

AccuRev 4.5.3 — Installation and Release Notes

June 16, 2007

AccuRev Installation Notes

- *Installing and Running AccuRev on an AIX Machine*
- *Upgrading from an Earlier Version?*
- *Getting Ready to Run the Wizard*
- *First-Time Installation — Client or Server*
- *Upgrading a Client (Non-Server) Machine*
- *Upgrading a Server Machine*
- *Non-Interactive Installation (Unix/Linux only)*

AccuRev Release Notes

- *Platform Support*
- *Compatibility with Previous Versions of AccuRev*
- *Compatibility with Versions of the Java Runtime Environment*
- *Notes on Cross-Links*
- *Internet Explorer 7 and Context-Sensitive Help*
- *Source Code Availability*
- *XML-Format Output of CLI Commands*
- *Principal Enhancements in Version 4.5.2*
 - *New Form of ‘In’ Command*
- *Principal Enhancements in Version 4.5.1*
 - *Support for Apple Macintosh*
 - *New Command: ‘accurev unmap’*
- *Principal Enhancements in Version 4.5*
 - *New Security Features*
 - *Version Control of File System Links*
 - *Cross-Links: Specifying the Backing Stream on a Per-Element Basis*
 - *Relationships between AccuWork Issue Records*
 - *New “Revert” Functionality*
 - *Per-Element Exclusive File Locking*
 - *Context-Sensitive Help for the AccuRev GUI*
- *Additional Enhancements in Version 4.5.1*
- *Additional Enhancements in Version 4.5*
- *Fixes in Version 4.5.3*
- *Fixes in Version 4.5.2*
- *Fixes in Version 4.5.1*
- *Fixes in Version 4.5*

AccuRev Installation Notes

Installation of AccuRev Configuration Management software on a machine involves running an installation “Wizard” program, after downloading it from the AccuRev Web site. One machine at your site should be installed as the “Server” machine — the one that runs the AccuRev Server process and hosts the AccuRev data repository. Any number of other machines can be installed as “Client” machines.

Each machine must be installed separately. There is no batch-installation capability, but you can perform a non-interactive installation, driven by a simple configuration file. (See *Non-Interactive Installation (Unix/Linux only)* below.)

We present an overview of the installation process here. The wizard aims to be self-explanatory, but if you’d like a detailed, step-by-step description of the installation process, see the “Quick Evaluation” chapters in *AccuRev Technical Notes*.

Installing and Running AccuRev on an AIX Machine

To install and run AccuRev Version 4.5.3 on an AIX machine, library file **libstdc++.a** must be present in a directory listed in environment variable LIBPATH. This environment variable must be set both in the environment of the installation Wizard and in the environment of AccuRev client and server programs.

Upgrading from an Earlier Version?

AccuRev Version 4.x use a different license key from previous versions. You need a new license key to upgrade successfully. Please contact support@accurev.com to obtain a new license key.

Upgrading a machine from an earlier AccuRev release is easy. You are not required to uninstall any existing software. You can install the new release “right on top of” the existing release — the installation Wizard lets you preserve your development data and configuration files. On the server machine — the one that runs the AccuRev Server process — installation of the new release automatically upgrades the data repository to the new release level.

Recommendation: Uninstall First on a Windows Client Machine

If you are upgrading the AccuRev client software on a Windows machine, we recommend that you uninstall the earlier version of AccuRev before installing AccuRev 4.5.3. Failing to first uninstall the earlier version has bothersome side-effects:

- The entry for the previous release is not removed from the **Add or Remove Programs** applet in the Windows control panel.
- The documentation entries under “AccuRev” in the Windows Start menu continue to point to the earlier version’s documentation; the AccuRev GUI’s **Help > Manuals** entries point to the new version’s documentation.

Upgrade Your AccuBridge Integrations, Too

As you upgrade your AccuRev installation, be sure check the AccuRev, Inc. Web site for upgrades to your AccuBridge integrations with issue-tracking systems and/or integrated development environments. Go to <http://www.accurev.com/download.htm> for AccuBridge installation packages.

