` after doing unicode encoding. (Robert Collins)
- `bzrlib.remote.RemoteBranch._set_parent_location` will use a new verb `Branch.set_parent_location` removing further VFS operations. (Robert Collins)
- `bzrlib.bzrdir.BzrDir._get_config` now returns a `TransportConfig` or similar when the dir supports configuration settings. The base class defaults to `None`. There is a matching new server verb `BzrDir.get-config_file` to reduce roundtrips for getting BzrDir configuration. (Robert Collins)
- `bzrlib.tests.ExtendedTestResult` has new methods `startTests` called before the first test is started, `done` called after the last test completes, and a new parameter `strict`. (Robert Collins)
- `-Dhpss` when passed to bzr will cause a backtrace to be printed when VFS operations are started on a smart server repository. This should not occur on regular push and pull operations, and is a key indicator for performance regressions. (Robert Collins)
- `-Dlock` when passed to selftest (e.g. `bzr -Dlock selftest`) will cause mismatched physical locks to cause test errors rather than just reporting to the screen. (Robert Collins)
- `-Dprogress` will cause pdb to start up if a progress view jumps backwards. (Robert Collins)
- Fallback `CredentialStore` instances registered with `fallback=True` are now able to provide credentials if obtaining credentials via `~/.bazaar/authentication.conf` fails. (Jelmer Vernooij, Vincent Ladeuil, #321918)
- New hook `Lock.lock_broken` which runs when a lock is broken. This is mainly for testing that lock/unlock are balanced in tests. (Vincent Ladeuil)
- New MergeDirective hook ‘merge_request_body’ allows hooks to supply or alter a body for the message produced by `bzr send`.
- New smart server verb `BzrDir.initialize_ex` which implements a refactoring to the core of clone allowing less round trips on new branches. (Robert Collins)
• New method `Tags.rename_revisions` that can rename revision ids tags are pointing at. (Jelmer Vernooij)
• Updated the bundled `ConfigObj` library to 4.6.0 (Matt Nordhoff)

13.1.11 Testing

• `bzr selftest` will now fail if lock/unlock are not correctly balanced in tests. Using `-Dlock` will turn the related failures into warnings. (Vincent Ladeuil, Robert Collins)
14.1 bzr 1.14

**Codename**  brisbane-core

1.14rc1  2009-04-06
1.14rc2  2009-04-19
1.14  2009-04-28
1.14.1  2009-05-01

New formats 1.14 and 1.14-rich-root supporting End-Of-Line (EOL) conversions, keyword templating (via the bzr-keywords plugin) and generic content filtering. End-of-line conversion is now supported for formats supporting content filtering.

14.1.1 Changes from 1.14final to 1.14.1

- Change api_minimum_version back to api_minimum_version = (1, 13, 0)

14.1.2 Changes from 1.14rc2 to 1.14final

- Fix a bug in the pure-python GroupCompress code when handling copies longer than 64KiB. (John Arbash Meinel, #364900)

14.1.3 Changes from 1.14rc1 to 1.14rc2

- Fix for bug 358037 Revision not in bzrlib.groupcompress.GroupCompressVersionedFiles (Brian de Alwis, John A Meinel)
- Fix for bug 354036 ErrorFromSmartServer - AbsentContentFactory object has no attribute ‘get_bytes_as’ exception while pulling from Launchpad (Jean-Francois Roy, Andrew Bennetts, Robert Collins)
- Fix for bug 355280 eol content filters are never loaded and thus never applied (Brian de Alwis, Ian Clatworthy)
- bzr.dev -r4280 Change _fetch_uses_deltas = False for CHK repos until we can write a better fix. (John Arbash Meinel, Robert Collins)
- Fix for bug 361574 uncommit recommends undefined –levels and -n options (Marius Kruger, Ian Clatworthy)
- bzr.dev r4289 as cherrypicked at lp:-spiv/bzr/stacking-cherrypick-1.14 (Andrew Bennetts, Robert Collins)
14.1.4 Compatibility Breaks

- A previously disabled code path to accelerate getting configuration settings from a smart server has been reinstated. We think this may cause a compatibility with servers older than bzr 0.15. We intend to issue a point release to address this if it turns out to be a problem. (Robert Collins, Andrew Bennetts)

- bzr no longer autodetects nested trees as ‘tree-references’. They must now be explicitly added tree references. At the commandline, use join –reference instead of add. (Aaron Bentley)

- The --long log format (the default) no longer shows merged revisions implicitly, making it consistent with the short and line log formats. To see merged revisions for just a given revision, use bzr log -n0 -rX. To see every merged revision, use bzr log -n0. (Ian Clatworthy)

14.1.5 New Features

- New formats 1.14 and 1.14-rich-root supporting End-Of-Line (EOL) conversions, keyword templating (via the bzr-keywords plugin) and generic content filtering. These formats replace the experimental development-wt5 and development-wt5-rich-root formats respectively, but have support for filtered views disabled. (Ian Clatworthy)

- New mv --auto option recognizes renames after they occur. (Aaron Bentley)

- bzr can now get passwords from stdin without requiring a controlling terminal (i.e. by redirecting stdin). (Vincent Ladeuil)

- bzr log now supports filtering of multiple files and directories and will show changes that touch any of them. Furthermore, directory filtering now shows the changes to any children of that directory, not just the directory object itself. (Ian Clatworthy, #97715)

- bzr shelve can now apply changes without storing anything on the shelf, via the new –destroy option. (Aaron Bentley)

- bzr send now accepts --body to specify an initial message body. (Aaron Bentley)

- bzr xxx --usage where xxx is a command now shows a usage message and the options without the descriptive help sections (like Description and Examples). A message is also given explaining how to see the complete help, i.e. bzr help xxx. (Ian Clatworthy)

- Content filters can now be used to provide custom conversion between the canonical format of content (i.e. as stored) and the convenience format of content (i.e. as created in working trees). See bzr help content-filters for further details. (Ian Clatworthy, Alexander Belchenko)

- End-of-line conversion is now supported for formats supporting content filtering. See bzr help eol for details. (Ian Clatworthy)

- Newly-blessed join command allows combining two trees into one. (Aaron Bentley)

14.1.6 Improvements

- A new format name alias default-rich-root has been added and points at the closest relative of the default format that supports rich roots. (Jelmer Vernooij, #338061)

- Branching from a stacked branch using bzr+:// will now stream the data when the target repository does not need topological ordering, reducing round trips and network overhead. This uses the existing smart server methods added in 1.13, so will work on any 1.13 or newer server. (Robert Collins, Andrew Bennetts)

- bzr cat and bzr export now supports a --filters option that displays/saves the content after content filters are applied. (Ian Clatworthy)
• `bzr ignore` gives a more informative message when existing version controlled files match the ignore pattern. (Neil Martinsen-Burrell, #248895)

• `bzr log` now has `--include-merges` as an alias for `--levels 0`. (Ian Clatworthy)

• `bzr send` is faster on repositories with deep histories. (Ian Clatworthy)

• IPv6 literals are accepted in URLs. (stlman, Martin Pool, Jelmer Vernooij, #165014)

• Progress bars now show the rate of network activity for `bzr+ssh://` and `bzr://` connections. (Andrew Bennetts)

• Prompt for user names if they are not in the configuration. (Jelmer Vernooij, #256612)

• Pushing to a stacked pack-format branch on a 1.12 or older smart server now takes many less round trips. (Andrew Bennetts, Robert Collins, #294479)

• Streaming push can be done to older repository formats. This is implemented using a new `Repository.insert_stream_locked` RPC. (Andrew Bennetts, Robert Collins)

• The “ignoring files outside view: ..” message has been re-worded to “Ignoring files outside view. View is ..” to reduce confusion about what was being considered and what was being ignored. (Ian Clatworthy)

• The `long` log formatter now shows [merge] indicators. If only one level of revisions is displayed and merges are found, the `long` and `short` log formatters now tell the user how to see the hidden merged revisions. (Ian Clatworthy)

• The `brisbane-core` project has delivered its beta format `development6-rich-root`. This format is suitable for judicious testing by early adopters. In particular if you are benchmarking bzr performance please be sure to test using this format. At this stage more information is best obtained by contacting the Bazaar mailing list or IRC channel if you are interested in using this format. We will make end user documentation available closer to blessing the format as production ready. (Robert Collins, John Arbash Meinel, Ian Clatworthy, Vincent Ladeuil, Andrew Bennetts, Martin Pool)

• Tildes are no longer escaped. No more `%7Euser/project/branch`! (Jonathan Lange)

### 14.1.7 Bug Fixes

• Pushing a new stacked branch will also push the parent inventories for revisions at the stacking boundary. This makes sure that the stacked branch has enough data to calculate inventory deltas for all of its revisions (without requiring the fallback branch). This avoids “‘AbsentContentFactory’ object has no attribute ‘get_bytes_as’” errors when fetching the stacked branch from a 1.13 (or later) smart server. This partially fixes #354036. (Andrew Bennetts, Robert Collins)

• End-Of-Line content filters are now loaded correctly. (Ian Clatworthy, Brian de Alwis, #355280)

• Authentication plugins now receive all the parameters from the request itself (aka host, port, realm, path, etc). Previously, only the authentication section name, username and encoded password were provided. (Jean-Francois Roy)

• `bzr` gives a better message if an invalid regexp is passed to `bzr log -m`. (Anne Mohsen, Martin Pool)

• `bzr split` now says “See also: join” (Aaron Bentley, #335015)

• `bzr version-info` now works in empty branches. (Jelmer Vernooij, #313028)

• Fix “is not a stackable format” error when pushing a stackable-format branch with an unstackable-format repository to a destination with a default stacking policy. (Andrew Bennetts)

• Fixed incorrect “Source format does not support stacking” warning when pushing to a smart server. (Andrew Bennetts, #334114)

• Fix ‘make check-dist-tarball’ failure by converting paths to unicode when needed. (Vincent Ladeuil, #355454)
• Fixed “Specified file ‘x/y/z’ is outside current view: ” occurring on bzr add x/y/z in formats supporting views when no view is defined. (Ian Clatworthy, #344708)

• It is no longer possible to fetch between repositories while the target repository is in a write group. This prevents race conditions that prevent the use of RPC’s to perform fetch, and thus allows optimising more operations. (Robert Collins, Andrew Bennetts)

• merge --force works again. (Robert Collins, #342105)

• No more warnings are issued about sha being deprecated under python-2.6. (Vincent Ladeuil, #346593)

• Pushing a new branch to a server that has a stacking policy will now upgrade from the local branch format when the stacking policy points at a branch which is itself stackable, because we know the client can read both branches, we know that the trunk for the project can be read too, so the upgrade will not inconvenience users. (Robert Collins, #345169)

• Pushing a new stacked branch will also push the parent inventories for revisions at the stacking boundary. This makes sure that the stacked branch has enough data to calculate inventory deltas for all of its revisions (without requiring the fallback branch). This avoids “‘AbsentContentFactory’ object has no attribute ‘get_bytes_as’” errors when fetching the stacked branch from a 1.13 (or later) smart server. This partially fixes #354036. (Andrew Bennetts, Robert Collins)

• The full test suite is passing again on OSX. Several minor issues (mostly test related) have been fixed. (Vincent Ladeuil, #355273).

• The GNU Changelog formatter is slightly improved in the case where the delta is empty, and now correctly claims not to support tags. (Andrea Bolognani)

• Shelve can now shelve changes to a symlink target. (James Westby, #341558)

• The help for the info command has been corrected. (Ian Clatworthy, #351931)

• Upgrade will now use a sensible default format if the source repository uses rich roots. (Jelmer Vernooij, #252908)

### 14.1.8 Documentation

• Expanded the index of the developer documentation. (Eric Siegerman)

• New topic bzr help debug-flags. (Martin Pool)

• The generated manpage now explicitly lists aliases as commands. (James Westby, #336998)

### 14.1.9 API Changes

• APIs deprecated in 1.6 and previous versions of bzr are now removed. (Martin Pool)

• CommitReporter is no longer called with unchanged status during commit - this was a full-tree overhead that bzr no longer performs. (Robert Collins)

• New abstract UIFactory method get_username which will be called to obtain the username to use when connecting to remote machines. (Jelmer Vernooij)

• New API Inventory.filter() added that filters an inventory by a set of file-ids so that only those fileids, their parents and their children are included. (Ian Clatworthy)

• New sort order for get_record_stream groupcompress which sorts optimally for use with groupcompress compressors. (John Arbash Meinel, Robert Collins)
• Repository APIs get_deltas_for_revisions() and get_revision_delta() now support an optional specific_fileids parameter. If provided, the deltas are filtered so that only those file-ids, their parents and their children are included. (Ian Clatworthy)

• The get_credentials and set_credentials methods of AuthenticationConfig now accept an optional realm argument. (Jean-Francois Roy)

• The pb argument to fetch() is deprecated. (Martin Pool)

• The Serializer class and the serializer format registry have moved from bzrlib.xml_serializer to bzrlib.serializer. (Jelmer Vernooij)

• The smart server jail now hooks into BzrDir.open to prevent any BzrDir that is not inside the backing transport from being opened. See the module documentation for bzrlib.smart.request for details. (Andrew Bennetts, Robert Collins)

• Tree.get_symlink_target now always returns a unicode string result or None. Previously it would return the bytes from reading the link which could be in any arbitrary encoding. (Robert Collins)

14.1.10 Testing

• bzrlib.tests.TestCase now fails the test if its own setUp and tearDown weren’t called. This catches faulty tests that forget to upcall when overriding setUp and tearDown. Those faulty tests were not properly isolated. (Andrew Bennetts, Robert Collins)

• Fix test_msgeditor.MsgEditorTest test isolation. (Vincent Ladeuil, #347130)

• medusa is not used anymore as an FTP test server starting with python2.6. A new FTP test server based on pyftplib can be used instead. This new server is a soft dependency as medusa which is still preferred if both are available (modulo python version). (Vincent Ladeuil)

14.1.11 Internals

• Added chk_map for fast, trie-based storage of tuple to string maps. (Robert Collins, John Arbash Meinel, Vincent Ladeuil)

• Added bzrlib.chk_map for fast, trie-based storage of tuple to string maps. (Robert Collins, John Arbash Meinel, Vincent Ladeuil)

• Added bzrlib.inventory_delta module. This will be used for serializing and deserializing inventory deltas for more efficient streaming on the network. (Robert Collins, Andrew Bennetts)

• Branch._get_config has been added, which splits out access to the specific config file from the branch. This is used to let RemoteBranch avoid constructing real branch objects to access configuration settings. (Robert Collins, Andrew Bennetts)

• Branch now implements set_stacked_on_url in the base class as the implementation is generic and should impact foreign formats. This helps performance for RemoteBranch push operations to new stacked branches. (Robert Collins, Andrew Bennetts)

• BtreeIndex._spill_mem_keys_to_disk() now generates disk index with optimizations turned off. This only has effect when processing > 100,000 keys during something like bzr pack. (John Arbash Meinel)

• bzr selftest now accepts --subunit to run in subunit output mode. Requires lp:subunit installed to work, but is not a hard dependency. (Robert Collins)

• BzrDir.open_branch now takes an optional ignore_fallbacks parameter for controlling opening of stacked branches. (Andrew Bennetts, Robert Collins)
• CommitBuilder has a new method, `record_iter_changes` which works in terms of an `iter_changes` iterator rather than full tree scanning. (Robert Collins)

• DirState can now be passed a custom SHA1Provider object enabling it to store the SHA1 and stat of the canonical (post content filtered) form. (Ian Clatworthy)

• New `assertLength` method based on one Martin has squirreled away somewhere. (Robert Collins, Martin Pool)

• New hook `BzrDir.pre_open` which runs before opening `BzrDir` objects, allowing better enforcement of the smart server jail when dealing with stacked branches. (Robert Collins, Andrew Bennetts)

• New hook `RioVersionInfoBuilder.revision`, allowing extra entries to be added to the stanza that is printed for a particular revision. (Jelmer Vernooij)

• New repository method `refresh_data` to cause any repository to make visible data inserted into the repository by a smart server fetch operation. (Robert Collins, Andrew Bennetts)

• `register_filter_stack_map` now takes an optional fallback parameter, a callable to invoke if a preference has a value not in the map of filter stacks. This enhancement allows, for example, bzr-svn to handle existing svn properties that define a list of keywords to be expanded. (Ian Clatworthy)

• `RemoteBranchConfig` will use a new verb `Branch.set_config_option` to write config settings to smart servers that support this, saving 5 round trips on the stacked streaming acceptance test. (Robert Collins, Andrew Bennetts)

• `RemoteBranch` now provides `_get_config` for access to just the branch specific configuration from a remote server, which uses the already existing `Branch.get_config_file` smart verb. (Robert Collins, Andrew Bennetts)

• `RemoteRepository` will now negatively cache missing revisions during `get_parent_map` while read-locked. Write-locks are unaffected. (Robert Collins, Andrew Bennetts)

• Removed `InterRemoteToOther`, `InterOtherToRemote` and `InterPackToRemotePack` classes, as they are now unnecessary. (Andrew Bennetts)

• `RepositoryFormat` as a new attribute `fast_deltas` to indicate whether the repository can efficiently generate deltas between trees regardless of tree size. (Robert Collins)

• `Repository.iter_files_bytes()` now properly returns an “iterable of byte strings” (aka ‘chunked’) for the content. It previously was returning a plain string, which worked, but performed very poorly when building a working tree (file.writelines(str) is very inefficient). This can have a large effect on `bzr checkout` times. (John Arbash Meinel)

• selftest now supports a `-parallel` option, with values of ‘fork’ or ‘subprocess’ to run the test suite in parallel. Currently only Linux machines work, other platforms need patches submitted. (Robert Collins, Vincent Ladeuil)

• `tests.run_suite` has a new parameter `suite_decorators`, a list of callables to use to decorate the test suite. Such decorators can add or remove tests, or even remote the test suite to another machine if desired. (Robert Collins)

• The smart server verb `Repository.get_parent_map` can now include information about ghosts when the special revision `include-missing:` is in the requested parents map list. With this flag, ghosts are included as `missing:REVISION_ID`. (Robert Collins, Andrew Bennetts)

• `_walk_to_common_revisions` will now batch up at least 50 revisions before calling `get_parent_map` on the target, regardless of `InterRepository`. (Andrew Bennetts, Robert Collins)
15.1  **bzr 1.13**

**Codename**  paraskavedekatriaphobia

**1.13**  2009-03-14

**1.13rc1**  2009-03-10

**1.13.1**  2009-03-23

**1.13.2**  2009-04-27

GNU Changelog output can now be produced by `bzr log --gnu-changelog`. Debug flags can now be set in `~/.bazaar/bazaar.conf`. Lightweight checkouts and stacked branches should both be much faster over remote connections.

15.1.1 **Changes From 1.13.1 to 1.13.2**

A regression was found in the 1.13.1 release. When bzr 1.13.1 and earlier push a stacked branch they do not take care to push all the parent inventories for the transferred revisions. This means that a smart server serving that branch often cannot calculate inventory deltas for the branch (because smart server does not/cannot open fallback repositories). Prior to 1.13 the server did not have a verb to stream revisions out of a repository, so that’s why this bug has appeared now.

15.1.2 **Bug Fixes**

- Fix for bug 354036 ErrorFromSmartServer - AbsentContentFactory object has no attribute ‘get_bytes_as’ exception while pulling from Launchpad (Jean-Francois Roy, Andrew Bennetts, Robert Collins)

15.1.3 **Changes From 1.13final to 1.13.1**

A couple regressions where found in the 1.13 release. The pyrex-generated C extensions are missing from the .tar.gz and .zip files. Documentation on how to generate GNU ChangeLogs is wrong.

15.1.4 **Bug Fixes**

- Change `./bzr's _script_version` to match `./bzrlib/__init__.py version_info`. (Bob Tanner, Martin Pool, #345232)
• Distribution archives for 1.13 do not contain generated C extension modules (Jean-Francois Roy, Bob Tanner, #344465)
• GNU ChangeLog output can now be produced by `bzr log --format gnu-changelog` is incorrect (Deejay, Bob Tanner, Martin Pool, Robert Collins, #343928)
• `merge --force` works again. (Robert Collins, #342105)

15.1.5 Changes From 1.13rc1 to 1.13final
• Fix “is not a stackable format” error when pushing a stackable-format branch with an unstackable-format repository to a destination with a default stacking policy. (Andrew Bennetts)
• Progress bars now show the rate of network activity for `bzr+ssh://` and `bzr://` connections. (Andrew Bennetts)

15.1.6 Compatibility Breaks
• `bzr log --line` now indicates which revisions are merges with `[merge]` after the date. Scripts which parse the output of this command may need to be adjusted. (Neil Martinsen-Burrell)

15.1.7 New Features
• `bzr reconfigure` now supports `--with-trees` and `--with-no-trees` options to change the default tree-creation policy of shared repositories. (Matthew Fuller, Marius Kruger, #145033)
• Debug flags can now be set in `~/.bazaar/bazaar.conf`. (Martin Pool)
• Filtered views provide a mask over the tree so that users can focus on a subset of a tree when doing their work. See Filtered views in chapter 7 of the User Guide and `bzr help view` for details. (Ian Clatworthy)
• GNU Changelog output can now be produced by `bzr log --gnu-changelog`. (Andrea Bolognani, Martin Pool)
• The `-Dmemory` flag now gives memory information on Windows. (John Arbash Meinel)
• Multiple authors for a commit can now be recorded by using the “`--author`” option multiple times. (James Westby, #185772)
• New clean-tree command, from bzrtools. (Aaron Bentley, Jelmer Vernooij)
• New command `bzr launchpad-open` opens a Launchpad web page for that branch in your web browser, as long as the branch is on Launchpad at all. (Jonathan Lange)
• New API for getting bugs fixed by a revision: Revision.iter_bugs(). (Jonathan Lange)

15.1.8 Improvements
• All `bzr Hooks` classes are now registered in `bzrlib.hooks.known_hooks`. This removes the separate list from `bzrlib.tests` and ensures that all hooks registered there are correctly isolated by the test suite (previously `MutableTreeHooks` were not being isolated correctly). Further, documentation for hooks is now dynamically generated from the present HookPoints. `bzr hooks` will now also report on all the hooks present in the `bzrlib.hooks.known_hooks` registry. (Robert Collins)
• `bzr add` no longer prints `add completed` on success. Failure still prints an error message. (Robert Collins)
• \texttt{bzr \ branch} now has a \texttt{--no-tree} option which turns off the generation of a working tree in the new branch. (Daniel Watkins, John Klinger, \#273993)

• Bazaar will now point out \texttt{bzr+ssh://} to the user when they use \texttt{ssh://}. (Jelmer Vernooij, \#330535)

• \texttt{bzr -v info} now omits the number of committers branch statistic, making it many times faster for large projects. To include that statistic in the output, use \texttt{bzr -vv info}. (Ian Clatworthy)

• \texttt{bzr push} to a \texttt{bzr url} (\texttt{bzr://}, \texttt{bzr+ssh://} etc) will stream if the server is version 1.13 or greater, reducing roundtrips significantly. (Andrew Bennetts, Robert Collins)

• Lightweight Checkouts and Stacked Branches should both be much faster over remote connections. Building the working tree now batches up requests into approx 5MB requests, rather than a separate request for each file. (John Arbash Meinel)

• Support for GSSAPI authentication when using HTTP or HTTPS. (Jelmer Vernooij)

• The \texttt{bzr shelve} prompt now includes a ‘?’ help option to explain the short options better. (Daniel Watkins, \#327429)

• \texttt{bzr lp-open} now falls back to the push location if it cannot find a public location. (Jonathan Lange, \#332372)

• \texttt{bzr lp-open} will try to find the Launchpad URL for the location passed on the command line. This makes \texttt{bzr lp-open lp:foo} work as expected. (Jonathan Lange, \#332705)

• \texttt{bzr send} now supports MH-E via \texttt{emacsclient}. (Eric Gillespie)

### 15.1.9 Bug Fixes

• Allows \texttt{bzr log <FILE>} to be called in an empty branch without backtracing. (Vincent Ladeuil, \#346431)

• Bazaar now gives a better message including the filename if it’s unable to read a file in the working directory, for example because of a permission error. (Martin Pool, \#338653)

• \texttt{bzr cat -r<old> <path>} doesn’t traceback anymore when \texttt{<path>} has a file id in the working tree different from the one in revision \texttt{<old>}. (Vincent Ladeuil, \#341517, \#253806)

• \texttt{bzr send help} is more specific about how to apply merge directives. (Neil Martinsen-Burrell, \#253470)

• \texttt{bzr missing} now uses \texttt{Repository.get_revision_delta()} rather than fetching trees and determining a delta itself. (Jelmer Vernooij, \#315048)

• \texttt{bzr push} to a smart server no longer causes “Revision \texttt{[set([\texttt{\{'null':\},]))]} not present ...” errors when the branch has multiple root revisions. (Andrew Bennetts, \#317654)

• \texttt{bzr shelve} now properly handle patches with no terminating newline. (Benoit PIERRE, \#303569)

• \texttt{bzr unshelve} gives a more palatable error if passed a non-integer shelf id. (Daniel Watkins)

• Export now handles files that are not present in the tree. (James Westby, \#174539)

• Fixed incorrect “Source format does not support stacking” warning when pushing to a smart server. (Andrew Bennetts, \#334114)

• Fixed “sprout() got an unexpected keyword argument ‘source_branch‘” error branching from old repositories. (Martin Pool, \#321695)

• Make \texttt{bzr push --quiet <non-local location>} less chatty. (Kent Gibson, \#221461)

• Many Branch hooks would not fire with \texttt{bzr://} and \texttt{bzr+ssh://} branches, and this was not noticed due to a bug in the test logic for branches. This is now fixed and a test added to prevent it reoccurring. (Robert Collins, Andrew Bennetts)
• Restore the progress bar on Windows. We were disabling it when TERM wasn’t set, but Windows doesn’t set TERM. (Alexander Belchenko, #334808)

• `setup.py build_ext` now gives a proper error when an extension fails to build. (John Arbash Meinel)

• Symlinks to non ascii file names are now supported. (Robert Collins, Vincent Ladeuil, #339055, #272444)

• Under rare circumstances (aka nobody reported a bug about it), the FTP transport could revert to ascii mode. It now stays in binary mode except when needed. (Vincent Ladeuil)

• Unshelve does not generate warnings about progress bars. (Aaron Bentley, #328148)

• `shelve` cleans up properly when unversioned files are specified. (Benoît Pierre, Aaron Bentley)

15.1.10 Documentation

• Added Organizing your workspace to the User Guide appendices, summarizing some common ways of organizing trees, branches and repositories and the processes/workflows implied/enabled by each. (Ian Clatworthy)

• Hooks can now be self documenting. `bzrlib.hooks.Hooks.create_hook` is the entry point for this feature. (Robert Collins)

• The documentation for `shelve` and `unshelve` has been clarified. (Daniel Watkins, #327421, #327425)

15.1.11 API Changes

• `bzr selftest` now fails if the bazaar sources contain trailing whitespace, non-unix style line endings and files not ending in a newline. About 372 files and 3243 lines with trailing whitespace was updated to comply with this. The code already complied with the other criteria, but now it is enforced. (Marius Kruger)

• `bzrlib.branch.PushResult` was renamed to `bzrlib.branch.BranchPushResult`. (Jelmer Vernooij)

• `Branch.fetch` and `Repository.fetch` now return None rather than a count of copied revisions and failed revisions. A while back we stopped ever reporting failed revisions because we started erroring instead, and the copied revisions count is not used in the UI at all - indeed it only reflects the repository status not changes to the branch itself. (Robert Collins)

• `Inventory.apply_delta` now raises an AssertionError if a file-id appears multiple times within the delta. (Ian Clatworthy)

• `MutableTree.commit` now favours the “authors” argument, with the old “author” argument being deprecated.

• Remove deprecated EmptyTree. (Martin Pool)

• `Repository.fetch` now accepts an optional `fetch_spec` parameter. A `SearchResult` or `MiniSearchResult` may be passed to `fetch_spec` instead of a `last_revision` to specify exactly which revisions to fetch. (Andrew Bennetts)

• `RepositoryAcquisitionPolicy.acquire_repository` now returns a tuple of `(repository, is_new_flag)`, rather than just the repository. (Andrew Bennetts)

• `Revision.get_apparent_author()` is now deprecated, replaced by `Revision.get_apparent_authors()`, which returns a list. The former now returns the first item that would be returned from the second.

• The `BranchBuilder` test helper now accepts a `timestamp` parameter to `build_commit` and `build_snapshot`. (Martin Pool)
• The `_fetch_*` attributes on `Repository` are now on `RepositoryFormat`, more accurately reflecting their intent (they describe a disk format capability, not state of a particular repository of that format). (Robert Collins)

15.1.12 Internals

• Branching from a non-stacked branch on a smart protocol is now free of virtual file system methods. (Robert Collins, Andrew Bennetts)

• Branch and Repository creation on a `bzr+ssh://server` are now done via RPC calls rather than VFS calls, reducing round trips for pushing new branches substantially. (Robert Collins)

• `Branch.clone` now takes the `repository_policy` formerly used inside `BzrDir.clone_on_transport`, allowing stacking to be configured before the branch tags and revision tip are set. This fixes a race condition cloning stacked branches that would cause plugins to have hooks called on non-stacked instances. (Robert Collins, #334187)

• `BzrDir.cloning_metadir` now has a RPC call. (Robert Collins)

• `BzrDirFormat.__str__` now uses the human readable description rather than the sometimes-absent disk label. (Robert Collins)

• `bzrlib.fetch` is now composed of a sender and a sink component allowing for decoupling over a network connection. Fetching from or into a RemoteRepository with a 1.13 server will use this to stream the operation. (Andrew Bennetts, Robert Collins)

• `bzrlib.tests.run_suite` accepts a runner_class parameter supporting the use of different runners. (Robert Collins)

• Change how file_ids and revision_ids are interned as part of inventory deserialization. Now we use the real `intern()`, rather than our own workaround that would also cache a Unicode copy of the string, and never emptied the cache. This should slightly reduce memory consumption. (John Arbash Meinel)

• New branch method `create_clone_on_transport` that returns a branch object. (Robert Collins)

• New hook Commands[‘extend_command’] to allow plugins to access a command object before the command is run (or help generated from it), without overriding the command. (Robert Collins)

• New version of the `BzrDir.find_repository` verb supporting `_network_name` to support removing more `_ensure_real` calls. (Robert Collins)

• `RemoteBranchFormat` no longer claims to have a disk format string. (Robert Collins)

• `Repository` objects now have `suspend_write_group` and `resume_write_group` methods. These are currently only useful with pack repositories. (Andrew Bennetts, Robert Collins)

• `BzrDirFormat`, `BranchFormat` and `RepositoryFormat` objects now have a `network_name` for passing the format across RPC calls. (Robert Collins, Andrew Bennetts)

• `RepositoryFormat` objects now all have a new attribute `_serializer` used by fetch when reserialising is required. (Robert Collins, Andrew Bennetts)

• Some methods have been pulled up from `BzrBranch` to `Branch` to aid branch types that are not `bzr` branch objects (like `RemoteBranch`). (Robert Collins, Andrew Bennetts)

• Test adaptation has been made consistent throughout the built in tests. `TestScenarioApplier`, `multiply_tests_from_modules`, `adapt_tests`, `adapt_modules` have all been deleted. Please use `multiply_tests`, or for lower level needs `apply_scenarios` and `apply_scenario`. (Robert Collins)
- TestSkipped is now detected by TestCase and passed to the TestResult by calling addSkip. For older TestResult objects, where addSkip is not available, addError is still called. This permits test filtering in subunit to strip out skipped tests resulting in a faster fix-shrink-list-run cycle. This is compatible with the testtools protocol for skips. (Robert Collins)

- The _index of KnitVersionedFiles now supports the ability to scan an underlying index that is going to be incorporated into the KnitVersionedFiles object, to determine if it has missing delta references. The method is scan_unvalidated_index. (Andrew Bennetts, Robert Collins)

- There is a RemoteSink object which handles pushing to smart servers. (Andrew Bennetts, Robert Collins)

- TransportTraceDecorator now logs put_bytes_non_atomic and rmdir calls. (Robert Collins)

- VersionedFiles record adapters have had their signature change from (record, record.get_bytes_as(record.storage_kind)) to (record) reducing excess duplication and allowing adapters to access private data in record to obtain content more efficiently. (Robert Collins)

- We no longer probe to see if we should create a working tree during clone if we cannot get a local_abspath for the new bzrdir. (Robert Collins)
BAZAAR RELEASE NOTES

16.1 bzr 1.12

Codename 1234567890
1.12 2009-02-13
1.12rc1 2009-02-10

This release of Bazaar contains many improvements to the speed, documentation and functionality of `bzr log` and the display of logged revisions by `bzr status`. bzr now also gives a better indication of progress, both in the way operations are drawn onto a text terminal, and by showing the rate of network IO.

16.1.1 Changes from RC1 to Final

- `bzr init --development-wt5[-rich-root]` would fail because of circular import errors. (John Arbash Meinel, #328135)

- Expanded the help for log and added a new help topic called `log-formats`. (Ian Clatworthy)

16.1.2 Compatibility Breaks

- By default, `bzr status` after a merge now shows just the pending merge tip revisions. This improves the signal-to-noise ratio after merging from trunk and completes much faster. To see all merged revisions, use the new `-v` flag. (Ian Clatworthy)

- `bzr log --line` now shows any tags after the date and before the commit message. If you have scripts which parse the output from this command, you may need to adjust them accordingly. (Ian Clatworthy)

- `bzr log --short` now shows any additional revision properties after the date and before the commit message. Scripts that parse output of the log command in this situation may need to adjust. (Neil Martinsen-Burrell)

- The experimental formats 1.12-preview and 1.12-preview-rich-root have been renamed `development-wt5` and `development-wt5-rich-root` respectively, given they are not ready for release in 1.12. (Ian Clatworthy)

- `read_bundle_from_url` has been deprecated. (Vincent Ladeuil)
16.1.3 New Features

• Add support for filtering `bzr missing` on revisions. Remote revisions can be filtered using `bzr missing -r -20..-10` and local revisions can be filtered using `bzr missing --my-revision -20..-10`. (Marius Kruger)

• `bzr log -p` displays the patch diff for each revision. When logging a file, the diff only includes changes to that file. (Ian Clatworthy, #202331, #227335)

• `bzr log` supports a new option called `--level N`. A value of 0 (zero) means “show all nested merge revisions” while a value of 1 (one) means “show just the top level”. Values above 1 can be used to see a limited amount of nesting. That can be useful for seeing the level or two below PQM submits for example. To force the `--short` and `--line` formats to display all nested merge revisions just like `--long` does by default, use a command like `bzr log --short --line --n0`. To display just the mainline using `--long` format, `bzr log --long --n1`. (Ian Clatworthy)

16.1.4 Improvements

• `bzr add` more clearly communicates success vs failure. (Daniel Watkins)

• `bzr init` will now print a little less verbose output. (Marius Kruger)

• `bzr log` is now much faster in many use cases, particularly at incrementally displaying results and filtering by a revision range. (Ian Clatworthy)

• `bzr log --short` and `bzr log --line` now show tags, if any, for each revision. The tags are shown comma-separated inside `{}`. For short format, the tags appear at the end of line before the optional [merge] indicator. For line format, the tags appear after the date. (Ian Clatworthy)

• Progress bars now show the rate of activity for some SFTP operations, and they are drawn different. (Martin Pool, #172741)

• Progress bars now show the rate of activity for urllib and pycurl based HTTP client implementations. The operations are tracked at the socket level for better precision. (Vincent Ladeuil)

• Rule-based preferences can now accept multiple patterns for a set of rules. (Marius Kruger)

• The ancestor: revision spec will now default to referring to the parent of the branch if no other location is given. (Daniel Watkins, #198417)

• The debugger started as a result of setting `$BZR_PDB` works around a bug in `pdb`, `http://bugs.python.org/issue4150`. The bug can cause truncated tracebacks in Python versions before 2.6. (Andrew Bennetts)

• VirtualVersionedFiles now implements `iter_lines_added_or_present_in_keys`. This allows the creation of new branches based on stacked `bzr-svn` branches. (#311997)

16.1.5 Bug Fixes

• `bzr annotate --show-ids` doesn’t give a backtrace on empty files anymore. (Anne Mohsen, Vincent Ladeuil, #314525)

• `bzr log FILE` now correctly shows mainline revisions merging a change to `FILE` when the `--short` and `--line` log formats are used. (Ian Clatworthy, #317417)

• `bzr log -rX..Y FILE` now shows the history of `FILE` provided it existed in `Y` or `X`, even if the file has since been deleted or renamed. If no range is given, the current/basis tree and initial tree are searched in that order. More generally, log now interprets filenames in their historical context. (Ian Clatworthy, #175520)
• **bzr status** now reports nonexistent files and continues, then errors (with code 3) at the end. (Karl Fogel, #306394)

• Don’t require the present compression base in knits to be the same when adding records in knits. (Jelmer Vernooij, #307394)

• Fix a problem with CIFS client/server lag on Windows colliding with an invariant-per-process algorithm for generating AtomicFile names (Adrian Wilkins, #304023)

• Many socket operations now handle EINTR by retrying the operation. Previously EINTR was treated as an unrecoverable failure. There is a new `until_no_eintr` helper function in `bzrlib.osutils`. (Andrew Bennetts)

• Support symlinks with non-ascii characters in the symlink filename. (Jelmer Vernooij, #319323)

• There was a bug in how we handled resolving when a file is deleted in one branch, and modified in the other. If there was a criss-cross merge, we would cause the deletion to conflict a second time. (Vincent Ladeuil, John Arbash Meinel)

• There was another bug in how we chose the correct intermediate LCA in criss-cross merges leading to several kind of changes be incorrectly handled. (John Arbash Meinel, Vincent Ladeuil)

• Unshelve now handles deleted paths without crashing. (Robert Collins)

### 16.1.6 Documentation

• Improved plugin developer documentation. (Martin Pool)

### 16.1.7 API Changes

• `ProgressBarStack` is deprecated; instead use `ui_factory.nested_progress_bar` to create new progress bars. (Martin Pool)

• `ForeignVcsMapping()` now requires a `ForeignVcs` object as first argument. (Jelmer Vernooij)

• `ForeignVcsMapping.show_foreign_revid()` has been moved to `ForeignVcs`. (Jelmer Vernooij)

• `read_bundle_from_url` is deprecated in favor of `read_mergeable_from_url`. (Vincent Ladeuil)

• Revision specifiers are now registered in `bzrlib.revisionspec.revspec_registry`, and the old list of revisionspec classes (`bzrlib.revisionspec.SPEC_TYPES`) has been deprecated. (Jelmer Vernooij, #321183)

• The progress and UI classes have changed; the main APIs remain the same but code that provides a new UI or progress bar class may need to be updated. (Martin Pool)

### 16.1.8 Internals

• Default User Interface (UI) is CLIUIFactory when bzr runs in a dumb terminal. It is sometimes desirable do override this default by forcing bzr to use TextUIFactory. This can be achieved by setting the BZR_USE_TEXT_UI environment variable (emacs shells, as opposed to compile buffers, are such an example). (Vincent Ladeuil)

• New API `Branch.iter_merge_sorted_revisions()` that iterates over `(revision_id, depth, revno, end_of_merge)` tuples. (Ian Clatworthy)

• New `Branch.dotted_revno_to_revision_id()` and `Branch.revision_id_to_dotted_revno()` APIs that pick the most efficient way of doing the mapping. (Ian Clatworthy)
• Refactor cmd_serve so that it’s a little easier to build commands that extend it, and perhaps even a bit easier to read. (Jonathan Lange)

• TreeDelta.show() now accepts a filter parameter allowing log formaters to restrict the output. (Vincent Ladeuil)
17.1 bzr 1.11

Codename  “Eyes up!”

