NAME

aegis – project change supervisor
Copyright © 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012 Peter Miller

Aegis is distributed under the terms of the GNU General Public License. See the LICENSE section, below, for more details.

aegis (ee.j.iz) *n.,* a protection, a defense.

DESCRIPTION

Aegis is a CASE tool with a difference. In the spirit of the UNIX Operating System, *Aegis* is a small component designed to work with other programs.

Many CASE systems attempt to provide everything, from bubble charts to source control to compilers. Users are trapped with the components supplied by the CASE system, and if you don't like one of the components (it may be too limited, for instance), then that is just tough.

In contrast, UNIX provides many components of a CASE system – compilers, editors, dependency maintenance tools (such as make), source control tools (such as RCS). You may substitute the tool of your choice if you don't like the ones supplied with the system – gcc, jove, cake, to name just a few. *Aegis* adds to this list with software configuration management, and true to UNIX philosophy, *Aegis* does not dictate the choice of any of the other tools (although it may stretch them to their limits).

Enough hype, what is it that *Aegis* does? Just what is software configuration management? This question is sufficiently broad as to require a book in answer. In essence, *Aegis* is a project change supervisor. It provides a framework within which a team of developers may work on many changes to a program independently, and *Aegis* coordinates integrating these changes back into the master source of the program, with as little disruption as possible. Resolution of contention for source files, a major headache for any project with more than one developer, is one of *Aegis'* major functions.

It should be noted that *Aegis* is a developer's tool, in the same sense as make or RCS are developer's tools. It is not a manager's tool – it does not provide progress tracking or manage work allocation.

BENEFITS

So why should you use *Aegis*?

Aegis uses a particular model of the development of software projects. This model has a master source (or baseline) of a project, and a team of developers creating changes to be made to this baseline. When a change is complete, it is integrated with the baseline, to become the new baseline. Each change must be atomic and self-contained, no change is allowed to cause the baseline to cease to work. "Working" is defined as passing it's own tests. The tests are considered part of the baseline. *Aegis* provides support for the developer so that an entire copy of the baseline need not be taken to change a few files, only those files which are to be changed need to be copied.

In order to ensure that changes are unable to cause the baseline to cease to work, *Aegis* mandates that changes be accompanied by at least one test, and that all such tests be known to complete successfully. These steadily accumulated tests form an ever increasing regression test suite for all later changes. There is also a mandatory review stage for each change to the baseline. While these requirements may be relaxed per-change or even per-project, doing so potentially compromises the "working" definition of the baseline.

The win in using *Aegis* is that there are $O(n)$ interactions between developers and the baseline. Contrast this with a master source which is being edited directly by the developers – there are $O(n!)$ interactions between developers – this makes adding "just one more" developer a potential disaster.

Another win is that the project baseline always works. Always having a working baseline means that a version is always available for demonstrations, or those "pre-release snapshots" we are always forced to provide.

The above advantages are all very well – for management types. Why should Joe Average Programmer use *Aegis*? Recall that RCS provides file locking, but only for one file at a time. *Aegis* provides the file locking, atomically, for the set of files in the change. Recall also that RCS locks the file the instant you start

editing it. This makes popular files a project bottleneck. *Aegis* allows concurrent editing, and a resolution mechanism just before the change must be integrated, meaning fewer delays for J.A.Programmer.

Aegis also has strong support for geographically distributed development. It supports both push and pull models, and many distribution topologies. Aegis' normal development process is used to validate received change sets before committing them.

ARCHIVE SITE

The latest version of *Aegis* is available by HTTP from:

```
URL:     http://miller.emu.id.au/pmiller/
File:    aegis.html              # the Aegis page
File:    aegis.4.25.README       # Description, from tar file
File:    aegis.4.25.lsm          # Description, in LSM format
File:    aegis.4.25.ae           # the complete source, aedist format
File:    aegis.4.25.spec         # RedHat package specification
File:    aegis.4.25.tar.gz       # the complete source
```

This directory also contains a few other pieces of software written by me. Some are referred to in the *Aegis* documentation. Please have a look if you are interested.

Mirrors

See http://miller.emu.id.au/pmiller/ for a list of mirror sites.

Aegis is also carried by `metalab.unc.edu` in its Linux archives. You will be able to find Aegis on any of its mirrors.

```
URL:     ftp://metalab.unc.edu/pub/Linux/devel/vc/
File:    aegis.4.25.README       # Description, from tar file
File:    aegis.4.25.lsm          # Description, in LSM format
File:    aegis.4.25.spec         # RedHat package specification
File:    aegis.4.25.ae           # the complete source, aedist format
File:    aegis.4.25.tar.gz       # the complete source
```

This site is extensively mirrored around the world, so look for a copy near you (you will get much better response).

MAILING LIST

A mailing list has been created so that users of *Aegis* may exchange ideas about how to use *Aegis*. Discussion may include, but is not limited to: bugs, enhancements, and applications. The list is not moderated.

The address of the mailing list is

```
aegis-users@auug.org.au
```

Please **do not** attempt to subscribe by sending email to this address. It is for content only.

How To Subscribe

To subscribe to this mailing list, visit the Aegis-users mailing list page (`http://www.auug.org.au/-mailman/listinfo/aegis-users`) and go through the **subscribe** dialogue.

Archive

The mailing list is archived at eGroups. The URL is `http://www.egroups.com/list/aegis-users/info.html`

No Files By EMail

The software which handles this mailing list **cannot** send you a copy of *Aegis*. Please use FTP or ftp-by-email, instead.

BUILDING

Instructions on how to build and test *Aegis* are to be found in the *BUILDING* file included in this distribution.

SOME HISTORY

The idea for *Aegis* did not come full-blown into my head in the shower, as some of my programs do, but rather from working in a software shop which used a simplistic form of something similar. That system was held together by chewing-gum and string, it was written in a disgusting variant of Basic, and by golly the damn thing worked (mostly). *Aegis* is nothing like it, owes none of its code to that system, and is far more versatile. It turns out that the system used is nothing new, and is described in many SCM textbooks; it is the result of systematically resolving development issues for large-ish teams.

Since that company decided to close down our section (the company was under attack by a hostile takeover bid) we all moved on simultaneously (all 60 of us), sometimes working together, and sometimes not, but always keeping in touch. With suggestions and conversations with some of them early in 1990, the manual entries for *Aegis* took shape, and formed most of the design document for *Aegis*.

Since getting the first glimmerings of a functional *Aegis* late in 1990 it is increasingly obvious that I never want to be without it ever again. All of my sources that I modify are instantly placed under *Aegis*, as is anything I distribute. All code I write for myself, and all new code I write for my employer, goes under *Aegis*. Why? Because it has fewer bugs!

Example: one of the sources I carry with me from job to job is "cook", my dependency maintenance tool. Cook had existed for 3 years before *Aegis* appeared on the scene, and I used it daily. When I placed cook under *Aegis*, I found 6 bugs! Since then I have found a few more. Not only are there now fewer bugs, but they never come back, because the regression test suite always grows.

Branching

In 1997 the full branching support was released (it took nearly 18 months to retro-fit. The underlying data structures for projects and change sets need to be merged. While I noticed back in 1990 that they were very similar, it wasn't until branch support design was well underways that they should have been the same data structure from the beginning.

Geographically Distributed Development

In 1999 a conversation on the `aegis-users` mailing list resulted in the creation of *aedist*, a program which packages and unpackages Aegis changes so they can be sent by e-mail, or WWW or whatever. With 20:20 hindsight, this could have been done way back in 1991, because the basic idea builds on Aegis change process model.

Windows NT

Aegis depends on the underlying security provided by the operating system (rather than re-invent yet another security mechanism). However, in order to do this, Aegis uses the POSIX *seteuid* system call, which has no direct equivalent on Windows NT. This makes porting difficult. Single-user ports are possible (*e.g.* using Cygwin), but are not usually what folks want.

Compounding this is the fact that many sites want to develop their software for both Unix and Windows NT simultaneously. This means that the security of the repository needs to be guaranteed to be handled in the same way by both operating systems, otherwise one can act as a "back door" into the repository. Many sites do not have the same users and permissions (sourced from the same network register of users) on both Unix and Windows NT, making the mapping almost impossible even if the security models did actually correspond.

Most sites using Aegis and Windows NT together do so by running Aegis on the Unix systems, but building and testing on the NT systems. The work areas and repository are accessed via Samba or NFS.

LICENSE

Aegis is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version. In addition, as a special exception, the copyright holders give permission to link the code of this program with the OpenSSL library, and distribute linked combinations including the two.

Aegis is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

It should be in the *LICENSE* file included in this distribution. The full test of the OpenSSL exception should be in the *LICENSE.openssl* file included in this distribution.

AUTHOR

Peter Miller   E-Mail: pmiller@opensource.org.au
/\/*   WWW: http://miller.emu.id.au/pmiller/

RELEASE NOTES

For excruciating detail, and also acknowledgments of those who generously sent me feedback, please see the *etc/CHANGES.** files included in this distribution.

Upgrading

In general, all the machines on your network need to be running the same release of Aegis. While the database format is backwards compatible, it is rarely forwards compatible in the face of new capabilities.

Version 4.26 (N-MMM-NNNN)

• Added OpenSSL license exception (LICENSE.openssl).

Version 4.25 (8-Mar-2008)
Version 4.24 (09-Mar-2008)

• Numerous portability improvements.

• Numerous improvements to the history reconstruction code.

• Numerous improvements and bug fixes to the distributed development code. See *aedist*(1), *aeget*(1), *aetar*(1) and *aepatch*(1) for more information.

• The site specific architecture information has been split into a separate file, maked with an entire-source-hide attribute, so that new Aegis-under-Aegis projects don't have such bad architecture problems.

• The license has been changed to GPLv3.

• A bug has been fixed the the change::pconf_get method which sometimes caused segfaults.

• A bug has been fixed in aeclone which caused aecp -delta to segfault when: a change set whith a removed file was cloned, and the clone change set subsequently integrated.

• The aediff command is now smarter about files which may have moved.

• A bug has been fixed in the aeannotate command, it no longer segfaults for some file histories.

• A Vietnamese error mesage translation has been added.

• There is a new Portugese (Brazillian) message catalogue translation.

• It is now possible to develop begin undo and new change undo in a single command.

• The *ael*(1) command now understands are much wider range of ways to specify changes.

• A segfault has been fixed in the use of --delta and --delta-data options.

• This change set fix a problem in the *aesvt*(1) checkout command that can fail to extract from a gzip compressed archive.

• A bug has been fixed in the *aechown*(1) command, to stop a segfault when printing some error messages.

• There is a new *aebisect*(1) command which helps to find project regressions not handled by the test suite.

• Many commands now cope with renames in more situations.

• A bug has been fixed in *aeclone*(1) which caused *aecp −delta* to segfault when a change set whith a removed file was cloned, and the clone change set subsequently integrated.

• The *aede-policy*(1) manual page has been updated to document the aede-policy-line-length file attribute.

• A vietnamese translation has been added.

• The *aereport*(1) command now understands more ways to specify changes.

• There is a new ${path_reduce} substitution which may be used to remove redundant elements from path lists, such as used by the $PATH environment variable.

• When the development directory style required actions to be perfomed on the development directory, the obsolete "creating symbolic links to baseline" message was produced. This was confusing. A more generic message is now used, which is intended to be less confusing.

• The aecp(1) -delta command now follows the whiteout preference when copying a "removed" file. • The test suite now runs much faster.

• A bug has been fixed in the *aedbu*(1) command. It no longer complains about permissions when the develop_begin_undo_command has been set.

• A bug has been fixed in the *aeimport*(1) command. It no longer uses the Attic portion of filenames when populating the history directory tree.

• The change details listing now prints comments in a wide column when the comments are lengthy.

• There is a new $Active_Directory substitution, used to obtain the development directory, or the integration directory, depending on the change state. This is rather like the default behaviour of the *aecd*(1) command.

• There is a new ${project version} substitution.

• The commands run by aeipass are now accompanied by more file name information, so that you can know which source file corresponds to which UUID history file, if there is a failure in the history commands.

• The *aeca*(1) command now checks for and discards duplicate architecture names. This fixes a bug with unsatifiable architecture dependencies.

• The *aeb*(1) command has been improved, it no longer keeps running the project_file_comand over and over again.

• A bug has been fixed in the *aenf*(1) command; it now preserves existing file contents if new files already exist in the development directory.

• A bug has been fixed in the "aet -regression" command, it no longer reports free()ing a non-existent string.

• A bug has been fixed in the *aed*(1) command, it no longer reports a bug when a cross branch merege is attempted for a file independently created in both branches.

Version 4.23
Version 4.22.2 (18-Oct-2007)
This is an update for the 4.22 stable release, it is meant to help Aegis users while the next release cycle ends.

• [1684820] Fixed a bug in aeclone that caused aecp −delta segfaults.

• The symlink farm now handle derived files registered within Aegis more like normal derived files.

• [1697199] The change_pconf_get function no longer looks for historical versions of files, if it can help it. This makes many things go faster and solved the problem of configuration fields redefinition. While this change does not make Aegis more time safe, it cures one of the symptoms.

• Fixed test 222 to work with recent releases of subversion.

• The *aepconf*(5) man page has been improved.

• [Debian 435422] The reference manual was wrongly referring to −Page-Headings instead of −Page-Header. The documentation has been updated to match the source code.

• [1704108] The *aecp*(1) −delta command now follows the whiteout preference when copying a "removed"file.

• [1704100] A bug has been fixed that caused *aecp*(1) −delta X to copy in a change also a file with the old name of a file *aemv*(1)ed before delta X.

• The generated Makefile now installs *aelock*(1) with the correct permissions.

• [1701701] A bug has been fixed in the *aetar*(1) command, it no longer creates tarballs that cause BSD and other tar to complain like this: tar: End of archive volume 1 reached tar: Unexpected EOF on archive file

• The configure script now handle correctly the datadir substitution.

Version 4.22.1 (14-Apr-2007)
• Test t0247a-walt.sh has been fixed, it was not exporting AEGIS_TEST_DIR. This make *aeintegratq*(1) leaving stuff in the home directory of the user.

• Some minor fix that prevented Aegis to build on RPM based distributions has been fixed.

• The t0011a.sh test script failed when *lex*(1) was missing, since it is not required to build Aegis the test script has been modified to pass even when *lex*(1) is missing.

• *aedist*(1) now handle certain renamed files correctly when receiving branches or entire-source.

• The t0011a.sh test script failed when *lex*(1) was missing, since it is not required to build Aegis the test script has been modified to pass even when *lex*(1) is missing.