Getting Ready to Run the Wizard

We assume that you've already downloaded a file containing the installation Wizard from the Downloads page of the AccuRev, Inc. Web site. If not, perform the download now, then return to these instructions.

If the file you downloaded is the Wizard itself (Windows: **AccuRevInstall.exe**, Unix: **AccuRevInstall.bin**), then you're all ready to run the Wizard.

If you downloaded a compressed package (a file with a **zip** or **gz** suffix), you must first extract the Wizard from the package. Many extraction tools are freely available. There are handy links to such tools on the Downloads page; and the **unzip** and **gunzip** programs are standard on many Unix systems.

Now that you're ready to run the Wizard, you can proceed to the appropriate section below. When we instruct you to "start the Wizard", keep in mind these tips:

- If you are using a window system (Microsoft Windows or the X Window System) and can see the file as an icon, double-click it to launch the wizard in graphical mode.
- If you're using a command shell, go to the directory where you downloaded the file, and type one of these commands:

```
AccuRevInstall           (on Windows)
sh AccuRevInstall.bin    (on Unix/Linux)
```

This always launches the Wizard in graphical mode on Windows. On Unix/Linux, it launches the Wizard in graphical mode only if the X Windows System DISPLAY environment variable is set. Otherwise, it launches the Wizard in console (text) mode.

First-Time Installation — Client or Server

Start the Wizard. You'll be up and running in just a few minutes!

Upgrading a Client (Non-Server) Machine

Start the Wizard. If you choose a Custom installation, be sure to select **No** when the Wizard asks, "Do you want to install new configuration files?"

Upgrading a Server Machine

1. Stop the AccuRev Server process. On Unix/Linux systems, use the command **acservctl stop**. The **acservctl** program is located in the AccuRev **bin** directory. On Windows systems,

you can use the Services applet to stop the AccuRev Server process, or the command **net stop accurev**.

2. Perform a full backup of the AccuRev data repository. For detailed instructions, see section *Backing Up the Repository* on page 3 of the *AccuRev Administrator's Guide*.
3. Start the Wizard. If you choose a Custom installation, be sure to select **No** when the Wizard asks, "Do you want to install new configuration files?". Similarly, select **No** when asked about replacing your license key file. Installation automatically upgrades the AccuRev data repository on the server machine. (There is no way to "downgrade" the data repository, for use with an earlier version of the AccuRev Server.)
4. Verify that the new AccuRev Server process has been started, by running the command **accurev info**. If you need to start the Server process manually, use the command **acservrctl start** on Unix/Linux systems. On Windows systems, use the Services applet or the command **net start accurev**.

Non-Interactive Installation (Unix/Linux only)

On a Unix or Linux, system, you can run the AccuRev installation program **AccuRevInstall.bin** as a non-interactive "console program" instead of as a wizard. Follow these steps:

1. Create an installation configuration file by copying the sample text below. This text is also available for download from the Downloads page of the AccuRev, Inc. Web site.
2. Revise the settings in the configuration file, as appropriate for the target machine.
3. Invoke the installation program as follows:

```
sh AccuRevInstall.bin -i SILENT -f <config-file-pathname>
```

You must use a full pathname to specify the installation configuration file. Use the following text as a template for the configuration file:

```
###
### AccuRev InstallAnywhere SILENT installation config file
###
#
# Unix/Linux usage:  sh AccuRevInstall.bin -i SILENT -f <config-file>
#

# this setting enables use of this config file; do not change.
INST_ROOT=true

# location of AccuRev installation directory (OK to already exist)
# installer creates subdirectories bin, doc, jre, etc.
USER_INSTALL_DIR=/opt/accurev

# un-comment exactly one of the following lines.
#CHOSEN_INSTALL_SET=Client
CHOSEN_INSTALL_SET=Full
```

```
# Client install: hostname of machine to run AccuRev Server process.
# Full install: this setting ignored, hostname of local machine used.
SERVER_HOSTNAME=myhost

# The traditional port for the AccuRev Server is 5050. But you can
# change it to any number. We recommend the range 5000 - 6000.
SERVER_PORT=5050

# Full install: full pathname of a valid license key file.
# installer copies this file into site_slice dir, as "keys.txt".

LICENSE_KEY_FILE=/path/to/license-key-file
```