Released  2009-01-19

This first monthly release of Bazaar for 2009 improves Bazaar’s operation in Windows, Mac OS X, and other situations where file names are matched without regard to capitalization: Bazaar tries to match the case of an existing file. This release of Bazaar also improves the efficiency of Tortoise Windows Shell integration and lets it work on 64-bit platforms.

The UI through which Bazaar supports historic formats has been improved, so ‘bzr help formats’ now gives a simpler and shorter list, with clear advice.

This release also fixes a number of bugs, particularly a glitch that can occur when there are concurrent writes to a pack repository.

17.1.1 Bug Fixes

• Fix failing test when CompiledChunksToLines is not available. (Vincent Ladeuil)
• Stacked branches don’t repeatedly open their transport connection. (John Arbash Meinel)

17.2 bzr 1.11rc1

Codename  “Eyes up!”

Released  2009-01-09

17.2.1 Changes

• Formats using Knit-based repository formats are now explicitly marked as deprecated. (Ian Clatworthy)

17.2.2 New Features

• Add support for *bzr tags -r 1..2*, that is we now support showing tags applicable for a specified revision range. (Marius Kruger)
• **authentication.conf** now accepts pluggable read-only credential stores. Such a plugin (netrc_credential_store) is now included, handles the $HOME/.netrc file and can server as an example to implement other plugins. (Vincent Ladeuil)

• **shelve --list** can now be used to list shelved changes. (Aaron Bentley)

## 17.2.3 Improvements

• Add trailing slash to directories in all output of b扎r ls, except b扎r ls --null. (Gordon P. Hemsley, #306424)

• **b扎r revision-info** now supports a -d option to specify an alternative branch. (Michael Hudson)

• Add connection to a C++ implementation of the Windows Shell Extension which is able to fully replace the current Python implemented one. Advantages include 64bit support and reduction in overhead for processes which drag in shell extensions. (Mark Hammond)

• Support the Claws mail client directly, rather than via xdg-email. This prevents the display of an unnecessary modal dialog in Claws, informing the user that a file has been attached to the message, and works around bug #291847 in xdg-utils which corrupts the destination address.

• When working on a case-insensitive case-preserving file-system, as commonly found with Windows, b扎r will often ignore the case of the arguments specified by the user in preference to the case of an existing item on the file-system or in the inventory to help prevent counter-intuitive behaviour on Windows. (Mark Hammond)

## 17.2.4 Bug Fixes

• Allow BzrDir implementation to implement backing up of control directory. (#139691)

• **b扎r push** creating a new stacked branch will now only open a single connection to the target machine. (John Arbash Meinel)

• Don’t call iteritems on transport_list_registry, because it may change during iteration. (Martin Pool, #277048)

• Don’t make a broken branch when pushing an unstackable-format branch that’s in a stackable shared repository to a location with default stack-on location. (Andrew Bennetts, #291046)

• Don’t require embedding user in HTTP(S) URLs do use authentication.conf. (Ben Jansen, Vincent Ladeuil, #300347)

• Fix a problem with CIFS client/server lag on windows colliding with an invariant-per-process algorithm for generating AtomicFile names (Adrian Wilkins, #304023)

• Fix bogus setUp signature in UnavailableFTPServer. (Gary van der Merwe, #313498)

• Fix compilation error in _dirstate_helpers_c on SunOS/Solaris. (Jari Aalto)

• Fix SystemError in _patiencediff_c module by calling PyErr_NoMemory() before returning NULL in PatienceSequenceMatcher_new. (Andrew Bennetts, #303206)

• Give proper error message for diff with non-existent dotted revno. (Marius Kruger, #301969)

• Handle EACCES (permission denied) errors when launching a message editor, and emit warnings when a configured editor cannot be started. (Andrew Bennetts)

• $HOME/.netrc file is now recognized as a read-only credential store if configured in authentication.conf with ‘password_encoding=netrc’ in the appropriate sections. (Vincent Ladeuil, #103029)

• Opening a stacked branch now properly shares the connection, rather than opening a new connection for the stacked-on branch. (John Arbush meinel)
• Preserve transport decorators while following redirections. (Vincent Ladeuil, #245964, #270863)
• Provides a finer and more robust filter for accepted redirections. (Vincent Ladeuil, #303959, #265070)
• shelve paths are now interpreted relative to the current working tree. (Aaron Bentley)
• Transport.readv() defaults to not reading more than 100MB in a single array. Further RemoteTransport.readv sets this to 5MB to work better with how it splits its requests. (John Arbash Meinel, #303538)
• Pack repositories are now able to reload the pack listing and retry the current operation if another action causes the data to be repacked. (John Arbash Meinel, #153786)
• pull -v now respects the log_format configuration variable. (Aaron Bentley)
• push -v now works on non-initial pushes. (Aaron Bentley)
• Use the short status format when the short format is used for log. (Vincent Ladeuil, #87179)
• Allow files to be renamed or moved via remove + add-by-id. (Charles Duffy, #314251)

17.2.5 Documentation

• Improved the formats help topic to explain why multiple formats exist and to provide guidelines in selecting one. Introduced two new supporting help topics: current-formats and other-formats. (Ian Clatworthy)

17.2.6 API Changes

• LRUCache(after_cleanup_size) was renamed to after_cleanup_count and the old name deprecated. The new name is used for clarity, and to avoid confusion with LRUSizeCache(after_cleanup_size). (John Arbash Meinel)
• New ForeignRepository base class, to help with foreign branch support (e.g. svn). (Jelmer Vernooij)
• node_distances and select_farthest can no longer be imported from bzrlib.graph. They can still be imported from bzrlib.deprecated_graph, which has been the preferred way to import them since before 1.0. (Andrew Bennetts)
• The logic in commit now delegates inventory basis calculations to the CommitBuilder object; this requires that the commit builder in use has been updated to support the new recording_deletes and record_delete methods. (Robert Collins)

17.2.7 Testing

• An HTTPS server is now available (it requires python-2.6). Future bzr versions will allow the use of the python-2.6 ssl module that can be installed for 2.5 and 2.4.
• bzr selftest now fails if new trailing white space is added to the bazaar sources. It only checks changes not committed yet. This means that PQM will now reject changes that introduce new trailing whitespace. (Marius Kruger)
• Introduced new experimental formats called 1.12-preview and 1.12-preview-rich-root to enable testing of related pending features, namely content filtering and filtered views. (Ian Clatworthy)
17.2.8 Internals

- Added an *InventoryEntry* cache when deserializing inventories. Can cut the time to iterate over multiple *RevisionsTrees* in half. (John Arbash Meinel)

- Added *bzrlib.fifo_cache.FIFOCache* which is designed to have minimal overhead versus using a plain dict for cache hits, at the cost of not preserving the ‘active’ set as well as an *LRUCache*. (John Arbash Meinel)

- *bzrlib.patience_diff.unified_diff* now properly uses a tab character to separate the filename from the date stamp, and doesn’t add trailing whitespace when a date stamp is not supplied. (Adeodato Simó, John Arbash Meinel)

- *DirStateWorkingTree* and *DirStateWorkingTreeFormat* added as base classes of *WorkingTree4* and *WorkingTreeFormat4* respectively. (Ian Clatworthy)

- *KnitVersionedFiles._check_should_delta()* now uses the *get_build_details* api to avoid multiple hits to the index, and to properly follow the *compression_parent* rather than assuming it is the left-hand parent. (John Arbash Meinel)

- *KnitVersionedFiles.get_record_stream()* will now chose a more optimal ordering when the keys are requested ‘unordered’. Previously the order was fully random, now the records should be returned from each pack in turn, in forward I/O order. (John Arbash Meinel)

- *mutter()* will now flush the ~/.bzr.log if it has been more than 2s since the last time it flushed. (John Arbash Meinel)

- New method *bzrlib.repository.Repository.add_inventory_by_delta* allows adding an inventory via an inventory delta, which can be more efficient for some repository types. (Robert Collins)

- *Repository CommitBuilder* objects can now accumulate an inventory delta. To enable this functionality call builder.recording_deletes and additionally call builder.record_delete when a delete against the basis occurs. (Robert Collins)

- The default HTTP handler has been changed from pycurl to urllib. The default is still pycurl for https connections. (The only advantage of pycurl is that it checks ssl certificates.) (John Arbash Meinel)

- *VersionedFiles.get_record_stream()* can now return objects with a storage_kind of chunked. This is a collection (list/tuple) of strings. You can use osutils.chunks_to_lines() to turn them into guaranteed ‘lines’ or you can use ”.join(chunks) to turn it into a fulltext. This allows for some very good memory savings when asking for many texts that share ancestry, as the individual chunks can be shared between versions of the file. (John Arbash Meinel)

- pull -v and push -v use new function *bzrlib.log.show_branch_change* (Aaron Bentley)
BAZAAR RELEASE NOTES

18.1 bzr 1.10

Released 2008-12-05

Bazaar 1.10 has several performance improvements for copying revisions (especially for small updates to large projects). There has also been a significant amount of effort in polishing stacked branches. The commands `shelve` and `unshelve` have become core commands, with an improved implementation.

The only changes versus bzr-1.10rc1 are bugfixes for stacked branches.

18.1.1 bug Fixes

• Don’t set a pack write cache size from RepoFetcher, because the cache is not coherent with reads and causes ShortReadvErrors. This reverses the change that fixed #294479. (Martin Pool, #303856)

• Properly handle when a revision can be inserted as a delta versus when it needs to be expanded to a fulltext for stacked branches. There was a bug involving merge revisions. As a method to help prevent future difficulties, also make stacked fetches sort topologically. (John Arbash Meinel, #304841)

18.2 bzr 1.10rc1

Released 2008-11-28

This release of Bazaar focuses on performance improvements when pushing and pulling revisions, both locally and to remote networks. The popular `shelve` and `unshelve` commands, used to interactively revert and restore work in progress, have been merged from bzrtools into the bzr core. There are also bug fixes for portability, and for stacked branches.

18.2.1 New Features

• New `commit_message_template` hook that is called by the commit code to generate a template commit message. (Jelmer Vernooij)

• New `shelve` and `unshelve` commands allow undoing and redoing changes. (Aaron Bentley)
18.2.2 Improvements
• (Remote)Branch.copy_content_into no longer generates the full revision history just to set the last revision info. (Andrew Bennetts, John Arbash Meinel)
• Fetches between formats with different serializers (such as pack-0.92-subtree and 1.9-rich-root) are faster now. This is due to operating on batches of 100 revisions at time rather than one-by-one. (Andrew Bennetts, John Arbash Meinel)
• Search index files corresponding to pack files we’ve already used before searching others, because they are more likely to have the keys we’re looking for. This reduces the number of iix and tix files accessed when pushing a new revision, for instance. (John Arbash Meinel)
• Signatures to transfer are calculated more efficiently in item_keys_introduced_by. (Andrew Bennetts, John Arbash Meinel)
• The generic fetch code can once again copy revisions and signatures without extracting them completely to fulltexts and then serializing them back down into byte strings. This is a significant performance improvement when fetching from a stacked branch. (John Arbash Meinel, #300289)
• When making a large readv() request over bzr+ssh, break up the request into manageable chunks. Because the RPC is still not capable of streaming, this helps keep us from buffering too much information at once. (John Arbash Meinel)

18.2.3 Bug Fixes
• Better message when the user needs to set their Launchpad ID. (Martin Pool, #289148)
• bzr commit --local doesn’t access the master branch anymore. This fixes a regression introduced in 1.9. (Marius Kruger, #299313)
• Don’t call the system chdir() with an empty path. Sun OS seems to give an error in that case. Also, don’t count on getcwd() being able to allocate a new buffer, which is a gnu extension. (John Arbash Meinel, Martin Pool, Harry Hirsch, #297831)
• Don’t crash when requesting log --forward <file> for a revision range starting with a dotted revno. (Vincent Ladeuil, #300055)
• Don’t create text deltas spanning stacked repositories; this could cause “Revision X not present in Y” when later accessing them. (Martin Pool, #288751)
• Pack repositories are now able to reload the pack listing and retry the current operation if another action causes the data to be repacked. (John Arbash Meinel, #153786)
• PermissionDenied errors from smart servers no longer cause “PermissionDenied: “None”” on the client. (Andrew Bennetts, #299254)
• Pushing to a stacked pack repository now batches writes, the same way writes are batched to ordinary pack repository. This makes pushing to a stacked branch over the network much faster. (Andrew Bennetts, #294479)
• TooManyConcurrentRequests no longer occur when a fetch fails and tries to abort a write group. This allows the root cause (e.g. a network interruption) to be reported. (Andrew Bennetts, #297014)
• RemoteRepository.get_parent_map now uses fallback repositories. (Aaron Bentley, #297991?, #293679?)

18.2.4 API Changes
• CommitBuilder now validates the strings it will be committing, to ensure that they do not have characters that will not be properly round-tripped. For now, it just checks for characters that are invalid in the XML form.
(John Arbash Meinel, #295161)

- Constructor parameters for NewPack (internal to pack repositories) have changed incompatibly.

- `Repository.abort_write_group` now accepts an optional `suppress_errors` flag. Repository implementations that override `abort_write_group` will need to be updated to accept the new argument. Subclasses that only override `_abort_write_group` don’t need to change.

- Transport implementations must provide `copy_tree_to_transport`. A default implementation is provided for Transport subclasses.

### 18.2.5 Testing

- `bzr selftest` now fails if no doctests are found in a module that’s expected to have them. (Martin Pool)

- Doctests now only report the first failure. (Martin Pool)
BAZAAR RELEASE NOTES

19.1  bzr 1.9

Released 2008-11-07

This release of Bazaar adds a new repository format, 1.9, with smaller and more efficient index files. This format can be specified when creating a new repository, or used to losslessly upgrade an existing repository. bzr 1.9 also speeds most operations over the smart server protocol, makes annotate faster, and uses less memory when making checkouts or pulling large amounts of data.

19.1.1 Bug Fixes

• Fix “invalid property value ‘branch-nick’ for None” regression with branches bound to svn branches. (Martin Pool, #293440)
• Fix SSL/https on Python2.6. (Vincent Ladeuil, #293054)
• SFTPTransport.readv() had a bug when requests were out-of-order. This only triggers some-of-the-time on Knit format repositories. (John Arbash Meinel, #293746)

19.2  bzr 1.9rc1

Released 2008-10-31

19.2.1 New Features

• New Branch hook transform_fallback_location allows a function to be called when looking up the stacked source. (Michael Hudson)
• New repository formats 1.9 and 1.9-rich-root. These have all the functionality of 1.6, but use the new btree indexes. These indexes are both smaller and faster for access to historical information. (John Arbash Meinel)

19.2.2 Improvements

• BTreeIndex code now is able to prefetch extra pages to help tune the tradeoff between bandwidth and latency. Should be tuned appropriately to not impact commands which need minimal information, but provide a
significant boost to ones that need more context. Only has a direct impact on the --development2 format which uses btree’s for the indexes. (John Arbash Meinel)

• **bzr dump-btree** is a hidden command introduced to allow dumping the contents of a compressed btree file. (John Arbash Meinel)

• **bzr pack** now tells the index builders to optimize for size. For btree index repositories, this can save 25% of the index size (mostly in the text indexes). (John Arbash Meinel)

• **bzr push** to an existing branch or repository on a smart server is faster, due to Bazaar making more use of the get_parent_map RPC when querying the remote branch’s revision graph. (Andrew Bennetts)

• default username for bzr+ssh:// and sftp:// can be configured in authentication.conf. (Aaron Bentley)

• launchpad-login now provides a default username for bzr+ssh and SFTP URLs, allowing username-free URLs to work for everyone. (Aaron Bentley)

• **lp:** lookups no longer include usernames, making them shareable and shorter. (Aaron Bentley)

• New PackRepository.autopack smart server RPC, which does autopacking entirely on the server. This is much faster than autopacking via plain file methods, which downloads a large amount of pack data and then re-uploads the same pack data into a single file. This fixes a major (although infrequent) cause of lengthy delays when using a smart server. For example, pushing the 10th revision to a repository with 9 packs now takes 44 RPCs rather than 179, and much less bandwidth too. This requires Bazaar 1.9 on both the client and the server, otherwise the client will fallback to the slower method. (Andrew Bennetts)

19.2.3 Bug Fixes

• A failure to load a plugin due to an IncompatibleAPI exception is now correctly reported. (Robert Collins, #279451)

• API versioning support now has a multiple-version checking api require_any_api. (Robert Collins, #279447)

• **bzr branch --stacked** from a smart server to a standalone branch works again. This fixes a regression in 1.7 and 1.8. (Andrew Bennetts, #270397)

• **bzr co** uses less memory. It used to unpack the entire WT into memory before writing it to disk. This was a little bit faster, but consumed lots of memory. (John Arbash Meinel, #269456)

• **bzr missing --quiet** no longer prints messages about whether there are missing revisions. The exit code indicates whether there were or not. (Martin Pool, #284748)

• Fixes to the annotate code. The fast-path which re-used the stored deltas was accidentally disabled all the time, instead of only when a branch was stacked. Second, the code would accidentally re-use a delta even if it wasn’t against the left-parent, this could only happen if **bzr reconcile** decided that the parent ordering was incorrect in the file graph. (John Arbash Meinel)

• “Permission denied” errors that occur when pushing a new branch to a smart server no longer cause tracebacks. (Andrew Bennetts, #278673)

• Some compatibility fixes for building the extensions with MSVC and for python2.4. (John Arbash Meinel, #277484)

• The index logic is now able to reload the list of pack files if and index ends up disappearing. We still don’t reload if the pack data itself goes missing after checking the index. This bug appears as a transient failure (file not found) when another process is writing to the repository. (John Arbash Meinel, #153786)

• **bzr switch** and **bzr bind** will now update the branch nickname if it was previously set. All checkouts will now refer to the bound branch for a nickname if one was not explicitly set. (Marius Kruger, #230903)
19.2.4 Documentation

- Improved hook documentation. (Michael Ernst)

19.2.5 API Changes

- commands.plugins_cmds is now a CommandRegistry, not a dict.

19.2.6 Internals

- New AuthenticationConfig.set_credentials method allows easy programmatic configuration of authentication credentials.
BAZAAR RELEASE NOTES

20.1 bzr 1.8

Released 2008-10-16

Bazaar 1.8 includes several fixes that improve working tree performance, display of revision logs, and merges. The bzr testsuite now passes on OS X and Python 2.6, and almost completely passes on Windows. The smartserver code has gained several bug fixes and performance improvements, and can now run server-side hooks within an HTTP server.

20.1.1 Bug Fixes

- Fix “Must end write group” error when another error occurs during bzr push. (Andrew Bennetts, #230902)

20.1.2 Portability

- Some Pyrex versions require the WIN32 macro defined to compile on that platform. (Alexander Belchenko, Martin Pool, #277481)

20.2 bzr 1.8rc1

Released 2008-10-07

20.2.1 Changes

- bzr log file has been changed. It now uses a different method for determining which revisions to show as merging the changes to the file. It now only shows revisions which merged the change towards your mainline. This simplifies the output, makes it faster, and reduces memory consumption. (John Arbash Meinel)

- bzr merge now defaults to having --reprocess set, whenever --show-base is not supplied. (John Arbash Meinel)

- bzr+http// will now optionally load plugins and write logs on the server. (Marius Kruger)

- bzrlib._dirstate_helpers_c.pyx does not compile correctly with Pyrex 0.9.4.1 (it generates C code which causes segfaults). We explicitly blacklist that version of the compiler for that extension. Packaged versions will include .c files created with pyrex >= 0.9.6 so it doesn’t effect releases, only users running from the source tree. (John Arbash Meinel, #276868)
20.2.2 Features

- bzr is now compatible with python-2.6. python-2.6 is not yet officially supported (nor released, tests were conducted with the dev version of python-2.6rc2), but all known problems have been fixed. Feedback welcome. (Vincent Ladeuil, #269535)

20.2.3 Improvements

- **bzr annotate** will now include uncommitted changes from the local working tree by default. Such uncommitted changes are given the revision number they would get if a commit was done, followed with a ? to indicate that its not actually known. (Robert Collins, #3439)

- **bzr branch** now accepts a `--standalone` option, which creates a standalone branch regardless of the presence of shared repositories. (Daniel Watkins)

- **bzr push** is faster in the case there are no new revisions to push. It is also faster if there are no tags in the local branch. (Andrew Bennetts)

- File changes during a commit will update the tree stat cache. (Robert Collins)

- Location aliases can now accept a trailing path. (Micheal Hudson)

- New hooks `Lock.hooks` when LockDirs are acquired and released. (Robert Collins, MartinPool)

- Switching in heavyweight checkouts uses the master branch’s context, not the checkout’s context. (Adrian Wilkins)

- **status** on large trees is now faster, due to optimisations in the walkdirs code. Of particular note, the walkdirs code now performs a temporary `chdir()` while reading a single directory: if your platform has non thread-local current working directories (and is not windows which has its own implementation), this may introduce a race condition during concurrent uses of bzrlib. The bzrlib CLI will not encounter this as it is single threaded for working tree operations. (Robert Collins)

- The C extensions now build on python 2.4 (Robert Collins, #271939)

- The `--Dhpss` debug flag now reports the number of smart server calls per medium to stderr. This is in addition to the existing detailed logging to the .bzr.log trace file. (Andrew Bennetts)

20.2.4 Bug Fixes

- Avoid random failures arising from misinterpreted `errno` values in `_readdir_pyx.read_dir`. (Martin Pool, #279381)

- Branching from a shared repository on a smart server into a new repository now preserves the repository format. (Andrew Bennetts, #269214)

- **bzr log** now accepts a `--change` option. (Vincent Ladeuil, #248427)

- **bzr missing** now accepts an `--include-merges` option. (Vincent Ladeuil, #233817)

- Don’t try to filter (internally) ‘.bzr’ from the files to be deleted if it’s not there. (Vincent Ladeuil, #272648)

- Fix ‘_in_buffer’ AttributeError when using the -Dhpss debug flag. (Andrew Bennetts)

- Fix TooManyConcurrentRequests errors caused by a connection failure when doing **bzr pull** or **bzr merge** from a bzr+ssh URL. (Andrew Bennetts, #246233)

- Fixed **bzr st -r branch:PATH_TO_BRANCH** where the other branch is in a different repository than the current one. (Lukáš Lalinsky, #144421)

- Make the first line of the manpage preamble a comment again. (David Futcher, #242106)
• Remove use of optional parameter in GSSAPI FTP support, since it breaks newer versions of Python-Kerberos. (Jelmer Vernooij)

• The autopacking logic will now always create a single new pack from all of the content which it deems is worth moving. This avoids the ‘repack a single pack’ bug and should result in better packing overall. (John Arbash Meinel, #242510, #172644)

• Trivial documentation fix. (John Arbash Meinel, #270471)

• `bzr switch` and `bzr bind` will now update the branch nickname if it was previously set. All checkouts will now refer to the bound branch for a nickname if one was not explicitly set. (Marius Kruger, #230903)

20.2.5 Documentation

• Explain revision/range identifiers. (Daniel Clemente)

20.2.6 API Changes

• `CommitBuilder.record_entry_contents` returns one more element in its result tuple - an optional file system hash for the hash cache to use. (Robert Collins)

• `dirstate.DirState.update_entry` will now only calculate the sha1 of a file if it is likely to be needed in determining the output of `iter_changes`. (Robert Collins)

• The `PackRepository`, `RepositoryPackCollection`, `NewPack` classes have a slightly changed interface to support different index types; as a result other users of these classes need to supply the index types they want. (Robert Collins)

20.2.7 Testing

• `bzrlib.tests.repository_implementations` has been renamed to `bzrlib.tests.per_repository` so that we have a common structure (and it is shorter). (John Arbash Meinel, #239343)

• `LocalTransport.abspath()` now returns a drive letter if the transport has one, fixing numerous tests on Windows. (Mark Hammond)

• `PreviewTree` is now tested via `intertree_implementations`. (Aaron Bentley)

• The full test suite is passing again on OSX. (Guillermo Gonzalez, Vincent Ladeuil)

• The full test suite passes when run with `-Eallow_debug`. (Andrew Bennetts)

20.2.8 Internals

• A new hook, `Branch.open`, has been added, which is called when branch objects are opened. (Robert Collins)

• `bzrlib.osutils._walkdirs_utf8` has been refactored into common tree walking, and modular directory listing code to aid future performance optimisations and refactoring. (Robert Collins)

• `bzrlib.trace.debug_memory` can be used to get a quick memory dump in the middle of processing. It only reports memory if `/proc/PID/status` is available. (John Arbash Meinel)

• New method `RevisionSpec.as_tree` for representing the revision specifier as a revision tree object. (Lukáš Lalinský)
• New race-free method on MutableTree `get_file_with_stat` for use when generating stat cache results. (Robert Collins)

• New win32utils.get_local_appdata_location() provides access to a local directory for storing data. (Mark Hammond)

• To be compatible with python-2.6 a few new rules should be observed. ‘message’ attribute can’t be used anymore in exception classes, ‘sha’ and ‘md5’ modules have been deprecated (use osutils.[md5|sha]), object.__init__ and object.__new__ don’t accept parameters anymore. (Vincent Ladeuil)
BAZAAR RELEASE NOTES

21.1 bzr 1.7.1

Released 2008-10-01
No changes from 1.7.1rc1.

21.2 bzr 1.7.1rc1

Released 2008-09-24
This release just includes an update to how the merge algorithm handles file paths when we encounter complex history.

21.2.1 Features

• If we encounter a criss-cross in history, use information from direct Least Common Ancestors to resolve inventory shape (locations of files, adds, deletes, etc). This is similar in concept to using --lca for merging file texts, only applied to paths. (John Arbash Meinel)

21.3 bzr 1.7

Released 2008-09-23
This release includes many bug fixes and a few performance and feature improvements. bzr rm will now scan for missing files and remove them, like how bzr add scans for unknown files and adds them. A bit more polish has been applied to the stacking code. The b-tree indexing code has been brought in, with an eye on using it in a future repository format. There are only minor installer changes since bzr-1.7rc2.

21.3.1 Features

• Some small updates to the win32 installer. Include localization files found in plugins, and include the builtin distutils as part of packaging qbzr. (Mark Hammond)
21.4 bzr 1.7rc2

Released 2008-09-17

A few bug fixes from 1.7rc1. The biggest change is a new `RemoteBranch.get_stacked_on_url` RPC. This allows clients that are trying to access a Stacked branch over the smart protocol, to properly connect to the stacked-on location.

21.4.1 Bug Fixes

- Branching from a shared repository on a smart server into a new repository now preserves the repository format. (Andrew Bennetts, #269214)
- Branching from a stacked branch via `bzr+ssh` can properly connect to the stacked-on branch. (Martin Pool, #261315)
- `bzr init` no longer re-opens the BzrDir multiple times. (Vincent Ladeuil)
- Fix ‘_in_buffer’ AttributeError when using the -Dhpss debug flag. (Andrew Bennetts)

21.5 bzr 1.7rc1

Released 2008-09-09

This release candidate for bzr 1.7 has several bug fixes and a few performance and feature improvements. `bzr rm` will now scan for missing files and remove them, like how `bzr add` scans for unknown files and adds them. A bit more polish has been applied to the stacking code. The b-tree indexing code has been brought in, with an eye on using it in a future repository format.

21.5.1 Changes

- `bzr export` can now export a subdirectory of a project. (Robert Collins)
- `bzr remove-tree` will now refuse to remove a tree with uncommitted changes, unless the --force option is specified. (Lukáš Lalinský, #74101)
- `bzr rm` will now scan for files that are missing and remove just them automatically, much as `bzr add` scans for new files that are not ignored and adds them automatically. (Robert Collins)

21.5.2 Features

- Support for GSSAPI authentication when using FTP as documented in RFC2228. (Jelmer Vernooij, #49623)
- Add support for IPv6 in the smart server. (Jelmer Vernooij, #165014)

21.5.3 Improvements

- A URL like `log+file:///tmp` will log all access to that Transport to `.bzr.log`, which may help in debugging or profiling. (Martin Pool)
- `bzr branch` and `bzr push` use the default stacking policy if the branch format supports it. (Aaron Bentley)
• `bzr init` and `bzr init-repo` will now print out the same as `bzr info` if it completed successfully. (Marius Kruger)

• `bzr uncommit` logs the old tip revision id, and displays how to restore the branch to that tip using `bzr pull`. This allows you to recover if you realize you uncommitted the wrong thing. (John Arbash Meinel)

• Fix problems in accessing stacked repositories over `bzr://`. (Martin Pool, #261315)

• `SFTPTransport.readv()` was accidentally using `list += string`, which ‘works’, but adds each character separately to the list, rather than using `list.append(string)`. Fixing this makes the SFTP transport a little bit faster (~20%) and use a bit less memory. (John Arbash Meinel)

• When reading index files, if we happen to read the whole file in a single request treat it as a `_buffer_all` request. This happens most often on small indexes over remote transports, where we default to reading 64kB. It saves a round trip for each small index during fetch operations. Also, if we have read more than 50% of an index file, trigger a `_buffer_all` on the next request. This works around some inefficiencies because reads don’t fall neatly on page boundaries, so we would ignore those bytes, but request them again later. This could trigger a total read size of more than the whole file. (John Arbash Meinel)

### 21.5.4 Bug Fixes

• `bzr rm` is now aliased to `bzr del` for the convenience of `svn` users. (Robert Collins, #205416)

• Catch the infamous “select/poll returned error” which occurs when `pycurl` try to send a body request to an HTTP/1.0 server which has already refused to handle the request. (Vincent Ladeuil, #225020)

• Fix `ObjectNotLocked` errors when using various commands (including `bzr cat` and `bzr annotate`) in combination with a smart server URL. (Andrew Bennetts, #237067)

• `FTPTransport.stat()` would return `0000` as the permission bits for the containing `.bzr/` directory (it does not implement permissions). This would cause us to set all subdirectories to `0700` and files to `0600` rather than leaving them unmodified. Now we ignore `0000` as the permissions and assume they are invalid. (John Arbash Meinel, #259855)

• Merging from a previously joined branch will no longer cause a traceback. (Jelmer Vernooij, #203376)

• Pack operations on windows network shares will work even with large files. (Robert Collins, #255656)

• Running `bzr st PATH_TO_TREE` will no longer suppress merge status. Status is also about 7% faster on mozilla sized trees when the path to the root of the tree has been given. Users of the internal `show_tree_status` function should be aware that the `show_pending` flag is now authoritative for showing pending merges, as it was originally. (Robert Collins, #255204)

• Set valid default `_param_name` for `Option` so that `ListOption` can embed `-'` in names. (Vincent Ladeuil, #263249)

• Show proper error rather than traceback when an unknown revision id is specified to `bzr cat-revision`. (Jelmer Vernooij, #175569)

• Trailing text in the dirstate file could cause the C dirstate parser to try to allocate an invalid amount of memory. We now properly check and test for parsing a dirstate with invalid trailing data. (John Arbash Meinel, #186014)

• Unexpected error responses from a smart server no longer cause the client to traceback. (Andrew Bennetts, #263527)

• Use a Windows api function to get a Unicode host name, rather than assuming the host name is ascii. (Mark Hammond, John Arbash Meinel, #256550)

• `WorkingTree4` trees will now correctly report missing-and-new paths in the output of `iter_changes`. (Robert Collins)
21.5.5 Documentation

• Updated developer documentation. (Martin Pool)

21.5.6 API Changes

• Exporters now take 4 parameters. (Robert Collins)
• `Tree.iter_changes` will now return False for the content change field when a file is missing in the basis tree and not present in the target tree. Previously it returned True unconditionally. (Robert Collins)
• The deprecated `Branch.abspath` and unimplemented `Branch.rename_one` and `Branch.move` were removed. (Jelmer Vernooij)
• `BzrDir.clone_on_transport` implementations must now accept a `stacked_on` parameter. (Aaron Bentley)
• `BzrDir.cloning_metadir` implementations must now take a `require_stacking` parameter. (Aaron Bentley)

21.5.7 Testing

• `addCleanup` now takes `*arguments` and `**keyword_arguments` which are then passed to the cleanup callable as it is run. In addition, `addCleanup` no longer requires that the callables passed to it be unique. (Jonathan Lange)
• Fix some tests that fail on Windows because files are deleted while still in use. (Mark Hammond)
• `selftest`'s `--starting-with` option can now use predefined prefixes so that one can say `bZR selftest -s bp.loom` instead of `bZR selftest -s bzrlib.plugins.loom`. (Vincent Ladeuil)
• `selftest`'s `--starting-with` option now accepts multiple values. (Vincent Ladeuil)

21.5.8 Internals

• A new plugin interface, `bzrlib.log.log_adapters`, has been added. This allows dynamic log output filtering by plugins. (Robert Collins)
• `bzrlib.btree_index` is now available, providing a b-tree index layer. The design is memory conservative (limited memory cache), faster to seek (approx 100 nodes per page, gives 100-way fan out), and stores compressed pages allowing more keys per page. (Robert Collins, John Arbash Meinel)
• `bzrlib.diff.DiffTree.show_diff` now skips changes where the kind is unknown in both source and target. (Robert Collins, Aaron Bentley)
• `GraphIndexBuilder.add_node` and `BTreeBuilder` have been streamlined a bit. This should make creating large indexes faster. (In benchmarking, it now takes less time to create a BTree index than it takes to read the GraphIndex one.) (John Arbash Meinel)
• Mail clients for `bzr send` are now listed in a registry. This allows plugins to add new clients by registering them with `bzrlib.mail_client.mail_client_registry`. All of the built-in clients now use this mechanism. (Neil Martinsen-Burrell)
22.1 bzr 1.6.1

Released 2008-09-05

A couple regressions were found in the 1.6 release. There was a performance issue when using bzr+ssh to branch large repositories, and some problems with stacking and rich-root capable repositories.

22.2 bzr 1.6.1rc2

Released 2008-09-03

22.2.1 Bug Fixes

• Copying between rich-root and rich-root-pack (and vice versa) was accidentally using the inter-model fetcher, instead of recognizing that both were 'rich root' formats. (John Arbash Meinel, #264321)

22.3 bzr 1.6.1rc1

Released 2008-08-29

This release fixes a few regressions found in the 1.6 client. Fetching changes was using an O(N^2) buffering algorithm, so for large projects it would cause memory thrashing. There is also a specific problem with the --1.6-rich-root format, which prevented stacking on top of --rich-root-pack repositories, and could allow users to accidentally fetch experimental data (-subtree) without representing it properly. The --1.6-rich-root format has been deprecated and users are recommended to upgrade to --1.6.1-rich-root immediately. Also we re-introduced a workaround for users who have repositories with incorrect nodes (not possible if you only used official releases). I should also clarify that none of this is data loss level issues, but still sufficient enough to warrant an updated release.