• *aedist*(1) now handle certain renamed files correctly when receiving branches or entire-source.

• [1691122] Newer versions of the autoconf tools introduced a new @datarootdir@, and complained loudly if it wasn't used. Aegis configure does not trigger anymore those warnings.

• The test suite does not use anymore *diff*(1) −u because not all systems have gnu diff, so the use of gnu diff's −u option is nor portable, and will give false negatives on some systems.

• The test suite does not use anymore *diff*(1) −u because not all systems have gnu diff, so the use of gnu diff's −u option is nor portable, and will give false negatives on some systems.

• *aedist*(1) −rec now save the UUID as the user defined original-UUID if the UUID is already present in the repository. This is especially useful when receiving changes in the same repository.

• *aeclone*(1) now preserve the the UUID of the original change as the original-UUID user defined attribute of the new change. It also copy any other used-defined attribute.

• Test 89 has been disable on HP-UX-10 because that system has a "vendor specific" (i.e. broken) *cpio*(1) archive format.

• Test 95 has been improved to be less sensitive to *libmagic*(3) differences.

• Test 207 has been changed to be less sensitive to *sort*(1) differences.

• The project_specific setenv:* variables are now exported only once.

• [1674882] The following bug as been fixed: if a file is created and renamed within a single branch, and that branch is integrated, then the file is not included in the output of 'aedist −send −es' from subsequent branches.

• A bug has been fixed in the *aedbu*(1) command. It no longer complains about permissions when the *develop_begin_undo_command* has been set.

• The *aedist*(1) −rec command now better handles file renamed (not aemved) to match the local repository state.

• A bug has been fixed that caused the *change_pconf_get* function terminate *aegis*(1) with a fatal error if applied to a branch without a config file (e.g. if the trunk does not contain any closed branch).

• The *aenpr(1) −keep* command now set the administrator recursively.

• A bug has been fixed in the ${project-specific} substitution, it now works correctly with the *aesub*(1) −bl command.

• *aedist(1) −received* has been modified to set the user defined attribute foreign-copyright to true when receiving a remote change set. This in order to avoid *aede-policy*(1) complain about incorrect copyright notice at *aede*(1) time.0

• A bug has been fixed that caused an *aemv*(1) followed by an *aenf*(1) to generate two different files with the same UUID.

• Avoid the "multiple permission set" error on quit.

• A bug has been fixed in the UUID generating code; it was running out of file descriptors.

• A bug has been fixed in the *aet*(1) −regression command, it no longer reports free()ing a non-existent string.

• A bug has been fixed in the *aed*(1) command, it no longer reports a bug when a cross branch merge is attempted for a file independently created in both branches.

• A bug has been fixed which caused *aeipass*(1) to assign UUID to files at branch integration pass time. This can happen if the files was created and integrated with an old Aegis release, lacking support for file's UUID. This bug make it possible to have the history for a file split into two part, one accessible via the file_name, the other accessible using the UUID.

• A segfault in *aeannotate*(1) has been fixed.

• A bug has been fixed related to the use of the *unchanged_file_integrate_pass_policy=remove* policy described in *aepconf*(5). In this case aeipass failed to reset the locked_by field from the project fstate file, this prevented subsequent changes to modify the removed file.

• A bug has been fixed in the handling of the symlink farm, for development directory styles which use them for derived files. Derived files in the baseline directory which were formerly source files, but then aerm-ed, are now included in the development directory when copy/link styles are used.

• A bug has been fixed in the *aenf*(1) command; it now preserves existing file contents if new files already exist in the file development directory.

• The ./configure script has been improved to stop with a fatal error if the bzip2 library is not available.

Version 4.22 (29-Mar-2006)

• A bug has been fixed in the *aeclean*(1) command, it now correctly resets the change build and test times.

• A bug has been fixed in writing of tar and cpio data, in cases where there was one byte too much padding.

• A bug has been fixed in the *aeintegratq*(1) command, it no longer ignores change number zero.

• A bug has been fixed in the *aepromptcmd*(1) comand, it now understands that when the build command is "exit 0" then no build is required.

• The *aede*(1) comand now runs the *review_pass_notify_command* (instead of the *develop_end_notify_command*) for projects configured to skip the *being reviewed* state.

• A bug has been fixed in the *aeannotate*(1) command, it no longer uses the wrong timestamp when creating histories for completed branches.

• A bug in the *aed*(1) command has been fixed, it no longer reports a bug when trying to merge a file that has been renamed.

• A bug has been fixed in the *aet*(1) command, it now correctly handles multiple architectures being reported for batch test results.

• A bug has been fixed in the *aet −regression* command, the *batch_test_command* now correctly handles multiple architectures in the results.

• The notification scripts distributed with Aegis have been fixed, they now correctly substitute recipients' email addresses.

• A bug has been fixed in the *aediff*(1) command, the *−change* option is now ablew to cope with degenerate forms of the delta name in cases like *aediff −change D001* and similar.

• A bug has been fixed in the *aenc*(1) command, it now takes more notice of project testing default settings.

• A bug has been fixed in the *aeget*(1) interface, the adjective for the alternate listing link at the bottom of the Integration Histogram pages has been inverted.

• A bug has been fixed in the *aeget*(1) command, is is now always possible to see the error produced by a script when the *noerror* modifier is specified.

• A bug has been fixed in the *aeget*(1) web interface, it now provides the correct links to the more and less detailed file history pages.

• The *aeget*(1) web interface no longer emits broken links to removed source files.

• A bug has been fixed in the *aenbr*(1) command, the *protect_database* project attribute is now correctly inherited from the parent branch.

• A bug has been fixed the the RSS feed, where HTML special characters were not rendered correctly.

• A bug has been fixed in the *aeipass*(1) command, it no longer fails if the *history_create_command* was not set, it uses the *history_put_command* instead, as it is supposed to.

• A bug has been fixed in the *aedist −send* command, it no longer attempts to include the source of removed files.

• A bug has been fixed in the *aedist*(1) command, it no longer segfaults when compiled with DEBUG defined.

• A bug has been fixed in the *aedist −replay* command, it no longer downloads change sets more than once.

• A bug has been fixed in the *aedist −send* command, it no longer obtains the wrong version of the project files when building patches for files which have been renamed.

• A bug has been fixed in the *aedist*(1) command, no longer attempts to include the source of removed files.

• A bug has been fixed in the *aedist −pending* command, it now resolves project aliases.

• A bug has been fixed in the *aedist*(1) command, it no longer segfaults on IRIX.

• A bug was fixed which caused the development_directory of a branch to be recorded as an absolute path in the Aegis meta-data, rather than relative to the home of the project. This problem make it difficult to move a project to a different location in the filesystem.

• There is a new open source project example on the web site, which allows tarballs to be unpacked and turned into an Aegis project in less than 30 minutes.

• There is a new *aefinish*(1) command which may be used to read the state of a change set and then run all of the Aegis commands necessary to to end development. See *aefinish*(1) for more information.

• The *aexml*(1) command now understands ".bz" output file suffix, in addition to the ".gz" suffix it already understood. The man page has been updated to cover the −output opion.

• The *aerevml −send* command is now able to produce bzip2 compressed output.

• The restrictions on project alias names have been eased. It is now possible to have any alias name you like, so long as it doesn't contain any shell special characters.

• It is now possible to set change attributes from the command line, without going via an editor. See *aeca*(1) for more information.

• The *aetar −send* command is now able to produce bzip2 compressed output.

• There is an new *aetar −exclude* command line option, allowing you to exclude files from the tarball being unpacked and used to for the change set. This is typically necessary when a tarball includes derived files (e.g. the ./configure script in most open source projects).

• There is a new *aetar −exclude-auto-tools* option, which can be used to exclude derived files commonly found in open source projects using the GNU Autoconf and GNU Automake tools.

• There is a new *aede-policy*(1) command which may be invoked by *develop_end_policy_command* to enforce additional local policies. See *aede-policy*(1) for more information.

• When symlinking files (source or derived) into the development directory, the last-modified time of the link is set to the last-modified time of the file being linked to, when the underlying filesystem supports it.

• The aefa(1) command now accepts *name=value* attribute assignments on the command line.

• The *aet*(1) command now understands *name=value* pairs on the command line, and passes them unchanged to the test command. The −force option implies a `force=1` variable setting.

• The *aepatch −send* command is now able to produce bzip2 compressed output.

• The *aesvt*(1) command now uses the *bzip2*(1) algorithm by default. There is a *aesvt −compression-algo-rithm=gzip* option for forwards compatibility.

• There is a new *ae-repo-ci*(1) command which may be used in an *integrate_pass_notify_command* to do a parallel check-in of a change set into a second parallel repository. It understands CVS and SVN at the moment; it is easy to extend to understand more repository types. The old *ae-cvs-ci*(1) script now invokes

the *ae-repo-ci*(1) command.

• The build step of the development process can now be made optional. Configuring a *build_command* of "exit 0" will tell Aegis your project does not need to be built.

• The *aedist −replay* command now adds a compatibility modifier to all of the downloads URLs, so that the change set received will be compatible with the version of aedist at the receiving end.

• The *aedist −send* command now accepts a *−no-mime-header* option, to make it easier to validate the *aedist*(1) output against the real *cpio*(1) command.

• The *aedist −send* command is now able to produce bzip2 compressed output.

• There is a new *entire-source-hide* file attribute which may be used to omit site-specific files from *aedist −send* change sets.

• The *aetar −remove-path-prefix* option now also accepts a numeric argument.

• The *aeannotate*(1) command now understands the *−change* and *−delta* options.

• The *aedb*(1) command has been enhanced to check that directory permissions above the development directory will be traversable by the integrator and the reviewers.

• The *aecpu*(1) comand now understands the *−read-only* option to mean uncopy all of the insulation files.

• There is a new *aelock*(1) command, which may be used to take read-only locks. This can be useful for backups, and other activities outside Aegis' scope which require a constant project state to operate correctly.

• The *aedist* command can now perform file merges with better results.

• The *aedist −receive* command now looks to see if the executing user has project admin priviledges, and if so does not cancel testing exemptions.

• The *aedist −receive* command now applies patches using the *patch*(1) command, rather than doing it less well itself.

• The *aedist −replay* command now attempts to use the same change number as on the remote system. A bug has been fixed in the way it looked for change numbers.

• There is a new *unchanged_file_integrate_pass_policy* field in the project configuration file, which controls what to do when a change set contains an unchanged file at integrate pass time.

• It is now possible for developers to edit a change description when a change is in the awaiting development state, if the project has *developers_may_create_changes* enabled.

• The *aed*(1) command is now optional. Configuring a *diff_command* of "exit 0" will tell Aegis your project does not need to be differenced.

• The *aeget*(1) interface now places HTML anchors in description text where it recognizes them.

• There is a new *aeget:inventory:hide* change attribute, which may be used to prevent strictly local change sets from being advertised in the *aeget*(1) change set inventory.

• The *aeget*(1) web interface file listings pages now link the edit numbers to file versions. When history is available there are also links to the previous verion, and the arrow is linked to a diff page.

• The *aeget*(1) presentation of file history has been improved to highlight renaming of files.

• The *aeget*(1) web interface now has a recursive option on its project integration history pages.

• The *aebuffy*(1) command is now able to run the *tkaer*(1) command from more states, and it now accepts 'q' to quit. The display of changes with double quotes (") in their brief description has been improved.

• A build problem with libcurl not being present has been fixed.

• A bug has been fixed which caused errors when Aegis was compiled with g++ 4.1

• A build problem has been fixed on Solaris.

• A build problem related to *bison*(1) using *libintl*(3) has been fixed.

- The `./configure` script has been improved to correctly detect installation of the OSSP UUID library.

- A build problem on HP/UX has been fixed.

- A build problem on MacOS X has been fixed.

- A build problem has been fixed where libraries required by the `./configure` script are located under `/usr/local/lib` or some other non-standard place.

Version 4.21 (10-Nov-2005)

You must have the *Gnome libxml2* library (`http://xmlsoft.org/`) installed in order to build Aegis. Please install the xml2 library version 1.8.17 or later. You do not have to install the rest of Gnome, the library can be used on its own. If you are using a package based install, you will need the *libxml2-devel* or *libxml2-dev* package in addition to the *libxml2* package.

Ideally, you would also install the *libmagic* package, used to determine file types, just as *file*(1) does. (This is not to be confused with the `libmagic6` image manipluation library. If you are using a package based install, you will need the *libmagic-devel* or *libmagic-dev* package in addition to the *libmagic* package.

- A bug has been fixed in the *aecp −independent −output* option, which resulted in an error when Aegis tried to chmod nothing.

- The auto file promote feature previously available in *aed*(1) has been added to the *aeb*(1), *aecp*(1), *aerm*(1) and *aenf*(1) commands.

- The *aedist −pending* and *aedist −missing* commands now print the number of changes in the remote inventory.

- A bug was fixed in the *aecp* command which caused a segfault sometimes when the user tries to copy a removed file.

- The *aedist −replay* command now accepts a *−maximum* option, which includes change sets not yet completed in the local change set inventory when considering what to download.

- There is a new *develop_end_policy_command* field in the project configuration file. It can be used to add addition constrains to change sets before they can complete *aede*(1) successfully.

- The *aedist −receive* command now annotates remote change sets (typically, change sets downloaded via the *aedist −replay* command) with their origin URL.

- A bug has been fixed in the *aebuffy* command where it would display incorrectly when the brief_description of a change contained double quotes.

- It is now possible to attach a comment to all commands which involve a change state transition, e.g. *aenc*, *aede*, etc. This is done using the **−reason** command line option, just as you are able to do for review fail, etc.

- A bug has been fixed in aenc, where it did not correctly copy user defined attributes.

- There is a new *aelcf*(1) command to efficiently generate lists of change source files for use by your build tool.

- There is a new *aelpf*(1) command to efficiently generate lists of project source files for use by your build tool.

- There is a new cache of state information attached to each delta, the project file state at the time of the delta. This has the potential to accelerate aecp −delta, and all other project_file_roll_forward-based operations. Large projects may want to turn this off, because each delta will produce another large project file state cache.

- There is support for generating RSS feeds from Aegis. See the Aegis project pages on the Aegis web interface for an example. See *aepconf*(5) and *aeget*(1) for more information.

- The ${change delta_uuid} substitution now allows access to the delta_uuid in the *being integrated* state.

- The "wrong file" error message from aedist has been improved, to say what was expected.

- There is a new optional *$filename* substitution for the *history_put_command*, so that you can attach the

current name of the check-in to the history file meta-data. There is a new optional *$uuid* substitution for history_put_command, so you can attach that as mete-data, too.