Note: the SERVER_PORT setting does not work in this release. The installation program places port number **5050** in the **acclient.cnf** and **acserver.cnf** files (if it creates them), no matter what value you specify in the configuration file. After installation, you can edit these files manually to configure another port number.

AccuRev Release Notes

Please read these Release Notes to familiarize yourself with the changes in AccuRev for Version 4.5.3.

Platform Support

AccuRev Version 4.5.3 is not supported on these versions of Microsoft Windows, which were supported in prior releases:

Windows 98 Windows ME Windows NT

These versions of Windows do not support *junction points* in the file system, a feature that is used in the implementation of AccuRev directory links. See [Version Control of File System Links](#).

Prior versions of AccuRev continue to be supported on these versions of Windows.

Compatibility with Previous Versions of AccuRev

AccuRev Version 4.5.3 is fully compatible with AccuRev Version 4.5. But it is incompatible with all prior AccuRev releases. If you are upgrading from a pre-V4.5 release of AccuRev, you must upgrade the machine that runs the AccuRev Server software, along with all the machines that run AccuRev client software.

No explicit conversion is required to bring the AccuRev data repository from any prior level to the Version 4.5.3 level. When you install Version 4.5.3 on the machine that runs the AccuRev Server software (this machine also hosts the data repository), the repository will be upgraded automatically, if necessary.

Compatibility with Versions of the Java Runtime Environment

Version 4.5.3 is fully compatible with both Versions 1.3.x, 1.4.x, and 1.5.x of the Java 2 Runtime Environment (JRE).

Notes on Cross-Links

This section includes information that clarifies and supplements the basic documentation of AccuRev’s cross-links feature, introduced in Version 4.5.

Cross-Link Direction and Terminology

A cross-link is created in a workspace by the **Include from Stream** command (CLI: **incl -b**). The command name implies that a connection is being established *from* a specified backing stream *to* the workspace. But an existing cross-link is listed by the CLI command **lsrules** like this:

```
xlink <pathname> from <workspace> to <backing-stream>
```

That is, the direction of the cross-link “arrow” is the opposite of the direction implied by the “include from” command name. When describing a cross-link, we use this terminology:

- The workspace (or stream) where the cross-link has been created is the cross-link’s source stream.
- The designated backing stream is the cross-link’s target stream.

In CLI messages, “cross-link” is abbreviated to “xlink”.

Cross-Links and Stream Namespaces

Each AccuRev stream (including snapshot streams and workspace streams) provides a namespace: a set of pathnames to some or all of the depot’s elements. For example:

```
\\.doc  
\\.src  
\\.tools  
\\.doc\chap01.doc  
\\.doc\chap02.doc  
\\.src\commands.c  
\\.src\topaz.c  
\\.src\topaz.h  
\\.tools\cmdshell  
\\.tools\perl  
\\.tools\python  
\\.tools\tools.readme  
\\.tools\cmdshell\bash  
\\.tools\cmdshell\csh  
\\.tools\cmdshell\bash\end.sh  
\\.tools\cmdshell\bash\start.sh  
\\.tools\cmdshell\csh\end.csh  
\\.tools\cmdshell\csh\start.csh  
\\.tools\perl\add_cr.pl  
\\.tools\perl\remove_cr.pl  
\\.tools\python\setup.py  
\\.tools\python\vars.py
```

Since this set of depot-relative pathnames defines a hierarchy, it's often clearer to list the pathnames component-by-component, like this:

```
\\.\  
  doc  
    chap01.doc  
    chap02.doc  
  src  
    commands.c  
    topaz.c  
    topaz.h  
  tools  
    tools.readme  
  ...
```

To locate an element, AccuRev interprets its specified pathname component-by-component (just like the operating system does). The cross-links facility provides a way to make AccuRev switch namespaces in the middle of the pathname-interpretation process.