22.3.1 Bug Fixes

• RemoteTransport.readv() was being inefficient about how it buffered the readv data and processed it. It would keep appending to the same string (causing many copies) and then pop bytes out of the start of the string (causing more copies). With this patch “bzr+ssh://local” can improve dramatically, especially for projects with large files. (John Arbash Meinel)
• Revision texts were always meant to be stored as fulltexts. There was a bug in a bzr.dev version that would accidentally create deltas when copying from a Pack repo to a Knit repo. This has been fixed, but to support those repositories, we know always request full texts for Revision texts. (John Arbash Meinel, #261339)

• The previous --1.6-rich-root format used an incorrect XML serializer, which would accidentally support fetching from a repository that supported subtrees, even though the local one would not. We deprecated that format, and introduced a new one that uses the correct serializer --1.6.1-rich-root. (John Arbhash Meinel, #262333)

22.4 bzr 1.6

Released 2008-08-25

Finally, the long awaited bzr 1.6 has been released. This release includes new features like Stacked Branches, improved weave merge, and an updated server protocol (now on v3) which will allow for better cross version compatibility. With this release we have deprecated Knit format repositories, and recommend that users upgrade them, we will continue to support reading and writing them for the forseeable future, but we will not be tuning them for performance as pack repositories have proven to be better at scaling. This will also be the first release to bundle TortoiseBzr in the standalone Windows installer.

22.5 bzr 1.6rc5

Released 2008-08-19

22.5.1 Bug Fixes

• Disable automatic detection of stacking based on a containing directory of the target. It interacted badly with push, and needs a bit more work to get the edges polished before it should happen automatically. (John Arbhash Meinel, #259275) (This change was reverted when merged to bzr.dev)

22.6 bzr 1.6rc4

Released 2008-08-18

22.6.1 Bug Fixes

• Fix a regression in knit => pack fetching. We had a logic inversion, causing the fetch to insert fulltexts in random order, rather than preserving deltas. (John Arbhash Meinel, #256757)

22.7 bzr 1.6rc3

Released 2008-08-14
22.7.1 Changes

• Disable reading `.bzrrules` as a per-branch rule preferences file. The feature was not quite ready for a full release. (Robert Collins)

22.7.2 Improvements

• Update the windows installer to bundle TortoiseBzr and qbzr into the standalone installer. This will be the first official windows release that installs Tortoise by default. (Mark Hammond)

22.7.3 Bug Fixes

• Fix a regression in `bzr+http` support. There was a missing function `_read_line` that needed to be carried over from `bzr+ssh` support. (Andrew Bennett)

• GraphIndex objects will internally read an entire index if more than 1/20th of their keyspace is requested in a single operation. This largely mitigates a performance regression in `bzr log FILE` and completely corrects the performance regression in `bzr log`. The regression was caused by removing an accommodation which had been supporting the index format in use. A newer index format is in development which is substantially faster. (Robert Collins)

22.8 `bzr 1.6rc2`

**Released** 2008-08-13

This release candidate has a few minor bug fixes, and some regression fixes for Windows.

22.8.1 Bug Fixes

• `bzr upgrade` on remote branches accessed via `bzr://` and `bzr+ssh://` now works. (Andrew Bennett)

• Change the `get_format_description()` strings for `RepositoryFormatKnitPack5` et al to be single line messages. (Aaron Bentley)

• Fix for a regression on Win32 where we would try to call `os.listdir()` on a file and not catch the exception properly. (Windows raises a different exception.) This would manifest in places like `bzr rm file` or `bzr switch`. (Mark Hammond, John Arbash Meinel)

• `Inventory.copy()` was failing to set the revision property for the root entry. (Jelmer Vernooij)

• SFTP transport: added missing `FileExists` case to `_translate_io_exception` (Christophe Troestler, #123475)

• The help for `bzr ignored` now suggests `bzr ls --ignored` for scripting use. (Robert Collins, #3834)

• The default `annotate` logic will now always assign the last-modified value of a line to one of the revisions that modified it, rather than a merge revision. This would happen when both sides claimed to have modified the line resulting in the same text. The choice is arbitrary but stable, so merges in different directions will get the same results. (John Arbash Meinel, #232188)
22.9 bzr 1.6rc1

Released 2008-08-06

This release candidate for bzr 1.6 solidifies the new branch stacking feature. Bazaar now recommends that users upgrade all knit repositories, because later formats are much faster. However, we plan to continue read/write and upgrade support for knit repositories for the foreseeable future. Several other bugs and performance issues were fixed.

22.9.1 Changes

• Knit format repositories are deprecated and bzr will now emit warnings whenever it encounters one. Use `bzr upgrade` to upgrade knit repositories to pack format. (Andrew Bennetts)

22.9.2 Improvements

• `bzr check` can now be told which elements at a location it should check. (Daniel Watkins)
• Commit now supports `--exclude` (or `-x`) to exclude some files from the commit. (Robert Collins, #3117)
• Fetching data between repositories that have the same model but no optimised fetcher will not reserialise all the revisions, increasing performance. (Robert Collins, John Arbash Meinel)
• Give a more specific error when target branch is not reachable. (James Westby)
• Implemented a custom `walkdirs_utf8` implementation for win32. This uses a pyrex extension to get direct access to the `FindFirstFileW` style apis, rather than using `listdir` + `lstat`. Shows a very strong improvement in commands like `status` and `diff` which have to iterate the working tree. Anywhere from 2x-6x faster depending on the size of the tree (bigger trees, bigger benefit.) (John Arbash Meinel)
• New registry for log properties handles and the method in LongLogFormatter to display the custom properties returned by the registered handlers. (Guillermo Gonzalez, #162469)

22.9.3 Bug Fixes

• Add more tests that stacking does not create deltas spanning physical repository boundaries. (Martin Pool, #252428)
• Better message about incompatible repositories. (Martin Pool, #206258)
• `bzr branch --stacked` ensures the destination branch format can support stacking, even if the origin does not. (Martin Pool)
• `bzr export` no longer exports `.bzrrules`. (Ian Clatworthy)
• `bzr serve --directory=/` now correctly allows the whole filesystem to be accessed on Windows, not just the root of the drive that Python is running from. (Adrian Wilkins, #240910)
• Deleting directories by hand before running `bzr rm` will not cause subsequent errors in `bzr st` and `bzr commit`. (Robert Collins, #150438)
• Fix a test case that was failing if encoding wasn’t UTF-8. (John Arbash Meinel, #247585)
• Fix “no buffer space available” error when branching with the new smart server protocol to or from Windows. (Andrew Bennetts, #246180)
• Fixed problem in branching from smart server. (#249256, Michael Hudson, Martin Pool)
• Handle a file turning in to a directory in TreeTransform. (James Westby, #248448)
22.9.4 API Changes

- `MutableTree.commit` has an extra optional keywork parameter `exclude` that will be unconditionally supplied by the command line UI - plugins that add tree formats may need an update. (Robert Collins)

- The API minimum version for plugin compatibility has been raised to 1.6 - there are significant changes throughout the code base. (Robert Collins)

- The generic fetch code now uses three attributes on Repository objects to control fetch. The streams requested are controlled via: `_fetch_order` and `_fetch_uses_deltas`. Setting these appropriately allows different repository implementations to receive data in their optimal form. If the `_fetch_reconcile` is set then a reconcile operation is triggered at the end of the fetch. (Robert Collins)

- `Repository.is_shared` doesn’t take a read lock. It didn’t need one in the first place (nobody cached the value, and `RemoteRepository` wasn’t taking one either). This saves a round trip when probing Pack repositories, as they read the `pack-names` file when locked. And during probe, locking the repo isn’t very useful. (John Arbash Meinel)

22.9.5 Internals

- `bzrlib.branchbuilder.BranchBuilder` is now much more capable of putting together a real history without having to create a full WorkingTree. It is recommended that tests that are not directly testing the WorkingTree use BranchBuilder instead. See `BranchBuilder.build_snapshot` or `TestCaseWithMemoryTree.make_branch_builder`. (John Arbash Meinel)

- `bzrlib.builtins.internal_tree_files` broken into two giving a new helper `safe_relpath_files` - used by the new `exclude` parameter to commit. (Robert Collins)

- Make it easier to introduce new WorkingTree formats. (Ian Clatworthy)

- The code for exporting trees was refactored not to use the deprecated `InventoryEntry` methods. (Ian Clatworthy)

- RuleSearchers return () instead of [] now when there are no matches. (Ian Clatworthy)

22.10 bzr 1.6beta3

Released 2008-07-17

This release adds a new ‘stacked branches’ feature allowing branches to share storage without being in the same repository or on the same machine. (See the user guide for more details.) It also adds a new hook, improved weaves, aliases for related locations, faster bzr+ssh push, and several bug fixes.

22.10.1 Features

- New `pre_change_branch_tip` hook that is called before the branch tip is moved, while the branch is write-locked. See the User Reference for signature details. (Andrew Bennetts)

- Rule-based preferences can now be defined for selected files in selected branches, allowing commands and plugins to provide custom behaviour for files matching defined patterns. See Rule-based preferences (part of Configuring Bazaar) in the User Guide and bzr help rules for more information. (Ian Clatworthy)

- Sites may suggest a branch to stack new branches on. (Aaron Bentley)
• Stacked branches are now supported. See `bzc help branch` and `bzc help push`. Branches must be in the `development1` format to stack, though the stacked-on branch can be of any format. (Robert Collins)

22.10.2 Improvements

• `bzc export --format=tgz --root=NAME` to export a gzipped tarball to stdout; also `tar` and `tbz2`. (Martin Pool)

• `bzc (re)merge --weave` will now use a standard Weave algorithm, rather than the annotation-based merge it was using. It does so by building up a Weave of the important texts, without needing to build the full ancestry. (John Arbash Meinel, #238895)

• `bzc send` documents and better supports `emacsclient` (proper escaping of mail headers and handling of the MUA Mew). (Christophe Troestler)

• Remembered locations can be specified by aliases, e.g. :parent, :public, :submit. (Aaron Bentley)

• The smart protocol now has improved support for setting branches’ revision info directly. This makes operations like push faster. The new request method name is `Branch.set_last_revision_ex`. (Andrew Bennetts)

22.10.3 Bug Fixes

• Bazaar is now able to be a client to the web server of IIS 6 and 7. The broken implementations of RFC822 in Python and RFC2046 in IIS combined with boundary-line checking in Bazaar previously made this impossible. (NB, IIS 5 does not suffer from this problem). (Adrian Wilkins, #247585)

• `bzc log --long` with a ghost in your mainline now handles that ghost properly. (John Arbash Meinel, #243536)

• `check` handles the split-up .bzr layout correctly, so no longer requires a branch to be present. (Daniel Watkins, #64783)

• Clearer message about how to set the PYTHONPATH if `bzrlib` can’t be loaded. (Martin Pool, #205230)

• Errors about missing libraries are now shown without a traceback, and with a suggestion to install the library. The full traceback is still in `.bzr.log` and can be shown with `-Derror`. (Martin Pool, #240161)

• Fetch from a stacked branch copies all required data. (Aaron Bentley, #248506)

• Handle URLs such as `ftp://user@host.com@www.host.com` where the user name contains an @. (Neil Martinsen-Burrell, #228058)

• `needs_read_lock` and `needs_write_lock` now suppress an error during `unlock` if there was an error in the original function. This helps most when there is a failure with a smart server action, since often the connection closes and we cannot unlock. (Andrew Bennetts, John Arbash Meinel, #125784)

• Obsolete hidden command `bzc fetch` removed. (Martin Pool, #172870)

• Raise the correct exception when doing `-rbefore:0` or `-c0`. (John Arbash Meinel, #239933)

• You can now compare file revisions in Windows diff programs from Cygwin Bazaar. (Matt McClure, #209281)

• `revision_history` now tolerates mainline ghosts for Branch format 6. (Aaron Bentley, #235055)

• Set locale from environment for third party libs. (Martin von Gagern, #128496)
22.10.4 Documentation

- Added Using stacked branches to the User Guide. (Ian Clatworthy)
- Updated developer documentation. (Martin Pool)

22.10.5 Testing

- --Dmemory will cause /proc/PID/status to be catted before bzr exits, allowing low-key analysis of peak memory use. (Robert Collins)
- testCaseWithTransport.make_branch_and_tree tries harder to return a tree with a branch attribute of the right format. This was preventing some RemoteBranch tests from actually running with RemoteBranch instances. (Andrew Bennetts)

22.10.6 API Changes

- Removed Repository.text_store, control_store, etc. Instead, there are new attributes texts, inventories, revisions, signatures, each of which is a VersionedFiles. See the Repository docstring for more details. (Robert Collins)
- Branch.pull now accepts an _override_hook_target optional parameter. If you have a subclass of Branch that overrides pull then you should add this parameter. (Andrew Bennetts)
- bzrlib.check.check() has been deprecated in favour of the more aptly-named bzrlib.check.check_branch(). (Daniel Watkins)
- Tree.print_file and Repository.print_file are deprecated. These methods are bad APIs because they write directly to sys.stdout. bzrlib does not use them internally, and there are no direct tests for them. (Alexander Belchenko)

22.10.7 Internals

- cat command no longer uses Tree.print_file() internally. (Alexander Belchenko)
- New class method BzrDir.open_containing_tree_branch_or_repository which eases the discovery of the tree, the branch and the repository containing a given location. (Daniel Watkins)
- New versionedfile.KeyMapper interface to abstract out the access to underlying .knit/.kndx etc files in repositories with partitioned storage. (Robert Collins)
- Obsolete developer-use command weave-join has been removed. (Robert Collins)
- RemoteToOtherFetcher and get_data_stream_for_search removed, to support new VersionedFiles layering. (Robert Collins)

22.11 bzr 1.6beta2

Released 2008-06-10

This release contains further progress towards our 1.6 goals of shallow repositories, and contains a fix for some user-affecting bugs in the repository layer. Building working trees during checkout and branch is now faster.
22.11.1 Bug Fixes

- Avoid KnitCorrupt error extracting inventories from some repositories. (The data is not corrupt; an internal check is detecting a problem reading from the repository.) (Martin Pool, Andrew Bennetts, Robert Collins, #234748)
- `bzr status` was breaking if you merged the same revision twice. (John Arbash Meinel, #235407)
- Fix infinite loop consuming 100% CPU when a connection is lost while reading a response body via the smart protocol v1 or v2. (Andrew Bennetts)
- Inserting a bundle which changes the contents of a file with no trailing end of line, causing a knit snapshot in a 'knits' repository will no longer cause KnitCorrupt. (Robert Collins)
- `RemoteBranch.pull` needs to return the `self._real_branch`'s pull result. It was instead just returning None, which breaks `bzr pull`. (John Arbash Meinel, #238149)
- Sanitize branch nick before using it as an attachment filename in `bzr send`. (Lukáš Lalinský, #210218)
- Squash `inv_entry.symlink_target` to a plain string when generating `DirState` details. This prevents from getting a `UnicodeError` when you have symlinks and non-ascii filenames. (John Arbash Meinel, #135320)

22.11.2 Improvements

- Added the ‘alias’ command to set/unset and display aliases. (Tim Penhey)
- `added`, `modified`, and `unknowns` behaviour made consistent (all three now quote paths where required). Added `--null` option to `added` and `modified` (for null-separated unknowns, use `ls --unknown --null`). (Adrian Wilkins)
- Faster branching (1.09x) and lightweight checkouts (1.06x) on large trees. (Ian Clatworthy, Aaron Bentley)

22.11.3 Documentation

- Added Bazaar Zen section to the User Guide. (Ian Clatworthy)

22.11.4 Testing

- Fix the test HTTPServer to be isolated from chdir calls made while it is running, allowing it to be used in blackbox tests. (Robert Collins)

22.11.5 API Changes

- `WorkingTree.set_parent_(ids/trees)` will now filter out revisions which are in the ancestry of other revisions. So if you merge the same tree twice, or merge an ancestor of an existing merge, it will only record the newest. (If you merge a descendent, it will replace its ancestor). (John Arbash Meinel, #235407)
- `RepositoryPolicy.__init__` now requires `stack_on` and `stack_on_pwd`, through the derived classes do not. (Aaron Bentley)
22.11.6 Internals

• **bzrlib.bzrdir.BzrDir.sprout** now accepts **stacked** to control creating stacked branches. (Robert Collins)

• Knit record serialisation is now stricter on what it will accept, to guard against potential internal bugs, or broken input. (Robert Collins)

22.12 **bazaar 1.6beta1**

**Released** 2008-06-02

Commands that work on the revision history such as push, pull, missing, uncommit and log are now substantially faster. This release adds a translation of some of the user documentation into Spanish. (Contributions of other translations would be very welcome.) Bazaar 1.6beta1 adds a new network protocol which is used by default and which allows for more efficient transfers and future extensions.

22.12.1 Notes When Upgrading

• There is a new version of the network protocol used for **bzr://, bzr+ssh://** and **bzr+http://** connections. This will allow more efficient requests and responses, and more graceful fallback when a server is too old to recognise a request from a more recent client. Bazaar 1.6 will interoperate with 0.16 and later versions, but servers should be upgraded when possible. Bazaar 1.6 no longer interoperates with 0.15 and earlier via these protocols. Use alternatives like SFTP or upgrade those servers. (Andrew Bennetts, #83935)

22.12.2 Changes

• Deprecation warnings will not be suppressed when running **bazaar selftest** so that developers can see if their code is using deprecated functions. (John Arbash Meinel)

22.12.3 Features

• Adding **-Derror** will now display a traceback when a plugin fails to load. (James Westby)

22.12.4 Improvements

• **bazaar branch/push/pull -r XXX** now have a helper function for finding the revno of the new revision (**Graph.find_distance_to_null**). This should make something like **bazaar branch -r -100** in a shared, no-trees repository much snappier. (John Arbash Meinel)

• **bazaar log --short -r X..Y** no longer needs to access the full revision history. This makes it noticeably faster when logging the last few revisions. (John Arbash Meinel)

• **bazaar ls** now accepts **-V** as an alias for **--versioned**. (Jerad Cramp, #165086)

• **bazaar missing** uses the new **Graph.find_unique_ancestors** and **Graph.find_differences** to determine missing revisions without having to search the whole ancestry. (John Arbash Meinel, #174625)

• **bazaar uncommit** now uses partial history access, rather than always extracting the full revision history for a branch. This makes it resolve the appropriate revisions much faster (in testing it drops uncommit from 1.5s => 0.4s). It also means **bazaar log --short** is one step closer to not using full revision history. (John Arbash Meinel, #172649)
22.12.5 Bugfixes

- `bzr merge --lca` should handle when two revisions have no common ancestor other than `NULL_REVISION`. (John Arbash Meinel, #235715)

- `bzr status` was breaking if you merged the same revision twice. (John Arbash Meinel, #235407)

- `bzr push` with both `--overwrite` and `-r NNN` options no longer fails. (Andrew Bennetts, #234229)

- Correctly track the base URL of a smart medium when using `bzr+http://` URLs, which was causing spurious “No repository present” errors with branches in shared repositories accessed over `bzr+http`. (Andrew Bennetts, #230550)

- Define `_remote_is_at_least_1_2` on `SmartClientMedium` so that all implementations have the attribute. Fixes ‘PyCurlTransport’ object has no attribute ‘_remote_is_at_least_1_2’ attribute errors. (Andrew Bennetts, #220806)

- Failure to delete an obsolete pack file should just give a warning message, not a fatal error. It may for example fail if the file is still in use by another process. (Martin Pool)

- Fix `MemoryError` during large fetches over HTTP by limiting the amount of data we try to read per `recv` call. The problem was observed with Windows and a proxy, but might affect other environments as well. (Eric Holmberg, #215426)

- Handle old merge directives correctly in `Merger.from_mergeable`. Stricter `get_parent_map` requirements exposed a latent bug here. (Aaron Bentley)

- Issue a warning and ignore passwords declared in `authentication.conf` when used for an SSH scheme (`sftp://` or `bzr+ssh://`). (Vincent Ladeuil, #203186)

- Make both HTTP implementations raise appropriate exceptions on 403 Forbidden when POSTing smart requests. (Vincent Ladeuil, #230223)

- Properly `title` header names in HTTP requests instead of capitalizing them. (Vincent Ladeuil, #229076)

- The “Unable to obtain lock” error message now also suggests using `bzr break-lock` to fix it. (Martin Albisetti, #139202)

- Treat an encoding of ‘’ as ascii; this can happen when `bzr` is run under `vim` on Mac OS X. (Neil Martinsen-Burrell)

- `VersionedFile.make_mpdiffs()` was raising an exception that wasn’t in scope. (Daniel Fischer #235687)

22.12.6 Documentation

- Added directory structure and started translation of docs in spanish. (Martin Albisetti, Lucio Albenga)

- Incorporate feedback from Jelmer Vernooij and Neil Martinsen-Burrell on the plugin and integration chapters of the User Guide. (Ian Clatworthy)

- More Bazaar developer documentation about packaging and release process, and about use of Python reprs. (Martin Pool, Martin Albisetti)

- Updated Tortise strategy document. (Mark Hammond)

22.12.7 Testing

- `bzrlib.tests.adapt_tests` was broken and unused - it has been fixed. (Robert Collins)
• Fix the test HTTPServer to be isolated from chdir calls made while it is running, allowing it to be used in blackbox tests. (Robert Collins)
• New helper function for splitting test suites split_suite_by_condition. (Robert Collins)

22.12.8 Internals

• Branch.missing_revisions has been deprecated. Similar functionality can be obtained using bzrlib.missing.find_unmerged. The api was fairly broken, and the function was unused, so we are getting rid of it. (John Arbash Meinel)

22.12.9 API Changes

• Branch.abspath is deprecated; use the Tree or Transport instead. (Martin Pool)
• Branch.update_revisions now takes an optional Graph object. This can be used by update_revisions when it is checking ancestry, and allows callers to prefer request to go to a local branch. (John Arbash Meinel)
• Branch, Repository, Tree and BzrDir should expose a Transport as an attribute if they have one, rather than having it indirectly accessible as .control_files._transport. This doesn’t add a requirement to support a Transport in cases where it was not needed before; it just simplifies the way it is reached. (Martin Pool)
• bzr missing --mine-only will return status code 0 if you have no new revisions, but the remote does. Similarly for --theirs-only. The new code only checks one side, so it doesn’t know if the other side has changes. This seems more accurate with the request anyway. It also changes the output to print ‘[This|Other] branch is up to date.’ rather than displaying nothing. (John Arbash Meinel)
• LockableFiles.put_utf8, put_bytes and controlfilename are now deprecated in favor of using Transport operations. (Martin Pool)
• Many methods on VersionedFile, Repository and in bzrlib.revision deprecated before bzrlib 1.5 have been removed. (Robert Collins)
• RevisionSpec.wants_revision_history can be set to False for a given RevisionSpec. This will disable the existing behavior of passing in the full revision history to self._match_on. Useful for specs that don’t actually need access to the full history. (John Arbash Meinel)
• The constructors of SmartClientMedium and its subclasses now require a base parameter. SmartClientMedium implementations now also need to provide a remote_path_from_transport method. (Andrew Bennetts)
• The default permissions for creating new files and directories should now be obtained from BzrDir._get_file_mode() and _get_dir_mode(), rather than from LockableFiles. The _set_file_mode and _set_dir_mode variables on LockableFiles which were advertised as a way for plugins to control this are no longer consulted. (Martin Pool)
• VersionedFile.join is deprecated. This method required local instances of both versioned file objects and was thus hostile to being used for streaming from a smart server. The new get_record_stream and insert_record_stream are meant to efficiently replace this method. (Robert Collins)
• WorkingTree.set_parent_(ids/trees) will now filter out revisions which are in the ancestry of other revisions. So if you merge the same tree twice, or merge an ancestor of an existing merge, it will only record the newest. (If you merge a descendent, it will replace its ancestor). (John Arbash Meinel, #235407)
• WorkingTreeFormat2.stub_initialize_remote is now private. (Martin Pool)
BAZAAR RELEASE NOTES

23.1 bzr 1.5

Released 2008-05-16

This release of Bazaar includes several updates to the documentation, and fixes to prepare for making rich root support the default format. Many bugs have been squashed, including fixes to log, bzr+ssh inter-operation with older servers.

23.1.1 Changes

• Suppress deprecation warnings when bzrlib is a ‘final’ release. This way users of packaged software won’t be bothered with DeprecationWarnings, but developers and testers will still see them. (John Arbash Meinel)

23.1.2 Documentation

• Incorporate feedback from Jelmer Vernooij and Neil Martinsen-Burrell on the plugin and integration chapters of the User Guide. (Ian Clatworthy)

23.2 bzr 1.5rc1

Released 2008-05-09

23.2.1 Changes

• Broader support of GNU Emacs mail clients. Set mail_client=emacsclient in your bazaar.conf and send will pop the bundle in a mail buffer according to the value of mail-user-agent variable. (Xavier Maillard)

23.2.2 Improvements

• Diff now handles revision specs like “branch:” and “submit:” more efficiently. (Aaron Bentley, #202928)
• More friendly error given when attempt to start the smart server on an address already in use. (Andrea Corbellini, #200575)
• Pull completes much faster when there is nothing to pull. (Aaron Bentley)
23.2.3 Bugfixes

- Authentication.conf can define sections without password. (Vincent Ladeuil, #199440)

- Avoid muttering every time a child update does not cause a progress bar update. (John Arbash Meinel, #213771)

- Branch.reconcile() is now implemented. This allows `bzr reconcile` to fix when a Branch has a non-canonical mainline history. `bzr check` also detects this condition. (John Arbash Meinel, #177855)

- `bzr log -r ..X bzr://` was failing, because it was getting a request for `revision_id=None` which was not a string. (John Arbash Meinel, #211661)

- `bzr commit` now works with Microsoft’s FTP service. (Andreas Deininger)

- Catch definitions outside sections in authentication.conf. (Vincent Ladeuil, #217650)

- Conversion from non-rich-root to rich-root(-pack) updates inventory sha1s, even when bundles are used. (Aaron Bentley, #181391)

- Conversion from non-rich-root to rich-root(-pack) works correctly even though search keys are not topologically sorted. (Aaron Bentley)

- Conversion from non-rich-root to rich-root(-pack) works even when a parent revision has a different root id. (Aaron Bentley, #177874)

- Disable strace testing until strace is fixed (see bug #103133) and emit a warning when selftest ends to remind us of leaking tests. (Vincent Ladeuil, #226769)

- Fetching all revisions from a repository does not cause pack collisions. (Robert Collins, Aaron Bentley, #212908)

- Fix error about “attempt to add line-delta in non-delta knit”. (Andrew Bennetts, #217701)

- Pushing a branch in “dirstate” format (Branch5) over bzr+ssh would break if the remote server was < version 1.2. This was due to a bug in the RemoteRepository.get_parent_map() fallback code. (John Arbash Meinel, #214894)

- Remove leftover code in `bzr_branch` that inappropriately creates a `branch-name` file in the branch control directory. (Martin Pool)

- Set SO_REUSEADDR on server sockets of `bzr serve` to avoid problems rebinding the socket when starting the server a second time. (John Arbash Meinel, Martin Pool, #164288)

- Severe performance degradation in fetching from knit repositories to knits and packs due to parsing the entire `revisions.kndx` on every graph walk iteration fixed by using the Repository.get_graph API. There was another regression in knit => knit fetching which re-read the index for every revision each side had in common. (Robert Collins, John Arbash Meinel)

- When logging the changes to a particular file, there was a bug if there were ghosts in the revision ancestry. (John Arbash Meinel, #209948)

- xs4all’s FTP server returns a temporary error when trying to list an empty directory, rather than returning an empty list. Adding a workaround so that we don’t get spurious failures. (John Arbash Meinel, #215522)

23.2.4 Documentation

- Expanded the User Guide to include new chapters on popular plugins and integrating Bazaar into your environment. The Best practices chapter was renamed to Miscellaneous topics as suggested by community feedback as well. (Ian Clatworthy)

- Document outlining strategies for TortoiseBzr. (Mark Hammond)
• Improved the documentation on hooks. (Ian Clatworthy)
• Update authentication docs regarding SSH agents. (Vincent Ladeuil, #183705)

23.2.5 Testing

• Add thread_name_suffix parameter to SmartTCPserver_for_testing, to make it easy to identify which test spawned a thread with an unhandled exception. (Andrew Bennetts)
• New --debugflag/-E option to bzr selftest for setting options for debugging tests, these are complementary to the -D options. The --selftest_debug global option has been replaced by the -E=allow_debug option for selftest. (Andrew Bennetts)
• Parameterised test ids are preserved correctly to aid diagnosis of test failures. (Robert Collins, Andrew Bennetts)
• selftest now accepts -starting-with <id> to load only the tests whose id starts with the one specified. This greatly speeds up running the test suite on a limited set of tests and can be used to run the tests for a single module, a single class or even a single test. (Vincent Ladeuil)
• The test suite modules have been modified to define load_tests() instead of test_suite(). That speeds up selective loading (via -load-list) significantly and provides many examples on how to migrate (grep for load_tests). (Vincent Ladeuil)

23.2.6 Internals

• Hooks.install_hook is now deprecated in favour of Hooks.install_named_hook which adds a required name parameter, to avoid having to call Hooks.name_hook. (Daniel Watkins)
• Implement xml8 serializer. (Aaron Bentley)
• New form @deprecated_method(deprecated_in(1, 5, 0)) for making deprecation wrappers. (Martin Pool)
• Repository.revision_parents is now deprecated in favour of Repository.get_parent_map([revid])[revid]. (Jelmer Vernooij)
• The Python assert statement is no longer used in Bazaar source, and a test checks this. (Martin Pool)

23.2.7 API Changes

• bzrlib.status.show_pending_merges requires the repository to be locked by the caller. Callers should have been doing it anyway, but it will now raise an exception if they do not. (John Arbash Meinel)
• Repository.get_data_stream, Repository.get_data_stream_for_search(), Repository.get_deltas_for_revisions(), Repository.revision_trees(), Repository.item_keys_introduced_by() no longer take read locks. (Aaron Bentley)
• LockableFiles.get_utf8 and .get are deprecated, as a start towards removing LockableFiles and .control_files entirely. (Martin Pool)
• Methods deprecated prior to 1.1 have been removed. (Martin Pool)
24.1 bzr 1.4

Released 2008-04-28

This release of Bazaar includes handy improvements to the speed of log and status, new options for several commands, improved documentation, and better hooks, including initial code for server-side hooks. A number of bugs have been fixed, particularly in interoperability between different formats or different releases of Bazaar over there network. There’s been substantial internal work in both the repository and network code to enable new features and faster performance.

24.1.1 Bug Fixes

• Pushing a branch in “dirstate” format (Branch5) over bzr+ssh would break if the remote server was < version 1.2. This was due to a bug in the RemoteRepository.get_parent_map() fallback code. (John Arbash Meinel, Andrew Bennetts, #214894)

24.2 bzr 1.4rc2

Released 2008-04-21

24.2.1 Bug Fixes

• bzr log -r ..X bzr:// was failing, because it was getting a request for revision_id=None which was not a string. (John Arbash Meinel, #211661)

• Fixed a bug in handling ghost revisions when logging changes in a particular file. (John Arbash Meinel, #209948)

• Fix error about “attempt to add line-delta in non-delta knit”. (Andrew Bennetts, #205156)

• Fixed performance degradation in fetching from knit repositories to knits and packs due to parsing the entire revisions.kndx on every graph walk iteration fixed by using the Repository.get_graph API. There was another regression in knit => knit fetching which re-read the index for every revision each side had in common. (Robert Collins, John Arbash Meinel)
24.3  bzr 1.4rc1

Released 2008-04-11

24.3.1 Changes

- bzr main script cannot be imported (Benjamin Peterson)
- On GNU/Linux bzr additionally looks for plugins in arch-independent site directory. (Toshio Kuratomi)
- The `set_rh` branch hook is now deprecated. Please migrate any plugins using this hook to use an alternative, e.g. `post_change_branch_tip`. (Ian Clatworthy)
- When a plugin cannot be loaded as the file path is not a valid python module name bzr will now strip a `bzr_` prefix from the front of the suggested name, as many plugins (e.g. bzr-svn) want to be installed without this prefix. It is a common mistake to have a folder named “bzr-svn” for that plugin, especially as this is what bzr branch lp:~bzr-svn will give you. (James Westby, Andrew Cowie)
- UniqueIntegerBugTracker now appends bug-ids instead of joining them to the base URL. Plugins that register bug trackers may need a trailing / added to the base URL if one is not already there. (James Westby, Andrew Cowie)

24.3.2 Features

- Added `start_commit` hook for mutable trees. (Jelmer Vernooij, #186422)
- `status` now accepts `--no-pending` to show the status without listing pending merges, which speeds up the command a lot on large histories. (James Westby, #202830)
- New `post_change_branch_tip` hook that is called after the branch tip is moved but while the branch is still write-locked. See the User Reference for signature details. (Ian Clatworthy, James Henstridge)
- Reconfigure can convert a branch to be standalone or to use a shared repository. (Aaron Bentley)

24.3.3 Improvements

- The smart protocol now has support for setting branches’ revision info directly. This should make operations like push slightly faster, and is a step towards server-side hooks. The new request method name is `Branch.set_last_revision_info`. (Andrew Bennetts)
- `bzr commit --fixes` now recognises “gnome” as a tag by default. (James Westby, Andrew Cowie)
- `bzr switch` will attempt to find branches to switch to relative to the current branch. E.g. `bzr switch branchname` will look for `current_branch/../branchname`. (Robert Collins, Jelmer Vernooij, Wouter van Heyst)
- Diff is now more specific about execute-bit changes it describes (Chad Miller)
- Fetching data over HTTP is a bit faster when urlib is used. This is done by forcing it to recv 64k at a time when reading lines in HTTP headers, rather than just 1 byte at a time. (Andrew Bennetts)
- Log –short and –line are much faster when -r is not specified. (Aaron Bentley)
- Merge is faster. We no longer check a file’s existence unnecessarily when merging the execute bit. (Aaron Bentley)
- `bzr status` on an explicit list of files no longer shows pending merges, making it much faster on large trees. (John Arbash Meinel)
• The launchpad directory service now warns the user if they have not set their launchpad login and are trying to resolve a URL using it, just in case they want to do a write operation with it. (James Westby)

• The smart protocol client is slightly faster, because it now only queries the server for the protocol version once per connection. Also, the HTTP transport will now automatically probe for and use a smart server if one is present. You can use the new `nosmart+` transport decorator to get the old behaviour. (Andrew Bennetts)

• The `version` command takes a `--short` option to print just the version number, for easier use in scripts. (Martin Pool)

• Various operations with revision specs and commands that calculate revnos and revision ids are faster. (John A. Meinel, Aaron Bentley)

### 24.3.4 Bugfixes

• Add `root_client_path` parameter to SmartWSGIAp and SmartServerRequest. This makes it possible to publish filesystem locations that don’t exactly match URL paths. SmartServerRequest subclasses should use the new `translate_client_path` and `transport_from_client_path` methods when dealing with paths received from a client to take this into account. (Andrew Bennetts, #124089)

• `bzr mv a b` can be now used also to rename previously renamed directories, not only files. (Lukáš Lalinský, #107967)

• `bzr uncommit --local` can now remove revisions from the local branch to be symmetric with `bzr commit --local`. (John Arbash Meinel, #93412)

• Don’t ask for a password if there is no real terminal. (Alexander Belchenko, #69851)

• Fix a bug causing a ValueError crash in `parse_line_delta_iter` when fetching revisions from a knit to pack repository or vice versa using `bzr://` (including over HTTP or SSH). (#208418, Andrew Bennetts, Martin Pool, Robert Collins)

• Fixed `_get_line` in `bzrlib.smart.medium`, which was buggy. Also fixed `_get_bytes` in the same module to use the push back buffer. These bugs had no known impact in normal use, but were problematic for developers working on the code, and were likely to cause real bugs sooner or later. (Andrew Bennetts)

• Implement handling of basename parameter for DefaultMail. (James Westby)

• Incompatibility with Paramiko versions newer than 1.7.2 was fixed. (Andrew Bennetts, #213425)

• Launchpad locations (lp: URLs) can be pulled. (Aaron Bentley, #181945)

• Merges that add files to deleted root directories complete. They do create conflicts. (Aaron Bentley, #210092)

• vsftpd’s return 550 RNFR command failed. supported. (Marcus Trautwig, #129786)

### 24.3.5 Documentation

• Improved documentation on send/merge relationship. (Peter Schuller)

• Minor fixes to the User Guide. (Matthew Fuller)

• Reduced the evangelism in the User Guide. (Ian Clatworthy)

• Added Integrating with Bazaar document for developers (Martin Albisetti)
24.3.6 API Breaks

- Attempting to pull data from a ghost aware repository (e.g. knits) into a non-ghost aware repository such as weaves will now fail if there are ghosts. (Robert Collins)

- KnitVersionedFile no longer accepts an access_mode parameter, and now requires the index and access_method parameters to be supplied. A compatible shim has been kept in the new function knit.make_file_knit. (Robert Collins)

- Log formatters must now provide log_revision instead of show and show_merge_revno methods. The latter had been deprecated since the 0.17 release. (James Westby)

- LoopbackSFTP is now called SocketAsChannelAdapter. (Andrew Bennetts)

- osutils.backup_file is removed. (Alexander Belchenko)

- Repository.get_revision_graph is deprecated, with no replacement method. The method was size(history) and not desirable. (Robert Collins)

- revision.revision_graph is deprecated, with no replacement function. The function was size(history) and not desirable. (Robert Collins)

- Transport.get_shared_medium is deprecated. Use Transport.get_smart_medium instead. (Andrew Bennetts)

- VersionedFile factories now accept a get_scope parameter rather than using a call to transaction_finished, allowing the removal of the fixed list of versioned files per repository. (Robert Collins)

- VersionedFile.annotate_iter is deprecated. While in principle this allowed lower memory use, all users of annotations wanted full file annotations, and there is no storage format suitable for incremental line-by-line annotation. (Robert Collins)

- VersionedFile.clone_text is deprecated. This performance optimisation is no longer used - reading the content of a file that is undergoing a file level merge to identical state on two branches is rare enough, and not expensive enough to special case. (Robert Collins)

- VersionedFile.clear_cache and enable_cache are deprecated. These methods added significant complexity to the VersionedFile implementation, but were only used for optimising fetches from knits - which can be done from outside the knit layer, or via a caching decorator. As knits are not the default format, the complexity is no longer worth paying. (Robert Collins)

- VersionedFile.create_empty is removed. This method presupposed a sensible mapping to a transport for individual files, but pack backed versioned files have no such mapping. (Robert Collins)

- VersionedFile.get_graph is deprecated, with no replacement method. The method was size(history) and not desirable. (Robert Collins)

- VersionedFile.get_graph_with_ghosts is deprecated, with no replacement method. The method was size(history) and not desirable. (Robert Collins)

- VersionedFile.get_parents is deprecated, please use VersionedFile.get_parent_map. (Robert Collins)

- VersionedFile.get_sha1 is deprecated, please use VersionedFile.get_sha1s. (Robert Collins)

- VersionedFile.has_ghost is now deprecated, as it is both expensive and unused outside of a single test. (Robert Collins)

- VersionedFile.iter_parents is now deprecated in favour of get_parent_map which can be used to instantiate a Graph on a VersionedFile. (Robert Collins)
• VersionedFileStore no longer uses the transaction parameter given to most methods; amongst other things this means that the get_weave_or_empty method no longer guarantees errors on a missing weave in a readonly transaction, and no longer caches versioned file instances which reduces memory pressure (but requires more careful management by callers to preserve performance). (Robert Collins)

24.3.7 Testing

• New -Dselftest_debug flag disables clearing of the debug flags during tests. This is useful if you want to use e.g. -Dhpss to help debug a failing test. Be aware that using this feature is likely to cause spurious test failures if used with the full suite. (Andrew Bennetts)

• selftest --load-list now uses a new more agressive test loader that will avoid loading unneeded modules and building their tests. Plugins can use this new loader by defining a load_tests function instead of a test_suite function. (a forthcoming patch will provide many examples on how to implement this). (Vincent Ladeuil)

• selftest --load-list now does some sanity checks regarding duplicate test IDs and tests present in the list but not found in the actual test suite. (Vincent Ladeuil)

• Slightly more concise format for the selftest progress bar, so there’s more space to show the test name. (Martin Pool)

[2500/10884, 1fail, 3miss in 1m29s] test_revisionnamespaces.TestRev

• The test suite takes much less memory to run, and is a bit faster. This is done by clearing most attributes of TestCases after running them, if they succeeded. (Andrew Bennetts)

24.3.8 Internals

• Added _build_client_protocol to _SmartClient. (Andrew Bennetts)

• Added basic infrastructure for automatic plugin suggestion. (Martin Albisetti)

• If a LockableFiles object is not explicitly unlocked (for example because of a missing try/finally block, it will give a warning but not automatically unlock itself. (Previously they did.) This sometimes caused knock-on errors if for example the network connection had already failed, and should not be relied upon by code. (Martin Pool, #109520)

• make dist target to build a release tarball, and also check-dist-tarball and dist-upload-escudero. (Martin Pool)

• The read_response_tuple method of SmartClientRequestProtocol* classes will now raise UnknownSmartMethod when appropriate, so that callers don’t need to try distinguish unknown request errors from other errors. (Andrew Bennetts)

• set_make_working_trees is now implemented provided on all repository implementations (Aaron Bentley)

• VersionedFile now has a new method get_parent_map which, like Graph.get_parent_map returns a dict of key:parents. (Robert Collins)
BAZAAR RELEASE NOTES

25.1 bsr 1.3.1

   Released  2008-04-09
   No changes from 1.3.1rc1.