• There is a new history tool bundled with Aegis. See *aesvt*(1) for more information.

• There is a new *default_regression_test_exempt* project attribute.

• The *aedist −receive* delta selection mechanism has been improved: previously the edit-origin-UUID attribute was considered in favour of the original-UUID attribute, with this change it is used the change set, bounded to the edit-origin-UUID or to original-UUID, more recently integrated. This should reduce the frequency of logical conflicts.

• There is a new *aerevml*(1) command, which can be used to send change sets in the RevML format. See *aerevml*(1) for more information. The *aeget*(1) web interface is also able to serve change sets in this format.

• A problem has been fixed which caused Aegis to fail on the hppa port of Debian.

• The *aetar −receive* program now uses the archive name as the brief description.

• A bug was fixed in *aedist −send* which caused segfaults when processing some files.

• A bug was fixed which caused *aedist −send* to produce an archive that can not be *aedist −receive* because of an operation impossible to replicate in a change set.

• There is an implementation of Robert Collins' subunit testing framework available. See *aesubunit*(1) for more information.

• A bug was fixed in *aedist* that caused an error when receiving a branch's archive generated with the *aedist −send −entire-source* option.

• A bug has been fixed in *aedist −receive* that caused a segfault in the rename handling code.

• The *aedist −missing* listing (and the *aedist −replay* behaviour) now check for branch UUIDs as well, just in case someone fetched a branch as a change set and applied it. However, *aeget* does not report these UUIDs, because that would be too confusing.

• It is now possible to specify any sufficiently unique leading prefix of a UUID rather than the full 36 characters.

• There is a new *${History_Path}* substitution available. It gives you the path name of the history file corresponding to the given filenames.

• A bug in *aedist −receive* which caused incorrect delta selection has been fixed.

• There is a new *aedist −pending* option which can print the list of local change sets missing from a remote repository.

• The *aedist −receive* command is now able to use the edit-origin-UUID attribute to copy modified files from the right origin.

• A bug has been fixed in *aedist −send* where some types of incomplete changes would fail an assert.

• There is a new *aexver*(1) command which can be used to view historical versions of files in an Aegis repository. See *aexver*(1) for more information.

• A bug as been fixed which caused *aemv*(1) to incorrectly rename a file to an existing directory

• It is now possible to specify user-defined user attributes in the ˜/.aegisrc file.

• The *aenf*(1) command now gives a warning if you specify the "config" file without the "−config" option. This is the old name for the project configuration file, the new name is "aegis.conf".

• The *aefind*(1) command now understands {+} to mean the resolved file name, and {−} as the unresolved file name.

• There was a bug where Aegis would exit with a fatal error if one of the directories on the AEGIS_PATH was read-only. Such directories are now ignored.

• The *aetar*(1) command has been improved to process modified and created files in a batched way; this improves the speed.

• Additional explanatory text has been added to the message printed when error message translation files can't be found.

• The *aenf*(1) command now understands the *−keep* and *−no-keep* options, to explicitly control the creation of new files in the development directory.

• A bug has been fixed in *aemv*(1) which failed to check the new name against the filename charset, *etc*.

Version 4.20 (28-Jan-2005)

Please Note: Users are advised to check the history command settings in their project configuration files. With the advent of file UUIDs, the history mechanism now decouples source file names from history file names. In particular, the assumption that the history file basename is the same as the source file basename is no longer true. Correct settings may be found in the *lib/config.example/* directory of the source distribution.

• The defaulting rules for the change number (if none was specified on the command line) have been altered. the current directory now takes precedence over the "only one" rule. This seems to meet user expectations better.

• A bug has been fixed in the *aecvsserver*(1) command which caused to to fail when accessed by some clients.

• A bug has been fixed which caused many of the programs to leave temporary files behind.

• A bug has been fixed in the *aedist −send −entire-source* command where it would hang for some cases. (Actually, it would dump core after using up all available swap space on an infinite recursion).

• A bug has been fixed in the *aedist* command (and other places) where the open of the project configuration file could fail, due to not properly reconstructing in historical circumstances.

• A bug has been fixed in the integration build which was removing files it should not, for *during_build_only = true* work area styles.

• The *aeb* command now complains *much* less about "directory not empty" when using the link farm.

• A bug has been fixed in the *aetar −send −entire-source* command where some files were missing when asking for a complete set of historical sources.

• A bug has been fixed in the *aedist −send −entire-source* where some files were zero length when asking for a complete set of historical sources.

• A bug in *aedist*(1) has been fixed, it was forcing regression test on the receiving side even if the change set does not require it and *default_test_exemption* was set to true. It was annoying especially if the test suite take a long time to run completely.

• Some bugs have been fixed in *aediff*(1) which caused it to mis-parse the command line in some cases, and it was also barfing on the expected exit status 1 when an actual difference was found.

• A bug has been fixed in the *aecpu*(1), *aemtu*(1), *aemvu*(1), *aenfu*(1), *aentu*(1) and *aermu*(1) commands. They were not repairing the symlinks (*etc*) required by the *development_directory_style* settings.

• A bug has been fixed in the *Change_Files* listing; it was not showing the locked-by information.

• A bug has been fixed in the code which updates the development directory symlinks. It was failing to make all the directories required.

• A bug has been fixed in the *aedist −send −entrie-source* command, where it would segfault in some cases.

• A bug has been fixed in reading plain *diff*(1) format patches. This was particularly obvious because *aeannotate*(1) uses this form of diff by default.

• A bug has been fixed in *aeget*(1) where it was showing removed source files as available for download.

• A bug has been fixed in *aeget*(1) where it produced invalid output if the SCRIPT_NAME environment was not set.

• A bug has been fixed in *aeget*(1) where it would sometimes ignore modifiers. This was particularly noticable in the download pages.

• A bug has been fixed in the *aeimport*(1) command. It was using the old work area style configuration file parameters, instead of the new *development_directory_style* settings.

• A memory leak has been fixed in the symbol table code.

• A bug has been fixed in the project_file_find_by_uuid function. In some cases it would SEGFAULT, particularly once the memory leak in the symbol table code was fixed.

• Several build problems have been fixed.

• The *aeintegratq*(1) command has a new −loop *options, which causes it to keep processing changes that become available while it is running.*

• The *aet*(1) command has a new −sugest-limit option which runs as many regression tests as possible (from most relevant to least relevant) but stops after the given number of minutes. This is a way for running the most relevant tests in a limited time. For example, this option could be used if a project has so many integrations in a day that it can only afford 20 minutes of integration testing for each one.

• The *aed*(1) man page has been updated to better describe the behaviour around the merge command.

• The *aetar −send* command now accepts an −include-build option that also add build files, registered with *aegis −new-file −build*, to the ouput archive. A −not-include-build option is also accepted.

• The *aetar −receive* command now avoids copying build files from the baseline because this operation is forbidden and the error stops the processing.

• There is a new *${Change_Attribute}* substitution, which is replaced by the values of the change attributes named.

• The history recapitulation code (project_file_roll_forward) now indexes by UUID rather than by file name (with backwards compatibility for UUID-less repositories). The user visable result is that file history reports and listings now accurately track renames.

• The *aet −nopersevere* option now also stops for *no result* as well as *fail*.

• The aedist −receive command now understands file UUIDs. This means that it will operate on the correct file even when one or the other repository has renamed the file.

• The *aedist −receive* command has been enhanced to perform file merges if necessary.

• There is a new *aedist −replay* option. When used in with an *aeget*(1) server, it can be used to synchronize two repositories. The *aedist −missing* option may be used to show what would be downloaded.

• The *aefa*(1) command, with the −edit option, now shows you the content type, rather than adding it silently.

• There is a new *aediff −command* option, allowing you to specify the command you want to use to display the difference. For example, you could use *tkdiff*(1) or *mgdiff*(1) to display the change graphically.

• The *aediff*(1) command now adds labels when it is producing a context or unified diff output.

• There is a new optional *review_policy_command* field in the project confioguration file. This allows for customised review policies for each project, including multiple reviewers and specific reviewers for portions of the sources.

• There is a new ${Change_Reviewer_List} substitution, which is replaced by a space separated list of reviewers of the current change, since the last develop end. This is of particular use to the *review_policy_command* field of the project configuartion file.

• There is a new ${Change_Developer_List} substitution, which is replaced by a space separated list of all the developers of the current change.

• There is a new ${quoted_email_address} substitution, which replaces it arguments with the email addresses of the names users. See *aesub*(5) for more information.

- The notification scripts have been updated to use the new ${quoted-email-address} substitution.

- The remaining *aegis.cgi*(1) functions have been reproduced in *aeget*(1). The *aegis.cgi*(1) script is now deprecated.

- When the UUID of a change is cleared it (because some operation on the change set invalidates it) is saved in a change attribute named *original-UUID*.

- The *aedist −receive* command is now able to use the original-UUID attribute of the incoming change set to select the delta to merge with.

- The "path unrelated" error message has been updated to make it more informative.

- All attribute names (project, change and file) are now case-insensitive.

- The *aedist −receive* command has been enhanced to allow you to select the branch of the delta to merge with.

- Several classes within the source have been refactored.

Version 4.19 (30-Sep-2004)

Please Note: Users are advised to check the history command settings in their project configuration files. With the advent of file UUIDs, the history mechanism now decouples source file names from history file names. In particular, the assumption that the history file basename is the same as the source file basename is no longer true. Correct settings may be found in the *lib/config.example/* directory of the source distribution.

- There is a new *development_directory_style* field of the project configuration file. This allows CVS-style and Arch-style work areas, in addition to the BCS-style and viewpath work areas already supported. These new work area styles permit many existing projects to use Aegis with no change to their build systems. The libsndfile and OpenLDAP projects, for example, have been imported and built without modification. See *aepconf*(5) and the *Dependency Maintenance Tool* chapter of the User Guide for more information.

- There is a new *aediff*(1) command, which may be used to obtain a *diff*(1) listing of a file for different deltas.

- There is a new *aepromptcmd*(1) command, used with bash's PROMPT_COMMAND environment variable. It can be used to obtain a colored prompt, simulating the process described in Kent Beck's book *Test Driven Development*.

- There is a new *signed_off_by* field of the project configuration file. Set it to true if you want "Signed-off-by" lines appended to change set descriptions as the changes pass through the Aegis process. The *aede*(1) and *aerpass*(1) commands now understand two new **–signed-off-by** and **–no-signed-off-by** options, to override the project setting. The *aedist −send* and *aepatch −send* commands also understand the new **–signed-off-by** option, to add the "Signed-off-by" line to the outgoing change set description. Conforming to: http://www.ussg.iu.edu/hypermail/linux/kernel/0405.2/1301.html and http://www.osdl.org/newsroom/press_releases/2004/2004_05_24_dco.html

- The *aet*(1) command has been enhanced to allow integrators to run specific tests.

- The *aesub*(1) command can now read the text to be substituted from a file or standard input.

- It is now possible to use the project-specific attributes to specify environment variables to be set for commands executed by Aegis. This can be used to set a predictable PATH, for example.

- It is now possible to customize the *aeget*(1) web interface using project specific attributes.

- The *ael*(1) command and the *aeget*(1) web interface now have file inventory pages, for the project file inventory and the change file inventory.

- There is a new "change set inventory" listing available via the *ael*(1) command and the *aeget*(1) web interface, which lists changes and their corresponding UUIDs, and links to an aedist download for each change. The idea is that the *aeget*(1) pages may be used to automate downloading change set your repository does not yet have.

- There are two new history commands in the project configuration file, the history_transaction_begin and

history_transaction_end fields. It is not an error if these fields are absent. If you need a transaction key, use the $version substitution.

• The *aedist*(1) command now runs all tests required for the change set, and honors test exemptions.

• The *aedist*(1) command now sleeps for a second to ensure that the last-time-modified of derived files will be strictly later than source files, and that the *aeb*(1) timestamp will also be strictly later then the last-time-modified for the source files.

• The tkdiff(1) man page has been updated to say how to use mgdiff(1) instead of tkdiff(1).

• All commands which accept the **–change** option may now be given a change set UUID. You can discover a change's UUID using the *ael cd* (list change details) or *ael inventory* listings.

• The *aed*(1) command now restores source file from backups (,B) when a merge fails. Previously this was not the case and subsquent aed invocations failed because the source file was missing.

• The *aetar –send* command now has an **–add-path-prefix** option, so that you can add a path prefix to all of the files in a tarball. The *aeget*(1) CGI interface now adds a path prefix to generated tarballs by default.

• Whenever you edit file attributes, there is a `Content-Type` attribute added automagically if none was there already. The idea is that this could be used by scripts to differentiate file types.

• The *aepatch*(1) command now uses *diff –u* by default.

• A number of build problems on different systems have been fixed.

• A number of minor problems with tests on different systems have been fixed.

• A bug has been fixed in the aepatch command; it was not parsing simple diff patches correctly.

• The example history commands have been updated to work better with the new UUID code.

• A bug has been fixed in aecp –delta, where it would fetch the wrong version of a file in some cases.

• A bug has been fixed in the handling of the executable bit.

• A bug has been fixed in *aede*(1), where is did not permit branches to end when they had a removed file (without a UUID) which has been subsequently recreated (with a UUID).

• A bug has been fixed in the *aeget*(1) command for file contents. It was giving a "multiple permissions set (bug)" error message.

• A bug in the *aedist –receive* command, where it was not accurately manipulating the incoming change set UUID.

• A bug has been fixed in *aed*(1) which caused it to SEGFAULT.

• A bug has been fixed in the *aede*(1) command, where it failed to copy the UUID when it promoted a file from "create" to "modify" automatically.

• A bug has been fixed in the `$date` substitution, it was not advancing properly when used in progress messages.

• A bug has been fixed in the command line processing of the *aefa*(1) command.

• A bug has been fixed in the *aegis –review-begin* command; it was not operating correctly when the change was in *awaiting_review* but the project was in *goto_being_reviewed*.

• A bug has been fixed in the `$basename` substitution; it now functions exactly like *basename*(1) command.

• A bug has been fixed in the *aet –bl* command; it erroneously stated that the `$Search_Path_Executable` substitution was mandatory, when it actually optional.

Version 4.18 (10-Jun-2004)

• A number of build problems have been fixed, particularly concerning GCC 3.3 and later.

• The *aemv*(1) command has been enhanced to accept more than two file names. You are now able to move serveral files in the one command.

• The *aedist −receive* command has been enhanced to process move operations in a batched way. This improved performance when receiving a change that renames many files.

• The `./configure` script has been changed to take note of the `--sysconfdir` option, used to specify the location of the /etc directory.