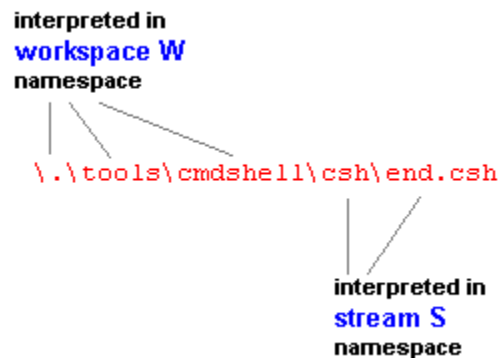
Note:

For example, consider this pathname:

```
\\.tools\cmdshell\csh\end.csh
```

And suppose you've created a cross-link at subdirectory **cmdshell**, with workspace **W** as the source stream and stream **S** as the target stream. AccuRev will process the pathname, component-by-component, as illustrated here:

- Pathname components up to *and including* the cross-linked component, are interpreted in the original (source stream) namespace.
- Additional pathname components, if any, are interpreted in the new (target stream) namespace.



Note that in workspace **W**, you continue to access the cross-linked element, subdirectory **cmdshell**, through its "local" name in the workspace's namespace. It's quite possible (but you don't need to know) that this element has a different name — even a different pathname — in the target stream:

```
\\.tools\shell_scripts  
\\.tools\common\scripts\  
\\.scripting  
... etc.
```

Pathname components below “cmdshell” are interpreted in the namespace of stream **S**, the target stream. For example, if script **end.csh** has been renamed in stream **S** to **topaz_exit.csh**, then that’s the name you must use in workspace **W**, as well:

```
\\.tools\cmdshell\csh\topaz_exit.csh
```

The File Browser and the CLI commands **stat** and **files** make this namespace-switching transparent: AccuRev shows you the element names and pathnames that will enable you to access the data from your current workspace or stream context.

Source Stream: Workspace vs. Dynamic Stream

The example in the preceding section uses a workspace as the “source stream”. The same pathname-interpretation principles apply if the source stream is a dynamic stream.

But the basic difference between workspace streams and dynamic streams affects the way cross-links work in them:

- In a dynamic stream, the **Include from Stream** command incorporates all changes from the target stream immediately. This reflects the fact that a dynamic stream inherits versions from its backing stream automatically and instantly.
- In a workspace, the **Include from Stream** command respects the workspace’s update level. That is, it incorporates only those changes that occurred in the target stream before the workspace’s most recent update. A subsequent **Update** command will bring in the more recent changes from the target stream.

Example: to see how cross-links work with a workspace’s update level, suppose that the following changes have been made in stream **topaz_mnt**:

- directory element **\\.tools\cmdshell\cmd** has been **Defunct**’ed
- directory element **\\.tools\cmdshell\csh** has been renamed to **\\.tools\cmdshell\c_shell**
- file element **\\.tools\cmdshell\c_shell\start.csh** has been edited

You use the **Include from Stream** command to create a cross-link from your workspace to stream **topaz_mnt**, at pathname **\\.tools\cmdshell**. The immediate change to your workspace depends on its update level:

- If the changes in stream **topaz_mnt** occurred after your workspace’s most recent update, you won’t see the changes immediately in your workspace: directory **cmd** will still exist, directory **csh** won’t be renamed to **c_shell**, and you won’t see the edits to file **start.csh**. But the status of these elements includes the **(stale)** indicator, showing that the changes are in the backing stream, waiting to be incorporated:

```
\\.tools\cmdshell\cmd topaz\1 (9\1) (backed) (xlinked) (stale)
\\.tools\cmdshell\csh topaz\1 (9\1) (backed) (xlinked) (stale)
\\.tools\cmdshell\csh\start.csh topaz\2 (9\4) (backed) (xlinked) (stale)
```

At this point, performing an **Update** will bring the changes into the workspace.