25.2 bsr 1.3.1rc1

   Released  2008-04-04

25.2.1 Bug Fixes

   • Fix a bug causing a ValueError crash in parse_line_delta_iter when fetching revisions from a knit to
     pack repository or vice versa using bsr:// (including over HTTP or SSH). (#208418, Andrew Bennetts, Martin
     Pool, Robert Collins)

25.3 bsr 1.3

   Released  2008-03-20
   Bazaar has become part of the GNU project <http://www.gnu.org>

   Many operations that act on history, including log and annotate are now substantially faster. Several bugs have
   been fixed and several new options and features have been added.

25.3.1 Testing

   • Avoid spurious failure of TestVersion.test_version matching directory names. (#202778, Martin
     Pool)

25.4 bsr 1.3rc1

   Released  2008-03-16
25.4.1 Notes When Upgrading

• The backup directory created by upgrade is now called backup.bzr, not .bzr.backup. (Martin Al-bisetti)

25.4.2 Changes

• A new repository format ‘development’ has been added. This format will represent the latest ‘in-progress’ format that the bzr developers are interested in getting early-adopter testing and feedback on. doc/developers/development-repo.txt has detailed information. (Robert Collins)

• BZR_LOG environment variable controls location of .bzr.log trace file. User can suppress writing messages to .bzr.log by using ‘/dev/null’ filename (on Unix) or ‘NUL’ (on Windows). If BZR_LOG variable is not defined but BZR_HOME is defined then default location for .bzr.log trace file is $BZR_HOME/.bzr.log. (Alexander Belchenko, #106117)

• launchpad builtin plugin now shipped as separate part in standalone bzr.exe, installed to C:\Program Files\Bazaar\plugins directory, and standalone installer allows user to skip installation of this plugin. (Alexander Belchenko)

• Restore auto-detection of plink.exe on Windows. (Dmitry Vasiliev)

• Version number is now shown as “1.2” or “1.2pr2”, without zeroed or missing final fields. (Martin Pool)

25.4.3 Features

• branch and checkout can hard-link working tree files, which is faster and saves space. (Aaron Bentley)

• bzs send will now also look at the child_submit_to setting in the submit branch to determine the email address to send to. (Jelmer Vernooij)

25.4.4 Improvements

• BzrBranch._lefthand_history is faster on pack repos. (Aaron Bentley)

• Branch6.generate_revision_history is faster. (Aaron Bentley)

• Directory services can now be registered, allowing special URLs to be dereferenced into real URLs. This is a generalization and cleanup of the lp: transport lookup. (Aaron Bentley)

• Merge directives that are automatically attached to emails have nicer filenames, based on branch-nick + revno. (Aaron Bentley)

• push has a --revision option, to specify what revision to push up to. (Daniel Watkins)

• Significantly reducing execution time and network traffic for trivial case of running bzr missing command for two identical branches. (Alexander Belchenko)

• Speed up operations that look at the revision graph (such as ‘bzr log’). KnitPackRepositor.get_revision_graph uses Graph.iter_ANCESTORY to extract the revision history. This allows filtering ghosts while stepping instead of needing to peek ahead. (John Arbash Meinel)

• The hooks command lists installed hooks, to assist in debugging. (Daniel Watkins)

• Updates to how annotate work. Should see a measurable improvement in performance and memory consumption for file with a lot of merges. Also, correctly handle when a line is introduced by both parents (it
should be attributed to the first merge which notices this, and not to all subsequent merges.) (John Arbash Meinel)

### 25.4.5 Bugfixes

- Autopacking no longer holds the full set of inventory lines in memory while copying. For large repositories, this can amount to hundreds of MB of ram consumption. (Ian Clatworthy, John Arbash Meinel)

- Cherrypicking when using `--format=merge3` now explicitly excludes BASE lines. (John Arbash Meinel, #151731)

- Disable plink’s interactive prompt for password. (#107593, Dmitry Vasiliev)

- Encode command line arguments from unicode to user_encoding before invoking external mail client in `bzr send` command. (#139318, Alexander Belchenko)

- Fixed problem connecting to `bzr+https://` servers. (#198793, John Ferlito)

- Improved error reporting in the Launchpad plugin. (Daniel Watkins, #196618)

- Include quick-start-summary.svg file to python-based installer(s) for Windows. (#192924, Alexander Belchenko)

- lca merge now respects specified files. (Aaron Bentley)

- Make version-info --custom imply --all. (#195560, James Westby)

- `merge --preview` now works for merges that add or modify symlinks (James Henstridge)

- Redirecting the output from `bzr merge` (when the remembered location is used) now works. (John Arbash Meinel)

- `setup.py` script explicitly checks for Python version. (Jari Aalto, Alexander Belchenko, #200569)

- UnknownFormatErrors no longer refer to branches regardless of kind of unknown format. (Daniel Watkins, #173980)

- Upgrade bundled ConfigObj to version 4.5.2, which properly quotes # signs, among other small improvements. (Matt Nordhoff, #86838)

- Use correct indices when emitting LCA conflicts. This fixes IndexError errors. (Aaron Bentley, #196780)

### 25.4.6 Documentation

- Explained how to use `version-info --custom` in the User Guide. (Neil Martinsen-Burrell)

### 25.4.7 API Breaks

- Support for loading plugins from zip files and `bzrlib.plugin.load_from_zip()` function are depre-cated. (Alexander Belchenko)

### 25.4.8 Testing

- Added missing blackbox tests for `modified` (Adrian Wilkins)

- The branch interface tests were invalid for branches using rich-root repositories because the empty string is not a valid file-id. (Robert Collins)
25.4.9 Internals

- `Graph.iter_ancestry` returns the ancestry of revision ids. Similar to `Repository.get_revision_graph()` except it includes ghosts and you can stop part-way through. (John Arbash Meinel)

- New module `tools/package_mf.py` provide custom module finder for python packages (improves standard python library's modulefinder.py) used by `setup.py` script while building standalone bzr.exe. (Alexander Belchenko)

- New remote method `RemoteBzrDir.find_repositoryV2` adding support for detecting external lookup support on remote repositories. This method is now attempted first when lookup up repositories, leading to an extra round trip on older bzr smart servers. (Robert Collins)

- Repository formats have a new supported-feature attribute `supports_external_lookups` used to indicate repositories which support falling back to other repositories when they have partial data. (Robert Collins)

- `Repository.get_revision_graph_with_ghosts` and `bzrlib.revision.(common_ancestor,MultipleRevisionSources,common_graph)` have been deprecated. (John Arbash Meinel)

- `Tree.iter_changes` is now a public API, replacing the work-in-progress `Tree._iter_changes`. The api is now considered stable and ready for external users. (Aaron Bentley)

- The bzrdir format registry now accepts an alias keyword to register_metadir, used to indicate that a format name is an alias for some other format and thus should not be reported when describing the format. (Robert Collins)
BAZAAR RELEASE NOTES

26.1 bzr 1.2

Released 2008-02-15

26.1.1 Bug Fixes

• Fix failing test in Launchpad plugin. (Martin Pool)

26.2 bzr 1.2rc1

Released 2008-02-13

26.2.1 Notes When Upgrading

• Fetching via the smart protocol may need to reconnect once during a fetch if the remote server is running Bazaar 1.1 or earlier, because the client attempts to use more efficient requests that confuse older servers. You may be required to re-enter a password or passphrase when this happens. This won’t happen if the server is upgraded to Bazaar 1.2. (Andrew Bennetts)

26.2.2 Changes

• Fetching via bzr+ssh will no longer fill ghosts by default (this is consistent with pack-0.92 fetching over SFTP). (Robert Collins)

• Formatting of bzr plugins output is changed to be more human-friendly. Full path of plugins locations will be shown only with --verbose command-line option. (Alexander Belchenko)

• merge now prefers to use the submit branch, but will fall back to parent branch. For many users, this has no effect. But some users who pull and merge on the same branch will notice a change. This change makes it easier to work on a branch on two different machines, pulling between the machines, while merging from the upstream. merge --remember can now be used to set the submit_branch. (Aaron Bentley)
26.2.3 Features

• `merge --preview` produces a diff of the changes merge would make, but does not actually perform the merge. (Aaron Bentley)

• New smart method `Repository.get_parent_map` for getting revision parent data. This returns additional parent information topologically adjacent to the requested data to reduce round trip latency impacts. (Robert Collins)

• New smart method, `Repository.stream_revisions_chunked`, for fetching revision data that streams revision data via a chunked encoding. This avoids buffering large amounts of revision data on the server and on the client, and sends less data to the server to request the revisions. (Andrew Bennetts, Robert Collins, #178353)

• The launchpad plugin now handles lp URLs of the form `lp://staging/`, `lp://demo/`, `lp://dev/` to use the appropriate launchpad instance to do the resolution of the branch identities. This is primarily of use to Launchpad developers, but can also be used by other users who want to try out Launchpad as a branch location without messing up their public Launchpad account. Branches that are pushed to the staging environment have an expected lifetime of one day. (Tim Penhey)

26.2.4 Improvements

• Creating a new branch no longer tries to read the entire revision-history unnecessarily over smart server operations. (Robert Collins)

• Fetching between different repository formats with compatible models now takes advantage of the smart method to stream revisions. (Andrew Bennetts)

• The `--coverage` option is now global, rather specific to `bzr selftest`. (Andrew Bennetts)

• The `register-branch` command will now use the public URL of the branch containing the current directory, if one has been set and no explicit branch is provided. (Robert Collins)

• Tweak the `reannotate` code path to optimize the 2-parent case. Speeds up `bzr annotate` with a pack repository by approx 3:2. (John Arbash Meinel)

26.2.5 Bugfixes

• Calculate remote path relative to the shared medium in `_SmartClient`. This is related to the problem in bug #124089. (Andrew Bennetts)

• Cleanly handle connection errors in smart protocol version two, the same way as they are handled by version one. (Andrew Bennetts)

• Clearer error when `version-info --custom` is used without `--template` (Lukáš Lalinský)

• Don’t raise `UnavailableFeature` during test setup when medusa is not available or tearDown is never called leading to nasty side effects. (#137823, Vincent Ladeuil)

• If a plugin’s test suite cannot be loaded, for example because of a syntax error in the tests, then `selftest` fails, rather than just printing a warning. (Martin Pool, #189771)

• List possible values for `BZR_SSH` environment variable in env-variables help topic. (Alexander Belchenko, #181842)

• New methods `push_log_file` and `pop_log_file` to intercept messages: popping the log redirection now precisely restores the previous state, which makes it easier to use `bzr log output` from other programs. `TestCaseInTempDir` no longer depends on a log redirection being established by the test framework, which lets `bzr` tests cleanly run from a normal unittest runner. (#124153, #124849, Martin Pool, Jonathan Lange)
• pull --quiet is now more quiet, in particular a message is no longer printed when the remembered pull location is used. (James Westby, #185907)

• reconfigure can safely be interrupted while fetching. (Aaron Bentley, #179316)

• reconfigure preserves tags when converting to and from lightweight checkouts. (Aaron Bentley, #182040)

• Stop polluting /tmp when running selftest. (Vincent Ladeuil, #123363)

• Switch from NFKC => NFC for normalization checks. NFC allows a few more characters which should be considered valid. (John Arbash Meinel, #185458)

• The launchpad plugin now uses the edge XML-RPC server to avoid interacting badly with a bug on the launchpad side. (Robert Collins)

• Unknown hostnames when connecting to a bzr:// URL no longer cause tracebacks. (Andrew Bennetts, #182849)

26.2.6 API Breaks

• Classes implementing Merge types like Merge3Merger must now accept (and honour) a do_merge flag in their constructor. (Aaron Bentley)

• Repository.add_inventory and add_revision now require the caller to previously take a write lock (and start a write group.) (Martin Pool)

26.2.7 Testing

• selftest now accepts --load-list <file> to load a test id list. This speeds up running the test suite on a limited set of tests. (Vincent Ladeuil)

26.2.8 Internals

• Add a new method get_result to graph search objects. The resulting SearchResult can be used to recreate the search later, which will be useful in reducing network traffic. (Robert Collins)

• Use convenience function to check whether two repository handles are referring to the same repository in Repository.get_graph. (Jelmer Vernooij, #187162)

• Fetching now passes the find_ghosts flag through to the InterRepository.missing_revision_ids call consistently for all repository types. This will enable faster missing revision discovery with bzr+ssh. (Robert Collins)

• Fix error handling in Repository.insert_data_stream. (Lukas Lalinsky)

• InterRepository.missing_revision_ids is now deprecated in favour of InterRepository.search_missing_revision_ids which returns a bzrlib.graph.SearchResult suitable for making requests from the smart server. (Robert Collins)

• New error NoPublicBranch for commands that need a public branch to operate. (Robert Collins)

• New method iter_inventories on Repository for access to many inventories. This is primarily used by the revision_trees method, as direct access to inventories is discouraged. (Robert Collins)

• New method next_with_ghosts on the Graph breadth-first-search objects which will split out ghosts and present parents into two separate sets, useful for code which needs to be aware of ghosts (e.g. fetching data cares about ghosts during revision selection). (Robert Collins)

• Record a timestamp against each mutter to the trace file, relative to the first import of bzrlib. (Andrew Bennetts)
• `Repository.get_data_stream` is now deprecated in favour of `Repository.get_data_stream_for_search` which allows less network traffic when requesting data streams over a smart server. (Robert Collins)

• `RemoteBzrDir._get_tree_branch` no longer triggers `_ensure_real`, removing one round trip on many network operations. (Robert Collins)

• `RemoteTransport`’s `recommended_page_size` method now returns 64k, like SFTPTransport and HttpTransportBase. (Andrew Bennetts)

• `Repository` has a new method `has_revisions` which signals the presence of many revisions by returning a set of the revisions listed which are present. This can be done by index queries without reading data for parent revision names etc. (Robert Collins)
BAZAAR RELEASE NOTES

27.1 bzr 1.1

Released 2008-01-15
(no changes from 1.1rc1)

27.2 bzr 1.1rc1

Released 2008-01-05

27.2.1 Changes

• Dotted revision numbers have been revised. Instead of growing longer with nested branches the branch number just increases. (eg instead of 1.1.1.1.1 we now report 1.2.1.) This helps scale long lived branches which have many feature branches merged between them. (John Arbash Meinel)

• The syntax `bzr diff branch1 branch2` is no longer supported. Use `bzr diff branch1 --new branch2` instead. This change has been made to remove the ambiguity where `branch2` is in fact a specific file to diff within `branch1`.

27.2.2 Features

• New option to use custom template-based formats in `bzr version-info`. (Lukáš Lalinský)

• `diff --using` allows an external diff tool to be used for files. (Aaron Bentley)

• New “lca” merge-type for fast everyday merging that also supports criss-cross merges. (Aaron Bentley)

27.2.3 Improvements

• `annotate` now doesn’t require a working tree. (Lukáš Lalinský, #90049)

• `branch` and `checkout` can now use files from a working tree to to speed up the process. For checkout, this requires the new `-files-from` flag. (Aaron Bentley)

• `bzr diff` now sorts files in alphabetical order. (Aaron Bentley)
• `bzr diff` now works on branches without working trees. Tree-less branches can also be compared to each other and to working trees using the new diff options `--old` and `--new`. Diffing between branches, with or without trees, now supports specific file filtering as well. (Ian Clatworthy, #6700)

• `bzr pack` now orders revision texts in topological order, with newest at the start of the file, promoting linear reads for `bzr log` and the like. This partially fixes #154129. (Robert Collins)

• Merge directives now fetch prerequisites from the target branch if needed. (Aaron Bentley)

• `pycurl` now handles digest authentication. (Vincent Ladeuil)

• `reconfigure` can now convert from repositories. (Aaron Bentley)

• `-l` is now a short form for `--limit` in `log`. (Matt Nordhoff)

• `merge` now warns when merge directives cause cherrypicks. (Aaron Bentley)

• `split` now supported, to enable splitting large trees into smaller pieces. (Aaron Bentley)

### 27.2.4 Bugfixes

• Avoid `AttributeError` when unlocking a pack repository when an error occurs. (Martin Pool, #180208)

• Better handle short reads when processing multiple range requests. (Vincent Ladeuil, #179368)

• `build_tree` acceleration uses the correct path when a file has been moved. (Aaron Bentley)

• `commit` now succeeds when a checkout and its master branch share a repository. (Aaron Bentley, #177592)

• Fixed error reporting of unsupported timezone format in `log --timezone`. (Lukáš Lalinský, #178722)

• Fixed Unicode encoding error in `ignored` when the output is redirected to a pipe. (Lukáš Lalinský)

• Fix traceback when sending large response bodies over the smart protocol on Windows. (Andrew Bennetts, #115781)

• Fix `urlutils.relative_url` for the case of two `file://` URLs pointed to different logical drives on Windows. (Alexander Belchenko, #90847)

• HTTP test servers are now compatible with the HTTP protocol version 1.1. (Vincent Ladeuil, #175524)

• `_KnitParentsProvider.get_parent_map` now handles requests for ghosts correctly, instead of erroring or attributing incorrect parents to ghosts. (Aaron Bentley)

• `merge --weave --uncommitted` now works. (Aaron Bentley)

• `pycurl` authentication handling was broken and incomplete. Fix handling of `user:pass` embedded in the URLs. (Vincent Ladeuil, #177643)

• Files inside non-directories are now handled like other conflict types. (Aaron Bentley, #177390)

• `reconfigure` is able to convert trees into lightweight checkouts. (Aaron Bentley)

• Reduce lockdir timeout to 0 when running `bzr serve`. (Andrew Bennetts, #148087)

• Test that the old `version_info_format` functions still work, even though they are deprecated. (John Arbash Meinel, ShenMaq, #177872)

• Transform failures no longer cause `ImmortalLimbo` errors (Aaron Bentley, #137681)

• `uncommit` works even when the commit messages of revisions to be removed use characters not supported in the terminal encoding. (Aaron Bentley)

• When dumb HTTP servers return whole files instead of the requested ranges, read the remaining bytes by chunks to avoid overflowing network buffers. (Vincent Ladeuil, #175886)
27.2.5 Documentation

- Minor tweaks made to the bug tracker integration documentation. (Ian Clatworthy)
- Reference material has now be moved out of the User Guide and added to the User Reference. The User Reference has gained 4 sections as a result: Authentication Settings, Configuration Settings, Conflicts and Hooks. All help topics are now dumped into text format in the doc/en/user-reference directory for those who like browsing that information in their editor. (Ian Clatworthy)
- *Using Bazaar with Launchpad* tutorial added. (Ian Clatworthy)

27.2.6 Internals

- find_* methods available for BzrDirs, Branches and WorkingTrees. (Aaron Bentley)
- Help topics can now be loaded from files. (Ian Clatworthy, Alexander Belchenko)
- get_parent_map now always provides tuples as its output. (Aaron Bentley)
- Parent Providers should now implement get_parent_map returning a dictionary instead of get_parents returning a list. Graph.get_parents is now deprecated. (John Arbash Meinel, Robert Collins)
- Patience Diff now supports arbitrary python objects, as long as they support hash(). (John Arbash Meinel)
- Reduce selftest overhead to establish test names by memoization. (Vincent Ladeuil)

27.2.7 API Breaks

27.2.8 Testing

- Modules can now customise their tests by defining a load_tests attribute. *pydoc bzrlib.tests.TestUtil.TestLoader.loadTestsFromModule* for the documentation on this attribute. (Robert Collins)
- New helper function *bzrlib.tests.condition_id_re* which helps filter tests based on a regular expression search on the tests id. (Robert Collins)
- New helper function *bzrlib.tests.condition_isinstance* which helps filter tests based on class. (Robert Collins)
- New helper function *bzrlib.tests.exclude_suite_by_condition* which generalises the exclude_suite_by_re function. (Robert Collins)
- New helper function *bzrlib.tests.filter_suite_by_condition* which generalises the filter_suite_by_re function. (Robert Collins)
- New helper method *bzrlib.tests.exclude_tests_by_re* which gives a new TestSuite that does not contain tests from the input that matched a regular expression. (Robert Collins)
- New helper method *bzrlib.tests.randomize_suite* which returns a randomized copy of the input suite. (Robert Collins)
- New helper method *bzrlib.tests.split_suite_by_re* which splits a test suite into two according to a regular expression. (Robert Collins)
- Parametrize all HTTP tests for the transport implementations, the HTTP protocol versions (1.0 and 1.1) and the authentication schemes. (Vincent Ladeuil)
- The *exclude_pattern* and *random_order* parameters to the function *bzrlib.tests.filter_suite_by_re* have been deprecated. (Robert Collins)
• The method `bzrlib.tests.sort_suite_by_re` has been deprecated. It is replaced by the new helper methods added in this release. (Robert Collins)
BAZAAR RELEASE NOTES

28.1 bzr 1.0

Released 2007-12-14

28.1.1 Documentation

• More improvements and fixes to the User Guide. (Ian Clatworthy)
• Add information on cherrypicking/rebasing to the User Guide. (Ian Clatworthy)
• Improve bug tracker integration documentation. (Ian Clatworthy)
• Minor edits to Bazaar in five minutes from David Roberts and to the rebasing section of the User
  Guide from Aaron Bentley. (Ian Clatworthy)

28.2 bzr 1.0rc3

Released 2007-12-11

28.2.1 Changes

• If a traceback occurs, users are now asked to report the bug through Launchpad (https://bugs.launchpad.net/bzr/),
  rather than by mail to the mailing list. (Martin Pool)

28.2.2 Bugfixes

• Fix Makefile rules for doc generation. (Ian Clatworthy, #175207)
• Give more feedback during long HTTP downloads by making readv deliver data as it arrives for urllib, and issue
  more requests for pycurl. High latency networks are better handled by urllib, the pycurl implementation give
  more feedback but also incur more latency. (Vincent Ladeuil, #173010)
• Implement _make_parents_provider on RemoteRepository, allowing generating bundles against branches on a
  smart server. (Andrew Bennetts, #147836)
28.2.3 Documentation

- Improved user guide. (Ian Clatworthy)
- The single-page quick reference guide is now available as a PDF. (Ian Clatworthy)

28.2.4 Internals

- readv urllib HTTP implementation is now a real iterator above the underlying socket and deliver data as soon as it arrives. ‘get’ still wraps its output in a StringIO. (Vincent Ladeuil)

28.3 bzr 1.0rc2

Released 2007-12-07

28.3.1 Improvements

- Added a --coverage option to selftest. (Andrew Bennetts)
- Annotate merge (merge-type=weave) now supports cherrypicking. (Aaron Bentley)
- bzr commit now doesn’t print the revision number twice. (Matt Nordhoff, #172612)
- New configuration option bugtracker_<tracker_abbreviation>_url to define locations of bug trackers that are not directly supported by bzr or a plugin. The URL will be treated as a template and {id} placeholders will be replaced by specific bug IDs. (Lukáš Lalinský)
- Support logging single merge revisions with short and line log formatters. (Kent Gibson)
- User Guide enhanced with suggested readability improvements from Matt Revell and corrections from John Arbash Meinel. (Ian Clatworthy)
- Quick Start Guide renamed to Quick Start Card, moved down in the catalog, provided in pdf and png format and updated to refer to send instead of bundle. (Ian Clatworthy, #165080)
- switch can now be used on heavyweight checkouts as well as lightweight ones. After switching a heavyweight checkout, the local branch is a mirror/cache of the new bound branch and uncommitted changes in the working tree are merged. As a safety check, if there are local commits in a checkout which have not been committed to the previously bound branch, then switch fails unless the --force option is given. This option is now also required if the branch a lightweight checkout is pointing to has been moved. (Ian Clatworthy)

28.3.2 Internals

- New -Dhttp debug option reports HTTP connections, requests and responses. (Vincent Ladeuil)
- New -Dmerge debug option, which emits merge plans for merge-type=weave.

28.3.3 Bugfixes

- Better error message when running bzr cat on a non-existant branch. (Lukáš Lalinský, #133782)
- Catch OSError 17 (file exists) in final phase of tree transform and show filename to user. (Alexander Belchenko, #111758)
• Catch ShortReadvErrors while using pycurl. Also make readv more robust by allowing multiple GET requests to be issued if too many ranges are required. (Vincent Ladeuil, #172701)

• Check for missing basis texts when fetching from packs to packs. (John Arbash Meinel, #165290)

• Fall back to showing e-mail in log --short/--line if the committer/author has only e-mail. (Lukáš Lalinský, #157026)

28.3.4 API Breaks

• Deprecate not passing a location argument to commit reporters’ started methods. (Matt Nordhoff)

28.4 bzr 1.0rc1

Released 2007-11-30

28.4.1 Notes When Upgrading

• The default repository format is now pack-0.92. This default is used when creating new repositories with init and init-repo, and when branching over bzr+ssh or bzr+hpss. (See https://bugs.launchpad.net/bugs/164626)

This format can be read and written by Bazaar 0.92 and later, and data can be transferred to and from older formats.

To upgrade, please reconcile your repository (bzr reconcile), and then upgrade (bzr upgrade).

pack-0.92 offers substantially better scaling and performance than the previous knits format. Some operations are slower where the code already had bad scaling characteristics under knits, the pack format makes such operations more visible as part of being more scalable overall. We will correct such operations over the coming releases and encourage the filing of bugs on any operation which you observe to be slower in a packs repository. One particular case that we do not intend to fix is pulling data from a pack repository into a knit repository over a high latency link; downgrading such data requires reinsertion of the file texts, and this is a classic space/time tradeoff. The current implementation is conservative on memory usage because we need to support converting data from any tree without problems. (Robert Collins, Martin Pool, #164476)

28.4.2 Changes

• Disable detection of plink.exe as possible SSH vendor. Plink vendor still available if user selects it explicitly with BZR_SSH environment variable. (Alexander Belchenko, workaround for bug #107593)

• The pack format is now accessible as “pack-0.92”, or “pack-0.92-subtree” to enable the subtree functions (for example, for bzr-svn). (Martin Pool)

28.4.3 Features

• New authentication.conf file holding the password or other credentials for remote servers. This can be used for SSH, SFTP, SMTP and other supported transports. (Vincent Ladeuil)

• New rich-root and rich-root-pack formats, recording the same data about tree roots that’s recorded for all other directories. (Aaron Bentley, #164639)

• pack-0.92 repositories can now be reconciled. (Robert Collins, #154173)
• **switch** command added for changing the branch a lightweight checkout is associated with and updating the tree to reflect the latest content accordingly. This command was previously part of the BzrTools plug-in. (Ian Clatworthy, Aaron Bentley, David Allouche)

• **reconfigure** command can now convert branches, trees, or checkouts to lightweight checkouts. (Aaron Bentley)

### 28.4.4 Performance

• Commit updates the state of the working tree via a delta rather than supplying entirely new basis trees. For commit of a single specified file this reduces the wall clock time for commit by roughly a 30%. (Robert Collins, Martin Pool)

• Commit with many automatically found deleted paths no longer performs linear scanning for the children of those paths during inventory iteration. This should fix commit performance blowing out when many such paths occur during commit. (Robert Collins, #156491)

• Fetch with pack repositories will no longer read the entire history graph. (Robert Collins, #88319)

• Revert takes out an appropriate lock when reverting to a basis tree, and does not read the basis inventory twice. (Robert Collins)

• Diff does not require an inventory to be generated on dirstate trees. (Aaron Bentley, #149254)

• New annotate merge (--merge-type=weave) implementation is fast on versionedfiles withough cached annotations, e.g. pack-0.92. (Aaron Bentley)

### 28.4.5 Improvements

• **bzr merge** now warns when it encounters a criss-cross merge. (Aaron Bentley)

• **bzr send** now doesn’t require the target e-mail address to be specified on the command line if an interactive e-mail client is used. (Lukáš Lalinský)

• **bzr tags** now prints the revision number for each tag, instead of the revision id, unless --show-ids is passed. In addition, tags can be sorted chronologically instead of lexicographically with --sort=time. (Adeodato Simó, #120231)

• Windows standalone version of bzr is able to load system-wide plugins from “plugins” subdirectory in installation directory. In addition standalone installer write to the registry (HKLMSOFTWAREBazaar) useful info about paths and bzr version. (Alexander Belchenko, #129298)

### 28.4.6 Documentation

### 28.4.7 Bug Fixes

• A progress bar has been added for knitpack -> knitpack fetching. (Robert Collins, #157789, #159147)

• Branching from a branch via smart server now preserves the repository format. (Andrew Bennetts, #164626)

• **commit** is now able to invoke an external editor in a non-ascii directory. (Daniel Watkins, #84043)

• Catch connection errors for FTP. (Vincent Ladeuil, #164567)

• **check** no longer reports spurious unreferenced text versions. (Robert Collins, John A Meinel, #162931, #165071)

• Conflicts are now resolved recursively by **revert**. (Aaron Bentley, #102739)
- Detect invalid transport reuse attempts by catching invalid URLs. (Vincent Ladeuil, #161819)
- Deleting a file without removing it shows a correct diff, not a traceback. (Aaron Bentley)
- Do no use timeout in HttpServer anymore. (Vincent Ladeuil, #158972).
- Don’t catch the exceptions related to the HTTP pipeline status before retrying an HTTP request or some pro-
programming errors may be masked. (Vincent Ladeuil, #160012)
- Fix `bzr rm` to not delete modified and ignored files. (Lukáš Lalinský, #172598)
- Fix exception when revisionspec contains merge revisions but log formatter doesn’t support merge revisions.
(Kent Gibson, #148908)
- Fix exception when ScopeReplacer is assigned to before any members have been retrieved. (Aaron Bentley)
- Fix multiple connections during checkout –lightweight. (Vincent Ladeuil, #159150)
- Fix possible error in insert_data_stream when copying between pack repositories over bzr+ssh or bzr+http.
KnitVersionedFile.get_data_stream now makes sure that requested compression parents are sent before any delta hunks that depend on them. (Martin Pool, #164637)
- Fix typo in limiting offsets coalescing for HTTP, leading to whole files being downloaded instead of parts.
(Vincent Ladeuil, #165061)
- FTP server errors don’t error in the error handling code. (Robert Collins, #161240)
- Give a clearer message when a pull fails because the source needs to be reconciled. (Martin Pool, #164443)
- It is clearer when a plugin cannot be loaded because of its name, and a suggestion for an acceptable name is
given. (Daniel Watkins, #103023)
- Leave port as None in transport objects if user doesn’t specify a port in URLs. (Vincent Ladeuil, #150860)
- Make sure Repository.fetch(self) is properly a no-op for all Repository implementations. (John Arbash Meinel,
#158333)
- Mark .bzr directories as “hidden” on Windows. (Alexander Belchenko, #71147)
- `merge --uncommitted` can now operate on a single file. (Aaron Bentley, Lukáš Lalinský, #136890)
- Obsolete packs are now cleaned up by pack and autopack operations. (Robert Collins, #153789)
- Operations pulling data from a smart server where the underlying repositories are not both annotated/both unan-
notated will now work. (Robert Collins, #165304).
- Reconcile now shows progress bars. (Robert Collins, #159351)
- `RemoteBranch` was not initializing `self._revision_id_to_revno_map` properly. (John Arbash Meinel, #162486)
- Removing an already-removed file reports the file does not exist. (Daniel Watkins, #152811)
- Rename on Windows is able to change filename case. (Alexander Belchenko, #77740)
- Return error instead of a traceback for `bzr log -r0`. (Kent Gibson, #133751)
- Return error instead of a traceback when bzr is unable to create symlink on some platforms (e.g. on Windows).
(Alexander Belchenko, workaround for #81689)
- Revert doesn’t crash when restoring a single file from a deleted directory. (Aaron Bentley)
- Stderr output via logging mechanism now goes through encoded wrapper and no more uses utf-8, but terminal
encoding instead. So all unicode strings now should be readable in non-utf-8 terminal. (Alexander Belchenko,
#54173)
• The error message when `move --after` should be used makes how to do so clearer. (Daniel Watkins, #85237)

• Unicode-safe output from `bzr info`. The output will be encoded using the terminal encoding and unrepresentable characters will be replaced by ‘?’. (Lukáš Lalinský, #151844)

• Working trees are no longer created when pushing into a local no-trees repo. (Daniel Watkins, #50582)

• Upgrade `util/configobj` to version 4.4.0. (Vincent Ladeuil, #151208).