• A bug has been fixed in the *aepatch*(1) command. It would SEGFAULT when a non-source file was patched.

• A bug has been fixed in the *aemeasure*(1) command. It would SEGFAULT when no files were named on the command line.

• The Russian error message catalogue has been updated.

• A subtle bug in the change file out-of-date tests have been fixed. It did not adequately address the transition case for projects containing files with and without UUIDs.

• The `./configure --sysconfdir` option is now honored. It is *very* important to set it to */etc* when you configure Aegis.

Version 4.17 (3-Jun-2004)

• Each new change set is now assigned a Universally Unique Identifier (UUID) to allow it to be tracked across geographically distributed development. The *aedist*(1) and *aepatch*(1) commands now send the change set UUIDs, and preserve them on receipt.

• Each file now has a Universally Unique Identifier (UUID) which allows tracking files across renames, even on geographically separate sites. (The *aedist*(1) and *aepatch*(1) commands send the file UUIDs, the next release will take advantage of them on receipt.)

• The history filename used to remember file history is now based in the file UUID, if the file has a UUID. This simplifies continuity of history across renames (this fucntion always been present in Aegis, but harder to access).

• As a consequence of the UUID being used to generate history file names, there is no longer the restriction that new files may not be named after the directory portion of a deleted file (or vice versa).

• There is a new *aecvsserver*(1) command, which presents Aegis projects and change sets as CVS modules. All of the core CVS functions are supported. This code needs to be exersized and tested by users.

• It is now possible to specify arbitrary attribute names and value for each source file. The *aefa*(1) command may be used to edit file attributes. The *aedist*(1) and *aepatch*(1) send these attributes; a future release will take advantage of the information on receipt.

• It is now possible to attach arbitrary attribute names and values to change sets. For example, you can use this to remember the bugzilla tracking number for a change.

• The *aepatch*(1) command now includes change set meta-data as a compressed BASE64 encoded block at the top of the patch, after the human-readable text but before the files. This means that *aepatch*(1) can be as effective as *aedist*(1) is transmitting chaneg sets. Patches without meta-data still work as before.

• There is a new report script which writes change logs in Debian format.

• The *aeget*(1) web interface has been improved. The *aepatch*(1) download now accepts `compat=`*N* modifier, and there is a new Project Staff page.

• There is a new *ae-cvs-ci*(1) support script which may be used as an *integrate_pass_notify_command* to commit change sets to CVS in parallel.

• There is more documentation in the User Guide about using GNU Diff, particularly using diff −U to provide whole-file listings with "change bars" on the left hand side.

• The files view of the *aeget*(1) web interface now accepts options to control the page contents. The

simplest view allows recursive fetch of project sources using wget or similar, with no extraneous links to confuse the results. Previous behavior is preserved by the aeget-generated links.

• You now receive a warning when you are seeing the short version of the error messages. These are terse and often quite cryptic. the long form of the error messages is to be preferred.

• The behaviour of the *aedeu*(1) command has changed slightly. When changes are in the *being reviewed* state, and Aegis has been configured to use the *awaiting review* state, the *aedeu*(1) command will now report an error. This is so that reviewers don't waste their time reviewing changes which have already been returned to the *being developed* state. Think of the change as "belonging" to the reviewer while in the *being reviewed* state.

• The *aedist −send* command has a new *−compatibility* option, use to indicate the version of the **receiving** aedist program. This, in turn, selects the features which may be added to or omitted form the generated .ae file.

• There is a new config file usage, and a corresponding *aenf −configure* command line option. It is now possible to move project configuration files. It is now possible to remove project configuration files, provided there is at least one left. The *aeimport*(1) command now avoids files which have the same name as the default project configuration file ("aegis.conf" or "config") and will use something else.

• The *aeipass*(1) command now adds a symlink from the delta directory to the baseline once it has been integrated. This helps lots of (idiotic) compilers which insist on burying absolute paths into executables.

• It is now possible to assign to some project configuration file array fields more than once. This can be useful where the configuration file is split into several pieces on several branches.

• The source language has been changed from C to C++. Future releases will take advantage of this.

• Several bugs have been fixed in the *aeget*(1) web interface where it would display "−42" instead of "0" for changes and branches numbered zero.

• A bug has been fixed in the *aed*(1) command when merging files which have been renamed. It now recognises they need merging.

• A bug has been fixed in the *aenf*(1) command. It now correctly ignores difference files when given a directory name.

• A bug has been fixed in *aedist*(1) where one of the *aegis −new-file* commands was missing a *−no-template* argument. Under some circumstances, this resulted in change sets which could not be *aedist −receive*d.

• A bug has been fixed in the way invalid sequences of multi-byte characters are handled by the internationalization code. This potentially affected all reports, listings and error messages. The symptom was that aeannotate listings could sometimes have a blank source code column.

• A bug has been fixed in the *aepatch*(1) command. It was creating empty patches for some changes in completed project branches. This also affected *aedist −send* and *aecp −delta* and *aecp −rescind* in some cases. It was caused by a subtle flaw in the non-detailed case for the *project_file_roll_forward* function.

• A bug has been fixed in the handling of the MANPATH enviromnet variable by the *profile* and *cshrc* scripts.

• A bug has been fixed in the *aedist −receive* and *aepatch −receive* commands has been fixed. There were cases where these commands could access off the end of an array and SEGFAULT.

• A bug has been fixed in the *aede*(1) command when it received pre-config-usage change sets. It used to try to remove the last project configuration file, which is a fatal error, and made it impossible to receive the change set.

• A bug has been fixed which caused the *aetar*(1) command to hang (actually, any thing which consulted LDAP or NIS) because the reserved symbol "send" was being overloaded. The reserved symbol "clone" was also being overloaded. Both have been fixed.

• A bug has been fixed which caused the *aedist*(1) command to reprt the wrong error when the input file did

not exist.

• A bug has been fixed in the *aenbru*(1) command which made project aliases disappear.

• A bug has been fixed in the *aede* command. It would fail with new build files already in the baseline.

Version 4.16 (14-Jan-2004)

• There is a new *aecp −keep* option, causing *aecp*(1) **not** to overwrite file contents in the development directory.

• The *aedist −receive* option now understands changing the type of a file.

• It is now possible to specify a URL to the **−file** option on the command line of *aedist*(1), *aepatch*(1) and *aetar*(1). The data will be downloaded and applied.

• More work has been done towards making the code compilable by a C++ compiler.

• The project list (see *ael*(1), *aeget*(1), etc) is now sorted in a slightly more natural way, as are the version statistics at the end of an *aeannotate*(1) listing.

• A bug has been fixed in *aede*(1) for branches, where Aegis would complain about build source files (created by the *aenf −build* command) being out-of-date. This, of course, was difficult or impossible to fix, and unnecessary because the next build would fix them.

• A bug has been fixed in the *aecp −independent* command, where it did not preserve the execute bit, nor honour the user's umask.

• The missing *aemt* and *aemtu* alias ve been added to the profile.

• More detail has been added to *aepconf*(5) detailing how to create the project configuration file for the first time.

• A bug has been fixed in *aedist*(1) and *aepatch*(1) which would cause an assertion failure (or segfault) when you tried to *aedist −send −delta −es* files which did not exist at that delta.

• A bug has been fixed in *aedist*(1) and *aepatch*(1) which caused an assert failure (or segfault) when you tried to send a file which had been created and removed in a branch, and after the branch was integrated only a remove record exists in the parent branch.

• The problem with test 134 failing has been fixed.

• A bug has been fixed in *aeipass*(1) which prevented changing a file's usage from being as straight-forward as it should have been.

• The source RPM (and the *spec* file) now has Build Prequisites specified.

• The **−Change** option now accepts more than just a change number. It now accepts many forms similar to those used by the *${version}* substitution, allowing its output to be used directly as command line input; forms such as −c 1.2.C34 and −c=5.6.D78 are now understood to imply a **−branch** option as well as either **−change** or **−change-from-delta**, respectively. In addition, you may prepend a project name, to imply the **−project** option as well; form such as −c aegis.4.15.C28 are understood.

• The *aemeasure*(1) program now also generates Halstead metrics.

• A bug has been fixed in the symbolic link handling code. In some cases it would report "multiple user permissions (bug)" and not complete correctly.

• A bug has been fixed in the test of *aedist*(1) for moved files. There was nothing wrong with *aedist*(1), the test itself was broken.

Version 4.15 (17-Nov-2003)

• A bug has been fixed in "ael cf". It used to fail an assertion when there were no files in the change.

• A bug has been fixed which caused aeipass to segfault when adjusting file modification time stamps in some circumstances.

• A bug has been fixed in the cross branch merging code. It would sometime erroneously complain about files no longer being in the baseline.

• A bug has been fixed which caused *aedist*(1) and *aeannotate*(1) to segfault. It was caused by the roll forward history mechanism ignoring some branches in some cases.

• A bugs has been fixed in the *aenrv −Descend_Project_Tree* option, which was free()ing a project twice, sometimes causing segfaults.

• The *aeget* CGI interface is now able to retrieve historical versions of files.

• The *aeget* CGI interface now has support for file metrics.

• The *aeget* CGI interface has been enhanced to provide more information about project files and change files: activity, conflicts, history.

• The *aeipass*(1) command now sets the AEGIS_INTEGRATION_DIRECTORY environment variable before running the *integrate_pass_notify_command*, so that you can add a symlink for compilers which insist on placing absolute paths into debugging information in object files.

• The *aeget* CGI interface has been enhanced to provide more information about project files and change files − activity, conflicts, history.

• The *aeget* CGI interface now reports more project information.

• The *aeget* CGI interface now has download links in many of its menus, allowing more and better downloads than the old *aegis.cgi*(1) script.

• The way *aenf*(1) and *aent*(1) work have been made more generous. It is now possible to *aerm*(1) a file and then *aenf*(1) or *aent*(1) the same file in the same change. This is useful for changing the type of a file. Previously this has to be done as two consecutive changes.

• The *aecp −independent* command has been enhanced to allow you to extract versions of built files (created with *aenf −build* and maintained at *aeipass*(1) time).

• Documentation has been added to *aer*(5) for the try/catch mechanism.

• There was a disagreement between the *aereport*(1), *aeannotate*(1), *aedist*(1), *aefind*(1), *aeimport*(1), *aels*(1), *aepatch*(1), *aerect*(1), *aetar*(1) and *aexml*(1) man pages and the commands themselves about the existence of the **−version** option. The commands now behave as documented.

• There is a new *Project_Branch_Dates* report, which may be used to see when branches of a project were begun and completed.

Version 4.12 (29-Sep-2003)

• A bug has been fixed in *aedist*(1) where it handled moved files incorrectly.

• There is a new experimental *aeget*(1) program. It is a potentially faster, potentially more capable replacement for the *aegis.cgi*(1) script. At the moment it isn't, it's experimental.

• A bug has been fixed in *aedist*(1) where it would sometimes segfault when sending transparent files.

• Command completion now works for the *aemt*(1) and *aemtu*(1) commands.

• A bug has been fixed where the symbolic link farm could point to the wrong place when change files are transparent.

• Change file notification commands have been added for the *aemt*(1) and *aemtu*(1) commands. See *aepconf*(5) for more information.

• A bug has been fixed in *aefind*(1) command where it could report files which had been removed.

• A bug has been fixed in the *aecp*(1) command where it would scramble the *aet −reg* exemption.

• A bug has been fixed in the *aede*(1) command. The problem manifested as an *aet −reg* command which terminated early.

• There is a new *aexml*(1) command. You can now obtain various pieces of the Aegis database as XML. See *aexml*(1) for more information.

• The the *new_file_command*, *copy_file_command* and *remove_file_command* fields of the project *config* file are now defaulted correctly.

• Theer is a new *$change_files* substitution. See *aesub*(5) for more information.

• The project config file has a new *architecture_discriminator_command* field. Now you can use an arbitrary command (rather than *uname*(2) information) to determine the architecture. See *aepconf*(5) for more information.

• The Russian message translation has been updated.

• The German message translation has been updated.

• The *ael*(1) command now has a new *incomplete* listing. It lists changes between *awaiting review* and *being integrated*. inclusive.

• The *ael*(1) command now accepts arguments for the listings. The *default-change*, *default-project*, *outstanding-changes* and *user-changes* lists now accept a user name argument.

• The *aemt*(1) command now understand the **−UNCHanged** option, so that files which are in the branch, but unchanged from the deeper branch, may be made transparent.

• A bug has been fixed in the *wecp*(1) command where the **−OverWrite** option did not honor the presence/absence of the **−ReadOnly** flag.

• There is a new *aeedit* script. See *aeedit*(1) for more information.

• A bug has been fixed in the file history mechanism (as used by the *−delta* options, *aeannotate*(1), *aedist*(1), *aepatch*(1), etc) which did not correctly understand transparent files.

• The *aeclean*(1) command now touches all of the source sfiles. It also accepts a **−NoTouch** option.

• There is a new *$change_files* substitution. See *aesub*(5) for more information.

• The *aeclean* command now touches the source files as well. Use the now **−no-touch option if you don't want this.**

• There have been several improvements to the output of the *aegis.cgi* script and the web site.

• For Aegis developers: all of the K&R insulation has been removed; you now need an ANSI C compiler to build Aegis. Some preparation has also been done to get the source ready for a C++ compiler.

Version 4.11 (29-Jan-2003)

• For Aegis developers: the developer build now uses *sudo*(8) to simplify and automate the tricky bit. The regular distribution build is unchanged.

• A bug has been fixed where the '*aet −reg*' command could not find any tests to run, cause by inconsistencies in the view path handling for project file searches.

• A partial Romainian translation has been added.

• A Spanish localization has been added. It needs work by a human.

• The French localization has been improved.

• The *aedist*(1) command now preserves the executable bit on files.

• There is a new *−descend-project-tree* option for the *aena*(1), *aera*(1), *aend*(1), *aerd*(1), *aeni*(1), *aeri*(1), *aenrv*(1), *aerrv*(1) and *aepa*(1) commands, to apply the action to all descendant branches of the project.

• A bug has been fixed in *tkaer*(1) which stopped it working on some systems.

• The *aeintegratq*(1) command now copes better with changes leaving the *awaiting integration* state.

• A bug has been fixed in the *aeimport*(1) command which misunderstood RCS branches.

• A bug has been fixed where there *aenf*(1) command would use the new *config* file about to be created, which was almost always wrong.

• There is a new *${substr}* substitution. See *aesub*(5) for more information.

• The *aeclone*(1) command now understands transparent files.

• The *aecpu*(1) command now restores test exemptions in some cases.