- If the changes in stream **topaz_mnt** occurred before your workspace’s most recent update, all those changes will be brought into the workspace immediately.

The procedure in the first bulleted paragraph can be described as “Include then Update”; the second bulleted paragraph’s case can be described as “Update then Include”. The final result is the same in both cases: the changes to the cross-linked elements in their new backing stream are incorporated into your workspace. We consider the second case to be an AccuRev best practice:

Best Practice:

Update your workspace before performing an **Include from Stream** command

If you **Update** first, other backing-stream changes won’t be “mixed in” with the **Include from Stream** changes during the next workspace update. Moreover, fully establishing the link from your workspace to the target stream will involve a single step (Include), rather than two steps (Include then Update).

Note: because it respects — but does not change — your workspace’s update level, **Include from Stream** more closely resembles the **Populate** command than the **Update** command.

Multiple Cross-Links: Chaining

AccuRev can traverse two or more cross-links in the same pathname. For example, you might use this pathname in workspace **W**:

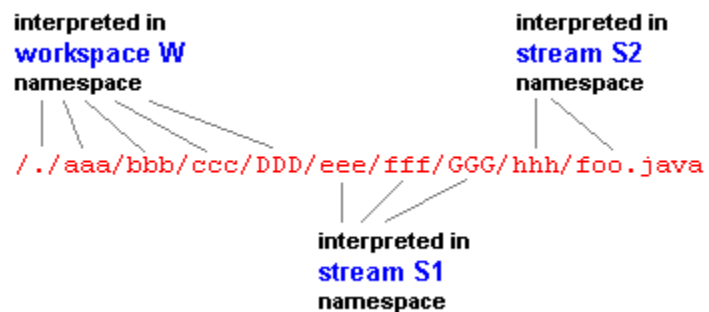
```
./aaa/bbb/ccc/DDD/eee/fff/GGG/hhh/foo.java
```

And suppose there are two cross-links:

- At subdirectory **DDD**, a cross-link from workspace **W** to stream **S1**
- At subdirectory **GGG**, a cross-link from stream **S1** to stream **S2**

As AccuRev traverses the pathname component-by-component, it interprets the components as illustrated here. As it progresses down the pathname, AccuRev also traverses a “chain” of cross-links:

- start in workspace **W**, then ...
- cross-link to stream **S1**, then ...
- cross-link to stream **S2**



“Chaining” of cross-links can continue to any number of levels. The same principle applies repeatedly: a cross-linked pathname component is interpreted in the *source* stream’s namespace; subsequent non-cross-linked components are interpreted in the target stream’s namespace.

But you must take care when “chaining” cross-links in this way. It is possible to create ambiguous configurations, which AccuRev handles by removing the affected elements. See [Cross-Link Overlaps](#) on page 12.

A special case of cross-link chaining occurs when you create a configuration in which two or more cross-links occur at the *same* pathname component. For example, consider again this pathname:

```
\\.tools\cmdshell\csh\end.csh
```

And suppose there is a chain of two cross-links at the same pathname component:

- At subdirectory **cmdshell**, a cross-link from workspace **W** to stream **S1**
- At subdirectory **cmdshell**, a cross-link from stream **S1** to stream **S2**

In workspace **W**, the subdirectory will continue to have its “original” name, **cmdshell**. But the subtree under the subdirectory will come from the stream **S2** namespace. By extension, you could chain any number of cross-links at the **cmdshell** component: **W > S1 > S2 > S3 > S4 ...** As above, the directory retains its “original” name in the workspace, and the workspace sees the directory’s subtree as it exists in the final target stream.