• Wrap medusa FTP test server as an FTPServer feature. (Vincent Ladeuil, #157752)

### 28.4.8 API Breaks

• `osutils.backup_file` is deprecated. Actually it’s not used in bzrlib during very long time. (Alexander Belchenko)

• The return value of `VersionedFile.iter_lines_added_or_present_in_versions` has been changed. Previously it was an iterator of lines, now it is an iterator of (line, version_id) tuples. This change has been made to aid reconcile and fetch operations. (Robert Collins)

• `bzrlib.repository.get_versioned_file_checker` is now private. (Robert Collins)

• The Repository format registry default has been removed; it was previously obsoleted by the bzrdir format default, which implies a default repository format. (Martin Pool)

### 28.4.9 Internals

• Added `ContainerSerialiser` and `ContainerPushParser` to `bzrlib.pack`. These classes provide more convenient APIs for generating and parsing containers from streams rather than from files. (Andrew Bennetts)

• New module `lru_cache` providing a cache for use by tasks that need semi-random access to large amounts of data. (John A Meinel)

• `InventoryEntry.diff` is now deprecated. Please use `diff.DiffTree` instead.
BAZAAR RELEASE NOTES

29.1 bzr 0.92

Released 2007-11-05

29.1.1 Changes

• New uninstaller on Win32. (Alexander Belchenko)

29.2 bzr 0.92rc1

Released 2007-10-29

29.2.1 Changes

• bzr now returns exit code 4 if an internal error occurred, and 3 if a normal error occurred. (Martin Pool)

• pull, merge and push will no longer silently correct some repository index errors that occurred as a result of the Weave disk format. Instead the reconcile command needs to be run to correct those problems if they exist (and it has been able to fix most such problems since bzr 0.8). Some new problems have been identified during this release and you should run bzr check once on every repository to see if you need to reconcile. If you cannot pull or merge from a remote repository due to mismatched parent errors - a symptom of index errors - you should simply take a full copy of that remote repository to a clean directory outside any local repositories, then run reconcile on it, and finally pull from it locally. (And naturally email the repositories owner to ask them to upgrade and run reconcile). (Robert Collins)

29.2.2 Features

• New knitpack-experimental repository format. This is interoperable with the dirstate-tags format but uses a smarter storage design that greatly speeds up many operations, both local and remote. This new format can be used as an option to the init.init-repository and upgrade commands. (Robert Collins)

• For users of bzr-svn (and those testing the prototype subtree support) that wish to try packs, a new knitpack-subtree-experimental format has also been added. This is interoperable with the dirstate-subtrees format. (Robert Collins)

• New reconfigure command. (Aaron Bentley)
• New `revert --forget-merges` command, which removes the record of a pending merge without affecting the working tree contents. (Martin Pool)

• New `bzr_remote_path` configuration variable allows finer control of remote bzr locations than BZR_REMOTE_PATH environment variable. (Aaron Bentley)

• New `launchpad-login` command to tell Bazaar your Launchpad user ID. This can then be used by other functions of the Launchpad plugin. (James Henstridge)

29.2.3 Performance

• Commit in quiet mode is now slightly faster as the information to output is no longer calculated. (Ian Clatworthy)

• Commit no longer checks for new text keys during insertion when the revision id was deterministically unique. (Robert Collins)

• Committing a change which is not a merge and does not change the number of files in the tree is faster by utilising the data about whether files are changed to determine if the tree is unchanged rather than recalculating it at the end of the commit process. (Robert Collins)

• Inventory serialisation no longer double-sha’s the content. (Robert Collins)

• Knit text reconstruction now avoids making copies of the lines list for interim texts when building a single text. The new `apply_delta` method on `KnitContent` aids this by allowing modification of the revision id such objects represent. (Robert Collins)

• Pack indices are now partially parsed for specific key lookup using a bisection approach. (Robert Collins)

• Partial commits are now approximately 40% faster by walking over the unselected current tree more efficiently. (Robert Collins)

• XML inventory serialisation takes 20% less time while being stricter about the contents. (Robert Collins)

• `Graph heads()` queries have been fixed to no longer access all history unnecessarily. (Robert Collins)

29.2.4 Improvements

• `bzr+https://` smart server across https now supported. (John Ferlito, Martin Pool, #128456)

• Mutt is now a supported mail client; set `mail_client=mutt` in your bazaar.conf and `send` will use mutt. (Keir Mierle)

• New option `-c/--change` for `merge` command for cherrypicking changes from one revision. (Alexander Belchenko, #141368)

• Show encodings, locale and list of plugins in the traceback message. (Martin Pool, #63894)

• Experimental directory formats can now be marked with `experimental = True` during registration. (Ian Clatworthy)

29.2.5 Documentation

• New Bazaar in Five Minutes guide. (Matthew Revell)

• The hooks reference documentation is now converted to html as expected. (Ian Clatworthy)
29.2.6 Bug Fixes

- Connection error reporting for the smart server has been fixed to display a user friendly message instead of a traceback. (Ian Clatworthy, #115601)

- Make sure to use O_BINARY when opening files to check their sha1sum. (Alexander Belchenko, John Arbash Meinel, #153493)

- Fix a problem with Win32 handling of the executable bit. (John Arbash Meinel, #149113)

- \texttt{bzr+ssh://} and \texttt{sftp://} URLs that do not specify ports explicitly no longer assume that means port 22. This allows people using OpenSSH to override the default port in their \texttt{~/.ssh/config} if they wish. This fixes a bug introduced in bzr 0.91. (Andrew Bennetts, #146715)

- Commands reporting exceptions can now be profiled and still have their data correctly dumped to a file. For example, a \texttt{bzr commit} with no changes still reports the operation as pointless but doing so no longer throws away the profiling data if this command is run with \texttt{--lsprof-file callgrind.out.ci} say. (Ian Clatworthy)

- Fallback to FTP when paramiko is not installed and SFTP can’t be used for \texttt{tests/commands} so that the test suite is still usable without paramiko. (Vincent Ladeuil, #59150)

- Fix commit ordering in corner case. (Aaron Bentley, #94975)

- Fix long standing bug in partial commit when there are renames left in tree. (Robert Collins, #140419)

- Fix selftest semi-random noise during HTTP related tests. (Vincent Ladeuil, #140614)

- Fix typo in ftp.py making the reconnection fail on temporary errors. (Vincent Ladeuil, #154259)

- Fix failing test by comparing real paths to cover the case where the \texttt{TMPDIR} contains a symbolic link. (Vincent Ladeuil, #141382).

- Fix log against smart server branches that don’t support tags. (James Westby, #140615)

- Fix pycurl HTTP implementation by defining error codes from pycurl instead of relying on an old curl definition. (Vincent Ladeuil, #147530)

- Fix ‘unprintable error’ message when displaying BzrCheckError and some other exceptions on Python 2.5. (Martin Pool, #144633)

- Fix \texttt{Inventory.copy()} and add test for it. (Jelmer Vernooij)

- Handles default value for ListOption in \texttt{cmd_commit}. (Vincent Ladeuil, #140432)

- \texttt{HttpServer} and \texttt{FtpServer} need to be closed properly or a listening socket will remain opened. (Vincent Ladeuil, #140055)

- Monitor the .bzr directory created in the top level test directory to detect leaking tests. (Vincent Ladeuil, #147986)

- The basename, not the full path, is now used when checking whether the profiling dump file begins with \texttt{callgrind.out} or not. This fixes a bug reported by Aaron Bentley on IRC. (Ian Clatworthy)

- Trivial fix for invoking command \texttt{reconfigure} without arguments. (Rob Weir, #141629)

- \texttt{WorkingTree.rename_one} will now raise an error if normalisation of the new path causes bzr to be unable to access the file. (Robert Collins)

- Correctly detect a NoSuchFile when using a filezilla server. (Gary van der Merwe)
29.2.7 API Breaks

• **bzrlib.index.GraphIndex** now requires a size parameter to the constructor, for enabling bisection searches. (Robert Collins)

• **CommitBuilder.record_entry_contents** now requires the root entry of a tree be supplied to it, previously failing to do so would trigger a deprecation warning. (Robert Collins)

• **KnitVersionedFile.add** will no longer cache added records even when enable_cache() has been called - the caching feature is now exclusively for reading existing data. (Robert Collins)

• **ReadOnlyLockError** is deprecated; **LockFailed** is usually more appropriate. (Martin Pool)

• Removed **bzrlib.transport.TransportLogger** - please see the new **trace** transport instead. (Robert Collins)

• Removed previously deprecated varargs interface to **TestCase.run_bzr** and deprecated methods **TestCase.capture** and **TestCase.run_bzr_captured**. (Martin Pool)

• Removed previous deprecated **basis_knit** parameter to the **KnitVersionedFile** constructor. (Robert Collins)

• Special purpose method **TestCase.run_bzr_decode** is moved to the test_non_ascii class that needs it. (Martin Pool)

• The class **bzrlib.repofmt.knitrepo.KnitRepository3** has been folded into **KnitRepository** by parameters to the constructor. (Robert Collins)

• The **VersionedFile** interface now allows content checks to be bypassed by supplying check_content=False. This saves nearly 30% of the minimum cost to store a version of a file. (Robert Collins)

• Tree’s with bad state such as files with no length or sha will no longer be silently accepted by the repository XML serialiser. To serialise inventories without such data, pass working=True to write_inventory. (Robert Collins)

• **VersionedFile.fix_parents** has been removed as a harmful API. **VersionedFile.join** will no longer accept different parents on either side of a join - it will either ignore them, or error, depending on the implementation. See notes when upgrading for more information. (Robert Collins)

29.2.8 Internals

• **bzrlib.transport.Transport.put_file** now returns the number of bytes put by the method call, to allow avoiding stat-after-write or housekeeping in callers. (Robert Collins)

• **bzrlib.xml_serializer.Serializer** is now responsible for checking that mandatory attributes are present on serialisation and deserialisation. This fixes some holes in API usage and allows better separation between physical storage and object serialisation. (Robert Collins)

• New class **bzrlib.errors.InternalBzrError** which is just a convenient shorthand for deriving from BzrError and setting internal_error = True. (Robert Collins)

• New method **bzrlib.mutabletree.update_to_one_parent_via_delta** for moving the state of a parent tree to a new version via a delta rather than a complete replacement tree. (Robert Collins)

• New method **bzrlib.osutils.minimum_path_selection** useful for removing duplication from user input, when a user mentions both a path and an item contained within that path. (Robert Collins)

• New method **bzrlib.repository.Repository.is_write_locked** useful for determining if a repository is write locked. (Robert Collins)

• New method on **bzrlib.tree.Tree path_content_summary** provides a tuple containing the key information about a path for commit processing to complete. (Robert Collins)
• New method on XML serialisers, write_inventory_to_lines, which matches the API used by knits for adding content. (Robert Collins)

• New module `bzrlib.bisect_multi` with generic multiple-bisection-at-once logic, currently only available for byte-based lookup (`bisect_multi_bytes`). (Robert Collins)

• New helper `bzrlib.tuned_gzip.bytes_to_gzip` which takes a byte string and returns a gzipped version of the same. This is used to avoid a bunch of api friction during adding of knit hunks. (Robert Collins)

• New parameter on `bzrlib.transport.Transport.readv` `adjust_for_latency` which changes `readv` from returning strictly the requested data to inserted return larger ranges and in forward read order to reduce the effect of network latency. (Robert Collins)

• New parameter `yield_parents` on `Inventory.iter_entries_by_dir` which causes the parents of a selected id to be returned recursively, so all the paths from the root down to each element of `selected_file_ids` are returned. (Robert Collins)

• Knit joining has been enhanced to support plain to annotated conversion and annotated to plain conversion. (Ian Clatworthy)

• The CommitBuilder method `record_entry_contents` now returns summary information about the effect of the commit on the repository. This tuple contains an inventory delta item if the entry changed from the basis, and a boolean indicating whether a new file graph node was recorded. (Robert Collins)

• The python path used in the Makefile can now be overridden. (Andrew Bennetts, Ian Clatworthy)

### 29.2.9 Testing

• New transport implementation `trace+` which is useful for testing, logging activity taken to its `_activity` attribute. (Robert Collins)

• When running bzr commands within the test suite, internal exceptions are not caught and reported in the usual way, but rather allowed to propagate up and be visible to the test suite. A new API `run_bzr_catch_user_errors` makes this behavior available to other users. (Martin Pool)

• New method `TestCase.call_catch_warnings` for testing methods that raises a Python warning. (Martin Pool)
BAZAAR RELEASE NOTES

30.1 bzr 0.91

Released 2007-09-26

30.1.1 Bug Fixes

• Print a warning instead of aborting the python setup.py install process if building of a C extension is not possible. (Lukáš Lalinský, Alexander Belchenko)
• Fix commit ordering in corner case (Aaron Bentley, #94975)
• Fix "bzr info bzr://host/" and other operations on "bzr://" URLs with an implicit port. We were incorrectly raising PathNotChild due to inconsistent treatment of the "_port" attribute on the Transport object. (Andrew Bennetts, #133965)
• Make RemoteRepository.sprout cope gracefully with servers that don’t support the Repository.tarball request. (Andrew Bennetts)

30.2 bzr 0.91rc2

Released 2007-09-11

• Replaced incorrect tarball for previous release; a debug statement was left in bzrlib/remote.py.

30.3 bzr 0.91rc1

Released 2007-09-11

30.3.1 Changes

• The default branch and repository format has changed to dirstate-tags, so tag commands are active by default. This format is compatible with Bazaar 0.15 and later. This incidentally fixes bug #126141. (Martin Pool)
• --quiet or -q is no longer a global option. If present, it must now appear after the command name. Scripts doing things like bzr -q missing need to be rewritten as bzr missing -q. (Ian Clatworthy)
30.3.2 Features

- New option `--author` in `bzr commit` to specify the author of the change, if it’s different from the committer. `bzr log` and `bzr annotate` display the author instead of the committer. (Lukáš Lalinský)

- In addition to global options and command specific options, a set of standard options are now supported. Standard options are legal for all commands. The initial set of standard options are:
  - `--help` or `-h` - display help message
  - `--verbose` or `-v` - display additional information
  - `--quiet` or `-q` - only output warnings and errors.

Unlike global options, standard options can be used in aliases and may have command-specific help. (Ian Clatworthy)

- Verbosity level processing has now been unified. If `--verbose` or `-v` is specified on the command line multiple times, the verbosity level is made positive the first time then increased. If `--quiet` or `-q` is specified on the command line multiple times, the verbosity level is made negative the first time then decreased. To get the default verbosity level of zero, either specify none of the above, `--no-verbose` or `--no-quiet`. Note that most commands currently ignore the magnitude of the verbosity level but do respect `quiet vs normal vs verbose` when generating output. (Ian Clatworthy)

- `Branch.hooks` now supports `pre_commit` hook. The hook’s signature is documented in `BranchHooks` constructor. (Nam T. Nguyen, #102747)

- New `Repository.stream_knit_data_for_revisions` request added to the network protocol for greatly reduced roundtrips when retrieving a set of revisions. (Andrew Bennetts)

30.3.3 Bug Fixes

- `bzr plugins` now lists the version number for each plugin in square brackets after the path. (Robert Collins, #125421)

- Pushing, pulling and branching branches with subtree references was not copying the subtree weave, preventing the file graph from being accessed and causing errors in commits in clones. (Robert Collins)

- Suppress warning “integer argument expected, got float” from Paramiko, which sometimes caused false test failures. (Martin Pool)

- Fix bug in bundle 4 that could cause attempts to write data to wrong versioned file. (Aaron Bentley)

- Diff’s generated using “diff -p” no longer break the patch parser. (Aaron Bentley)

- `get_transport` treats an empty possible_transports list the same as a non-empty one. (Aaron Bentley)

- `patch` verification for merge directives is reactivated, and works with CRLF and CR files. (Aaron Bentley)

- Accept `..` as a path in revision specifiers. This fixes for example “-r branch:..other-branch” on Windows. (Lukáš Lalinský)

- `BZR_PLUGIN_PATH` may now contain trailing slashes. (Blake Winton, #129299)

- man page no longer lists hidden options (#131667, Aaron Bentley)

- `uncommit --help` now explains the `-r` option adequately. (Daniel Watkins, #106726)

- Error messages are now better formatted with parameters (such as filenames) quoted when necessary. This avoids confusion when directory names ending in a `.` at the end of messages were confused with a full stop that may or not have been there. (Daniel Watkins, #129791)

- Fix `status FILE -r X..Y`. (Lukáš Lalinský)
• If a particular command is an alias, help will show the alias instead of claiming there is no help for said alias. (Daniel Watkins, #133548)

• TreeTransform-based operations, like pull, merge, revert, and branch, now roll back if they encounter an error. (Aaron Bentley, #67699)

• `bzr commit` now exits cleanly if a character unsupported by the current encoding is used in the commit message. (Daniel Watkins, #116143)

• `bzr send` uses default values for ranges when only half of an elipsis is specified ("-r..5" or "-r5.."). (#61685, Aaron Bentley)

• Avoid trouble when Windows SSH calls itself ‘plink’ but no plink binary is present. (Martin Albisetti, #107155)

• `bzr remove` should remove clean subtrees. Now it will remove (without needing `--force`) subtrees that contain no files with text changes or modified files. With `--force` it removes the subtree regardless of text changes or unknown files. Directories with renames in or out (but not changed otherwise) will now be removed without needing `--force`. Unknown ignored files will be deleted without needing `--force`. (Marius Kruger, #111665)

• When two plugins conflict, the source of both the losing and now the winning definition is shown. (Konstantin Mikhaylov, #5454)

• When committing to a branch, the location being committed to is displayed. (Daniel Watkins, #52479)

• `bzr --version` takes care about encoding of stdout, especially when output is redirected. (Alexander Belchenko, #131100)

• Prompt for an FTP password if none is provided. (Vincent Ladeuil, #137044)

• Reuse bound branch associated transport to avoid multiple connections. (Vincent Ladeuil, #128076, #131396)

• Overwrite conflicting tags by `push` and `pull` if the `--overwrite` option is specified. (Lukáš Lalinský, #93947)

• In checkouts, tags are copied into the master branch when created, changed or deleted, and are copied into the checkout when it is updated. (Martin Pool, #93856, #93860)

• Print a warning instead of aborting the `python setup.py install` process if building of a C extension is not possible. (Lukáš Lalinský, Alexander Belchenko)

### 30.3.4 Improvements

• Add the option “--show-diff” to the commit command in order to display the diff during the commit log creation. (Goffredo Baroncelli)

• `pull` and `merge` are much faster at installing bundle format 4. (Aaron Bentley)

• `pull` with `--v` no longer includes deltas, making it much faster. (Aaron Bentley)

• `send` now sends the directive as an attachment by default. (Aaron Bentley, Lukáš Lalinský, Alexander Belchenko)

• Documentation updates (Martin Albisetti)

• Help on debug flags is now included in `help global-options`. (Daniel Watkins, #124853)

• Parameters passed on the command line are checked to ensure they are supported by the encoding in use. (Daniel Watkins)

• The compression used within the bzr repository has changed from zlib level 9 to the zlib default level. This improves commit performance with only a small increase in space used (and in some cases a reduction in space). (Robert Collins)
• Initial commit no longer SHAs files twice and now reuses the path rather than looking it up again, making it faster. (Ian Clatworthy)

• New option --change for diff and status to show changes in one revision. (Lukáš Lalinský)

• If versioned files match a given ignore pattern, a warning is now given. (Daniel Watkins, #48623)

• bzr status now has -S as a short name for --short and -V as a short name for --versioned. These have been added to assist users migrating from Subversion: bzr status -SV is now like svn status -q. (Daniel Watkins, #115990)

• Added C implementation of PatienceSequenceMatcher, which is about 10x faster than the Python version. This speeds up commands that need file diffing, such as bzr commit or bzr diff. (Lukáš Lalinský)

• HACKING has been extended with a large section on core developer tasks. (Ian Clatworthy)

• Add branches and standalone-trees as online help topics and include them as Concepts within the User Reference. (Paul Moore, Ian Clatworthy)

• check can detect versionedfile parent references that are inconsistent with revision and inventory info, and reconcile can fix them. These faulty references were generated by 0.8-era releases, so repositories which were manipulated by old bzrs should be checked, and possibly reconciled ASAP. (Aaron Bentley, Andrew Bennetts)

### 30.3.5 API Breaks

- Branch.append_revision is removed altogether; please use Branch.set_last_revision_info instead. (Martin Pool)

- CommitBuilder now advertises itself as requiring the root entry to be supplied. This only affects foreign repository implementations which reuse CommitBuilder directly and have changed record_entry_contents to require that the root not be supplied. This should be precisely zero plugins affected. (Robert Collins)

- The add_lines methods on VersionedFile implementations has changed its return value to include the sha1 and length of the inserted text. This allows the avoidance of double-sha1 calculations during commit. (Robert Collins)

- Transport.should_cache has been removed. It was not called in the previous release. (Martin Pool)

### 30.3.6 Testing

- Tests may now raise TestNotApplicable to indicate they shouldn’t be run in a particular scenario. (Martin Pool)

- New function multiply_tests_from_modules to give a simpler interface to test parameterization. (Martin Pool, Robert Collins)

- Transport.should_cache has been removed. It was not called in the previous release. (Martin Pool)

- NULL_REVISION is returned to indicate the null revision, not None. (Aaron Bentley)

- Use UTF-8 encoded StringIO for log tests to avoid failures on non-ASCII committer names. (Lukáš Lalinský)

### 30.3.7 Internals

- bzrlib.plugin.all_plugins has been deprecated in favour of bzrlib.plugin.plugins() which returns PlugIn objects that provide useful functionality for determining the path of a plugin, its tests, and its version information. (Robert Collins)
• Add the option user_encoding to the function ‘show_diff_trees()’ in order to move the user encoding at the UI level. (Goffredo Baroncelli)

• Add the function make_commit_message_template_encoded() and the function edit_commit_message_encoded() which handle encoded strings. This is done in order to mix the commit messages (which is a unicode string), and the diff which is a raw string. (Goffredo Baroncelli)

• CommitBuilder now defaults to using add_lines_with_ghosts, reducing overhead on non-weave repositories which don’t require all parents to be present. (Robert Collins)

• Deprecated method find_previous_heads on bzrlib.inventory.InventoryEntry. This has been superseded by the use of parent_candidates and a separate heads check via the repository API. (Robert Collins)

• New trace function mutter_callsite will print out a subset of the stack to the log, which can be useful for gathering debug details. (Robert Collins)

• bzrlib.pack.ContainerWriter now tracks how many records have been added via a public attribute records_written. (Robert Collins)

• New method bzrlib.transport.Transport.get_recommended_page_size. This provides a hint to users of transports as to the reasonable minimum data to read. In principle this can take latency and bandwidth into account on a per-connection basis, but for now it just has hard coded values based on the URL. (E.g., http:// has a large page size, file:// has a small one.) (Robert Collins)

• New method on bzrlib.transport.Transport open_write_stream allows incremental addition of data to a file without requiring that all the data be buffered in memory. (Robert Collins)

• New methods on bzrlib.knit.KnitVersionedFile: get_data_stream(versions), insert_data_stream(stream) and get_format_signature(). These provide some infrastructure for efficiently streaming the knit data for a set of versions over the smart protocol.

• Knits with no annotation cache still produce correct annotations. (Aaron Bentley)

• Three new methods have been added to bzrlib.trace: set_verbosity_level, get_verbosity_level and is_verbos. set_verbosity_level expects a numeric value: negative for quiet, zero for normal, positive for verbose. The size of the number can be used to determine just how quiet or verbose the application should be. The existing be_quiet and is_quiet routines have been integrated into this new scheme. (Ian Clatworthy)

• Options can now be declared with a custom_callback parameter. If set, this routine is called after the option is processed. This feature is now used by the standard options verbose and quiet so that setting one implicitly resets the other. (Ian Clatworthy)

• Rather than declaring a new option from scratch in order to provide custom help, a centrally registered option can be decorated using the new bzrlib.Option.custom_help routine. In particular, this routine is useful when declaring better help for the verbose and quiet standard options as the base definition of these is now more complex than before thanks to their use of a custom callback. (Ian Clatworthy)

• Tree._iter_changes(specific_file=[]) now iterates through no files, instead of iterating through all files. None is used to iterate through all files. (Aaron Bentley)

• WorkingTree.revert() now accepts None to revert all files. The use of [] to revert all files is deprecated. (Aaron Bentley)
CHAPTER
THIRTYONE

BAZAAR RELEASE NOTES

31.1 bzr 0.90

Released 2007-08-28

31.1.1 Improvements

• Documentation is now organized into multiple directories with a level added for different languages or locales. Added the Mini Tutorial and Quick Start Summary (en) documents from the Wiki, improving the content and readability of the former. Formatted NEWS as Release Notes complete with a Table of Contents, one heading per release. Moved the Developer Guide into the main document catalog and provided a link from the developer document catalog back to the main one. (Ian Clatworthy, Sabin Iacob, Alexander Belchenko)

31.1.2 API Changes

• The static convenience method BzrDir.create_repository is deprecated. Callers should instead create a BzrDir instance and call create_repository on that. (Martin Pool)

31.2 bzr 0.90rc1

Released 2007-08-14

31.2.1 Bugfixes

• bzr init should connect to the remote location one time only. We have been connecting several times because we forget to pass around the Transport object. This modifies BzrDir.create_branch_convenience, so that we can give it the Transport we already have. (John Arbash Meinel, Vincent Ladeuil, #111702)

• Get rid of SFTP connection cache (get rid of the FTP one too). (Vincent Ladeuil, #43731)

• bzr branch {local|remote} remote don’t try to create a working tree anymore. (Vincent Ladeuil, #112173)

• All identified multiple connections for a single bzr command have been fixed. See bzrlib/tests/commands directory. (Vincent Ladeuil)

• bzr rm now does not insist on --force to delete files that have been renamed but not otherwise modified. (Marius Kruger, #111664)
• `bzr selftest --bench` no longer emits deprecation warnings (Lukáš Lalinský)
• `bzr status` now honours FILE parameters for conflict lists (Aaron Bentley, #127606)
• `bzr checkout` now honours `-r` when reconstituting a working tree. It also honours `-r 0`. (Aaron Bentley, #127708)
• `bzr add *` no more fails on Windows if working tree contains non-ascii file names. (Kuno Meyer, #127361)
• allow `easy_install bzr` runs without fatal errors. (Alexander Belchenko, #125521)
• Graph._filter_candidate_lca does not raise KeyError if a candidate is eliminated just before it would normally be examined. (Aaron Bentley)
• SMTP connection failures produce a nice message, not a traceback. (Aaron Bentley)

31.2.2 Improvements

• Don’t show “dots” progress indicators when run non-interactively, such as from cron. (Martin Pool)
• `info` now formats locations more nicely and lists “submit” and “public” branches (Aaron Bentley)
• New `pack` command that will trigger database compression within the repository (Robert Collins)
• Implement `_KnitIndex._load_data` in a pyrex extension. The pyrex version is approximately 2-3x faster at parsing a `.kndx` file. Which yields a measurable improvement for commands which have to read from the repository, such as a 1s => 0.75s improvement in `bzr diff` when there are changes to be shown. (John Arbash Meinel)
• Merge is now faster. Depending on the scenario, it can be more than 2x faster. (Aaron Bentley)
• Give a clearer warning, and allow `python setup.py install` to succeed even if pyrex is not available. (John Arbash Meinel)
• `DirState._read_dirblocks` now has an optional Pyrex implementation. This improves the speed of any command that has to read the entire DirState. (diff, status, etc, improve by about 10%). `bisect_dirblocks` has also been improved, which helps all `_get_entry` type calls (whenever we are searching for a particular entry in the in-memory DirState). (John Arbash Meinel)
• `bzr pull` and `bzr push` no longer do a complete walk of the branch revision history for ui display unless -v is supplied. (Robert Collins)
• `bzr log -rA..B` output shifted to the left margin if the log only contains merge revisions. (Kent Gibson)
• The `plugins` command is now public with improved help. (Ian Clatworthy)
• New bundle and merge directive formats are faster to generate, and
• Annotate merge now works when there are local changes. (Aaron Bentley)
• Commit now only shows the progress in terms of directories instead of entries. (Ian Clatworthy)
• Fix `KnitRepository.get_revision_graph` to not request the graph 2 times. This makes `get_revision_graph` 2x faster. (John Arbash Meinel)
• Fix `VersionedFile.get_graph()` to avoid using `set.difference_update(other)`, which has bad scaling when `other` is large. This improves `VF.get_graph([version_id])` for a 12.5k graph from 2.9s down to 200ms. (John Arbash Meinel)
• The `--lsprof-file` option now generates output for KCacheGrind if the file starts with `callgrind.out`. This matches the default file filtering done by KCacheGrind’s Open Dialog. (Ian Clatworthy)
• Fix `bzr update` to avoid an unnecessary `branch.get_master_branch` call, which avoids 1 extra connection to the remote server. (Partial fix for #128076, John Arbash Meinel)
• Log errors from the smart server in the trace file, to make debugging test failures (and live failures!) easier. (Andrew Bennetts)

• The HTML version of the man page has been superseded by a more comprehensive manual called the Bazaar User Reference. This manual is completed generated from the online help topics. As part of this change, limited reStructuredText is now explicitly supported in help topics and command help with ‘unnatural’ markup being removed prior to display by the online help or inclusion in the man page. (Ian Clatworthy)

• HTML documentation now use files extension *.html (Alexander Belchenko)

• The cache of ignore definitions is now cleared in WorkingTree.unlock() so that changes to .bzrignore aren’t missed. (#129694, Daniel Watkins)

• `bzr selftest --strict` fails if there are any missing features or expected test failures. (Daniel Watkins, #111914)

• Link to registration survey added to README. (Ian Clatworthy)

• Windows standalone installer show link to registration survey when installation finished. (Alexander Belchenko)

### 31.2.3 Library API Breaks

• Deprecated dictionary `bzrlib.option.SHORT_OPTIONS` removed. Options are now required to provide a help string and it must comply with the style guide by being one or more sentences with an initial capital and final period. (Martin Pool)

• `KnitIndex.get_parents` now returns tuples. (Robert Collins)

• Ancient unused `Repository.text_store` attribute has been removed. (Robert Collins)

• The `bzrlib.pack` interface has changed to use tuples of byte strings rather than just byte strings, making it easier to represent multiple element names. As this interface was not used by any internal facilities since it was introduced in 0.18 no API compatibility is being preserved. The serialised form of these packs is identical with 0.18 when a single element tuple is in use. (Robert Collins)

### 31.2.4 Internals

• `merge` now uses `iter_changes` to calculate changes, which makes room for future performance increases. It is also more consistent with other operations that perform comparisons, and reduces reliance on `Tree.inventory`. (Aaron Bentley)

• Refactoring of transport classes connected to a remote server. `ConnectedTransport` is a new class that serves as a basis for all transports needing to connect to a remote server. `transport.split_url` have been deprecated, use the static method on the object instead. URL tests have been refactored too. (Vincent Ladeuil)

• Better connection sharing for `ConnectedTransport` objects. `transport.get_transport()` now accepts a ‘possible_transports’ parameter. If a newly requested transport can share a connection with one of the list, it will. (Vincent Ladeuil)

• Most functions now accept `bzrlib.revision.NULL_REVISION` to indicate the null revision, and consider using `None` for this purpose deprecated. (Aaron Bentley)

• New `index` module with abstract index functionality. This will be used during the planned changes in the repository layer. Currently the index layer provides a graph aware immutable index, a builder for the same index type to allow creating them, and finally a composer for such indices to allow the use of many indices in a single query. The index performance is not optimised, however the API is stable to allow development on top of the index. (Robert Collins)
• bzrlib.dirstate.cmp_by_dirs can be used to compare two paths by their directory sections. This is equivalent to comparing path.split('/'), only without having to split the paths. This has a Pyrex implementation available. (John Arbash Meinel)

• New transport decorator 'unlistable+' which disables the list_dir functionality for testing.

• Deprecated change_entry in transform.py. (Ian Clatworthy)

• RevisionTree.get_weave is now deprecated. Tree.plan_merge is now used for performing annotate-merge. (Aaron Bentley)

• New EmailMessage class to create email messages. (Adeodato Simó)

• Unused functions on the private interface KnitIndex have been removed. (Robert Collins)

• New knit.KnitGraphIndex which provides a KnitIndex layered on top of a index.GraphIndex. (Robert Collins)

• New knit.KnitVersionedFile.iter_parents method that allows querying the parents of many knit nodes at once, reducing round trips to the underlying index. (Robert Collins)

• Graph now has an is_ancestor method, various bits use it. (Aaron Bentley)

• The -Dhpss flag now includes timing information. As well as logging when a new connection is opened. (John Arbash Meinel)

• bzrlib.pack.ContainerWriter now returns an offset, length tuple to callers when inserting data, allowing generation of readv style access during pack creation, without needing a separate pass across the output pack to gather such details. (Robert Collins)

• bzrlib.pack.make_readv_reader allows readv based access to pack files that are stored on a transport. (Robert Collins)

• New Repository.has_same_location method that reports if two repository objects refer to the same repository (although with some risk of false negatives). (Andrew Bennetts)

• InterTree.compare now passes require_versioned on correctly. (Marius Kruger)

• New methods on Repository - start_write_group, commit_write_group, abort_write_group and is_in_write_group - which provide a clean hook point for transactional Repositories - ones where all the data for a fetch or commit needs to be made atomically available in one step. This allows the write lock to remain while making a series of data insertions. (e.g. data conversion). (Robert Collins)

• In bzrlib.knit the internal interface has been altered to use 3-tuples (index, pos, length) rather than 2-tuples (pos, length) to describe where data in a knit is, allowing knits to be split into many files. (Robert Collins)

• bzrlib.knit._KnitData split into cache management and physical access with two access classes - _PackAccess and _KnitAccess defined. The former provides access into a .pack file, and the latter provides the current production repository form of .knit files. (Robert Collins)

31.2.5 Testing

• Remove selftest --clean-output, --numbered-dirs and --keep-output options, which are obsolete now that tests are done within directories in STMPDIR. (Martin Pool)

• The SSH_AUTH_SOCK environment variable is now reset to avoid interaction with any running SSH agents. (Jelmer Vernooij, #125955)

• run_bzr_subprocess handles parameters the same way as run_bzr: either a string or a list of strings should be passed as the first parameter. Varargs-style parameters are deprecated. (Aaron Bentley)
BAZAAR RELEASE NOTES

32.1 bzr 0.18

Released 2007-07-17

32.1.1 Bugfixes

• Fix ‘bzr add’ crash under Win32 (Kuno Meyer)

32.2 bzr 0.18rc1

Released 2007-07-10

32.2.1 Bugfixes

• Do not suppress pipe errors, etc. in non-display commands (Alexander Belchenko, #87178)
• Display a useful error message when the user requests to annotate a file that is not present in the specified revision. (James Westby, #122656)
• Commands that use status flags now have a reference to ‘help status-flags’. (Daniel Watkins, #113436)
• Work around python-2.4.1 inabilility to correctly parse the authentication header. (Vincent Ladeuil, #121889)
• Use exact encoding for merge directives. (Adeodato Simó, #120591)
• Fix tempfile permissions error in smart server tar bundling under Windows. (Martin _, #119330)
• Fix detection of directory entries in the inventory. (James Westby)
• Fix handling of HTTP code 400: Bad Request When issuing too many ranges. (Vincent Ladeuil, #115209)
• Issue a CONNECT request when connecting to an https server via a proxy to enable SSL tunneling. (Vincent Ladeuil, #120678)
• Fix bzr log -r to support selecting merge revisions, both individually and as part of revision ranges. (Kent Gibson, #4663)
• Don’t leave cruft behind when failing to acquire a lockdir. (Martin Pool, #109169)
• Don’t use the ‘-f’ strace option during tests. (Vincent Ladeuil, #102019).
• Warn when setting `push_location` to a value that will be masked by `locations.conf`. (Aaron Bentley, #122286)
• Fix commit ordering in corner case (Aaron Bentley, #94975)
• Make `annotate` behave in a non-ASCII world (Adeodato Simó).