• There is a new *aemeasure*(1) command, which procudes simple file metrics for use with Aegis.

• There is a new *project ancestors* report.

• Trunk version number no longer have a leading dot.

• Command line completion now works for *zsh*(1).

• The *aetar*(1) command now preserves the executable bit on files.

• A bug has been fixed which caused *aetar*(1) to hang.

• The *aereport*(1) and *aesub*(1) commands now gave the same email address for users.

• The *aeannotate*(1) command now olny prints caption columns if their value changes. This highlights the differences, and is less distracting.

Version 4.10 (24-Dec-2002)

• There is a new *aemt*(1) command, used to make branch files "transparent". This is like an *aecpu*(1) command for branches, but done through the agency of a change set.

Note: The behaviour of the view path in the presence of transparent files is complete, however full support for *aecp −delta* and reports is not. Support will be present in the next release. File transparency information stored by this release will be able to be used by *aecp −delta* and reports in the next release.

• There is a new *aemtu*(1) command, to undo the effects of the *aemt*(1) command.

• It is now possible to use the *aeclone*(1) command on changes in the *awaiting development* state.

• The problematics directory permissions check has been removed from the *aeintegratq*(1) command.

• A bug has been fixed in *aecp*(1) when retrieving deltas before files were removed.

• There are new *${split}* and *${unsplit}* substitutions for manipulating search paths (*etc*). See *aesub*(5) for more information.

• A bug has been fixed where test time stamps were not updated for batch tests which covered multiple architectures.

• The *aedist*(1) program now includes a change number, which will be used on receipt if possible. Note that this produces .ae files which are not backwards compatible; the −nopatch option will suppress inclusion of the change number in the archive.

• A German translation of Recursive Make Considered Harmful has been added, courtesy of CM Magazin.

• A bug with *aeimport*(1) and removed files has been fixed.

• A problem has been fixed with the transition case when a project changed from `develop_end_action = goto_being_reviewed` to `goto_awaiting_review` while having changes in the *being reviewed* state.

• A problem with long command lines has been fixed in the *aedist −receive*, *aepatch −receive* and *aetar −receive* commands.

• A problem with *aeimport*(1) and binary files has been fixed.

Version 4.9 (23-Oct-2002)

• The *aepatch*(1) and *aetar*(1) commands now accept −add-path-prefix and −remove-path-prefix options, for manipulating the filenames when unpacking an creating a change set. The *aepatch*(1) documentation has been significantly improved.

• There is a new *aecp −rescind* option, which may be used to rescind (roll back) a completed change. See *aecp*(1) for more information.

• The Debian */etc/mailname* file is now understood by the *${user email}* substitution.

• There is a new *project_gantt* report, which produces comma-separated-value (CSV) output, for extracting data to import into Ms. Project. Unfortunately, Mr. Project does not yet know how to import CSV files.

• It is now possible to provide a comment to the *aerpass*(1) command, just as you always could to the *aerfail*(1) command.

• The *aet*(1) program now has a −progress option, to tell you where it is up to. See *aet*(1) for more information.

• The Russian error messages have been updated.

• The *aeimport*(1) program now understands the CVS Attic directory.

• There are new *perl*, *PLural_Forms*, *capitalize*, *downcase* and *upcase* substitutions. See *aer*(5) for more information.

• A work-around for the aeimport/delta bug has been added, for projects which were imported with the buggy aeimport.

• Aegis developers will need to upgrade to GNU Autoconf 2.53 or later, as the GNU Autoconf files have been updated to work with that version. This does not affect normal users.

- Many typos have been fixed in the documentation, and some improvements have been made.

- Some build problems have been fixed.

- Numerous improvements have been made to the web interface.

Version 4.8 (19-Aug-2002)

- A bugs has been fixed in the *aetar −receive* command, where it incorrectly complained about shorty input files.

- Numerous changes have been made to the web interface. They now use cascading style sheets, have more navigation links, and inclde tarball downloads.

- Several build issues have been resolved.

- A bug has been fixed in the *aeimport*(1) command. The symptom was that the *aecp −delta* command misbehaved. The probelm was that the first delta needed a timestamp *prior to* the first change set taken from the import sets.

- A bug has been fixed in the *aepatch −send* command, where it would add `Index` lines for files with no differences.

- A bug has been fixed in the `protect_development_directory = true;` handling, where it would cause a "multiple user permissions setting" error message.

Version 4.7 (6-Aug-2002)

- The *aefind*(1) command now has −resolve as the default. To get the previous behaviour, use the −NoResolve *option.*

- In the *aeca −e* and *aepa −e* commands, it is now possible to quote strings with at-signs (@) instead of double quotes. This type of string allows newlines within the string. See *aegis*(5) for more information.

- For the benefit of Aegis developers, there is now HTML documentation genaretd by Doxygen (if you have Doxygen installed). When developing an Aegis change, in your development directory, point your browser at `doxygen-html/index.html`. The `common/str.h` file is an example of the style desired, should you wish to contribute to the effort to get all of the header files suitably annotated. Also, the removal of the K&R C support has started, see the files in `common/*.[ch]` for examples. Also <varargs.h> is not longer used anywhere.

- The *aedist*(1) command has two new options, −patch and −nopatch, which may be used to control how and when *aedist* uses patches. See *aepatch*(1) for more information.

- A bug has been fixed inthe *strncasecmp* function. This only affected you if your system did not have a native version of this function.

- The *aeca*(1) command now accepts a −fix-architecture option. This option may be used to correct the architecture list of a change automatically.

- The *aedist −receive* command now runs the *aeca −fixarch* command when a change set arrives which modified the project *config* file. This should fix many of the "architecture not in project configuration file" problems when seeding new projects.

- Some deficiencies on the "How to Become a Developer " instructions have been addressed. The native Aegis build (but not the Makefile.in) now builds the "tags" and "TAGS" files so that it easier to navigate the sources.

- There is a new *aetar*(1) command. It may be used to send and receive tarballs as Aegis change sets. See *aetar*(1) for more information.

- Missing documentation on the *aepconf*(5) man page about the fine grained file change notification commands has been added.

- Some changes have been made to the Aegis web interface, with more back links. Also uses *html2diff*(1) if available.

- It is now possible for reviewers to use the *aet*(1) command to run tests against the changes they are

reviewing.

• The command completion for the *aet*(1) command now works better; it now completes project test names as well as change test names.

• The *aepatch*(1) and *aedist*(1) commands now cope with a wider range of input vagueries, including some weird things done by MTAs and more content transfer encoding synonyms.

Version 4.6 (11-Jul-2002)

/* vim: set ts=8 sw=4 et : */

• The *aeipass*(1) command now sleeps, rather than issue the rather alarming "warning: file modification times extend into the future" message. There is a new project *config* file field, *build_time_adjust*, which controls this behaviour, but it is strongly recommended that you leave it on the default setting.

• There is a new *${base_relative}* substitiontion, almost the inverse of *${source}*. See *aesub*(5) for more information.

• A bug has been fixed with the *aeca* and *aepa* −*edit* option. It was caused by the change in the previous release which added editor user preferences.

• A few build problems have been fixed.

• A bugs has been fixed in the *tkaepa* script. It would sometimes fail the "OK" button.

• A bug has been fixed in the "user changes" list. It was not explicitly passing the project name when it accessed the list of user owned changes.

Version 4.5 (26-Jun-2002)

• It is now possible to set pager and editor preferences in your *.aegisrc* file. See *aeuconf*(5) for more information.

• A bug in *aepatch* −*receive* has been fixed, where it would sometimes misapply a patch. The search used to determine the patch position (when it needs to be offset) has been improved.

• The *aedist*(1) and *aepatch*(1) commands now accept −delta and −delta-date options.

• The *integrate_q.sh* shell script has been replaced by the *aeintegratq*(1) Perl script. It can now lots more useful things. See *aeintegratq*(1) for more information.

• A bug has been fixed in the date parsing code (used by the −delta-date option). There was the potential to mis-calculate dates after February 2000.

• A bug has been fixed in *aepatch* −*receive*, where it sometimes complain of "no uudecode data in file", for files which did not require uudecoding.

• There are more change-specific substitutions available. See *${change ...}* within *aesub*(5) for more information.

• The *aepatch*(1) command now understands ordinary diff listings, in addition to the context and unified differences it already understood.

• There is a new *aeannotate*(1) command, used to produce annotated source file listings. See *aeannotate*(1) for more information.

Version 4.4 (12-May-2002)

• It is now possible to specify system wide user preferences. See *aeuconf*(5) for more information.

• The *aepatch*(1) command now understands the quoted-printable content transfer encoding.

• The *aepatch*(1) is more robust when receiving patches that want to use a change number that has already been used.

• The Dutch error message translations have been updated.

• There was a problem with the way the install directory for *aegis.cgi* was being determined. The *aegis.cgi* script is now installed into $bindir by default. There is a *aegis.cgi.i* helper script to find your web server's *cgi-bin* directory and copy *aegis.cgi* there, but this is not done automatically. See *aegis.cgi*(1) for more information.

• Another change has been made to cope with still more Bison changes.

• A French error message translation has been contributed.

• A problem with *aedist −receive* has been fixed, where the new *configuration_directory* could interact with the order of file creation.

• A big has been fixed in the uuencode output, which could occasionally miss the "begin" line.

• A bug has been fiexed in the context diff parsing, where it would get the last hunk wrong if it was a hunk which deleted lines, due to incorrect end-of-file handling. This affected both *aepatch −receive* and *aedist −receive*, because *aedist*(1) now includes patches for better merge behaviour.

• Numerous issues concenting the new GNU Gettext versions have been addressed.

• A number of Solaris build problems have been fixed, and one genuine bug buried in the warning messages (change completion time was wrong for changes not yet completed).

• More information about the "mod times extend into the future" warning issued by *aeipass*(1) has been added to the man page.

• Some improvements have been made to the web pages.

Version 4.3 (16-Apr-2002)

• The notification shell scripts all now use sendmail consistently. Autoconf support for locating sendmail is not yet present.

• A problem which caused a core dump on Cygwin has been fixed.

• The *aede*(1) command now gives a more informative error message when files in a branch require merging.

• There is now an interconnection between the *aeib*(1) and the *aeb*(1) command. When you specify a minimum integrate begin, you also get a minimum integrate build.

• A bug has been fixed which caused *aenf*(1) to dump core if you used the file name accept pattern.

• The executability or otherwise of each source file is now remembered. If any of the execute bits are set at *aede*(1) time, the file is remembered as executable. When an executable file is copied into an integration directory or development directory, all of the execute bits (minus the project umask) are set.

• A bug has been fixed in the "*aecp −ind*" command, where it would give a "there is no development directory" error when you tried to extract a file version from history of a completed branch.

• Many of the web pages have been updated to provide a more consistent and intuitive interface. It is also possible to get patches, via the *aepatch* command.

• Interrupts are now ignored during database writes. This should alleviate some of the problems induced by Ctrl-C. (It would be nice to find the actual cause.)

• The *aedist*(1) command has been enhanved to include a patch fragment for modified files, as well as the whole source files. On receipt, if the patch applies cleanly the whole source is ignored. If the file does not exist at the receiving end, or the patch does not apply cleanly, the whole source file is included. The incrimental cost is very low, because all of the patch pieces appear in the source file, and thus compress

exceptionally well. The net result is to greatly reduce merge costs on recipt of `.ae` files. However, this change to *aedist*(1) is only backwards compatiple. Previous versions of *aedist*(1) will give a fatal error if they see a `.ae` file generated by this version of *aedist*(1).

• File name resolution is now more robust in the face of permission problems.

• Some error message translations have been improved.

• A small bug has been fixed in the history labeling.

• You can now use shell (#) and C++ (//) comments in your project *config* file, if you prefer them to C comments.

• A bug has been fixed in the maintenance of the symlink farm. It would often fail to make all of the necessary symlinks.

• There is a new project attribute, *protect_development_directory*, which when true causes the development directory to be read-only in states between *awaiting_review* and *being_integrated*.

• A problem has been fixed where some reports would fail is users had made their *.aegisrc* files unreadable.

• A number of small build problems have been fixed.

• Command completion has been added for the *aeb*(1), *ae_c*(1), *aeca*(1), *aecd*(1), *aechown*(1), *aeclean*(1), *aecp*(1), *aecpu*(1), *aedb*(1), *aedbu*(1), *aede*(1), *aedeu*(1), *aedn*(1), *aeib*(1), *aeibu*(1), *aeipass*(1), *aeifail*(1), *aena*(1), *aencu*(1), *aend*(1), *aenf*(1), *aenfu*(1), *aeni*(1), *aenrv*(1), *aentu*(1), *ae_p*(1), *aepa*(1), *aera*(1), *aerb*(1), *aerbu*(1), *aerd*(1), *aerfail*(1), *aeri*(1), *aerm*(1), *aermu*(1), *aerpass*(1), *aerpu*(1), *aerrv*(1) and *aet*(1) commands. More will be added in the future.

• It is now possible to specify a directory to contain project *config* file fragments. These fragments are then read in as if catenated as a single project *config* file. See *aepfonf*(5) for more information.

Version 4.2 (26-Feb-2002)
// vim: set ts=8 sw=4 et :

• There is a new "–No-Page-Headers" option which may be used to suppress page and column headers in listings and reports.

• There is a new "aecp –delta-from-change" option, allowing the specification of a delta number by specifying the number of a completed change.

• The "aecp –ind –delta" command now omits files which did not exist at the given delta.

• There is a new *history_label_command* which may be used to label your history files at each integration. See *aepconf*(5) for more information.

• The code which guesses which change you are working on, based on your current directory, has been enhanced to cover far more cases. It can recognize the integration directory, too.

• There is a new *Change_Log* report, which generates reports in the style of common Internet change logs.

• The web interface is now able to show you file differences between deltas.

• A bug has been fixed in the "*aecp –delta*" command (for all delta variants). The problem occurred when you wanted to copy a version of the file before the file has been modified by the branch (but it wasonly a problem for files modified later in the branch, files never modified by the branch were OK). As a side-effect of the bug fix, "*aecp –delta*" now goes significantly faster (*N* times faster, where *N* is the number of files you are copying).

• Build problems caused by new Bison releases have been fixed.

• A number of oversights in handling the new *awaiting review* state have been corrected.

• The *${expr}* substitution has been enhanced to include modulo, logical not and the six relative operators. All using the usual C syntax and precedences. See *aesub*(5) for more information.

• There is a new *${switch}* substitution, see *aesub*(5) for more information.

• A Russian localization of the error messages has been contributed.

• A bug has been fixed in the "aecp –output" code, which sometimes incorrectly created directories.

• A bug has been fixed in the symbolic link maintenance code. It now repairs links which point to a file which is too deep in the ancestor tree, and has been subsequently replaced. It now uses a single pass, rather than two passes.

• The *change_file_command* field of the project *config* file is now available at a finer granularity. There are 8 new commands (the *copy_file_command*, *copy_file_undo_command*, *new_file_command*, *new_file_-undo_command*, *new_test_command*, *new_test_undo_command*, *remove_file_command* and *remove_file_-undo_command* fields) which may be individually configured. They default to the previous behaviour, for backwards compatibility. See *aepconf*(5) for more information.

• A bug has been fixed in the *aepatch*(1) command, which prevented it form constructing patches for changes on completed branches.

• The *aeipass*(1) command now issues an error message if the build changes a source file. (Previously it erroneously reported that the history tool had done the damage.)

• A bug has been fixed in "aecpu –unchanged" in the case where the change had no files. (It tried to uncopy a file called the empty thring.)

• The missing *aemvu*(1) man page has been added.

Version 4.1 (6-Dec-2001)
    Note: You will need to upgrade all of your Aegis machines simultaneously for this release. It introduces database changes which older Aegis release will not be able to cope with.

• A bug has been fixed in *aed*(1), which tried to access a nonexistent files under some circumstances.

• A bug has been fixed in *aede*. When two changes created the same file, the second change received a misleading message from *aede*.

• There is a new German message translation.

• There is a new *tkaepa*(1) command, giving GUI access to the *aepa*(1) command.

• The *aeclone*(1) command now runs the *change_file_command* and *project_file_command* from the project *config* file. This is in order to be more consistent withthe *aecp*(1) command.

• The "time safe" property described by Damon Poole mostly applies to Aegis' operation. One last area related to future times and the –delta options. There is now a warning in the instance where non-time-safe behavior may occur.

• The *history_put_command* and *history_create_command* field of the project *config* file are strongly recommended to be identical. It is now possible to only specify the first one, and the second will default to it.

• A bug has been fixed in the *aeib*(1) command, when the *link_integration_directory* field in the project *config* file is false.

• There is a new *awaiting review* state, and new *aerb*(1) and *aerbu*(1) commands to go with it. It is now possible to configure your project to have changes enter the *awaiting review* state after *aede*(1), rather than the *being reviewed* state. It is also possible to skip the review states altogether and immediatelt enter the *awaiting integration* state.

• There is a new *modeP field for the specification of architectures in the project config file. The means that you can designate some architectures as mandatory and some as optional. See aepconf(5) for more information.*

• The *aenbr*(1) command now populates the new branch's baseline with symlinks if the project *config* file is set so that they would remain after an integration build. This is more consistent with the *aedb*(1) behaviour in the same situation.

• There have been a number of changes to the web pages, accompanying the move to SourceForge, along with some corrections.

• There is a new *aels*(1) command, which may be used to list directories, annotated with Aegis' file

attribute information.

• The *aeclean*(1) command now accepts the **–Keep** option, so that it reports what it would do, rather than actually do anything.

• A problem with the CGI interface, which reported a bug to the user, has been fixed.

Version 3.29 (31-Oct-2001)

• The *aeimport*(1) command can now import CVS repositories which contain binary files.

• There is a new *${Read_File_Simple}* substritution. It is like *${Read_File}*, but is does not substitute into the file contents.

• The *aecp –independent* command now accepts a *–output* option.

• There is a new *${environment}* substitution, allowing you to access environment variables within substitutions. See *aesub*(5) for more information.

• There is a new *${project-specific}* substitution, allowing you to define project specific value to be inserted into various commands. See *aesub*(5) and *aepconf*(5) for more information.

• The *aefind*(1) command now works with completed change. It searches the baseline.

• A problem with using the ${source} substitution within the *integrate_pass_notify_command* has been fixed. It was getting the path wrong.

• The batch test command is only ever invoked of there are tests to run. (This fixes a problem where it would simethimes run with no arguments.)

• The web reports now bahave themselves when the names of non-longer-here user appear.

• A number of errors and typos have been fixed in the documentation.

Version 3.28 (21-Aug-2001)

• There is a new *aepatch*(1) program, which may be used to send an receive changes using the classic open source patch format. See *aepatch*(1) for more information.

• The general output mechanism (for listings and reports) has been rewritten to be significantly faster.

• Numerous small things have been improved in and around the `./configure` script and the `Makefile`.

• The web interface has been improved. It should result in better save file locations being suggested for `.ae` files. cgi vs downloads

• Aegis now takes a baseline read lock during tests, so that the baseline doesn't move out from under your tests, causing mysterious failures.

• A bug has been fixed in the `subst` function of the report generator. It was free()in a string twice.

• There is a new *${developer email}* substitution, for inserting users' preferred email addresses into commands. Useful for the state transition notification commands.

• There is now more text in the *aepconf*(5) man page, explaining how each of the pattern fields are applied to file names. It is now explicit when patterns applied to whole file names, and when they only applied to path name elements.

• A segfault has been fixed in the removed file whiteout code.

• The *aesub $source* substitution now works in combination with the **–BaseLine** option.

• The *aegis.spec* file now mentions the executables again.

Version 3.27 (26-Jun-2001)

- A bug has been fixed in the *aesub*(1) $delta substitution. It now works correctly for completed changes.

- A bug has been fixed in *aermu*(1), when used in combination with the symlink farm. It no longer complains about "multiple user permissions set".

- A serious bug has been fixed in the locking code. The bug meant that only one build per project could happen at a time. (There was never any risk of repository or Aegis database damage.)

- A bug has been fixed in the *aedist*(1) command. It failed to correctly recognise files produced using the *aedist −send −no-ascii-armour* option.

- The *aecpu −unch* command now deals more gracefully with files which have been removed from the project in the mean time.

- There is a new *change file history* listing, similar to the *file history* report. It is much faster, much more informatuve, and less selective.

Version 3.26 (21-Jun-2001)

- Some optimizations have been done to the input parsing. Depending on your architecture, this will or won't be noticable.

- The locking has been changed so that *aeipass*(1) takes precedence over new development builds, so that there is a guarantee that *aeipass*(1) will succeed in finite time. Current development builds will run to completion, but new development builds will block until the *aeipass*(1) gets the basline lock and subsequently completes.

- The "file format error" bug in *aedist*(1) has been fixed.

- There is a new *project activity* report, which is useful to project leaders to see what has been happing in the project, sorted by time and then by user name.

- Aegis can now transparently cope with binary files, even if the history tool cannot. It does this by using a MIME encoding for binary files. (This can be configured away, if your history tool correctly handles binary files.) See *aepconf*(5) and the *User Guide* for more information.

- There is a fix for the "file unrelated" error commonly seen on Solaris and BSD when combined with an automounter, in come cases. It relies on the *bash*(1) behaviour which sets the $PWD environment variable. (GNU libc does this internally to the *getcwd*(3) function, not all libcs do.)

- The *aer*(1) report generator now has access to the project *config* file fields, through a new config field in the report generator's concept of the project state.

- There is a new *aer*(1) $comdir substitution, which gives access to the shared state directory, configured at build time.

- The *aebuffy*(1) now accepts a project name on the command line.

- There is a new *build_covers_all_architectures* field in the project *config* file, so that you can tell Aegis that the build tool builds all architectures wimultaneouasly. See *aepconf*(5) for more information.

- The *tkaer*(1) command now has a comment editor, so that you can edit your review fail comments from within the GUI.

- A bug has been fixed which was caused *aenbru*(1) to delete one directory level too deep when the branch was removed.

- There is a new getuid() function in the report generator.

- This change fixed a bug in *aede*(1) where it would not allow a branch, created with *aeimport*(1), with new files which had subsequently been modified to end development, when those files had never existed in the baseline.

- It is now possible for project administrators to nominate the developer in the *tkaenc*(1) dialog. You are presented with a pick list.

- There is a new *aesub*(1) $history_directory substitution. This may be used in scripts which

access the history tool's files directly.

• There is a new *change_file_undo_command* field of the project *config* file. It is similar to the *change_-file_command* field (it defaults to it if unset), but is executed by all of the "undo" file commands.

• The *aede*(1) command no longer cancels your build and test time stamps. This means that you don't need to re-build if you don't change anything, after *aedeu*(1) or *aerfail*(1).

Version 3.25 (3-Apr-2001)

• It is now possible to remove users who's accounts have been removed (the affects the *aera*(1), *aerd*(1), *aeri*(1) and *aerrv*(1) commands).

• There is a new *−description-only* option to the *aeca*(1) command. This is useful for editing only the descrioption, and also for use within scripts.

• The *−file* option has now been generalized to accept "−" to mean the standard input. This is useful in scripts.

• There is a new *aebuffy*(1) command, which may be used to see what changes a user has outstanding. It needs X11 (Tk/Tcl) to work. Named after the *xbuffy*(1) command.

• The *tkaer*(1) command now presents you with a "detail" button, so that you may see the change details when pwerforming a review.

• The restriction that placed the function name at the start of the command line (*e.g.* the "−cp" of *aecp*) has been relaxed. This may now appear anywhere on the command line.

• The Bourne / BASH shell aliases have been improved, so that they now preserve quoting of special characters and white space. This dates from the earliest days of Aegis. It's wonderful to have it fixed at last.

• There is a new *aemvu*(1) command, which may be used to undo the effects of an *aemv*(1) command. This should prove less confusing than the previous method.

• A big has been fixed in the *aemv*(1) command. It failed to accept the −base-relative option, even though it was documented to do so.

• A bug has been fixed in the quote_tcl() report function. Is fixes the problem with getting the dollars sign into descriptions when using the *tkaenc*(1) command.

• The SCCS section of the *User Guide* and example configurations have been updated and confirmed to work correctly, however I've only tested this with GNU CSSC.

• A bug in the file name handling has been fixed. This was most obvious around the *aecpu*(1) command when you had *create_symlinks_before_build* turned on and you were using an automounter, but it occured at other times as well.

• The *aeimport*(1) command now understands the SCCS format. If the comments in GNU CSSC are accurate, this also means you can import BitKeeper repositories, however I am unable to confirm this.

Version 3.24 (10-Mar-2001)

• There is a new *aeimport*(1) command, which may be used to import CVS archives into Aegis.

• The cross branch merge has been improved so that it uses an earlier version number than it was using, resulting in a more sensable merge.

• A bug has been fixed in the *${quote}* substitution which incorrectly quoted the exclamation mark (!). Unfortunately, quoting isn't at all simple, because you can't exclusively use single quotes *or* double quotes *or* backslash.

• There is now a *${change description}* substitution, allowing you access to the brief description of a change in a substitution. (The suggested RCS history command have been changed to use it.)

• A Dutch localization of the error messages has been contributed.

• Project administrators can also use the *aeibu*(1) command. Handy for abandoned integrations which inconvenience everyone else.

• There is a new project *config* file field, called `build_covers_all_architectures`, which allows you to tell Aegis that your build process can cover all architectures simultaneously.

• The *${quote}* substitution has been fixed to correctly quote more characters. It now prefers the single quote (but is is npt possible to use this exclusively).

• The web site now uses PDF files for documentation, rather than gzipped PostScript. This was for lots of reasons, including the fact theat many folks couldn't work out how to print them, and also IE decompressed them "for free" but left the .gz suffix.

• The report generator, *aereport*(1), can now access fields of the *.aegisrc* file. This is important for accessing the preferred email address in various reports. • The "*aecp −delta*" command now adds removed files to the change as removed files instead of adding them as copied-but-empty files. This should make reproducing projects more accurate, but you need to use *aermu*(1) to get rid of them ,rather than *aecpu*(1).

• The *aedist* program now adds a "Content-Disposition" header to the files it generates. This means MIME programs will unpack it into a correctly named file more often.

Numerous build problems have been fixed, both for Unix and for Cygwin (Windows). There have benn some test script improvements, too.

Contributions have started to roll in using the "aedist" format. This is very encouraging. The instructions for how to do this are contained in the "Howto", in the *Developer* section.

Version 3.23 (29-Oct-2000)

• A bug has been fixed which caused the report generator *change_number* function to give garbage answers for change number zero (fortunalely, not very common).

• There is a new *mtime* function in the report generator.

• There is a new *aecomp* utility, which may be used to compare two active changes, using *tkdiff*.

• A bug in "aesub ${dd}" which reported the wrong directory when applied to branches, has been fixed.

• The project *config* file now contains two new fields, *create_symlinks_before_integration_build* and *remove_symlinks_after_integration_build*, which may be used to better control the behavior of the symlink farm at integration time. (Default behavior is backwards compatible.)

• A new utility called *tkaer* has been contributed. It is for reviewing, and shows you lists of files. When you click on one, it launches *tkdiff*(1) to examine it. You're going to like this one, folks!

• The *aedist −receive* command now preserves the testing exemptions, if possible.

• A problem with very very large test runs and the −no-persevere option has been fixed.

• The *aenf*(1) and *aent*(1) commands now accept **−template** and **−no-template** options, to control the use of new file templates.

• A nasty Catch-22 in the *aedist*(1) command has been fixed, involving the (unnecessary) use of new file templates, when the actual template files don't yet exist in the −receive development directory.

Version 3.22 (13-May-2000)

• **Please Note:** Some code has been added to Aegis to assist in diagnosing problems when restoring projects from backups. If you see a message "aegis: *project-path*: has been tampered with (fatal)" this means there are problems with the project file ownerships. The project owner needs to be >= AEGIS_UID (defaults to 100), and the project group needs to be >= AEGIS_GID (defaults to 10). Use chown –R and/or chgrp –R to fix these problems.

• The *aesub*(1) command now accepts the ${arch} substitution in combination with the –baseline option.

• A bugs has been fixed in the *aedist −receive* command, when one of the files was also locked for review.

• A bug in *aeclone*(1) has been fixed, where it dropped file move information.

• The *aeib*(1) command now correctly validates that youare actually allowed to do this integration. This may win the prize for the *oldest* Aegis bug.

• There is a new ${search_path_executable} substitution. See *aesub*(5) and *aet*(1) for more information.

• Line wrapping in reports works properly again for lines with no white space. The previous release broke it when the wide output generalization was added.

• The *aet −nopersever* option works again. The previous release broke it when the batch test support has implemented.

• A problem with the *aeb*(1) command which made it difficult to use with th symbolic link farm (in some cases) has been fixed.

• A new report is available from the web interface, showing a change-of-state histogram over time for all state transitions (not just the integrate pass transitions).

• A problem with the *aenf*(1) command which made it difficult to use with th symbolic link farm (in some cases) has been fixed.