### 32.2.2 Improvements

• The `--lsprof-file` option now dumps a text rendering of the profiling information if the filename ends in `".txt"`. It will also convert the profiling information to a format suitable for KCacheGrind if the output filename ends in `".callgrind"`. Fixes to the `lsprofcalltree` conversion process by Jean Paul Calderone and Itamar were also merged. See [http://ddaa.net/blog/python/lsprof-calltree](http://ddaa.net/blog/python/lsprof-calltree). (Ian Clatworthy)

• `info` now defaults to non-verbose mode, displaying only paths and abbreviated format info. `info -v` displays all the information formerly displayed by `info`. (Aaron Bentley, Adeodato Simó)

• `bzr missing` now has better option names --this and --other. (Elliot Murphy)

• The internal `weave-list` command has become `versionedfile-list`, and now lists knits as well as weaves. (Aaron Bentley)

• Automatic merge base selection uses a faster algorithm that chooses better bases in criss-cross merge situations (Aaron Bentley)

• Progress reporting in `commit` has been improved. The various logical stages are now reported on as follows, namely:
  - Collecting changes [Entry x/y] - Stage n/m
  - Saving data locally - Stage n/m
  - Uploading data to master branch - Stage n/m
  - Updating the working tree - Stage n/m
  - Running post commit hooks - Stage n/m

  If there is no master branch, the 3rd stage is omitted and the total number of stages is adjusted accordingly. Each hook that is run after commit is listed with a name (as hooks can be slow it is useful feedback). (Ian Clatworthy, Robert Collins)

• Various operations that are now faster due to avoiding unnecessary topological sorts. (Aaron Bentley)

• Make merge directives robust against broken bundles. (Aaron Bentley)

• The `lsprof` filename note is emitted via `trace.note()`, not standard output. (Aaron Bentley)

• `bzrlib` now exports explicit API compatibility information to assist library users and plugins. See the `bzrlib.api` module for details. (Robert Collins)

• Remove unnecessary lock probes when acquiring a lockdir. (Martin Pool)

• `bzr --version` now shows the location of the `bzr log file`, which is especially useful on Windows. (Martin Pool)

• `-D` now supports hooks to get debug tracing of hooks (though its currently minimal in nature). (Robert Collins)

• Long log format reports deltas on merge revisions. (John Arbash Meinel, Kent Gibson)

• Make initial push over FTP more resilient. (John Arbash Meinel)

• Print a summary of changes for update just like pull does. (Daniel Watkins, #113990)

• Add a `-Dhpss` option to trace smart protocol requests and responses. (Andrew Bennetts)
32.2.3 Library API Breaks

- Testing cleanups - bzrlib.repository.RepositoryTestProviderAdapter has been moved to bzrlib.tests.repository_implementations; bzrlib.repository.InterRepositoryTestProviderAdapter has been moved to bzrlib.tests.interrepository_implementations; bzrlib.transport.TransportTestProviderAdapter has moved to bzrlib.tests.test_transport_implementations. bzrlib.branch.BranchTestProviderAdapter has moved to bzrlib.tests.branch_implementations. bzrlib.bzrdir.BzrDirTestProviderAdapter has moved to bzrlib.tests.bzrdir_implementations. bzrlib.versionedfile.InterVersionedFileTestProviderAdapter has moved to bzrlib.tests.interversionedfile_implementations. bzrlib.store.revision.RevisionStoreTestProviderAdapter has moved to bzrlib.tests.revisionstore_implementations. bzrlib.workingtree.WorkingTreeTestProviderAdapter has moved to bzrlib.tests.workingtree_implementations. These changes are an API break in the testing infrastructure only. (Robert Collins)
- Relocate TestCaseWithRepository to be more central. (Robert Collins)
- bzrlib.add.smart_add_tree will no longer perform glob expansion on win32. Callers of the function should do this and use the new MutableTree.smart_add method instead. (Robert Collins)
- bzrlib.add.glob_expand_for_win32 is now bzrlib.win32utils.glob_expand. (Robert Collins)
- bzrlib.add.FastPath is now private and moved to bzrlib.mutabletree._FastPath. (Robert Collins, Martin Pool)
- LockDir.wait removed. (Martin Pool)
- The SmartServer hooks API has changed for the server_started and server_stopped hooks. The first parameter is now an iterable of backing URLs rather than a single URL. This is to reflect that many URLs may map to the external URL of the server. E.g. the server interally may have a chrooted URL but also the local file:// URL will be at the same location. (Robert Collins)

32.2.4 Internals

- New SMTPConnection class to unify email handling. (Adeodato Simó)
- Fix documentation of BzrError. (Adeodato Simó)
- Make BzrBadParameter an internal error. (Adeodato Simó)
- Remove use of ‘assert False’ to raise an exception unconditionally. (Martin Pool)
- Give a cleaner error when failing to decode knit index entry. (Martin Pool)
- TreeConfig would mistakenly search the top level when asked for options from a section. It now respects the section argument and only searches the specified section. (James Westby)
- Improve make api-docs output. (John Arbash Meinel)
- Use os.lstat rather than os.stat for osutils.make_readonly and osutils.make_writeable. This makes the difftools plugin more robust when dangling symlinks are found. (Elliot Murphy)
- New -Dlog option to log (to ~/.bzr.log) information on when lockdirs are taken or released. (Martin Pool)
- bzrlib Hooks are now nameable using Hooks.name_hook. This allows a nicer UI when hooks are running as the current hook can be displayed. (Robert Collins)
• `Transport.get` has had its interface made more clear for ease of use. Retrieval of a directory must now fail with either ‘PathError’ at open time, or raise ‘ReadError’ on a read. (Robert Collins)

• New method `_maybe_expand_globs` on the `Command` class for dealing with unexpanded glob lists - e.g. on the win32 platform. This was moved from `bzrlib.add._prepare_file_list`. (Robert Collins)

• `bzrlib.add.smart_add` and `bzrlib.add.smart_add_tree` are now deprecated in favour of `MutableTree.smart_add`. (Robert Collins, Martin Pool)

• New method `external_url` on `Transport` for obtaining the URL to hand to external processes. (Robert Collins)

• Teach windows installers to build pyrex/C extensions. (Alexander Belchenko)

### 32.2.5 Testing

• Removed the `--keep-output` option from `selftest` and clean up test directories as they’re used. This reduces the IO load from running the test suite and cuts the time by about half. (Andrew Bennetts, Martin Pool)

• Add scenarios as a public attribute on the `TestAdapter` classes to allow modification of the generated scenarios before adaption and easier testing. (Robert Collins)

• New testing support class `TestScenarioApplier` which multiplies out a single teste by a list of supplied scenarios. (Robert Collins)

• Setting `repository_to_test_repository` on a `repository_implementations` test will cause it to be called during repository creation, allowing the testing of repository classes which are not based around the Format concept. For example a repository adapter can be tested in this manner, by altering the repository scenarios to include a scenario that sets this attribute during the test parameterisation in `bzrlib.tests.repository.repository_implementations`. (Robert Collins)

• Clean up many of the APIs for blackbox testing of Bazaar. The standard interface is now `self.run_bzr`. The command to run can be passed as either a list of parameters, a string containing the command line, or (deprecated) `vargs` parameters. (Martin Pool)

• The base `TestCase` now isolates tests from `-D` parameters by clearing `debug.debug_flags` and restores it afterwards. (Robert Collins)

• Add a relpath parameter to `get_transport` methods in test framework to avoid useless cloning. (Vincent Ladeuil, #110448)
33.1 bzs 0.17

Released 2007-06-18

33.1.1 Bugfixes

• Fix crash of commit due to wrong lookup of filesystem encoding. (Colin Watson, #120647)
• Revert logging just to stderr in commit as broke unicode filenames. (Aaron Bentley, Ian Clatworthy, #120930)

33.2 bzs 0.17rc1

Released 2007-06-12

33.2.1 Notes When Upgrading

• The kind() and is_executable() APIs on the WorkingTree interface no longer implicitly (read) locks and unlocks the tree. This might impact some plug-ins and tools using this part of the API. If you find an issue that may be caused by this change, please let us know, particularly the plug-in/tool maintainer. If encountered, the API fix is to surround kind() and is_executable() calls with lock_read() and unlock() like so:

    work_tree.lock_read()
    try:
        kind = work_tree.kind(...)
    finally:
        work_tree.unlock()

33.2.2 Internals

• Rework of LogFormatter API to provide beginning/end of log hooks and to encapsulate the details of the revision to be logged in a LogRevision object. In long log formats, merge revision ids are only shown when –show-ids is specified, and are labelled “revision-id:”, as per mainline revisions, instead of “merged:”. (Kent Gibson)
• New BranchBuilder API which allows the construction of particular histories quickly. Useful for testing and potentially other applications too. (Robert Collins)
33.2.3 Improvements

- There are two new help topics, working-trees and repositories that attempt to explain these concepts. (James Westby, John Arbash Meinel, Aaron Bentley)
- Added `bzr log --limit` to report a limited number of revisions. (Kent Gibson, #3659)
- Revert does not try to preserve file contents that were originally produced by reverting to a historical revision. (Aaron Bentley)
- `bzr log --short` now includes `[merge]` for revisions which have more than one parent. This is a small improvement to help understanding what changes have occurred (John Arbash Meinel, #83887)
- TreeTransform avoids many renames when constructing large trees, improving speed. 3.25x speedups have been observed for construction of kernel-sized-trees, and checkouts are 1.28x faster. (Aaron Bentley)
- Commit on large trees is now faster. In my environment, a commit of a small change to the Mozilla tree (55k files) has dropped from 66 seconds to 32 seconds. For a small tree of 600 files, commit of a small change is 33% faster. (Ian Clatworthy)
- New `--create-prefix` option to `bzr init`, like for push. (Daniel Watkins, #56322)

33.2.4 Bugfixes

- `bzr push` should only connect to the remote location one time. We have been connecting 3 times because we forget to pass around the Transport object. This adds `BzrDir.clone_on_transport()`, so that we can pass in the Transport that we already have. (John Arbash Meinel, #75721)
- `DirState.set_state_from_inventory()` needs to properly order based on split paths, not just string paths. (John Arbash Meinel, #115947)
- Let TestUIFactoy encode the password prompt with its own stdout. (Vincent Ladeuil, #110204)
- pycurl should take use the range header that takes the range hint into account. (Vincent Ladeuil, #112719)
- `WorkingTree4.get_file_sha1` no longer raises an exception when invoked on a missing file. (Aaron Bentley, #118186)
- `WorkingTree.remove` works correctly with tree references, and when `pwd` is not the tree root. (Aaron Bentley)
- Merge no longer fails when a file is renamed in one tree and deleted in the other. (Aaron Bentley, #110279)
- `revision-info` now accepts dotted revnos, doesn’t require a tree, and defaults to the last revision (Matthew Fuller, #90048)
- Tests no longer fail when `BZR_REMOTE_PATH` is set in the environment. (Daniel Watkins, #111958)
- `bzr branch -r revid:foo` can be used to branch any revision in your repository. (Previously Branch6 only supported revisions in your mainline). (John Arbash Meinel, #115343)
CHAPTER
THIRTYFOUR

BAZAAR RELEASE NOTES

34.1 bzt 0.16

Released 2007-05-07

34.1.1 Bugfixes

• Handle when you have 2 directories with similar names, but one has a hyphen (‘abc’ versus ‘abc-2’). The WT4._iter_changes iterator was using direct comparison and ‘abc/a’ sorts after ‘abc-2’, but (‘abc’, ‘a’) sorts before (‘abc-2’, ). (John Arbash Meinel, #111127)

• Handle when someone renames a file on disk without telling bzr. Previously we would report the first file as missing, but not show the new unknown file. (John Arbash Meinel, #111288)

• Avoid error when running hooks after pulling into or pushing from a branch bound to a smartserver branch. (Martin Pool, #111968)

34.1.2 Improvements

• Move developer documentation to doc/developers/. This reduces clutter in the root of the source tree and allows HACKING to be split into multiple files. (Robert Collins, Alexander Belchenko)

• Clean up the WorkingTree4._iter_changes() internal loops as well as DirState.update_entry(). This optimizes the core logic for bzr diff and bzr status significantly improving the speed of both. (John Arbash Meinel)

34.2 bzt 0.16rc2

Released 2007-04-30

34.2.1 Bugfixes

• Handle the case when you delete a file, and then rename another file on top of it. Also handle the case of bzt rm --keep foo.bzt status should show the removed file and an unknown file in its place. (John Arbash Meinel, #109993)

• Bundles properly read and write revision properties that have an empty value. And when the value is not ASCII. (John Arbash Meinel, #109613)
• Fix the bzr commit message to be in text mode. (Alexander Belchenko, #110901)
• Also handle when you rename a file and create a file where it used to be. (John Arbash Meinel, #110256)
• WorkingTree4._iter_changes should not descend into unversioned directories. (John Arbash Meinel, #110399)

34.3 bzr 0.16rc1

Released 2007-04-26

34.3.1 Notes When Upgrading

• bzr remove and bzr rm will now remove the working file, if it could be recovered again. This has been done for consistency with svn and the unix rm command. The old remove behaviour has been retained in the new option bzr remove --keep, which will just stop versioning the file, but not delete it. bzr remove --force have been added which will always delete the files. bzr remove is also more verbose. (Marius Kruger, #82602)

34.3.2 Improvements

• Merge directives can now be supplied as input to merge and pull, like bundles can. (Aaron Bentley)
• Sending the SIGQUIT signal to bzr, which can be done on Unix by pressing Control-Backslash, drops bzr into a debugger. Type ‘c’ to continue. This can be disabled by setting the environment variable BZR_SIGQUIT_PDB=0. (Martin Pool)
• selftest now supports –list-only to list tests instead of running them. (Ian Clatworthy)
• selftest now supports –exclude PATTERN (or -x PATTERN) to exclude tests with names that match that regular expression. (Ian Clatworthy, #102679)
• selftest now supports –randomize SEED to run tests in a random order. SEED is typically the value ‘now’ meaning ‘use the current time’. (Ian Clatworthy, #102686)
• New option --fixes to commit, which stores bug fixing annotations as revision properties. Built-in support for Launchpad, Debian, Trac and Bugzilla bug trackers. (Jonathan Lange, James Henstridge, Robert Collins)
• New API, bzrlib.bugtracker.tracker_registry, for adding support for other bug trackers to fixes. (Jonathan Lange, James Henstridge, Robert Collins)
• selftest has new short options -f and -l. (Martin Pool)
• bzrlib.tsort.MergeSorter optimizations. Change the inner loop into using local variables instead of going through self._var. Improves the time to merge_sort a 10k revision graph by approximately 40% (~700->400ms). (John Arbash Meinel)
• make docs now creates a man page at man1/bzr.1 fixing bug 107388. (Robert Collins)
• bzr help now provides cross references to other help topics using the _see_also facility on command classes. Likewise the bzr_man documentation, and the bzr.1 man page also include this information. (Robert Collins)
• Tags are now included in logs, that use the long log formatter. (Erik Bågfors, Alexander Belchenko)
• bzr help provides a clearer message when a help topic cannot be found. (Robert Collins, #107656)
• `bzr help` now accepts optional prefixes for command help. The help for all commands can now be found at `bzr help commands/COMMANDNAME` as well as `bzr help COMMANDNAME` (which only works for commands where the name is not the same as a more general help topic). (Robert Collins)

• `bzr help PLUGINNAME` will now return the module docstring from the plugin PLUGINNAME. (Robert Collins, #50408)

• New help topic `urlspec` which lists the available transports. (Goffredo Baroncelli)

• `doc/server.txt` updated to document the default `bzr://` port and also update the blurb about the hpss' current status. (Robert Collins, #107125).

• `bzr serve` now listens on interface 0.0.0.0 by default, making it serve out to the local LAN (and anyone in the world that can reach the machine running `bzr serve`). (Robert Collins, #98918)

• A new smart server protocol version has been added. It prefixes requests and responses with an explicit version identifier so that future protocol revisions can be dealt with gracefully. (Andrew Bennetts, Robert Collins)

• The `bzr` protocol version 2 indicates success or failure in every response without depending on particular commands encoding that consistently, allowing future client refactorings to be much more robust about error handling. (Robert Collins, Martin Pool, Andrew Bennetts)

• The smart protocol over HTTP client has been changed to always post to the same `.bzr/smart` URL under the original location when it can. This allows HTTP servers to only have to pass URLs ending in `.bzr/smart` to the smart server handler, and not arbitrary `.bzr/*/smart` URLs. (Andrew Bennetts)

• digest authentication is now supported for proxies and HTTP by the urllib based HTTP implementation. Tested against Apache 2.0.55 and Squid 2.6.5. Basic and digest authentication are handled coherently for HTTP and proxy: if the user is provided in the URL (bzr command line for HTTP, proxy environment variables for proxies), the password is prompted for (only once). If the password is provided, it is taken into account. Once the first authentication is successful, all further authentication roundtrips are avoided by preventively setting the right authentication header(s). (Vincent Ladeuil).

### 34.3.3 Internals

• bzrlib API compatibility with 0.8 has been dropped, cleaning up some code paths. (Robert Collins)

• Change the format of chroot URLs so that they can be safely manipulated by generic URL utilities without causing the resulting URLs to have escaped the chroot. A side effect of this is that creating a chroot requires an explicit action using a ChrootServer. (Robert Collins, Andrew Bennetts)

• Deprecate `Branch.get_root_id()` because branches don't have root ids, rather than fixing bug #96847. (Aaron Bentley)

• `WorkingTree.apply_inventory_delta` provides a better alternative to `WorkingTree._write_inventory`. (Aaron Bentley)

• Convenience method `TestCase.expectFailure` ensures that known failures do not silently pass. (Aaron Bentley)

• `Transport.local_abspath` now raises `NotLocalUrl` rather than `TransportNotPossible`. (Martin Pool, Ian Clatworthy)

• New SmartServer hooks facility. There are two initial hooks documented in `bzrlib.transport.smart.SmartServerHooks`. The two initial hooks allow plugins to execute code upon server startup and shutdown. (Robert Collins).

• SmartServer in standalone mode will now close its listening socket when it stops, rather than waiting for garbage collection. This primarily fixes test suite hangs when a test tries to connect to a shutdown server. It may also help improve behaviour when dealing with a server running on a specific port (rather than dynamically assigned ports). (Robert Collins)
• Move most SmartServer code into a new package, bzrlib/smart. bzrlib/transport/remote.py contains just the Transport classes that used to be in bzrlib/transport/smart.py. (Andrew Bennetts)

• urllib HTTP implementation avoid roundtrips associated with 401 (and 407) errors once the authentication succeeds. (Vincent Ladeuil).

• urllib HTTP now supports querying the user for a proxy password if needed. Realm is shown in the prompt for both HTTP and proxy authentication when the user is required to type a password. (Vincent Ladeuil).

• Renamed SmartTransport (and subclasses like SmartTCPTTransport) to RemoteTransport (and subclasses to RemoteTCPTTransport, etc). This is more consistent with its new home in bzrlib/transport/remote.py, and because it’s not really a “smart” transport, just one that does file operations via remote procedure calls. (Andrew Bennetts)

• The lock_write method of LockableFiles, Repository and Branch now accept a token keyword argument, so that separate instances of those objects can share a lock if it has the right token. (Andrew Bennetts, Robert Collins)

• New method get_branch_reference on BzrDir allows the detection of branch references - which the smart server component needs.

• The Repository API make_working_trees is now permitted to return False when set_make_working_trees is not implemented - previously an unimplemented set_make_working_trees implied the result True from make_working_trees. This has been changed to accomodate the smart server, where it does not make sense (at this point) to ever make working trees by default. (Robert Collins)

• Command objects can now declare related help topics by having _see_also set to a list of related topic. (Robert Collins)

• bzrlib.help now delegates to the Command class for Command specific help. (Robert Collins)

• New class TransportListRegistry, derived from the Registry class, which simplifies tracking the available Transports. (Goffredo Baroncelli)

• New function Branch.get_revision_id_to_revno_map which will return a dictionary mapping revision ids to dotted revnos. Since dotted revnos are defined in the context of the branch tip, it makes sense to generate them from a Branch object. (John Arbash Meinel)

• Fix the ‘Unprintable error’ message display to use the repr of the exception that prevented printing the error because the str value for it is often not useful in debugging (e.g. KeyError(‘foo’) has a str() of ‘foo’ but a repr of ‘KeyError(‘foo’)’ which is much more useful. (Robert Collins)

• urlutils.normalize_url now unescapes unreserved characters, such as “~”. (Andrew Bennetts)

### 34.3.4 Bugfixes

• Don’t fail bundle selftest if email has ‘two’ embedded. (Ian Clatworthy, #98510)

• Remove --verbose from bzr bundle. It didn’t work anyway. (Robert Widhopf-Fenk, #98591)

• Remove --basis from the checkout/branch commands - it didn’t work properly and is no longer beneficial. (Robert Collins, #53675, #43486)

• Don’t produce encoding error when adding duplicate files. (Aaron Bentley)

• Fix bzr log <file> so it only logs the revisions that changed the file, and does it faster. (Kent Gibson, John Arbash Meinel, #51980, #69477)

• Fix InterDirstateTre._iter_changes to handle when we come across an empty versioned directory, which now has files in it. (John Arbash Meinel, #104257)
• Teach `common_ancestor` to shortcut when the tip of one branch is inside the ancestry of the other. Saves a lot of graph processing (with an ancestry of 16k revisions, `bzr merge ../already-merged changes from 2m10s to 13s`). (John Arbash Meinel, #103757)

• Fix `show_diff_trees` to handle the case when a file is modified, and the containing directory is renamed. (The file path is different in this versus base, but it isn’t marked as a rename). (John Arbash Meinel, #103870)

• FTP now works even when the FTP server does not support atomic rename. (Aaron Bentley, #89436)

• Correct handling in bundles and merge directives of timezones with that are not an integer number of hours offset from UTC. Always represent the epoch time in UTC to avoid problems with formatting earlier times on win32. (Martin Pool, Alexander Belchenko, John Arbash Meinel)

• Typo in the help for `register-branch` fixed. (Robert Collins, #96770)

• “dirstate” and “dirstate-tags” formats now produce branches compatible with old versions of bzr. (Aaron Bentley, #107168)

• Handle moving a directory when children have been added, removed, and renamed. (John Arbash Meinel, #105479)

• Don’t preventively use basic authentication for proxy before receiving a 407 error. Otherwise people willing to use other authentication schemes may expose their password in the clear (or nearly). This add one roundtrip in case basic authentication should be used, but plug the security hole. (Vincent Ladeuil)

• Handle HTTP and proxy digest authentication. (Vincent Ladeuil, #94034).

### 34.3.5 Testing

• Added `bzrlib.strace.strace` which will strace a single callable and return a StraceResult object which contains just the syscalls involved in running it. (Robert Collins)

• New test method `reduceLockdirTimeout` to drop the default (ui-centric) default time down to one suitable for tests. (Andrew Bennetts)

• Add new `vfs_transport_factory` attribute on tests which provides the common vfs backing for both the readonly and readwrite transports. This allows the RemoteObject tests to back onto local disk or memory, and use the existing `transport_server` attribute all tests know about to be the smart server transport. This in turn allows tests to differentiate between ‘transport to access the branch’, and ‘transport which is a VFS’ - which matters in Remote* tests. (Robert Collins, Andrew Bennetts)

• The `make_branch_and_tree` method for tests will now create a lightweight checkout for the tree if the `vfs_transport_factory` is not a `LocalURLServer`. (Robert Collins, Andrew Bennetts)

• Branch implementation tests have been audited to ensure that all URLs passed to Branch APIs use proper URLs, except when local-disk paths are intended. This is so that tests correctly access the test transport which is often not equivalent to local disk in Remote* tests. As part of this many tests were adjusted to remove dependencies on local disk access. (Robert Collins, Andrew Bennetts)

• Mark `bzrlib.tests` and `bzrlib.tests.TestUtil` as providing assertFOO helper functions by adding a `__unittest` global attribute. (Robert Collins, Andrew Bennetts, Martin Pool, Jonathan Lange)

• Refactored proxy and authentication handling to simplify the implementation of new auth schemes for both HTTP and proxy. (Vincent Ladeuil)
BAZAAR RELEASE NOTES

35.1  bzr 0.15

Released  2007-04-01

35.1.1 Bugfixes

- Handle incompatible repositories as a user issue when fetching. (Aaron Bentley)
- Don’t give a recommendation to upgrade when branching or checking out a branch that contains an old-format working tree. (Martin Pool)

35.2  bzr 0.15rc3

Released  2007-03-26

35.2.1 Changes

- A warning is now displayed when opening working trees in older formats, to encourage people to upgrade to WorkingTreeFormat4. (Martin Pool)

35.2.2 Improvements

- HTTP redirections are now taken into account when a branch (or a bundle) is accessed for the first time. A message is issued at each redirection to inform the user. In the past, HTTP redirections were silently followed for each request which significantly degraded the performances. The HTTP redirections are not followed anymore by default, instead a RedirectRequested exception is raised. For bzrlib users needing to follow HTTP redirections anyway, bzrlib.transport.do_catching_redirections provide an easy transition path. (vila)

35.2.3 Internals

- Added ReadLock temporary_write_lock() to allow upgrading an OS read lock to an OS write lock. Linux can do this without unlocking, Win32 needs to unlock in between. (John Arbash Meinel)
• New parameter `recommend_upgrade` to `BzrDir.open_workingtree` to silence (when false) warnings about opening old formats. (Martin Pool)

• Fix minor performance regression with bzr-0.15 on pre-dirstate trees. (We were reading the working inventory too many times). (John Arbash Meinel)

• Remove `Branch.get_transaction()` in favour of a simple cache of `revision_history`. Branch subclasses should override `_gen_revision_history` rather than `revision_history` to make use of this cache, and call `_clear_revision_history_cache` and `_cache_revision_history` at appropriate times. (Andrew Bennetts)

### 35.2.4 Bugfixes

• Take `smtp_server` from user config into account. (vila, #92195)

• Restore Unicode filename handling for versioned and unversioned files. (John Arbash Meinel, #92608)

• Don’t fail during `bzr commit` if a file is marked removed, and the containing directory is auto-removed. (John Arbash Meinel, #93681)

• `bzr status` `FILENAME` failed on Windows because of an uncommon errno. (ERROR_DIRECTORY == 267 != ENOTDIR). (Wouter van Heyst, John Arbash Meinel, #90819)

• `bzr checkout` `source` should create a local branch in the same format as source. (John Arbash Meinel, #93854)

• `bzr commit` with a kind change was failing to update the last-changed-revision for directories. The InventoryDirectory._unchanged only looked at the `parent_id` and name, ignoring the fact that the kind could have changed, too. (John Arbash Meinel, #90111)

• `bzr mv` `dir/subdir` `other` was incorrectly updating files inside the directory. So that there was a chance it would break commit, etc. (John Arbash Meinel, #94037)

• Correctly handles multiple permanent HTTP redirections. (vila, #88780)

### 35.3 bzr 0.15rc2

**Released** 2007-03-14

#### 35.3.1 Notes When Upgrading

• Release 0.15rc2 of bzr changes the `bzr init-repo` command to default to `--trees` instead of `--no-trees`. Existing shared repositories are not affected.

#### 35.3.2 Improvements

• New `merge-directive` command to generate machine- and human-readable merge requests. (Aaron Bentley)

• New `submit: revision specifier` makes it easy to diff against the common ancestor with the submit location (Aaron Bentley)

• Added support for Putty’s SSH implementation. (Dmitry Vasiliev)

• Added `bzr status --versioned` to report only versioned files, not unknowns. (Kent Gibson)
• Merge now autodetects the correct line-ending style for its conflict markers. (Aaron Bentley)

35.3.3 Internals

• Refactored SSH vendor registration into SSHVendorManager class. (Dmitry Vasiliev)

35.3.4 Bugfixes

• New \texttt{--numbered-dirs} option to \texttt{bzr selftest} to use numbered dirs for \texttt{TestCaseInTempDir}. This is default behavior on Windows. Anyone can force named dirs on Windows with \texttt{--no-numbered-dirs}. (Alexander Belchenko)

• Fix \texttt{RevisionSpec_revid} to handle the Unicode strings passed in from the command line. (Marien Zwart, \#90501)

• Fix \texttt{TreeTransform._iter_changes} when both the source and destination are missing. (Aaron Bentley, \#88842)

• Fix commit of merges with symlinks in dirstate trees. (Marien Zwart)

• Switch the \texttt{bzr init-repo} default from \texttt{--no-trees} to \texttt{--trees}. (Wouter van Heyst, \#53483)

35.4 \texttt{bzr 0.15rc1}

\textbf{Released} 2007-03-07

35.4.1 Surprises

• The default disk format has changed. Please run ‘bzr upgrade’ in your working trees to upgrade. This new default is compatible for network operations, but not for local operations. That is, if you have two versions of \texttt{bzr} installed locally, after upgrading you can only use the \texttt{bzr 0.15} version. This new default does not enable tags or nested-trees as they are incompatible with \texttt{bzr} versions before 0.15 over the network.

• For users of \texttt{bzrlib}: Two major changes have been made to the working tree api in \texttt{bzrlib}. The first is that many methods and attributes, including the inventory attribute, are no longer valid for use until one of \texttt{lock_read/lock_write/lock_tree_write} has been called, and become invalid again after unlock is called. This has been done to improve performance and correctness as part of the dirstate development. (Robert Collins, John A Meinel, Martin Pool, and others).

• For users of \texttt{bzrlib}: The attribute ‘tree.inventory’ should be considered readonly. Previously it was possible to directly alter this attribute, or its contents, and have the tree notice this. This has been made unsupported - it may work in some tree formats, but in the newer dirstate format such actions will have no effect and will be ignored, or even cause assertions. All operations possible can still be carried out by a combination of the tree API, and the \texttt{bzrlib.transform API}. (Robert Collins, John A Meinel, Martin Pool, and others).

35.4.2 Improvements

• Support for OS Windows 98. Also .bzr.log on any windows system saved in My Documents folder. (Alexander Belchenko)
• `bzr mv` enhanced to support already moved files. In the past the `mv` command would have failed if the source file doesn’t exist. In this situation `bzr mv` would now detect that the file has already moved and update the repository accordingly, if the target file does exist. A new option `--after` has been added so that if two files already exist, you could notify Bazaar that you have moved a (versioned) file and replaced it with another. Thus in this case `bzr move --after` will only update the Bazaar identifier. (Steffen Eichenberg, Marius Kruger)

• `ls` now works on treeless branches and remote branches. (Aaron Bentley)

• `bzr help global-options` describes the global options. (Aaron Bentley)

• `bzr pull --overwrite` will now correctly overwrite checkouts. (Robert Collins)

• Files are now allowed to change kind (e.g. from file to symlink). Supported by `commit`, `revert` and `status` (Aaron Bentley)

• `inventory` and `unknowns` hidden in favour of `ls` (Aaron Bentley)

• `bzr help checkouts` describes what checkouts are and some possible uses of them. (James Westby, Aaron Bentley)

• A new `-d` option to push, pull and merge overrides the default directory. (Martin Pool)

• Branch format 6: smaller, and potentially faster than format 5. Supports `append_history_only` mode, where the log view and revnos do not change, except by being added to. Stores policy settings in `~/.bzr/branch/branch.conf`.

• `append_only` branches: Format 6 branches may be configured so that log view and revnos are always consistent. Either create the branch using “`bzr init --append-revisions-only`” or edit the config file as described in docs/configuration.txt.

• rebind: Format 6 branches retain the last-used bind location, so if you “`bzr unbind`”, you can “`bzr bind`” to bind to the previously-selected bind location.

• Built-in tags support, created and deleted by the `tag` command and stored in the branch. Tags can be accessed with the `revisionspec -rtag:`, and listed with `bzr tags`. Tags are not versioned at present. Tags require a network incompatible upgrade. To perform this upgrade, run `bzr upgrade --dirstate-tags` in your branch and repositories. (Martin Pool)

• The `bzr://` transport now has a well-known port number, 4155, which it will use by default. (Andrew Benett, Martin Pool)

• Bazaar now looks for user-installed plugins before looking for site-wide plugins. (Jonathan Lange)

• `bzr resolve` now detects and marks resolved text conflicts. (Aaron Bentley)

### 35.4.3 Internals

• Internally revision ids and file ids are now passed around as utf-8 bytestrings, rather than treating them as Unicode strings. This has performance benefits for Knits, since we no longer need to decode the revision id for each line of content, nor for each entry in the index. This will also help with the future dirstate format. (John Arbash Meinel)

• Reserved ids (any revision-id ending in a colon) are rejected by versionedfiles, repositories, branches, and working trees (Aaron Bentley)

• Minor performance improvement by not creating a ProgressBar for every KnitIndex we create. (about 90ms for a bzr.dev tree) (John Arbash Meinel)

• New easier to use Branch hooks facility. There are five initial hooks, all documented in `bzr-lib.branch.BranchHooks.__init__` - ‘set_rh’, ‘post_push’, ‘post_pull’, ‘post_commit’, ‘post_uncommit’. These hooks fire after the matching operation on a branch has taken place, and were originally added for the branchrss plugin. (Robert Collins)
• New method Branch.push() which should be used when pushing from a branch as it makes performance and policy decisions to match the UI level command push. (Robert Collins).

• Add a new method Tree.revision_tree which allows access to cached trees for arbitrary revisions. This allows the in development dirstate tree format to provide access to the callers to cached copies of inventory data which are cheaper to access than inventories from the repository. (Robert Collins, Martin Pool)

• New Branch.last_revision_info method, this is being done to allow optimization of requests for both the number of revisions and the last revision of a branch with smartservers and potentially future branch formats. (Wouter van Heyst, Robert Collins)

• Allow ‘import bzrlib.plugins.NAME’ to work when the plugin NAME has not yet been loaded by load_plugins(). This allows plugins to depend on each other for code reuse without requiring users to perform file-renaming gymnastics. (Robert Collins)

• New Repository method ‘gather_stats’ for statistic data collection. This is expected to grow to cover a number of related uses mainly related to bzr info. (Robert Collins)

• Log formatters are now managed with a registry. log.register_formatter continues to work, but callers accessing the FORMATTERS dictionary directly will not.

• Allow a start message to be passed to the edit_commit_message function. This will be placed in the message offered to the user for editing above the separator. It allows a template commit message to be used more easily. (James Westby)

• GPGStrategy.sign() will now raise BzrBadParameterUnicode if you pass a Unicode string rather than an 8-bit string. Callers need to be updated to encode first. (John Arbash Meinel)

• Branch.push, pull, merge now return Result objects with information about what happened, rather than a scattering of various methods. These are also passed to the post hooks. (Martin Pool)

• File formats and architecture is in place for managing a forest of trees in bzr, and splitting up existing trees into smaller subtrees, and finally joining trees to make a larger tree. This is the first iteration of this support, and the user-facing aspects still require substantial work. If you wish to experiment with it, use bzr upgrade --dirstate-with-subtree in your working trees and repositories. You can use the hidden commands split and join and to create and manipulate nested trees, but please consider using the nested-trees branch, which contains substantial UI improvements, instead. http://code.aaronbentley.com/bzr/bzrrepo/nested-trees/ (Aaron Bentley, Martin Pool, Robert Collins).

35.4.4 Bugfixes

• bzr annotate now uses dotted revnos from the viewpoint of the branch, rather than the last changed revision of the file. (John Arbash Meinel, #82158)

• Lock operations no longer hang if they encounter a permission problem. (Aaron Bentley)

• bzr push can resume a push that was canceled before it finished. Also, it can push even if the target directory exists if you supply the --use-existing-dir flag. (John Arbash Meinel, #30576, #45504)

• Fix HTTP proxy authentication when user and an optional password appears in the *_proxy vars. (Vincent Ladeuil, #83954).

• bzr log branch/file works for local treeless branches (Aaron Bentley, #84247)

• Fix problem with UNC paths on Windows 98. (Alexander Belchenko, #84728)

• Searching location of CA bundle for PyCurl in env variable (CURL_CA_BUNDLE), and on win32 along the PATH. (Alexander Belchenko, #82086)

• bzr init works with unicode argument LOCATION. (Alexander Belchenko, #85599)

• Raise DependencyNotPresent if pycurl do not support https. (Vincent Ladeuil, #85305)
• Invalid proxy env variables should not cause a traceback. (Vincent Ladeuil, #87765)
• Ignore patterns normalised to use ‘/’ path separator. (Kent Gibson, #86451)
• bzr rocks. It sure does! Fix case. (Vincent Ladeuil, #78026)
• Fix bzrtools shelve command for removed lines beginning with “–” (Johan Dahlberg, #75577)

35.4.5 Testing

• New --first option to bzr selftest to run specified tests before the rest of the suite. (Martin Pool)
BAZAAR RELEASE NOTES

36.1 bzr 0.14

Released 2007-01-23

36.1.1 Improvements

• bzr help global-options describes the global options. (Aaron Bentley)

36.1.2 Bug Fixes

• Skip documentation generation tests if the tools to do so are not available. Fixes running selftest for installed copies of bzr. (John Arbash Meinel, #80330)

• Fix the code that discovers whether bzr is being run from its working tree to handle the case when it isn’t but the directory it is in is below a repository. (James Westby, #77306)

36.2 bzr 0.14rc1

Released 2007-01-16

36.2.1 Improvements

• New connection: bzr+http:// which supports tunnelling the smart protocol over an HTTP connection. If writing is enabled on the bzr server, then you can write over the HTTP connection. (Andrew Bennetts, John Arbash Meinel)

• Aliases now support quotation marks, so they can contain whitespace (Marius Kruger)

• PyCurlTransport now use a single curl object. By specifying explicitly the ‘Range’ header, we avoid the need to use two different curl objects (and two connections to the same server). (Vincent Ladeuil)

• bzr commit does not prompt for a message until it is very likely to succeed. (Aaron Bentley)

• bzr conflicts now takes –text to list pathnames of text conflicts (Aaron Bentley)
• Fix `iter_lines_added_or_present_in_versions` to use a set instead of a list while checking if a revision id was requested. Takes 10s off of the `fileids_affected_by_revision_ids` time, which is 10s of the `bzr branch` time. Also improve `fileids_...` time by filtering lines with a regex rather than multiple `str.find()` calls. (saves another 300ms) (John Arbash Meinel)

• Policy can be set for each configuration key. This allows keys to be inherited properly across configuration entries. For example, this should enable you to do:

```bash
$ /home/user/project
push_location = sftp://host/srv/project/
push_location:policy = appendpath
```

And then a branch like `/home/user/project/mybranch` should get an automatic push location of `sftp://host/srv/project/mybranch`. (James Henstridge)

• Added `bzr status --short` to make status report svn style flags for each file. For example:

```bash
$ bzr status --short
A foo
A bar
D baz
? wooley
```

• ‘bzr selftest --clean-output’ allows easily clean temporary tests directories without running tests. (Alexander Belchenko)

• `bzr help hidden-commands` lists all hidden commands. (Aaron Bentley)

• `bzr merge` now has an option `--pull` to fall back to pull if local is fully merged into remote. (Jan Hudec)

• `bzr help formats` describes available directory formats. (Aaron Bentley)

### 36.2.2 Internals

• A few tweaks directly to `fileids_affected_by_revision_ids` to help speed up processing, as well allowing to extract unannotated lines. Between the two `fileids_affected_by_revision_ids` is improved by approx 10%. (John Arbash Meinel)

• Change Revision serialization to only write out millisecond resolution. Rather than expecting floating point serialization to preserve more resolution than we need. (Henri Weichers, Martin Pool)

• Test suite ends cleanly on Windows. (Vincent Ladeuil)

• When `encoding_type` attribute of class Command is equal to ‘exact’, force sys.stdout to be a binary stream on Windows, and therefore keep exact line-endings (without LF -> CRLF conversion). (Alexander Belchenko)

• Single-letter short options are no longer globally declared. (Martin Pool)

• Before using detected user/terminal encoding `bzr` should check that Python has corresponding codec. (Alexander Belchenko)

• Formats for end-user selection are provided via a FormatRegistry (Aaron Bentley)

### 36.2.3 Bug Fixes

• `bzr missing --verbose` was showing adds/removals in the wrong direction. (John Arbash Meinel)

• `bzr annotate` now defaults to showing dotted revnos for merged revisions. It cuts them off at a depth of 12 characters, but you can supply `--long` to see the full number. You can also use `--show-ids` to display the original revision ids, rather than revision numbers and committer names. (John Arbash Meinel, #75637)
• bzr now supports Win32 UNC path (e.g. \HOST\path. (Alexander Belchenko, #57869)

• Win32-specific: output of cat, bundle and diff commands don’t mangle line-endings (Alexander Belchenko, #55276)

• Replace broken fnmatch based ignore pattern matching with custom pattern matcher. (Kent Gibson, Jan Hudec #57637)

• pycurl and urllib can detect short reads at different places. Update the test suite to test more cases. Also detect HTTP error code 416 which was raised for that specific bug. Also enhance the urllib robustness by detecting invalid ranges (and pycurl’s one by detecting short reads during the initial GET). (Vincent Ladeuil, #73948)

• The urllib connection sharing interacts badly with urllib2 proxy setting (the connections didn’t go thru the proxy anymore). Defining a proper ProxyHandler solves the problem. (Vincent Ladeuil, #74759)

• Use urlutils to generate relative URLs, not osutils (Aaron Bentley, #76229)

• bzr status in a readonly directory should work without giving lots of errors. (John Arbash Meinel, #76299)

• Mention the revisionspec topic for the revision option help. (Wouter van Heyst, #31663)

• Allow plugins import from zip archives. (Alexander Belchenko, #68124)
CHAPTER

THIRTYSEVEN

BAZAAR RELEASE NOTES

37.1  bzt 0.13

Released 2006-12-05

No changes from 0.13rc

37.2  bzt 0.13rc1

Released 2006-11-27

37.2.1 Improvements

• New command bzt remove-tree allows the removal of the working tree from a branch. (Daniel Silverstone)

• urllib uses shared keep-alive connections, so HTTP operations are substantially faster. (Vincent Ladeuil, #53654)

• bzt export allows an optional branch parameter, to export a bzt tree from some other URL. For example: bzt export bzt.tar.gz http://bazaar-vcs.org/bzt/bzt.dev (Daniel Silverstone)

• Added bzt help topics to the bzt help system. This gives a location for general information, outside of a specific command. This includes updates for bzt help revisionspec the first topic included. (Goffredo Baroncelli, John Arbash Meinel, #42714)

• WSGI-compatible HTTP smart server. See doc/http_smart_server.txt. (Andrew Bennetts)

• Knit files will now cache full texts only when the size of the deltas is as large as the size of the fulltext. (Or after 200 deltas, whichever comes first). This has the most benefit on large files with small changes, such as the inventory for a large project. (eg For a project with 2500 files, and 7500 revisions, it changes the size of inventory.knit from 11MB to 5.4MB) (John Arbash Meinel)

37.2.2 Internals

• New -D option given before the command line turns on debugging output for particular areas. -Derror shows tracebacks on all errors. (Martin Pool)

• Clean up bzt selftest --benchmark bundle to correct an import, and remove benchmarks that take longer than 10min to run. (John Arbash Meinel)
• Use `time.time()` instead of `time.clock()` to decide on progress throttling. Because `time.clock()` is actually CPU time, so over a high-latency connection, too many updates get throttled. (John Arbash Meinel)

• `MemoryTransport.list_dir()` would strip the first character for files or directories in root directory. (John Arbash Meinel)

• New method `get_branch_reference` on ‘BzrDir’ allows the detection of branch references - which the smart server component needs.

• New `ChrootTransportDecorator`, accessible via the `chroot+` URL prefix. It disallows any access to locations above a set URL. (Andrew Bennetts)

### 37.2.3 Bug Fixes

• Now `_KnitIndex` properly decode revision ids when loading index data. And optimize the knit index parsing code. (Dmitry Vasiliev, John Arbash Meinel)

• `bzrlib/bzrdir.py` was directly referencing `bzrlib.workingtree`, without importing it. This prevented `bzr upgrade` from working unless a plugin already imported `bzrlib.workingtree` (John Arbash Meinel, #70716)

• Suppress the traceback on invalid URLs (Vincent Ladeuil, #70803).

• Give nicer error message when an HTTP server returns a 403 error code. (Vincent Ladeuil, #57644).

• When a multi-range HTTP GET request fails, try a single range one. If it fails too, forget about ranges. Remember that until the death of the transport and propagates that to the clones. (Vincent Ladeuil, #62276, #62029).

• Handles user/passwords supplied in URL from command line (for the `urllib` implementation). Don’t request already known passwords (Vincent Ladeuil, #42383, #44647, #48527)

• `_KnitIndex.add_versions()` dictionary compresses revision ids as they are added. This fixes bug where fetching remote revisions records them as full references rather than integers. (John Arbash Meinel, #64789)

• `bzr ignore` strips trailing slashes in patterns. Also `bzr ignore` rejects absolute paths. (Kent Gibson, #4559)

• `bzr ignore` takes multiple arguments. (Cheuksan Edward Wang, #29488)

• `mv` correctly handles paths that traverse symlinks. (Aaron Bentley, #66964)

• Give nicer looking error messages when failing to connect over SSH. (John Arbash Meinel, #49172)

• Pushing to a remote branch does not currently update the remote working tree. After a remote push, `bzr status` and `bzr diff` on the remote machine now show that the working tree is out of date. (Cheuksan Edward Wang #48136)

• Use `patiencediff` instead of `difflib` for determining deltas to insert into knits. This avoids the O(N^3) behavior of `difflib`. Patience diff should be O(N^2). (Cheuksan Edward Wang, #65714)

• Running `bzr log` on nonexistent file gives an error instead of the entire log history. (Cheuksan Edward Wang #50793)

• `bzr cat` can look up contents of removed or renamed files. If the pathname is ambiguous, i.e. the files in the old and new trees have different id’s, the default is the file in the new tree. The user can use “--name-from-revision” to select the file in the old tree. (Cheuksan Edward Wang, #30190)

### 37.2.4 Testing

• Testing `HTTPRequestHandler` really handles the Range header (previously it was ignoring it and returning the whole file.).
BAZAAR RELEASE NOTES

38.1 bzr 0.12

Released 2006-10-30

38.1.1 Internals

• Clean up bzr selftest --benchmark bundle to correct an import, and remove benchmarks that take longer than 10min to run. (John Arbash Meinel)

38.2 bzr 0.12rc1

Released 2006-10-23

38.2.1 Improvements

• bzr log now shows dotted-decimal revision numbers for all revisions, rather than just showing a decimal revision number for revisions on the mainline. These revision numbers are not yet accepted as input into bzr commands such as log, diff etc. (Robert Collins)

• revisions can now be specified using dotted-decimal revision numbers. For instance, bzr diff -r 1.2.1..1.2.3. (Robert Collins)

• bzr help commands output is now shorter (Aaron Bentley)

• bzr now uses lazy importing to reduce the startup time. This has a moderate effect on lots of actions, especially ones that have little to do. For example bzr rocks time is down to 116ms from 283ms. (John Arbash Meinel)

• New Registry class to provide name-to-object registry-like support, for example for schemes where plugins can register new classes to do certain tasks (e.g. log formatters). Also provides lazy registration to allow modules to be loaded on request. (John Arbash Meinel, Adeodato Simó)

38.2.2 API Incompatability

• LogFormatter subclasses show now expect the ‘revno’ parameter to show() to be a string rather than an int. (Robert Collins)
38.2.3 Internals

- `TestCase.run_bzr`, `run_bzr_captured`, and `run_bzr_subprocess` can take a `working_dir='foo'` parameter, which will change directory for the command. (John Arbash Meinel)

- `bzrlib.lazy_regex.lazy_compile` can be used to create a proxy around a regex, which defers compilation until first use. (John Arbash Meinel)

- `TestCase.run_bzr_subprocess` defaults to supplying the `--no-plugins` parameter to ensure test reproducability, and avoid problems with system-wide installed plugins. (John Arbash Meinel)

- Unique tree root ids are now supported. Newly created trees still use the common root id for compatibility with bzr versions before 0.12. (Aaron Bentley)

- `WorkingTree.set_root_id(None)` is now deprecated. Please pass in `inventory.ROOT_ID` if you want the default root id value. (Robert Collins, John Arbash Meinel)

- New method `WorkingTree.flush()` which will write the current memory inventory out to disk. At the same time, `read_working_inventory` will no longer trash the current tree inventory if it has been modified within the current lock, and the tree will now `flush()` automatically on `unlock()`. `WorkingTree.set_root_id()` has been updated to take advantage of this functionality. (Robert Collins, John Arbash Meinel)

- `bzrlib.tsort.merge_sorted` now accepts `generate_revnos`. This parameter will cause it to add another column to its output, which contains the dotted-decimal revno for each revision, as a tuple. (Robert Collins)

- `LogFormatter.show_merge` is deprecated in favour of `LogFormatter.show_merge_revno`. (Robert Collins)

38.2.4 Bug Fixes

- Avoid circular imports by creating a deprecated function for `bzrlib.tree.RevisionTree`. Callers should have been using `bzrlib.revisontree.RevisionTree` anyway. (John Arbash Meinel, #66349)

- Don’t use `socket.MSG_WAITALL` as it doesn’t exist on all platforms. (Martin Pool, #66356)

- Don’t require `Content-Type` in range responses. Assume they are a single range if `Content-Type` does not exist. (John Arbash Meinel, #62473)

- `bzr branch/pull` no longer complain about progress bar cleanup when interrupted during fetch. (Aaron Bentley, #54000)

- `WorkingTree.set_parent_trees()` uses the trees to directly write the basis inventory, rather than going through the repository. This allows us to have 1 inventory read, and 2 inventory writes when committing a new tree. (John Arbash Meinel)

- When reverting, files that are not locally modified that do not exist in the target are deleted, not just unversioned (Aaron Bentley)

- When trying to acquire a lock, don’t fail immediately. Instead, try a few times (up to 1 hour) before timing out. Also, report why the lock is unavailable (John Arbash Meinel, #43521, #49556)

- Leave `HttpTransportBase` daughter classes decides how they implement cloning. (Vincent Ladeuil, #61606)

- `diff3` does not indicate conflicts on clean merge. (Aaron Bentley)

- If a commit fails, the commit message is stored in a file at the root of the tree for later commit. (Cheuksan Edward Wang, Stefan Metzmacher, #32054)
38.2.5 Testing

- New test base class `TestCaseWithMemoryTransport` offers memory-only testing facilities: it’s not suitable for tests that need to mutate disk state, but most tests should not need that and should be converted to `TestCaseWithMemoryTransport`. (Robert Collins)

- `TestCase.make_branch_and_memory_tree` now takes a format option to set the BzrDir, Repository and Branch formats of the created objects. (Robert Collins, John Arbash Meinel)
CHAPTER

THIRTYNINE

BAZAAR RELEASE NOTES

39.1 bazaar 0.11

   Released  2006-10-02
   • Smart server transport test failures on windows fixed. (Lukáš Lalinský).

39.2 bazaar 0.11rc2

   Released  2006-09-27

39.2.1 Bug Fixes

   • Test suite hangs on windows fixed. (Andrew Bennets, Alexander Belchenko).
   • Commit performance regression fixed. (Aaron Bentley, Robert Collins, John Arbash Meinel).

39.3 bazaar 0.11rc1

   Released  2006-09-25

39.3.1 Improvements

   • Knit files now wait to create their contents until the first data is added. The old code used to create an empty .knit
     and a .kndx with just the header. However, this caused a lot of extra round trips over SFTP. This can change the
     time for bazaar push to create a new remote branch from 160s down to 100s. This also affects bazaar commit
     performance when adding new files, bazaar commit on a new kernel-like tree drops from 50s down to 40s (John
     Arbash Meinel, #44692)

   • When an entire subtree has been deleted, commit will now report that just the top of the subtree has been deleted,
     rather than reporting all the individual items. (Robert Collins)

   • Commit performs one less XML parse. (Robert Collins)

   • bazaar checkout now operates on readonly branches as well as readwrite branches. This fixes bug #39542.
     (Robert Collins)
Bazaar Release Notes, Release 2.8.0dev1

• bzr bind no longer synchronises history with the master branch. Binding should be followed by an update
or push to synchronise the two branches. This is closely related to the fix for bug #39542. (Robert Collins)
• bzrlib.lazy_import.lazy_import function to create on-demand objects. This allows all imports to
stay at the global scope, but modules will not actually be imported if they are not used. (John Arbash Meinel)
• Support bzr:// and bzr+ssh:// URLS to work with the new RPC-based transport which will be used with
the upcoming high-performance smart server. The new command bzr serve will invoke bzr in server mode,
which processes these requests. (Andrew Bennetts, Robert Collins, Martin Pool)
• New command bzr version-info which can be used to get a summary of the current state of the tree. This
is especially useful as part of a build commands. See doc/version_info.txt for more information (John
Arbash Meinel)

39.3.2 Bug Fixes
• ’bzr inventory [FILE...]’ allows restricting the file list to a specific set of files. (John Arbash
Meinel, #3631)
• Don’t abort when annotating empty files (John Arbash Meinel, #56814)
• Add Stanza.to_unicode() which can be passed to another Stanza when nesting stanzas. Also, add
read_stanza_unicode to handle when reading a nested Stanza. (John Arbash Meinel)
• Transform._set_mode() needs to stat the right file. (John Arbash Meinel, #56549)
• Raise WeaveFormatError rather than StopIteration when trying to read an empty Weave file. (John Arbash
Meinel, #46871)
• Don’t access e.code for generic URLErrors, only HTTPErrors have .code. (Vincent Ladeuil, #59835)
• Handle boundary=”” lines properly to allow access through a Squid proxy. (John Arbash Meinel, #57723)
• revert now removes newly-added directories (Aaron Bentley, #54172)
• bzr upgrade sftp:// shouldn’t fail to upgrade v6 branches if there isn’t a working tree. (David Allouche,
#40679)
• Give nicer error messages when a user supplies an invalid –revision parameter. (John Arbash Meinel, #55420)
• Handle when LANG is not recognized by python. Emit a warning, but just revert to using ‘ascii’. (John Arbash
Meinel, #35392)
• Don’t use preexec_fn on win32, as it is not supported by subprocess. (John Arbash Meinel)
• Skip specific tests when the dependencies aren’t met. This includes some setup.py tests when python-dev
is not available, and some tests that depend on paramiko. (John Arbash Meinel, Mattheiu Moy)
• Fallback to Paramiko properly, if no ssh executable exists on the system. (Andrew Bennetts, John Arbash
Meinel)
• Branch.bind(other_branch) no longer takes a write lock on the other branch, and will not push or pull
between the two branches. API users will need to perform a push or pull or update operation if they require
branch synchronisation to take place. (Robert Collins, #43744)
• When creating a tarball or zipfile export, export unicode names as utf-8 paths. This may not work perfectly on
all platforms, but has the best chance of working in the common case. (John Arbash Meinel, #56815)
• When committing, only files that exist in working tree or basis tree may be specified (Aaron Bentley, #50793)

250

Chapter 39. Bazaar Release Notes


39.3.3 Portability

- Fixes to run on Python 2.5 (Brian M. Carlson, Martin Pool, Marien Zwart)

39.3.4 Internals

- TestCaseInTempDir now creates a separate directory for HOME, rather than having HOME set to the same location as the working directory. (John Arbash Meinel)

- run_bzr_subprocess() can take an optional env_changes={} parameter, which will update os.environ inside the spawned child. It also can take a universal_newlines=True, which helps when checking the output of the command. (John Arbash Meinel)

- Refactor SFTP vendors to allow easier re-use when SSH is used. (Andrew Bennetts)

- Transport.list_dir() and Transport.iter_files_recursive() should always return unlescaped paths. This is now tested (there were bugs in a few of the transports) (Andrew Bennetts, David Allouche, John Arbash Meinel)

- New utility function symbol_versioning.deprecation_string. Returns the formatted string for a callable, deprecation format pair. (Robert Collins)

- New TestCase helper applyDeprecated. This allows you to call a callable which is deprecated without it spewing to the screen, just by supplying the deprecation format string issued for it. (Robert Collins)

- Transport.append and Transport.put have been deprecated in favor of .append_bytes, .append_file, .put_bytes, and .put_file. This removes the ambiguity in what type of object the functions take. Transport.non_atomic_put_{bytes,file} has also been added. Which works similarly to Transport.append() except for SFTP, it doesn’t have a round trip when opening the file. Also, it provides functionality for creating a parent directory when trying to create a file, rather than raise NoSuchFile and forcing the caller to repeat their request. (John Arbash Meinel)

- WorkingTree has a new api unversion which allow the unversioning of entries by their file id. (Robert Collins)

- WorkingTree.pending_merges is deprecated. Please use the get_parent_ids (introduced in 0.10) method instead. (Robert Collins)

- WorkingTree has a new lock_tree_write method which locks the branch for read rather than write. This is appropriate for actions which only need the branch data for reference rather than mutation. A new decorator needs_tree_write_lock is provided in the workingtree module. Like the needs_read_lock and needs_write_lock decorators this allows static declaration of the locking requirements of a function to ensure that a lock is taken out for casual scripts. (Robert Collins, #54107)

- All WorkingTree methods which write to the tree, but not to the branch have been converted to use needs_tree_write_lock rather than needs_write_lock. Also converted is the revert, conflicts and tree transform modules. This provides a modest performance improvement on metadir style trees, due to the reduce lock-acquisition, and a more significant performance improvement on lightweight checkouts from remote branches, where trivial operations used to pay a significant penalty. It also provides the basis for allowing readonly checkouts. (Robert Collins)

- Special case importing the standard library ‘copy’ module. This shaves off 40ms of startup time, while retaining compatibility. See: bzrlib/inspect_for_copy.py for more details. (John Arbash Meinel)

- WorkingTree has a new parent class MutableTree which represents the specialisations of Tree which are able to be altered. (Robert Collins)

- New methods mkdir and put_file_bytes_non_atomic on MutableTree that mutate the tree and its contents. (Robert Collins)
Transport behaviour at the root of the URL is now defined and tested. (Andrew Bennetts, Robert Collins)

### 39.3.5 Testing

- New test helper class MemoryTree. This is typically accessed via `self.make_branch_and_memory_tree()` in test cases. (Robert Collins)

- Add `start_bzr_subprocess` and `stop_bzr_subprocess` to allow test code to continue running concurrently with a subprocess of bzr. (Andrew Bennetts, Robert Collins)

- Add a new method `Transport.get_smart_client()`. This is provided to allow upgrades to a richer interface than the VFS one provided by Transport. (Andrew Bennetts, Martin Pool)
BAZAAR RELEASE NOTES

40.1 bzr 0.10

Released 2006-08-29

40.1.1 Improvements

• ‘merge’ now takes --uncommitted, to apply uncommitted changes from a tree. (Aaron Bentley)

• ‘bzr add --file-ids-from’ can be used to specify another path to use for creating file ids, rather than generating all new ones. Internally, the ‘action’ passed to smart_add_tree() can return file_ids that will be used, rather than having bzrlib generate new ones. (John Arbash Meinel, #55781)

• bzr selftest --benchmark now allows a --cache-dir parameter. This will cache some of the intermediate trees, and decrease the setup time for benchmark tests. (John Arbash Meinel)

• Inverse forms are provided for all boolean options. For example, --strict has --no-strict, --no-recurse has --recurse (Aaron Bentley)

• Serialize out Inventories directly, rather than using ElementTree. Writing out a kernel sized inventory drops from 2s down to ~350ms. (Robert Collins, John Arbash Meinel)

40.1.2 Bug Fixes

• Help diffutils 2.8.4 get along with binary tests (Marien Zwart: #57614)

• Change LockDir so that if the lock directory doesn’t exist when lock_write() is called, an attempt will be made to create it. (John Arbash Meinel, #56974)

• bzr uncommit preserves pending merges. (John Arbash Meinel, #57660)

• Active FTP transport now works as intended. (ghozzy, #56472)

• Really fix mutter() so that it won’t ever raise a UnicodeError. It means it is possible for ~/.bzr.log to contain non UTF-8 characters. But it is a debugging log, not a real user file. (John Arbash Meinel, #56947, #53880)

• Change Command handle to allow Unicode command and options. At present we cannot register Unicode command names, so we will get BzrCommandError(‘unknown command’), or BzrCommandError(‘unknown option’) But that is better than a UnicodeError + a traceback. (John Arbash Meinel, #57123)

• Handle TZ=UTC properly when reading/writing revisions. (John Arbash Meinel, #55783, #56290)

• Use GPG_TTY to allow gpg --cl to work with gpg-agent in a pipeline, (passing text to sign in on stdin). (John Arbash Meinel, #54468)
• External diff does the right thing for binaries even in foreign languages. (John Arbash Meinel, #56307)

• Testament handles more cases when content is unicode. Specific bug was in handling of revision properties. (John Arbash Meinel, Holger Krekel, #54723)

• The bzr selftest was failing on installed versions due to a bug in a new test helper. (John Arbash Meinel, Robert Collins, #58057)

40.1.3 Internals

• bzrlib.cache_utf8 contains encode() and decode() functions which can be used to cache the conversion between utf8 and Unicode. Especially helpful for some of the knit annotation code, which has to convert revision ids to utf8 to annotate lines in storage. (John Arbash Meinel)

• setup.py now searches the filesystem to find all packages which need to be installed. This should help make the life of packagers easier. (John Arbash Meinel)
BAZAAR RELEASE NOTES

41.1  bzr 0.9.0

Released  2006-08-11

41.1.1  Surprises

• The hard-coded built-in ignore rules have been removed. There are now two rule sets which are enforced. A
  user global one in ~/.bazaar/ignore which will apply to every tree, and the tree specific one ‘.bzrignore’. ~/.bazaar/ignore
  will be created if it does not exist, but with a more conservative list than the old default. This fixes bugs with
default rules being enforced no matter what. The old list of ignore rules from bzr is available by running ‘bzr ignore
–old-default-rules’. (Robert Collins, Martin Pool, John Arbash Meinel)

• ‘branches.conf’ has been changed to ‘locations.conf’, since it can apply to more locations than just branch
  locations. (Aaron Bentley)

41.1.2  Improvements

• The revision specifier “revno:” is extended to accept the syntax revno:N:branch. For example,
  revno:42:http://bazaar-vcs.org/bzr/bzr.dev/ means revision 42 in bzr.dev. (Matthieu Moy)

• Tests updates to ensure proper URL handling, UNICODE support, and proper printing when the user’s terminal
  encoding cannot display the path of a file that has been versioned. bzr branch can take a target URL rather
  than only a local directory. Branch.get_parent() / set_parent() now save a relative path if possible,
  and normalize the parent based on root, allowing access across different transports. (John Arbash Meinel,
  Wouter van Heyst, Martin Pool) (Malone #48906, #42699, #40675, #5281, #3980, #36363, #43689, #42517,
  #42514)

• On Unix, detect terminal width using an ioctl not just $COLUMNS. Use terminal width for single-line logs from
  bzr log --line and pending-merge display. (Robert Widhopf-Fenk, Gustavo Niemeyer) (Malone #3507)

• On Windows, detect terminal width using GetConsoleScreenBufferInfo. (Alexander Belchenko)

• Speedup improvement for ‘date:’-revision search. (Guillaume Pinot).

• Show the correct number of revisions pushed when pushing a new branch. (Robert Collins).

• ‘bzr selftest’ now shows a progress bar with the number of tests, and progress made. ‘make check’ shows tests
  in -v mode, to be more useful for the PQM status window. (Robert Collins). When using a progress bar, failed
  tests are printed out, rather than being overwritten by the progress bar until the suite finishes. (John Arbash
  Meinel)
• ‘bzr selftest –benchmark’ will run a new benchmarking selftest. ‘bzr selftest –benchmark –lsprof-timed’ will use lspy profile to generate profile data for the individual profiled calls, allowing for fine grained analysis of performance. (Robert Collins, Martin Pool).

• ‘bzr commit’ shows a progress bar. This is useful for commits over SFTP where commit can take an appreciable time. (Robert Collins)

• ‘bzr add’ is now less verbose in telling you what ignore globs were matched by files being ignored. Instead it just tells you how many were ignored (because you might reasonably be expecting none to be ignored). ‘bzr add -v’ is unchanged and will report every ignored file. (Robert Collins).

• FTP now has a test server if medusa is installed. As part of testing, FTP support has been improved, including support for supplying a non-standard port. (John Arbash Meinel).

• ‘bzr log –line’ shows the revision number, and uses only the first line of the log message (#5162, Alexander Belchenko; Matthieu Moy)

• ‘bzr status’ has had the –all option removed. The ‘bzr ls’ command should be used to retrieve all versioned files. (Robert Collins)

• ‘bzr bundle OTHER/BRANCH’ will create a bundle which can be sent over email, and applied on the other end, while maintaining ancestry. This bundle can be applied with either ‘bzr merge’ or ‘bzr pull’, the same way you would apply another branch. (John Arbash Meinel, Aaron Bentley)

• ‘bzr whoami’ can now be used to set your identity from the command line, for a branch or globally. (Robey Pointer)

• ‘bzr checkout’ now aliased to ‘bzr co’, and ‘bzr annotate’ to ‘bzr ann’. (Michael Ellerman)

• ‘bzr revert DIRECTORY’ now reverts the contents of the directory as well. (Aaron Bentley)

• ‘bzr get sftp://foo’ gives a better error when paramiko is not present. Also updates things like ‘http+pycurl://’ if pycurl is not present. (John Arbash Meinel) (Malone #47821, #52204)

• New env variable BZR_PROGRESS_BAR, sets the default progress bar type. Can be set to ‘none’ or ‘dummy’ to disable the progress bar, ‘dots’ or ‘tty’ to create the respective type. (John Arbash Meinel, #42197, #51107)

• Improve the help text for ‘bzr diff’ to explain what various options do. (John Arbash Meinel, #6391)

• ‘bzr uncommit -r 10’ now uncommits revisions 11.. rather than uncommitting revision 10. This makes -r10 more in line with what other commands do. ‘bzr uncommit’ also now saves the pending merges of the revisions that were removed. So it is safe to uncommit after a merge, fix something, and commit again. (John Arbash Meinel, #32526, #31426)

• ‘bzr init’ now also works on remote locations. (Wouter van Heyst, #48904)

• HTTP support has been updated. When using pycurl we now support connection keep-alive, which reduces dns requests and round trips. And for both urllib and pycurl we support multi-range requests, which decreases the number of round-trips. Performance results for bzr branch http://bazaar-vcs.org/bzr/bzr.dev/ indicate HTTP branching is now 2-3x faster, and bzr pull in an existing branch is as much as 4x faster. (Michael Ellerman, Johan Rydberg, John Arbash Meinel, #46768)

• Performance improvements for SFTP. Branching and pulling are now up to 2x faster. Utilize paramiko.readv() support for async requests if it is available (paramiko > 1.6) (John Arbash Meinel)

41.1.3 Bug Fixes

• Fix shadowed definition of TestLocationConfig that caused some tests not to run. (Erik Bågfors, Michael Ellerman, Martin Pool, #32587)
• Fix unnecessary requirement of sign-my-commits that it be run from a working directory. (Martin Pool, Robert Collins)
• ‘bzr push location’ will only remember the push location if it succeeds in connecting to the remote location. (John Arbash Meinel, #49742)
• ‘bzr revert’ no longer toggles the executable bit on win32 (John Arbash Meinel, #45010)
• Handle broken pipe under win32 correctly. (John Arbash Meinel)
• SFTP tests now work correctly on win32 if you have a newer paramiko (John Arbash Meinel)
• Cleanup win32 test suite, and general cleanup of places where file handles were being held open. (John Arbash Meinel)
• When specifying filenames for ‘diff -r x..y’, the name of the file in the working directory can be used, even if its name is different in both x and y.
• File-ids containing single- or double-quotes are handled correctly by push. (Aaron Bentley, #52227)
• Normalize unicode filenames to ensure cross-platform consistency. (John Arbash Meinel, #43689)
• The argument parser can now handle ‘-’ as an argument. Currently no code interprets it specially (it is mostly handled as a file named ‘-’). But plugins, and future operations can use it. (John Arbash meinel, #50984)
• Bundles can properly read binary files with a plain ‘r’ in them. (John Arbash Meinel, #51927)
• Tuning iter_entries() to be more efficient (John Arbash Meinel, #5444)
• Lots of win32 fixes (the test suite passes again). (John Arbash Meinel, #50155)
• Handle openbsd returning None for sys.getfilesystemencoding() (#41183)
• Support FTP APPE (append) to allow Knits to be used over FTP (#42592)
• Removals are only committed if they match the filespec (or if there is no filespec). (#46635, Aaron Bentley)
• smart-add recurses through all supplied directories (John Arbash Meinel, #52578)
• Make the bundle reader extra lines before and after the bundle text. This allows you to parse an email with the bundle inline. (John Arbash Meinel, #49182)
• Change the file id generator to squash a little bit more. Helps when working with long filenames on windows. (Also helps for unicode filenames not generating hidden files). (John Arbash Meinel, #43801)
• Restore terminal mode on C-c while reading SFTP password. (#48923, Nicholas Allen, Martin Pool)
• Timestamps are rounded to 1ms, and revision entries can be recreated exactly. (John Arbash Meinel, Jamie Wilkinson, #40693)
• Branch.base has changed to a URL, but ~/.bazaar/locations.conf should use local paths, since it is user visible (John Arbash Meinel, #53653)
• bzr status foo when foo was unversioned used to cause a full delta to be generated (John Arbash Meinel, #53638)
• When reading revision properties, an empty value should be considered the empty string, not None (John Arbash Meinel, #47782)
• bzr diff --diff-options can now handle binary files being changed. Also, the output is consistent when --diff-options is not supplied. (John Arbash Meinel, #54651, #52930)
• Use the right suffixes for loading plugins (John Arbash Meinel, #51810)
• Fix Branch.get_parent() to handle the case when the parent is not accessible (John Arbash Meinel, #52976)
41.1.4 Internals

- Combine the ignore rules into a single regex rather than looping over them to reduce the threshold where \(N^2\) behaviour occurs in operations like status. (Jan Hudec, Robert Collins)

- Appending to `bzrlib.DEFAULT_IGNORE` is now deprecated. Instead, use one of the add functions in `bzrlib.ignores`. (John Arbash Meinel)

- ‘bzr push’ should only push the ancestry of the current revision, not all of the history in the repository. This is especially important for shared repositories. (John Arbash Meinel)

- `bzrlib.delta.compare_trees` now iterates in alphabetically sorted order, rather than randomly walking the inventories. (John Arbash Meinel)

- Doctests are now run in temporary directories which are cleaned up when they finish, rather than using special ScratchDir/ScratchBranch objects. (Martin Pool)

- Split `check` into separate methods on the branch and on the repository, so that it can be specialized in ways that are useful or efficient for different formats. (Martin Pool, Robert Collins)

- Deprecate `Repository.all_revision_ids`; most methods don’t really need the global revision graph but only that part leading up to a particular revision. (Martin Pool, Robert Collins)

- Add a BzrDirFormat `control_formats` list which allows for control formats that do not use ‘.bzr’ to store their data - i.e. ‘.svn’, ‘.hg’ etc. (Robert Collins, Jelmer Vernooij).

- `bzrlib.diff.external_diff` can be redirected to any file-like object. Uses subprocess instead of `spawnvp`. (James Henstridge, John Arbash Meinel, #4047, #48914)

- New command line option ‘--profile-imports’, which will install a custom importer to log time to import modules and regex compilation time to `sys.stderr` (John Arbash Meinel)

- ‘EmptyTree’ is now deprecated. please use `repository.revision_tree(None)` instead. (Robert Collins)

- “RevisionTree” is now in `bzrlib/revisiontree.py`. (Robert Collins)
BAZAAR RELEASE NOTES

42.1  bzr 0.8.2

Released  2006-05-17

42.1.1 Bug Fixes

• setup.py failed to install launchpad plugin. (Martin Pool)

42.2  bzr 0.8.1

Released  2006-05-16

42.2.1 Bug Fixes

• Fix failure to commit a merge in a checkout. (Martin Pool, Robert Collins, Erik Bågfors, #43959)
• Nicer messages from ‘commit’ in the case of renames, and correct messages when a merge has occurred. (Robert Collins, Martin Pool)
• Separate functionality from assert statements as they are skipped in optimized mode of python. Add the same check to pending merges. (Olaf Conradi, #44443)

42.2.2 Changes

• Do not show the None revision in output of bzr ancestry. (Olaf Conradi)
• Add info on standalone branches without a working tree. (Olaf Conradi, #44155)
• Fix bug in knits when raising InvalidRevisionId. (Olaf Conradi, #44284)

42.2.3 Changes

• Make editor invocation comply with Debian Policy. First check environment variables VISUAL and EDITOR, then try editor from alternatives system. If that all fails, fall back to the pre-defined list of editors. (Olaf Conradi, #42904)
42.2.4 New Features

- New ‘register-branch’ command registers a public branch into Launchpad.net, where it can be associated with bugs, etc. (Martin Pool, Bjorn Tillenius, Robert Collins)

42.2.5 Internals

- New public api in InventoryEntry - describe_change(old, new) which provides a human description of the changes between two old and new. (Robert Collins, Martin Pool)

42.2.6 Testing

- Fix test case for bzr info in upgrading a standalone branch to metadir, uses bzrlib api now. (Olaf Conradi)

42.3 bzr 0.8

Released 2006-05-08

42.3.1 Notes When Upgrading

Release 0.8 of bzr introduces a new format for history storage, called ‘knit’, as an evolution of to the ‘weave’ format used in 0.7. Local and remote operations are faster using knits than weaves. Several operations including ‘init’, ‘init-repo’, and ‘upgrade’ take a –format option that controls this. Branching from an existing branch will keep the same format.

It is possible to merge, pull and push between branches of different formats but this is slower than moving data between homogenous branches. It is therefore recommended (but not required) that you upgrade all branches for a project at the same time. Information on formats is shown by ‘bzr info’.

bzr 0.8 now allows creation of ‘repositories’, which hold the history of files and revisions for several branches. Previously bzr kept all the history for a branch within the .bzr directory at the root of the branch, and this is still the default. To create a repository, use the new ‘bzr init-repo’ command. Branches exist as directories under the repository and contain just a small amount of information indicating the current revision of the branch.

bzr 0.8 also supports ‘checkouts’, which are similar to in cvs and subversion. Checkouts are associated with a branch (optionally in a repository), which contains all the historical information. The result is that a checkout can be deleted without losing any already-committed revisions. A new ‘update’ command is also available.

Repositories and checkouts are not supported with the 0.7 storage format. To use them you must upgrad to either knits, or to the ‘metaweave’ format, which uses weaves but changes the .bzr directory arrangement.

42.3.2 Improvements

- SFTP paths can now be relative, or local, according to the lftp convention. Paths now take the form:
  sftp://user:pass@host:port/~/relative/path
  or
  sftp://user:pass@host:port/absolute/path

- The FTP transport now tries to reconnect after a temporary failure. FTP put is made atomic. (Matthieu Moy)
• The FTP transport now maintains a pool of connections, and reuses them to avoid multiple connections to the same host (like SFTP did). (Daniel Silverstone)

• The `bzr_man.py` file has been removed. To create the man page now, use `./generate_docs.py man`. The new program can also create other files. Run `python generate_docs.py --help` for usage information. (Hans Ulrich Niedermann & James Blackwell)

• Man Page now gives full help (James Blackwell). Help also updated to reflect user config now being stored in `.bazaar` (Hans Ulrich Niedermann)

• It’s now possible to set aliases in bazaar.conf (Erik Bågfors)

• Pull now accepts a `--revision` argument (Erik Bågfors)

• `bzr re-sign` now allows multiple revisions to be supplied on the command line. You can now use the following command to sign all of your old commits:

```
find .bzr/revision-store// -name my@email-* \\
| sed 's/.\*/\/.\///' \\
| xargs bzr re-sign
```

• Upgrade can now upgrade over the network. (Robert Collins)

• Two new commands ‘bzr checkout’ and ‘bzr update’ allow for CVS/SVN-alike behaviour. By default they will cache history in the checkout, but with `--lightweight` almost all data is kept in the master branch. (Robert Collins)

• ‘revert’ unversions newly-versioned files, instead of deleting them.

• ‘merge’ is more robust. Conflict messages have changed.

• ‘merge’ and ‘revert’ no longer clobber existing files that end in ‘~’ or ‘.moved’.

• Default log format can be set in configuration and plugins can register their own formatters. (Erik Bågfors)

• New ‘reconcile’ command will check branch consistency and repair indexes that can become out of sync in pre 0.8 formats. (Robert Collins, Daniel Silverstone)

• New ‘bzr init --format’ and ‘bzr upgrade --format’ option to control what storage format is created or produced. (Robert Collins, Martin Pool)

• Add parent location to ‘bzr info’, if there is one. (Olaf Conradi)

• New developer commands ‘weave-list’ and ‘weave-join’. (Martin Pool)

• New ‘init-repository’ command, plus support for repositories in ‘init’ and ‘branch’ (Aaron Bentley, Erik Bågfors, Robert Collins)

• Improve output of ‘info’ command. Show all relevant locations related to working tree, branch and repository. Use kibibytes for binary quantities. Fix off-by-one error in missing revisions of working tree. Make ‘info’ work on branches, repositories and remote locations. Show locations relative to the shared repository, if applicable. Show locking status of locations. (Olaf Conradi)

• Diff and merge now safely handle binary files. (Aaron Bentley)

• ‘pull’ and ‘push’ now normalise the revision history, so that any two branches with the same tip revision will have the same output from ‘log’. (Robert Collins)

• ‘merge’ accepts `--remember` option to store parent location, like ‘push’ and ‘pull’. (Olaf Conradi)

• `bzr status` and `diff` when files given as arguments do not exist in the relevant trees. (Martin Pool, #3619)

• Add `.hg` to the default ignore list. (Martin Pool)

• ‘knit’ is now the default disk format. This improves disk performance and utilization, increases incremental pull performance, robustness with SFTP and allows checkouts over SFTP to perform acceptably. The initial
Knit code was contributed by Johan Rydberg based on a specification by Martin Pool. (Robert Collins, Aaron Bentley, Johan Rydberg, Martin Pool).