• The *aeipass*(1) command now preserve file mod times across history updates, if the history tool gratuitously changes them.

• The Solaris and IRIX build problems (wputc, *et al*) has been fixed.

• Numerous documentation patches were received and have been applied.

Version 3.21 (12-Mar-2000)

• A couple of minor bugs have been fixed in *aedist*, especially the problem with sending an baseline image while a change is being reviewed.

• A couple of bugs have been fixed in the tkae* commands, in partucular they no longer leave temporary files lying around.

• Lots of stuff has been added to the HOWTO: a cheat sheet, how to change a project's owner, how to use distributed development, how to become a developer.

• The problem which caused 'aesub ${copyright_years}' to contain duplicates has been fixed.

• There have been Y2K fixes: the date parsing for the –delta-date option has been fixed, and the web page data has also been fixed.

• The *aet*(1) command can now run more than one test at once, if configured appropriately. This is of most use on systems with more than one CPU.

• The –UNFormatted option no longer truncates column values.

• The *aesub*(1) command now accepts the –baseline option, so that you can get project-specific substitution in shell scripts.

• A bug has been fixed in *tkaenc*(1) which gave incorrect testing ssttings. It now also tracks the project testing exemptions.

• A bug in *aenf(1) has been fixed which allowed multiple instances of the same file to be created.*

• A bug has been fixed which caused 'aesub ${search_path}' to fail in some cases.

- A bug has been fixed in *aenf*(1) which allowed you to create the same file multiple times, corrupting Aegis' database and causing *aede*(1) to report mysterious errors. Use *aenfu*(1) multiple times to untangle things.

- Information has been added to the section 5 manual pages, detailing how to access state information from within the report generator. This should make writing report scripts a little easier.

- A bug has been fixed which caused Aegis to misbehave when launched by some versions of *cron*(8) or *at*(1).

Version 3.20 (19-Oct-1999)

- The *aeib* command is now more robust about "foreign" files in the baseline (*e.g.* root-owned core files).

- A bug has been fixed in the *${administrator_list}* substitution.

- A bug has been fixed in the *aedist −delta* option, which caused it to dump core.

- There is now a section in the *History Tool* chapter of the User Guide describing how to add checksums to your history files, in order to detect file corruptions. It is a general technique which applies to most history tools (including RCS).

- A bug has been fixed which caused *aeclone* to misbehave badly when dealing with removed files.

- There is now an embryonic "How To" document for Aegis. Please feel free to contribute subjects.

- You can now say "**−BRanch −**" as a synonym for the "**−TRunk** option, for those commands which accept it.

- The report generator now copes with more types of empty lists.

- A bug has been fixed which caused a core dump instead of a useful error message if you tried to create an alias with an illegal name.

- A bug has been removed which left undeletable branch aliases if a branch was removed.

- A bug has been fixed in *aenbru* which failed to remove the branch development directory .

- The *aenf*(1) command now behaves better when you do horrible things like turn the files you created into directories without telling Aegis first.

- A couple of small bugs have been fixed in the *aenpa*(1) command, both in error situations.

- A bug with the **−interactive** option has been fixed. It will actually ask you, now.

Version 3.19 (4-Aug-1999)

- You can now run a command to generate new file templates if you want, rather than using a simple string substitution. See *aenf*(1), *aent*(1) and *aepconf*(5) for more information. The existing functionality is still there.

- There is a new ${SUBSTitute} substitution, which provides regular expression substitutions. This is useful in new file templates.

- A bug has been fixed which allowed *aede* of a branch when there were some kinds of outstanding changes.

- The automatic change number guessing has been improved slightly, and will cope with some more variation in the *development_directory_template* field.

- There are two new commands, *aenpa*(1) and *aerpa*(1) for creating and removing project aliases. This means that you can give project branches more meaningful names.

- There is a new *aesub*(1) command. It substitutes its arguments and prints them, rather like the *echo*(1) command. This is useful when you need access to the Aegis substitutions in a script.

- The command line option "−−" is now understood. It means "the rest of the arguments on the command line are filenames or strings". Because this makes the options on the command line more "order sensitive" than usual, use with care.

- There is a new *tkaenc*(1) command, allowing you to create new change via a Tcl/Tk GUI. (And a

problem with TCL special characters in description text has been fixed.)

• The *aenf*(1) command now does the right thing with directories named on the command line. In particular, you can now use "aenf ." to import whole directory trees.

• There is a new *State-File-Name* list type, useful when writing cookbooks or makefiles to keep a web page in sync with a change.

• There is a new ${capitalize} substitution, useful for putting in new file templates.

• A bug has been fixed which caused *aeclean* to delete the development directory of changes with no files.

Version 3.18 (8-Jul-1999)

• A bug has been fixed which caused *aecp −delta* to dump core in some cases.

• A bug has been fixed which caused the create-symlinks-before-build functionality to create symlinks to deleted files.

• Still more typos and minor errors have been corrected in the documentation.

• The *aerp*(1) man page has been moved to *aerpass*(1). Similarly for *aerfail*(1), *aeipass*(1) and *aeifail*(1). This should make things easier for users to find the man pages.

Version 3.17 (22-Jun-1999)

• Another *aedist* bug has been fixed – unfortunately it was introduced while trying to fix the last one.

• A Cygwin 20.1 portability bug has been fixed.

• There is a new *${dirname_relative}* substitution. This is useful in new file templates, and also some configured commands.

Version 3.16 (15-Jun-1999)

• There is a new *tkaeca* command. It is a GUI interface to the *aeca*(1) command, using Tcl/Tk.

• There are two new reports available: the *Project-Branches* and *Project-Active-Branches* reports may be used to query about branches within a project.

• A bug has been fixed in the *aedist −receive* duplicate suppression code. It was complaining about user permissions.

• A bug has been fixed in *aeb*(1), which did strange things if you tried to build an unbuildable change.

• There is a new *−No-WhiteOut* option for the *aerm*(1) and *aemv*(1) commands, letting you suppress the "whiteout" files, along with some explanation in the man page about why they are there. See *aerm*(1) for more information.

• The default value of the "*maximum_filename_length*" field of the project *config* file has been raised from 14 to 255. If your project depends on the old default value, you will need to set it explicitly.

• The *aedist −receive* command now accepts a *−directory* option, so you can specifiy the location of the development directory.

Version 3.15 (2-May-1999)

- The "*aedist −receive*" command now accepts a **−delta** option, allowing a received change set to be applied to an historical version.

- There is now some information about managing super-projects and sub-projects in the *Branching* chapter of the User Guide.

- The *aenpr*(1) command now accepts a *−keep* option, so that you can re-attach projects moved after using the *aermpr −keep* command. See *aenpr*(1) for more information.

- The *aenpr*(1) command now accepts *−edit* and *−file* options, allowing you to specify project attributes when creating the project. See *aenpr*(1) for more information.

- If the project *developers_may_create_changes* attribute is true, the *aencu*(1) command now allows developers to destroy changes they created.

- There is a new *add_path_suffix* substitution, for manipulating search paths. See *aesub*(5) for more information.

- There are 3 new substitutions: ${bindir}, ${datadir} and ${libdir}. These are replaced by the *./configure* options of the same name (or the values calculated, if none were given to *./configure*). The old ${lib} substitution is deprecated in favour of the new ${datadir} substitution. See *aesub*(5) for more information.

- Some changes have been made which increases portability, particularly the Linux libc5 *vs* libc6 differences.

- Some changes have been made which increases portability, particularly for Windows NT. This isn't to say Aegis works under Windows NT yet, but it helps the porting efforts. Don't forget to run the mkpasswd and mkgroup utilities included in the Cygwin system.

Version 3.12 (26-Mar-1999)

- The way the Apache configuration files are scanned for and read has been changed, to adapt to recent Apache changes. The *./configure* script will now find it more often.

- The "*aedist −receive*" command has been enhanced to be more robust about change sets without headers (some browsers *generously* strip them all off.

- A bug has been fixed in the "*aedist −receive*" command which sometimes caused decompression failures. An unfortunate interaction with the Windows NT support caused CRLF sequences in the compressed data to be mangled in some cases.

- The wrong include file was being used for zlib. This has been fixed, so it should build more easily now.

- The way MANPATH is handled on Linux has been improved in the *chsrc* and *profile* commands. It will not over-ride */etc/man.config* now.

- The *aegis.cgi* script has been made more robust in coping with *aedist* errors.

- The *symlink_exception* field of the project *config* file now accepts filename patterns, not simply literal filenames.

- There was a problem compiling with gcc 2.8, involving the <stdarg.h> header. This has been fixed.

Version 3.11 (17-Mar-1999)

- The *aet*(1) command now accepts a −force option, forcing tests to be run, even if Aegis doesn't think they need to.

- The Aegis CGI interface has been enhanced so that you can download changes from the generated web pages listing the changes, using the *aedist* command.

- The *aedist −send* command now accepts a −no-ascii-armor option, which leaves off the MIME base 64 encoding. Useful for binary distributions and web servers.

- There is a new *trojan_horse_suspect* field in the project *config* file. This is used by *aedist −receive* to check for files which could be abused to carry Trojan horse attacks.

- The *aedist −receive* command now accepts a −trojan option which treats the incoming change set as suspect, and a −no-trojan option which treats the incoming change set as benign.

- The *aedist −receive* command now quotes filenames (if necessary) when executing commands, thus defending against filenames which contain semicolons.

- The *aenbru*(1) command has been implemented at last. At last! You no longer need to use the *aedbu* work-around.

- The *aedbu*(1) command now gives an error if you attempt to apply it to a branch.

- The *aermpr*(1) command may now be applied to a project with active branches, and will remove the branches as well (provided there are no active changes on any of the branches).

- The dos_filename_required and windows_filename_required fields of the project *config* file have been enhanced to reject the brain-dead Windows special filenames such as "aux" *et al*.

- The ${user} and ${project} substitutions have been enhanced to provide additional information when given an additional argument. Useful for file templates. See *aesub*(5) for more information.

- Several portability enhancements, notably the Windows filename incompatibility has been fixed, and also the Linux *stdlib.h* problem.

Version 3.10 (6-Mar-1999)

- As of this release you must have **zlib** installed before you can build Aegis.

- There is a new *reuse_change_numbers* project attribute, letting you control whether *aenc* fills in holes in the change number sequence. Defaults to true if not set. See *aepattr*(5) for more information.

- There is a new *integrate_begin_exceptions* field in the project *config* file. This permits the user to specify file to be omitted when the integration directory copy/link is performed.

- The *aet*(1) command has been changed so that it does not exit with an error if you have a test exemption but no tests. This is no longer an error.

- There is a new *aedist*(1) command, which may be used to send and receive Aegis change sets via e-mail and the web.

- The *aeclone*(1), *aenbr*(1), *aenc*(1) commands now accept a −output option, a file to contain the automatically generated change number. This greatly assists in writing scripts. See the man pages for more information.

- The *aent*(1) command now accepts a −output option, a file to contain the automatically generated file name. See the man pages for more information.

- There is a new *compres_database* field in the project attributes, allowing the Aegis database to be stored in a compressed form (using the GNU Zip algorthm). Unless you have an exceptionally large project, coupled with fast CPUs and high network latency, there is probably very little benefit in using this feature. (The database is usually less than 5% of the size of the repository.) On slow networks, however, this can sometimes improve the preformance of file-related commands.

Version 3.9 (7-Feb-1999)

• A bug in the merge command has been fixed. It no longer deletes all of your change source files if one of the merge commands fails.

• There is a new *tkaegis* command, using Tk/Tcl to give Aegis a GUI. Contributed by Graham Wheeler <gram@cdsec.com>. Please report *tkaegis* bugs and suggestions to Graham.

• The integrate pass command has been enhanced to cope with RCS and SCCS expanding keywords in source files (modifying the repository) on check-in. This can be ignored, or a warning can be issued, or it can be a fatal error (this is the default). See *aeipass*(1) for more information.

• The worked example in the User Guide has (finally!) been updated to use the new branch numbering. Numerous spelling errors have been corrected.

• The developer section of the worked example chapter now also includes discussion of some common questions raised by folks evaluating Aegis. It covers insulating development directories from the baseline, partial check-in and collaboration.

• The *aesub*(5) man page now brings attention to the fact that the ${Copyright_Years} substitution contains spaces. You often need to quote it.

• The man pages which mention filename limitations, now also note that where underlying file-system has stricter filename length limitations than the *filename_maximum_length* field in the project *config* file, the file-system wins. Mention of this is now also present in *aedb*(1), *etc*; Linux UMSDOS is highlighted as problematic.

• Aegis can now collect code metrics. See *aeb*(1) and *aeipass*(1) for more information.

• There are three new report functions available: *quote_url*, *quote_html* and *unquote_url*. These are all for use when creating Aegis reports for the CGI interface. See *aer*(5) for more information.

• There are several new substitutions available. These include `subst`, `trim_extension`, `trim_directory`, and `trim_filename`. See *aesub*(5) for more information.

• The integrate_q.sh script now works correctly for branches.

• Numerous configure, make and install problems have been fixed for a variety of portability targets.

• The RPM spec file has been corrected to use appropriate file attributes.

Version 3.8 (1-Oct-1998)

• Some users were unable to build the previous release, due to inconsistent wide character support by the various UNIX vendors. This has now been fixed.

• There are two new substitutions, *trim_directory* and *trim_extension*, which are useful for constructing file templates. These can be very useful in constructing skeletons of C++ classes.

• Some changes have been made to pathname handling to better cope with automounters. See *aegis*(1) for more information (see discussion of the AEGIS_AUTOMOUNT_POINTS environment variable). This assumes that paths below the automounter's mount directory are echoes of paths without it (*e.g.* `/home` is the trigger, and `/tmp_mnt/home` is where the NFS mount is performed, with `/home` appearing to be a symlink).

Version 3.7 (22-Sep-1998)

• The *aeifail*(1) and *aerfail*(1) commands now have a new −reason option, to specify the failure reason on the command line, rather than in a file.

• Some file operations are now faster. Mostly, this applies to operations which mention many files, and to projects with large numbers of files. Smaller projects may not notice any improvement.

• There is a new **−delta-date** option to the *aecp*(1) command, allowing deltas to be extracted by date. This change also had the side-effect of making extraction by delta number more accurate on branches.

• There is a new **−base-relative** option to most of the file manipulation commands, *aecp*(1), *aenf*(1), *etc*. This option may be used to specify that relative filenames are relative to the base of the source tree, rather than the current directory. There is also a related user preference, see *aeuconf*(5) for more information.

• There is a new "aeclean" command. It can be used to clean your development directories of non-source files. See *aeclean*(1) for more information.

• The *aeb*(1) command now passes through arguments of the form *name*=*value*, on the assumption that these are variable assignments for th ebuild tool. Previously, they were "resolved" as if they were file names.

• A serious bug in the error and interrupt handling has been fixed. This bug would sometimes case Aegis to hang, and eventually run out of stack, when the user attempted to interrupt Aegis using ˆC.

Version 3.6 (5-Jul-1998)

• The *diff3_command* field of the project *config* file has been replaced by a *merge_command* field. It works exactly the same way, but Aegis moves the files around first, so that the output replaces the change source file. This results in fewer "lost" merges. Those of you who have been hacking the *diff3_command* to move the files around will need to take the moves **out** when you rename the *diff3_command* field to be the new *merge_command* field.

• The columnizing functions used by the report generator and the listings has been enhanced to understand international character sets. This allows native character sets to be used in comments and descriptions, without getting gibberish (C escapes) in the output.

• There is a new *shell_safe_filenames* field in the project *config* file. This field controls whether filenames are required to be free of shell special characters. This field defaults to true if not set, so if you are using any "interesting" filenames, you may need to explicitly set this field to false. (You still can't use spaces or international characters in filenames.)

• There is a new ${quote} substitution for insulating shell special characters in filenames in the commands in the project *config* file.

• A number of bugs relating to environment variables have been corrected; this will make the aereport and aefind commands behave more consistently, with respect to the aegis command.

• A bug has been fixed which caused the final newline of new test files to be omitted.

• A bug which prevented the "aeb −minimum" option from working in any non-trivial case has been fixed.

Version 3.5 (28-May-1998)

• A bug was fixed in the lock waiting code. Aegis will now correctly wait for locks when there are several users blocking on the same lock.

Version 3.4 (22-May-1998)

• There is a new "aegis –clone" command, used to replicate changes across branches. See *aeclone*(1) for more information.

• There is a new "–No-Wait" command line option, which asks for a fatal error if a lock cannot be obtained immediately; this applies to all commands which takes locks. See *aeuconf(5) for more information.*

Version 3.3 (4-Apr-1998)

This release is a bug fix release, and mostly install and portability bug, at that.

• The problem with *errno* defines messing up *glue.c* has been fixed.

• Numerous fixes to the wide character support, to cope with the vagueries of wide character support on many platforms.

• The problem with the LINES and COLS environment variables messing up testing have been fixed. Some tests gave false negatives because of this.

• There is a new *aeb –minimum* option, for use with symbolic links, only, which has a minimal set of source file links, rather than everything in the baseline.

Version 3.2 (22-Mar-1998)

• There are some additional reports available via the web interface. They are mainly to extract error causes and trends from the project history statistics.

• There have been a number of minor bug fixes concerning the handling of old 2.3 projects. This should ease transition for users with existing 2.3 projects.

• A bug in aecp –delta has been fixed, where Aegis was trying to find change state files one branch level too high.

• There is now a re-try performed when a stale NFS file handle error is detected. This should make it easier for some sites which are heavily networked.

• There have been some improvements to the way Control-C is handled. It should be more responsive when waiting for locks.

• Project administrators may now end development of a branch. Since branches can endure for months or years, the orginal branch creator may have moved on. This copes with this situation.

Version 3.1 (15-Jan-1998)

Version 3.0 was not used by many sites. It was available as beta software for about a year, in numerous incarnations. Version 3.1 is the first completely stable version since adding full branching support.

Version 3.0

Version 3.0 is fully backwards compatible with earlier versions, however once a project has been used under 3.0, it will not be possible to revert, e.g. to version 2.3, without restoring the project's "info" directory from backup. While this was generally true of previous releases, any additional state information was usually undo-able with *vi*(1). This time the process is much more involved because the project state files and the change state files have been combined as a necessary step in implementing branches.

Version 3.0 Major New Features

• Aegis now has a feature known in the literature as long transactions, also known as branches. This allows appropriately created changes to be treated as if they were projects, and thus to have changes made to them. This allows a hierachy of changes within changes, to any desired depth. See the *Branching* chapter of the *Aegis User Guide* for more information.

• The project state files have been merged with the change state files. This is part of the implementation of branching. If you have written your own reports, you may need to alter them slightly. For example, in version 2.3 and earlier, reports accessed the project state file using

```
auto p, ps;
p = project[project_name()];
ps = p.state;
```

Because the project state has been moved into a change state, the *state* field above now points at a change state description, and most of the old project information is contained in the *branch* field within it. Reports access this information as

```
auto p, ps;
p = project[project_name()];
ps = p.state.branch;
```

Except for files, which were already present in the *cstate*, so access to the project file list need not change. See the new *aecstate*(5) for more information.

• The new project command now creates branches to match the version number specified. See *aenpr*(1) for more information.

• The error messages of Aegis have been internationalized. This affects how you build Aegis, and the environment Aegis runs in. See the *BUILDING* file for more information. The *cshrc* and *profile* shipped with this release set the LANG environment variable to "en" (for English) if you have not set it; otherwise ther error messages would be terse and uninformative.

• The *aet (1)* command can now suggest tests to be run. This is done by correlating the source files and test files from each change. See *aet*(1) for more information.

• There is now an *aereport*(1) command. The separates out the report functionality from the main body of the Aegis code, allowing the report generator to be used in places where more trust is required.

• There is an intranet Web interface, which is installed automaticly when the install script discovers a web server. This interface allows browsing of much of the Aegis meta-data, of all publicly accessible projects.

• There is now an *aefind*(1) command. This is very similar to the UNIX *find*(1) command, except that it finds in the unified directory stack of a change and its project. The introduction of full branch support can sometimes mean that finding a file may require looking in more than two directories; the *aefind*(1) command makes this simple again.

Version 3.0 Minor New Features

• There is now a –No_Pager option, to prevent listings and help from being redirected to a pager. There is also a user preference to more thoroughly disable paging, and a –PAGer option to override it. See *aegis*(1) and *aeuconf* (5) for more information.

• There is now a –No_PErsevere option to *aet*(1), allowing you to request that *aet*(1) stop after the first test failure. There is also a user preference to set this permanently if desired, and a –PErsever option to override it. See *aet*(1) and *aeuconf* (5) for more information.

• The copyright years attribute has been moved from being a project attribute to a change attribute. This is consistent with a number of other fields which have transparently moved from the project state files into the

change state files, as a result of branching support. See *aeca*(1) and *aecattr*(5) for more information.

• There is a new *Search_Path* substitution, to support builds on branches. See *aesub*(5) for more information. As a side effect, you can also use it in the *test_command* field of the project *config* file, and thus have a search path to look down for data files for your tests.

• Test times are now remembered, so that tests are only run if they need to be. This allows you to keep working on a test, and Aegis only runs those that have not yet passed.

• Aegis now uses "fingerprints" to tell if files have changed, rather than simply relying on file modification time stamps. While this makes Aegis more robust, there is one caveat: it is recommended that 3.0 be installed when there are no changes in the 'being reviewed' or 'awaiting integration' states, in any project.

• There is now a log file prefernces control, allowing users to set their preferred logging behaviour. See *aeuconf*(5) for more information.

• It is now possible to specify the filename for new tests on the command line. See *aent*(1) for more information.

• It is not possible to specify a pattern for test filenames. See *aepconf*(5) for more information.

• There is now a **−MAXimum** option to the *aeib* command, allowing you to keep obsolete derived files at integrate begin. This can avoid long integration build times for large projects.

Version 3.0 Bug Fixes

• Architecture names are now checked a 'develop end' time, to ensure there are no unknown variants. This fixes the mysterious "you must build again" problem.

• The *aecp*(1) and *aed*(1) commands now take a baseline read lock, to be more symetric with the *aeb*(1) command which has always done so. The *aeipass*(1) command takes the complementary baseline write lock, ensuring the the baseline remains constant for the duration of builds, file copies, differences and merges. The manual entries for these commands have all been improved to document this behaviour. See *aeb*(1), *aecp*(1), *aed*(1) and *aeipass*(1) for more information.

• There are now some reminder scripts in the library, which can be run from *crontab*(1). These are installed into the */remind* directory. These scripts can be used to remind users of changes in various states, such as those being developed or being reviewed.

• All of the commands which accept the *−Edit* option now also accept a *−Edit_BackGround* option, allowing edit commands to be piped in from the standard input.

• The *aecp*(1) command now accepts a *−INDependent* option, allowing files to be copied independent of any change (similar to the *−INDependent* option of the command.) See *aecp*(1) for more information.

• The *aecp*(1) command now accepts a *−Read_Only* option, allowing files to be copied into a change specifically to insulate it from baseline changes. Such files must be uncopied before development may end. See *aecp*(1) for more information.

• The *aenrls*(1) command is now used *only* to convert pre-3.0 projects into post-3.0 projects. This is because the full branching support in 3.0 makes it more useful to create a new release of a project by ending development on the branch of the previous release and starting development of a new branch numbered for the new release. See the *Branching* chapter of the *User Guide* for more information.

Version 2.3

• The merging behaviour of the *aed*(1) command has changed. If any files require merging, it only merges. In this way, merged files are not lost in the rest of the output. Also, there are now command line options and user preferences so that you can select to only merge or only difference. See *aed*(1) and *aeuconf*(5) for more nformation.

• It is now possible to assign symbolic names to project deltas. This means that you may now recreate earlier project baselines by name.

• All commands which accept a **–Edit** option now check for most errors before commencing the edit. This avoids wasted edits in many error cases.

• Fuzzy file name matches are now used to improve the error messages from aecp, aerm, etc.

• Version number separators in project names are preserved across new releases. Particularly, you can use a minus ('−') between the name and the major version number.

• A new "copyright_years" project attribute has been added. This is a list of years maintained at integrate begin time, to automate the insertion of list of copyright years into copyright messages and documentation. There is a new ${Copyright_Years} substitution and the copyright years are also listed in the "aegis –list version" listing. See *aesub*(5) and *ael*(1) for more information.

• It is now possible to specify patterns for acceptable and unacceptable filenames in the project configuration file. See *aepconf*(5) for more information.

• Four more functions have been added to the report language: *length*, *split*, *substr* and *wrap*. See *aer*(5) for more information.

• The tests distributed with are now more stable on very fast hosts. See the environment variables section of *aeb*(1) for more information.

• The *lib/config.example* directory of the distribution now contains files with example portions of the project *config* file. May thanks to David R Shue <shue@ll.mit.edu> for this suggestion.

Changes made in the previous release included:

Version 2.2

This release of Aegis provides 3 of the most commonly requested features: support for heterogeneous development, support for a greater range of DMTs, support for user-defined reports.

• Aegis now supports heterogeneous development. Now you can be sure that your project not only always builds and tests sucessfully, but that it does so across a configurable set of system or hardware architectures. See the *Heterogeneous Development* secion of the *Tips and Traps* chapter of the User Guide for more information.

• Aegis can now cope with a wider range of Dependency Maintenance Tools (DMTs). It now has the ability to fill development directories with symbolic links to all files in the baseline which are not present in the development directory. This allows DMTs to assume all files are present below the current directory, allowing DMTs such as *cake* and *GNU Make* to be used. See the *Dependency Maintenance Tool* section of the User Guide and *aeb*(1) for more information.

• Aegis now has a report generator, so you can create your own reports. Many "canned" reports are included in this distribution; of particular interest to many will be the *File_Activity* report, which details currently active files. See *aer*(1) for more information.

• Aegis is now configured using a shell script called *configure*, distributed with the package. This shell script is generated using GNU Autoconf. See the *BUILDING* file for more information.

• The *AEGIS* environment variable has been renamed *AEGIS_PATH*, to bring it in line with the AEGIS_PROJECT and AEGIS_CHANGE environment variable names. The old name will keep working for some time, but aegis will warn you.

• Filename lengths are now configurable. The 14 character portability limit is still the default, but a higher limit is configurable for each project, up to the filesystem filename limit. See *aepconf*(5) for more information.

• It is now possible to specify that filenames must be within the minimum character set mandated by POSIX. The default is as before, to allow any printing character. See *aepconf* (5) for more information.

• Limits on the length of project names have been relaxed. Project names are now only limited by the filesystem filename limit.

• It is now possible to specify the command to run tests, allowing a project to use a specialized test facility, rather than be forced to use shell scripts. See *aet*(1) and for more information.

• The commands which accept the *–Edit* now preserve the edited text in the event of a failure.

• The commands which delete files now accept a *–Interactive* option, which causes them to prompt the user for confirmation of file deletion. This can be made the default by an appropriate setting of the aliases or individual users preferences files. See *aenfu*(1), *aentu*(1), *aecpu*(1), and *aeuconf* (5) for more information.

• The *aecp*(1) command now accepts directory names, allowing whole directory trees to be copied into a change. The *aecpu*(1) command now has a **–UNChanged** option which allows the unchanged files to be uncopied.

• The *aeb* command now accepts file names, allowing partial builds to be performed. See *aeb*(1) for more information.

• There is a new *aechown*(1) command to facilitate reassigning the developer of a change which is in the *being developed* state.

• It is now possible for project administrators to assign changes to specific developers. See *aedb*(1) for more information.

Version 2.1

• Can now ask for history to maintained for file generated by the build. This is useful for generating patch files.

Version 2.0

• A new command has been added to facilitate changing the name of a file as part of a change. See *aemv*(1) for more information.

• It is now possible to list the locks currently held. See *ael*(1) for more information.

• If no other defaulting mechanism is specified, aegis will now attempt to guess the project name and change number from the pathname of the current directory. This only works from within development directories.

• The *aenc*, *aeca*, *aerfail*, *aeifail* and *aepa* commands now accept a **–Edit** command line argument. See the relevant manual pages for more information.

• The *aenpr* command now understands the **–MAJor** and **–MINOr** options, allowing the initial version of a project to be something other than 1.0.

• The *aed* command now understands the **–ANticipate** option. See *aed*(1) for more information.

• It is now possible to list all the outstanding changes of a project, or of all projects. See *ael*(1) for more information.

Version 1.4

• Support has been added for systems without the *seteuid* system call, or those with crippled implementations.

• Most of the unimplemented command variants have been finished. These include *New Change Undo*, *Develop Begin Undo* and *ReMove PRoject*. Most notable of the exceptions is *-ANticipate* option for the *-CoPy_file* and *-DIFFerence* command.

• The User Guide has been added to, making it a little more complete. It still needs more work, sigh.

• The code to handle automounters has been made more robust.

• The command substitutions have been vastly improved, and are now documented.