- New tool to generate all-in-one html version of the manual. (Alexander Belchenko)
- Hitting CTRL-C while doing an SFTP push will no longer cause stale locks to be left in the SFTP repository. (Robert Collins, Martin Pool).
- New option ‘diff –prefix’ to control how files are named in diff output, with shortcuts ‘-p0’ and ‘-p1’ corresponding to the options for GNU patch. (Alexander Belchenko, Goffredo Baroncelli, Martin Pool)
- Add –revision option to ‘annotate’ command. (Olaf Conradi)
- If bzr shows an unexpected revision-history after pulling (perhaps due to a reweave) it can now be corrected by ‘bzr reconcile’. (Robert Collins)

**42.3.3 Changes**

- Commit is now verbose by default, and shows changed filenames and the new revision number. (Robert Collins, Martin Pool)
- Unify ‘mv’, ‘move’, ‘rename’. (Matthew Fuller, #5379)
- ‘bzr -h’ shows help. (Martin Pool, Ian Bicking, #35940)
- Make ‘pull’ and ‘push’ remember location on failure using –remember. (Olaf Conradi)
- For compatibility, make old format for using weaves inside metadir available as ‘metawave’ format. Rename format ‘metadir’ to ‘default’. Clean up help for option –format in commands ‘init’, ‘init-repo’ and ‘upgrade’. (Olaf Conradi)

**42.3.4 Internals**

- The internal storage of history, and logical branch identity have now been split into Branch, and Repository. The common locking and file management routines are now in bzrlib.lockablefiles. (Aaron Bentley, Robert Collins, Martin Pool)
- Transports can now raise DependencyNotPresent if they need a library which is not installed, and then another implementation will be tried. (Martin Pool)
- Remove obsolete (and no-op) decode parameter to Transport.get. (Martin Pool)
- Using Tree Transform for merge, revert, tree-building
- WorkingTree.create, Branch.create, WorkingTree.create_standalone, Branch.initialize are now deprecated. Please see BzrDir.create_* for replacement API’s. (Robert Collins)
- New BzrDir class represents the .bzr control directory and manages formatting issues. (Robert Collins)
- New repository.InterRepository class encapsulates Repository to Repository actions and allows for clean selection of optimised code paths. (Robert Collins)
- bzrlib.fetch.fetch and bzrlib.fetch.greedy_fetch are now deprecated, please use branch.fetch or repository.fetch depending on your needs. (Robert Collins)
- deprecated methods now have a is_deprecated flag on them that can be checked, if you need to determine whether a given callable is deprecated at runtime. (Robert Collins)
- Progress bars are now nested - see bzrlib.ui.ui_factory.nested_progress_bar. (Robert Collins, Robey Pointer)
- New API call get_format_description() for each type of format. (Olaf Conradi)
• Changed `branch.set_parent()` to accept None to remove parent. (Olaf Conradi)
• Deprecated BzrError AmbiguousBase. (Olaf Conradi)
• WorkingTree.branch is now a read only property. (Robert Collins)
• `bzrlib.ui.text.TextUIFactory` now accepts a `bar_type` parameter which can be None or a factory that will create a progress bar. This is useful for testing or for overriding the `bzrlib.progress` heuristic. (Robert Collins)
• New API method `get_physical_lock_status()` to query locks present on a transport. (Olaf Conradi)
• `Repository.reconcile` now takes a thorough keyword parameter to allow requesting an indepth reconciliation, rather than just a data-loss check. (Robert Collins)
• `bzrlib.ui.ui_factory protocol` now supports `get_boolean` to prompt the user for yes/no style input. (Robert Collins)

42.3.5 Testing

• SFTP tests now shortcut the SSH negotiation, reducing test overhead for testing SFTP protocol support. (Robey Pointer)
• Branch formats are now tested once per implementation (see `bzrlib.tests.branch_implementations`). This is analagous to the transport interface tests, and has been followed up with working tree, repository and BzrDir tests. (Robert Collins)
• New test base class `TestCaseWithTransport` provides a transport aware test environment, useful for testing any transport-interface using code. The test suite option `–transport` controls the transport used by this class (when its not being used as part of implementation contract testing). (Robert Collins)
• Close logging handler on disabling the test log. This will remove the handler from the internal list inside python’s logging module, preventing shutdown from closing it twice. (Olaf Conradi)
• Move test case for uncommit to blackbox tests. (Olaf Conradi)
• `run_bzr` and `run_bzr_captured` now accept a `‘stdin=’foo”` parameter which will provide String(“foo”) to the command as its stdin.
BAZAAR RELEASE NOTES

43.1  bzr 0.7

Released 2006-01-09

43.1.1  Changes

- .bzrignore is excluded from exports, on the grounds that it’s a bzr internal-use file and may not be wanted. (Jamie Wilkinson)

- The “bzr directories” command were removed in favor of the new –kind option to the “bzr inventory” command. To list all versioned directories, now use “bzr inventory –kind directory”. (Johan Rydberg)

- Under Windows configuration directory is now %APPDATA%\bazaar\2.0 by default. (John Arbash Meinel)

- The parent of Bzr configuration directory can be set by BZR_HOME environment variable. Now the path for it is searched in BZR_HOME, then in HOME. Under Windows the order is: BZR_HOME, APPDATA (usually points to C:\Documents and Settings\User Name\Application Data), HOME. (John Arbash Meinel)

- Plugins with the same name in different directories in the bzr plugin path are no longer loaded: only the first successfully loaded one is used. (Robert Collins)

- Use system’s external SSH command to open connections if possible. This gives better integration with user settings such as ProxyCommand. (James Henstridge)

- Permissions on files underneath .bzr/ are inherited from the .bzr directory. So for a shared repository, simply doing ‘chmod -R g+w .bzr/’ will mean that future file will be created with group write permissions.

- configure.in and config.guess are no longer in the builtin default ignore list.

- ‘.sw[nop]’ pattern ignored, to ignore vim swap files for nameless files. (John Arbash Meinel, Martin Pool)

43.1.2  Improvements

- “bzr INIT dir” now initializes the specified directory, and creates it if it does not exist. (John Arbash Meinel)

- New remerge command (Aaron Bentley)

- Better zsh completion script. (Steve Borho)

- ‘bzr diff’ now returns 1 when there are changes in the working tree. (Robert Collins)

- ‘bzr push’ now exists and can push changes to a remote location. This uses the transport infrastructure, and can store the remote location in the ~/.bazaar/branches.conf configuration file. (Robert Collins)
• Test directories are only kept if the test fails and the user requests that they be kept.
• Tweaks to short log printing
• Added branch nicks, new nick command, printing them in log output. (Aaron Bentley)
• If $BZR_PDB$ is set, pop into the debugger when an uncaught exception occurs. (Martin Pool)
• Accept ‘bzr resolved’ (an alias for ‘bzr resolve’), as this is the same as Subversion. (Martin Pool)
• New FTP transport support (on ftplib), for ftp:// and aftp:// URLs. (Daniel Silverstone)
• Commit editor temporary files now start with bzar log, to allow text editors to match the file name and set up appropriate modes or settings. (Magnus Therning)
• Improved performance when integrating changes from a remote weave. (Goffredo Baroncelli)
• Sftp will attempt to cache the connection, so it is more likely that a connection will be reused, rather than requiring multiple password requests.
• bzr revno now takes an optional argument indicating the branch whose revno should be printed. (Michael Ellerman)
• bzr cat defaults to printing the last version of the file. (Matthieu Moy, #3632)
• New global option ‘bzr –lsprof COMMAND’ runs bzr under the lsprof profiler. (Denys Duchier)
• Faster commits by reading only the headers of affected weave files. (Denys Duchier)
• ‘bzr add’ now takes a –dry-run parameter which shows you what would be added, but doesn’t actually add anything. (Michael Ellerman)
• ‘bzr add’ now lists how many files were ignored per glob. add –verbose lists the specific files. (Aaron Bentley)
• ‘bzr missing’ now supports displaying changes in diverged trees and can be limited to show what either end of the comparison is missing. (Aaron Bently, with a little prompting from Daniel Silverstone)

43.1.3 Bug Fixes

• SFTP can walk up to the root path without index errors. (Robert Collins)
• Fix bugs in running bzr with ‘python -O’. (Martin Pool)
• Error when run with -OO
• Fix bug in reporting HTTP errors that don’t have an HTTP error code. (Martin Pool)
• Handle more cases of pipe errors in display commands
• Change status to 3 for all errors
• Files that are added and unlinked before committing are completely ignored by diff and status
• Stores with some compressed texts and some uncompressed texts are now able to be used. (John A Meinel)
• Fix for bzr pull failing sometimes under windows
• Fix for SFTP transport under windows when using interactive auth
• Show files which are both renamed and modified as such in ‘bzr status’ output. (Daniel Silverstone, #4503)
• Make annotate cope better with revisions committed without a valid email address. (Marien Zwart)
• Fix representation of tab characters in commit messages. (Harald Meland)
• List of plugin directories in BZR_PLUGIN_PATH environment variable is now parsed properly under Windows. (Alexander Belchenko)
• Show number of revisions pushed/pulled/merged. (Robey Pointer)
• Keep a cached copy of the basis inventory to speed up operations that need to refer to it. (Johan Rydberg, Martin Pool)
• Fix bugs in bzr status display of non-ascii characters. (Martin Pool)
• Remove Makefile.in from default ignore list. (Tollef Fog Heen, Martin Pool, #6413)
• Fix failure in ‘bzr added’. (Nathan McCallum, Martin Pool)

43.1.4 Testing

• Fix selftest asking for passwords when there are no SFTP keys. (Robey Pointer, Jelmer Vernooij)
• Fix selftest run with ‘python -O’. (Martin Pool)
• Fix HTTP tests under Windows. (John Arbash Meinel)
• Make tests work even if HOME is not set (Aaron Bentley)
• Updated build_tree to use fixed line-endings for tests which read the file contents and compare. Make some tests use this to pass under Windows. (John Arbash Meinel)
• Skip stat and symlink tests under Windows. (Alexander Belchenko)
• Delay in selftest/testhashcash is now issued under win32 and Cygwin. (John Arbash Meinel)
• Use terminal width to align verbose test output. (Martin Pool)
• Blackbox tests are maintained within the bzrlib.tests.blackbox directory. If adding a new test script please add that to bzrlib.tests.blackbox.__init__. (Robert Collins)
• Much better error message if one of the test suites can’t be imported. (Martin Pool)
• Make check now runs the test suite twice - once with the default locale, and once with all locales forced to C, to expose bugs. This is not trivially done within python, so for now its only triggered by running Make check. Integrators and packagers who wish to check for full platform support should run ‘make check’ to test the source. (Robert Collins)
• Tests can now run TestSkipped if they can’t execute for any reason. (Martin Pool) (NB: TestSkipped should only be raised for correctable reasons - see the wiki spec ImprovingBzrTestSuite).
• Test SFTP with relative, absolute-in-homedir and absolute-not-in-homedir paths for the transport tests. Introduce blackbox remote SFTP tests that test the same permutations. (Robert Collins, Robey Pointer)
• Transport implementation tests are now independent of the local file system, which allows tests for esoteric transports, and for features not available in the local file system. They also repeat for variations on the URL scheme that can introduce issues in the transport code, see bzrlib.transport.TransportTestProviderAdapter() for this. (Robert Collins).
• TestCase.build_tree uses the transport interface to build trees, pass in a transport parameter to give it an existing connection. (Robert Collins).

43.1.5 Internals

• WorkingTree.pull has been split across Branch and WorkingTree, to allow Branch only pulls. (Robert Collins)
• commands.display_command now returns the result of the decorated function. (Robert Collins)
• LocationConfig now has a set_user_option(key, value) call to save a setting in its matching location section (a new one is created if needed). (Robert Collins)
• Branch has two new methods, getPushLocation and setPushLocation to respectively, get and set the push location. (Robert Collins)

• commands.register_command now takes an optional flag to signal that the registrant is planning to decorate an existing command. When given multiple plugins registering a command is not an error, and the original command class (whether built in or a plugin based one) is returned to the caller. There is a new error ‘MustUseDecorated’ for signalling when a wrapping command should switch to the original version. (Robert Collins)

• Some option parsing errors will raise ‘BzrOptionError’, allowing granular detection for decorating commands. (Robert Collins).

• Branch.readWorkingInventory has moved to WorkingTree.readWorkingInventory. This necessitated changes to Branch.getRootId, and a move of Branch.setInventory to WorkingTree as well. To make it clear that a WorkingTree cannot always be obtained Branch.working_tree() will raise errors.NoWorkingTree if one cannot be obtained. (Robert Collins)

• All pending merges operations from Branch are now on WorkingTree. (Robert Collins)

• The follow operations from Branch have moved to WorkingTree:

  add()
  commit()
  move()
  rename_one()
  unknowns()

  (Robert Collins)

• bzrlib.add.smart_add_branch is now smart_add_tree. (Robert Collins)

• New “rio” serialization format, similar to rfc-822. (Martin Pool)

• Rename selftests to bzrlib.tests.test_foo. (John A Meinel, Martin Pool)

• bzrlib.plugin.all_plugins has been changed from an attribute to a query method. (Robert Collins)

• New options to read only the table-of-contents of a weave. (Denys Duchier)

• Raise NoSuchFile when someone tries to add a non-existent file. (Michael Ellerman)

• Simplify handling of DivergedBranches in cmd_pull(). (Michael Ellerman)

• Branch.controlfile* logic has moved to lockablefiles.LockableFiles, which is exposed as Branch().control_files. Also this has been altered with the controlfile pre/suffix replaced by simple method names like ‘get’ and ‘put’. (Aaron Bentley, Robert Collins).

• Deprecated functions and methods can now be marked as such using the bzrlib.symbol_versioning module. Marked method have their docstring updated and will issue a DeprecationWarning using the warnings module when they are used. (Robert Collins)

• bzrlib.osutils.safe_unicode now exists to provide parameter coercion for functions that need unicode strings. (Robert Collins)
BAZAAR RELEASE NOTES

44.1  bzr 0.6

Released 2005-10-28

44.1.1  Improvements

- pull now takes --verbose to show you what revisions are added or removed (John A Meinel)
- merge now takes a --show-base option to include the base text in conflicts. (Aaron Bentley)
- The config files are now read using ConfigObj, so '=' should be used as a separator, not ':'. (Aaron Bentley)
- New 'bzr commit --strict' option refuses to commit if there are any unknown files in the tree. To commit, make sure all files are either ignored, added, or deleted. (Michael Ellerman)
- The config directory is now ~/.bazaar, and there is a single file ~/.bazaar/bazaar.conf storing email, editor and other preferences. (Robert Collins)
- 'bzr add' no longer takes a --verbose option, and a --quiet option has been added that suppresses all output.
- Improved zsh completion support in contrib/zsh, from Clint Adams.
- Builtin 'bzr annotate' command, by Martin Pool with improvements from Goffredo Baroncelli.
- 'bzr check' now accepts -v for verbose reporting, and checks for ghosts in the branch. (Robert Collins)
- New command 're-sign' which will regenerate the gpg signature for a revision. (Robert Collins)
- If you set check_signatures=require for a path in ~/.bazaar/branches.conf then bzr will invoke your gpg_signing_command (defaults to gpg) and record a digital signature of your commit. (Robert Collins)
- New SFTP transport, based on Paramiko. (Robey Pointer)
- 'bzr pull' now accepts --clobber which will discard local changes and make this branch identical to the source branch. (Robert Collins)
- Just give a quieter warning if a plugin can’t be loaded, and put the details in .bzr.log. (Martin Pool)
- 'bzr branch' will now set the branch-name to the last component of the output directory, if one was supplied.
- If the option post_commit is set to one (or more) python function names (must be in the bzrlib namespace), then they will be invoked after the commit has completed, with the branch and revision_id as parameters. (Robert Collins)
- Merge now has a retcode of 1 when conflicts occur. (Robert Collins)
• –merge-type weave is now supported for file contents. Tree-shape changes are still three-way based. (Martin Pool, Aaron Bentley)

• ‘bzr check’ allows the first revision on revision-history to have parents - something that is expected for cheap checkouts, and occurs when conversions from baz do not have all history. (Robert Collins).

• ‘bzr merge’ can now graft unrelated trees together, if you specify 0 as a base. (Aaron Bentley)

• ‘bzr commit branch’ and ‘bzr commit branch/file1 branch/file2’ now work (Aaron Bentley)

• Add ‘.sconsign***’ to default ignore list. (Alexander Belchenko)

• ‘bzr merge –reprocess’ minimizes conflicts

### 44.1.2 Testing

• The ‘bzr selftest –pattern’ option for has been removed, now test specifiers on the command line can be simple strings, or regexps, or both. (Robert Collins)

• Passing -v to selftest will now show the time each test took to complete, which will aid in analysing performance regressions and related questions. (Robert Collins)

• ‘bzr selftest’ runs all tests, even if one fails, unless ‘–one’ is given. (Martin Pool)

• There is a new method for TestCaseInTempDir, assertFileEqual, which will check that a given content is equal to the content of the named file. (Robert Collins)

• Fix test suite’s habit of leaving many temporary log files in $TMPDIR. (Martin Pool)

### 44.1.3 Internals

• New ‘testament’ command and concept for making gpg-signatures of revisions that are not tied to a particular internal representation. (Martin Pool).

• Per-revision properties (‘revprops’) as key-value associated strings on each revision created when the revision is committed. Intended mainly for the use of external tools. (Martin Pool).

• Config options have moved from bzrlib.osutils to bzrlib.config. (Robert Collins)

• Improved command line option definitions allowing explanations for individual options, among other things. Contributed by Magnus Therning.

• Config options have moved from bzrlib.osutils to bzrlib.config. Configuration is now done via the config.Config interface: Depending on whether you have a Branch, a Location or no information available, construct a *Config, and use its signature_checking, username and user_email methods. (Robert Collins)

• Plugins are now loaded under bzrlib.plugins, not bzrlib.plugin, and they are made available for other plugins to use. You should not import other plugins during the __init__ of your plugin though, as no ordering is guaranteed, and the plugins directory is not on the python path. (Robert Collins)

• Branch.relpath has been moved to WorkingTree.relpath. WorkingTree no no longer takes an inventory, rather it takes an option branch parameter, and if None is given will open the branch at basedir implicitly. (Robert Collins)

• Cleaner exception structure and error reporting. Suggested by Scott James Remnant. (Martin Pool)

• Branch.remove has been moved to WorkingTree, which has also gained lock_read, lock_write and unlock methods for convenience. (Robert Collins)

• Two decorators, needs_read_lock and needs_write_lock have been added to the branch module. Use these to cause a function to run in a read or write lock respectively. (Robert Collins)
• `Branch.open_containing` now returns a tuple `(Branch, relative-path)`, which allows direct access to the common case of ‘get me this file from its branch’. (Robert Collins)

• Transports can register using `register_lazy_transport`, and they will be loaded when first used. (Martin Pool)

• ‘pull’ has been factored out of the command as `WorkingTree.pull()`. A new option to `WorkingTree.pull` has been added, clobber, which will ignore diverged history and pull anyway. (Robert Collins)

• `config.Config` has a `get_user_option` call that accepts an option name. This will be looked up in `branches.conf` and `bazaar.conf` as normal. It is intended that this be used by plugins to support options - options of built in programs should have specific methods on the config. (Robert Collins)

• `merge.merge_inner` now has tempdir as an optional parameter. (Robert Collins)

• `Tree.kind` is not recorded at the top level of the hierarchy, as it was missing on EmptyTree, leading to a bug with merge on EmptyTrees. (Robert Collins)

• `WorkingTree.__del__` has been removed, it was non deterministic and not doing what it was intended to. See `WorkingTree.__init__` for a comment about future directions. (Robert Collins/Martin Pool)

• `bzrlib.transport.http` has been modified so that only 404 urllib errors are returned as NoSuchFile. Other exceptions will propagate as normal. This allows debugging of actual errors. (Robert Collins)

• `bzrlib.transport.Transport` now accepts `ONLY URL-escaped relative paths to apis like ‘put’, ‘get’ and ‘has’`. This is to provide consistent behaviour - it operates on URLs only. (Robert Collins)

• Transports can register using `register_lazy_transport`, and they will be loaded when first used. (Martin Pool)

• `merge_flex` no longer calls `conflict_handler.finalize()`, instead that is called by `merge_inner`. This is so that the conflict count can be retrieved (and potentially manipulated) before returning to the caller of `merge_inner`. Likewise ‘merge’ now returns the conflict count to the caller. (Robert Collins)

• `revision.revision_graph` can handle having only partial history for a revision - that is no revisions in the graph with no parents. (Robert Collins).

• New `builtins.branch_files` uses the standard `file_list` rules to produce a branch and a list of paths, relative to that branch (Aaron Bentley)

• New `TestCase.addCleanup` facility.

• New `bzrlib.version_info` tuple (similar to `sys.version_info`), which can be used by programs importing `bzrlib`.

### 44.1.4 Bug Fixes

• Better handling of branches in directories with non-ascii names. (Joel Rosdahl, Panagiotis Papadakos)

• Upgrades of trees with no commits will not fail due to accessing [-1] in the revision-history. (Andres Salomon)
baazaar release notes

45.1 bzr 0.1.1

released 2005-10-12

45.1.1 bug fixes

- fix problem in pulling over http from machines that do not allow directories to be listed.
- avoid harmless warning about invalid hash cache after upgrading branch format.

45.1.2 performance

- avoid some unnecessary http operations in branch and pull.

45.2 bzr 0.1

released 2005-10-11

45.2.1 notes

- ‘bzr branch’ over http initially gives a very high estimate of completion time but it should fall as the first few revisions are pulled in. branch is still slow on high-latency connections.

45.2.2 bug fixes

- bzr-man.py has been updated to work again. contributed by rob weir.
- locking is now done with fcntl.lockf which works with nfs file systems. contributed by harald meland.
- when a merge encounters a file that has been deleted on one side and modified on the other, the old contents are written out to foo.base and foo.side, where side is this or other. contributed by aaron bentley.
- export was choosing incorrect file paths for the content of the tarball, this has been fixed by aaron bentley.
- commit will no longer commit without a log message, an error is returned instead. contributed by jelmer vernooij.
• If you commit a specific file in a sub directory, any of its parent directories that are added but not listed will be automatically included. Suggested by Michael Ellerman.

• bzr commit and upgrade did not correctly record new revisions for files with only a change to their executable status. bzr will correct this when it encounters it. Fixed by Robert Collins.

• HTTP tests now force off the use of http_proxy for the duration. Contributed by Gustavo Niemeyer.

• Fix problems in merging weave-based branches that have different partial views of history.

• Symlink support: working with symlinks when not in the root of a bzr tree was broken, patch from Scott James Remnant.

45.2.3 Improvements

• ‘branch’ now accepts a –basis parameter which will take advantage of local history when making a new branch. This allows faster branching of remote branches. Contributed by Aaron Bentley.

• New tree format based on weave files, called version 5. Existing branches can be upgraded to this format using ‘bzr upgrade’.

• Symlinks are now versionable. Initial patch by Erik Toubro Nielsen, updated to head by Robert Collins.

• Executable bits are tracked on files. Patch from Gustavo Niemeyer.

• ‘bzr status’ now shows unknown files inside a selected directory. Patch from Heikki Paajanen.

• Merge conflicts are recorded in .bzr. Two new commands ‘conflicts’ and ‘resolve’ have needed added, which list and remove those merge conflicts respectively. A conflicted tree cannot be committed in. Contributed by Aaron Bentley.

• ‘rm’ is now an alias for ‘remove’.

• Stores now split out their content in a single byte prefixed hash, dropping the density of files per directory by 256. Contributed by Gustavo Niemeyer.

• ‘bzr diff -r branch:URL’ will now perform a diff between two branches. Contributed by Robert Collins.

• ‘bzr log’ with the default formatter will show merged revisions, indented to the right. Initial implementation contributed by Gustavo Niemeyer, made incremental by Robert Collins.

45.2.4 Internals

• Test case failures have the exception printed after the log for your viewing pleasure.

• InventoryEntry is now an abstract base class, use one of the concrete InventoryDirectory etc classes instead.

• Branch raises an UnsupportedFormatError when it detects a bzr branch it cannot understand. This allows for precise handling of such circumstances.

• Remove RevisionReference class; Revision.parent_ids is now simply a list of their ids and parent_sha1s is a list of their corresponding sha1s (for old branches only at the moment.)

• New method-object style interface for Commit() and Fetch().

• Renamed Branch.last_patch() to Branch.last_revision(), since we call them revisions not patches.

• Move copy_branch to bzrlib.clone.copy_branch. The destination directory is created if it doesn’t exist.

• Inventories now identify the files which were present by giving the revision of that file.
• Inventory and Revision XML contains a version identifier. This must be consistent with the overall branch version but allows for more flexibility in future upgrades.

### 45.2.5 Testing

- Removed testsweet module so that tests can be run after bzr installed by ‘bzr selftest’.
- ‘bzr selftest’ command-line arguments can now be partial ids of tests to run, e.g. `bzr selftest test_weave`

### 45.3 bzr 0.0.9

**Released** 2005-09-23

#### 45.3.1 Bug Fixes

- Fixed “branch -r” option.
- Fix remote access to branches containing non-compressed history. (Robert Collins).
- Better reliability of HTTP server tests. (John Arbash-Meinel)
- Merge graph maximum distance calculation fix. (Aaron Bentley)
- Various minor bug in windows support have been fixed, largely in the test suite. Contributed by Alexander Belchenko.

#### 45.3.2 Improvements

- Status now accepts a -r argument to give status between chosen revisions. Contributed by Heikki Paajanen.
- Revision arguments no longer use +/-= to control ranges, instead there is a ‘before’ namespace, which limits the successive namespace. For example ‘$ bzr log -r date:yesterday..before:date:today’ will select everything from yesterday and before today. Contributed by Robey Pointer
- There is now a bzr.bat file created by distutils when building on Windows. Contributed by Alexander Belchenko.

#### 45.3.3 Internals

- Removed uuid() as it was unused.
- Improved ‘fetch’ code for pulling revisions from one branch into another (used by pull, merged, etc.)

### 45.4 bzr 0.0.8

**Released** 2005-09-20
45.4.1 Improvements

- Adding a file whose parent directory is not versioned will implicitly add the parent, and so on up to the root. This means you should never need to explicitly add a directory, they’ll just get added when you add a file in the directory. Contributed by Michael Ellerman.
- Ignore `.DS_Store` (contains Mac metadata) by default. (Nir Soffer)
- If you set `BZR_EDITOR` in the environment, it is checked in preference to `EDITOR` and the config file for the interactive commit editing program. Related to this is a bugfix where a missing program set in `EDITOR` would cause editing to fail, now the fallback program for the operating system is still tried.
- Files that are not directories/symlinks/regular files will no longer cause `bzr` to fail, it will just ignore them by default. You cannot add them to the tree though - they are not versionable.

45.4.2 Internals

- Refactor XML packing/unpacking.

45.4.3 Bug Fixes

- Fixed `bzr mv` by Ollie Rutherfurd.
- Fixed strange error when trying to access a nonexistent HTTP branch.
- Make sure that the hashcache gets written out if it can’t be read.

45.4.4 Portability

- Various Windows fixes from Ollie Rutherfurd.
- Quieten warnings about locking; patch from Matt Lavin.

45.5 bzr-0.0.7

Released 2005-09-02

45.5.1 New Features

- `bzr shell-complete` command contributed by Clint Adams to help with intelligent shell completion.
- New expert command `bzr find-merge-base` for debugging merges.

45.5.2 Enhancements

- Much better merge support.
- `merge3` conflicts are now reported with markers like ‘<<<<<<<’ (seven characters) which is the same as CVS and pleases things like `emacs` smerge.
45.5.3 Bug Fixes

- `bzr upgrade` no longer fails when trying to fix trees that mention revisions that are not present.
- Fixed bugs in listing plugins from `bzr plugins`.
- Fix case of `$EDITOR` containing options for the editor.
- Fix `log -r` refusing to show the last revision. (Patch from Goffredo Baroncelli.)

45.5.4 Changes

- `bzr log --show-ids` shows the revision ids of all parents.
- Externally provided commands on your `$BZRPATH` no longer need to recognize `--bzr-usage` to work properly, and can just handle `--help` themselves.

45.5.5 Library

- Changed trace messages to go through the standard logging framework, so that they can more easily be redirected by libraries.

45.6 bzr-0.0.6

Released 2005-08-18

45.6.1 New Features

- Python plugins, automatically loaded from the directories on `BZR_PLUGIN_PATH` or `~/.bzr.conf/plugins` by default.
- New `bzr mkdir` command.
- Commit message is fetched from an editor if not given on the command line; patch from Torsten Marek.
- `bzr log -m FOO` displays commits whose message matches regexp FOO.
- `bzr add` with no arguments adds everything under the current directory.
- `bzr mv` does move or rename depending on its arguments, like the Unix command.
- `bzr missing` command shows a summary of the differences between two trees. (Merged from John Arbash-Meinel.)
- An email address for commits to a particular tree can be specified by putting it into `.bzr/email` within a branch. (Based on a patch from Heikki Paajanen.)

45.6.2 Enhancements

- Faster working tree operations.
45.6.3 Changes

- 3rd-party modules shipped with bzr are copied within the bzrlib python package, so that they can be installed by the setup script without clashing with anything already existing on the system. (Contributed by Gustavo Niemeyer.)

- Moved plugins directory to bzrlib/, so that there’s a standard plugin directory which is not only installed with bzr itself but is also available when using bzr from the development tree. BZR_PLUGIN_PATH and DEFAULT_PLUGIN_PATH are then added to the standard plugins directory.

- When exporting to a tarball with `bzr export --format tgz`, put everything under a top directory rather than dumping it into the current directory. This can be overridden with the `--root` option. Patch from William Dodé and John Meinel.

- New `bzr upgrade` command to upgrade the format of a branch, replacing `bzr check --update`.

- Files within store directories are no longer marked readonly on disk.

- Changed `bzr log` output to a more compact form suggested by John A Meinel. Old format is available with the `--long` or `-l` option, patched by William Dodé.

- By default the commit command refuses to record a revision with no changes unless the `--unchanged` option is given.

- The `--no-plugins`, `--profile` and `--builtin` command line options must come before the command name because they affect what commands are available; all other options must come after the command name because their interpretation depends on it.

- `branch` and `clone` added as aliases for `branch`.

- Default log format is back to the long format; the compact one is available with `--short`.

45.6.4 Bug Fixes

- Fix bugs in committing only selected files or within a subdirectory.

45.7 bzr-0.0.5

Released 2005-06-15

45.7.1 Changes


- `bzr status` output format changed, because svn-style output doesn’t really match the model of bzr. Now files are grouped by status and can be shown with their IDs. `bzr status --all` shows all versioned files and unknown files but not ignored files.

- `bzr log` runs from most-recent to least-recent, the reverse of the previous order. The previous behaviour can be obtained with the `--forward` option.

- `bzr inventory` by default shows only filenames, and also ids if `--show-ids` is given, in which case the id is the second field.
45.7.2 Enhancements

- New ‘bzr whoami -email’ option shows only the email component of the user identification, from Jo Vermeulen.
- New `bzr ignore PATTERN` command.
- Nicer error message for broken pipe, interrupt and similar conditions that don’t indicate an internal error.
- Add `.*.sw[nop] .git .*tmp`,v to default ignore patterns.
- Per-branch locks keyed on `.bzr/branch-lock`, available in either read or write mode.
- New option `bzr log --show-ids` shows revision and file ids.
- New usage `bzr log FILENAME` shows only revisions that affected that file.
- Changed format for describing changes in `bzr log -v`.
- New option `bzr commit --file` to take a message from a file, suggested by LarstiQ.
- New syntax `bzr status [FILE...]` contributed by Bartosz Oler. File may be in a branch other than the working directory.
- `bzr log` and `bzr root` can be given an HTTP URL instead of a filename.
- Commands can now be defined by external programs or scripts in a directory on `$BZRPATH`.
- New “stat cache” avoids reading the contents of files if they haven’t changed since the previous time.
- If the Python interpreter is too old, try to find a better one or give an error. Based on a patch from Fredrik Lundh.
- New optional parameter `bzr info [BRANCH]`.
- New form `bzr commit SELECTED` to commit only selected files.
- New form `bzr log -r FROM:TO` shows changes in selected range; contributed by John A Meinel.
- New option `bzr diff --diff-options ’OPTS’` allows passing options through to an external GNU diff.
- New option `bzr add --no-recurse` to add a directory but not their contents.
- `bzr --version` now shows more information if bzr is being run from a branch.

45.7.3 Bug Fixes

- Fixed diff format so that added and removed files will be handled properly by patch. Fix from Lalo Martins.
- Various fixes for files whose names contain spaces or other metacharacters.

45.7.4 Testing

- Converted black-box test suites from Bourne shell into Python; now run using `./testbzr`. Various structural improvements to the tests.
- `testbzr` by default runs the version of bzr found in the same directory as the tests, or the one given as the first parameter.
- `testbzr` also runs the internal tests, so the only command required to check is just `./testbzr`.
- `testbzr` requires python2.4, but can be used to test bzr running under a different version.
- Tests added for many other changes in this release.
45.7.5 Internal

- Included ElementTree library upgraded to 1.2.6 by Fredrik Lundh.
- Refactor command functions into Command objects based on HCT by Scott James Remnant.
- Better help messages for many commands.
- Expose `bzrlib.open_tracefile()` to start the tracefile; until this is called trace messages are just discarded.
- New internal function `find_touching_revisions()` and hidden command touching-revisions trace the changes to a given file.
- Simpler and faster `compare_inventories()` function.
- `bzrlib.open_tracefile()` takes a tracefilename parameter.
- New AtomicFile class.
- New developer commands added, modified.

45.7.6 Portability

- Cope on Windows on python2.3 by using the weaker random seed. 2.4 is now only recommended.

45.8 bzr-0.0.4

Released 2005-04-22

45.8.1 Enhancements

- ‘bzr diff’ optionally takes a list of files to diff. Still a bit basic. Patch from QuantumG.
- More default ignore patterns.
- New ‘bzr log --verbose’ shows a list of files changed in the changeset. Patch from Sebastian Cote.
- Roll over ~/.bzr.log if it gets too large.
- New ‘bzr help commands’ based on a patch from Denys Duchier.

45.8.2 Changes

- User email is determined by looking at $BZREMAIL or ~/.bzr.email or $EMAIL. All are decoded by the locale preferred encoding. If none of these are present user@hostname is used. The host’s fully-qualified name is not used because that tends to fail when there are DNS problems.
- New ‘bzr whoami’ command instead of username user-email.
45.8.3 Bug Fixes

- Make commit safe for hardlinked bzr trees.
- Some Unicode/locale fixes.
- Partial workaround for difflib.unified_diff not handling trailing newlines properly.

45.8.4 Internal

- Allow docstrings for help to be in PEP0257 format. Patch from Matt Brubeck.
- More tests in test.sh.
- Write profile data to a temporary file not into working directory and delete it when done.
- Smaller .bzr.log with process ids.

45.8.5 Portability

- Some improvements in handling paths on Windows, based on a patch from QuantumG.

45.9 bzr-0.0.3

Released 2005-04-06

45.9.1 Enhancements

- New “directories” internal command lists versioned directories in the tree.
- Can now say “bzr commit –help”.
- New “rename” command to rename one file to a different name and/or directory.
- New “move” command to move one or more files into a different directory.
- New “renames” command lists files renamed since base revision.
- New cat command contributed by janmar.

45.9.2 Changes

- .bzr.log is placed in $HOME (not pwd) and is always written in UTF-8. (Probably not a completely good long-term solution, but will do for now.)

45.9.3 Portability

- Workaround for difflib bug in Python 2.3 that causes an exception when comparing empty files. Reported by Erik Toubro Nielsen.
45.9.4 Internal

- Refactored inventory storage to insert a root entry at the top.

45.9.5 Testing

- Start of shell-based black-box testing in test.sh.

45.10 bzr-0.0.2.1

45.10.1 Portability

- Win32 fixes from Steve Brown.

45.11 bzr-0.0.2

Codename “black cube”

Released 2005-03-31

45.11.1 Enhancements

- Default ignore list extended (see bzrlib/__init__.py).
- Patterns in .bzrignore are now added to the default ignore list, rather than replacing it.
- Ignore list isn’t reread for every file.
- More help topics.
- Reinstate the ‘bzr check’ command to check invariants of the branch.
- New ‘ignored’ command lists which files are ignored and why; ‘deleted’ lists files deleted in the current working tree.
- Performance improvements.
- New global –profile option.
- Ignore patterns like ‘./config.h’ now correctly match files in the root directory only.

45.12 bzr-0.0.1

Released 2005-03-26
45.12.1 Enhancements

- More information from info command.
- Can now say “bzr help COMMAND” for more detailed help.
- Less file flushing and faster performance when writing logs and committing to stores.
- More useful verbose output from some commands.

45.12.2 Bug Fixes

- Fix inverted display of ‘R’ and ‘M’ during ‘commit -v’.

45.12.3 Portability

- Include a subset of ElementTree-1.2.20040618 to make installation easier.
- Fix time.localtime call to work with Python 2.3 (the minimum supported).

45.13 bzr-0.0.0.69

Released 2005-03-22

45.13.1 Enhancements

- First public release.
- Storage of local versions: init, add, remove, rm, info, log, diff, status, etc.