Double Vision: Seeing an Element Multiple Times in a Workspace

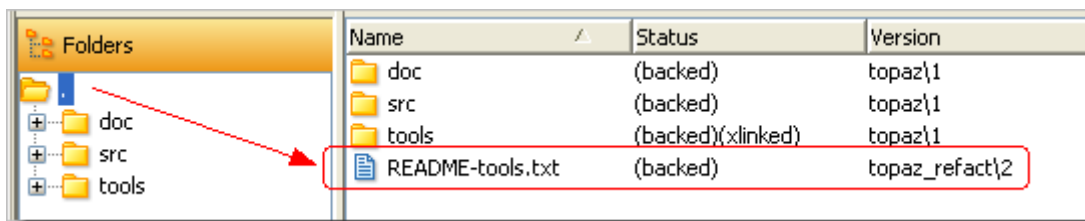
One consequence of AccuRev’s cross-link facility is that two (or more) different versions of the same element can appear at different pathnames in the same workspace or stream. We call this phenomenon double vision. This is not an error — at least, not from AccuRev’s perspective. Seeing the same element twice might be exactly what you intended, or it might signify that you’ve left some refactoring work unfinished.

Here’s an example: suppose you are tasked with doing some cleanup on the Topaz project’s development tree:

- Flatten out the subdirectories under **tools**.
- Move file **tools.readme** to the depot’s root directory, and rename it to **README-tools.txt**.
- Improve the source file comments in the **src** directory.

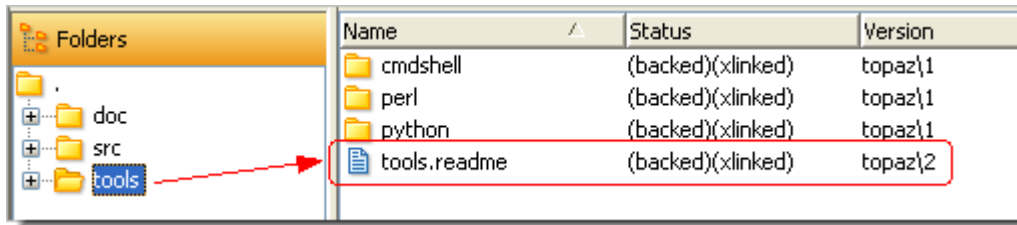
You perform this work in your workspace, named **topaz_refact**. But when the dust settles, you find that the programs in the **tools** subdirectory no longer work. You are not sure whether the problem is in the **tools** directory or the **src** directory. So you decide to “back out” your refactoring of the **tools** directory, by cross-linking to the known-to-work version of the **tools** directory in snapshot stream **topaz_2.3.9**.

Now, you have two different versions of the “README” element in your workspace! In your refactoring, you created a new version in your workspace, at pathname **\\.README-tools.txt**:



Name	Status	Version
doc	(backed)	topaz\1
src	(backed)	topaz\1
tools	(backed)(xlinked)	topaz\1
README-tools.txt	(backed)	topaz_refact\2

But your workspace now cross-links to the Release 2.3.9 version of the **tools** subdirectory, which contains the Release 2.3.9 version of the same element, at pathname `\\.tools\tools.readme`:



This case of double-vision is clearly an error, reflecting the fact that your refactoring work is still ongoing. In other cases, you might want two (or more) versions of a commonly used source file, say **topaz.h**, to appear in a workspace. Perhaps several different versions of the file are required, in order to build different executables using that file. Version skew is the executables' other dependencies might mandate the different versions of **topaz.h**.

Double Vision and the 'accurev name' Command

The **accurev name** command lists the pathname for a given element (specified by element-ID) in your workspace. It can also list the pathname for a specific version of an element, or the version in a specific stream:

```
accurev name -e 28
accurev name -v topaz_mnt -e 116
```

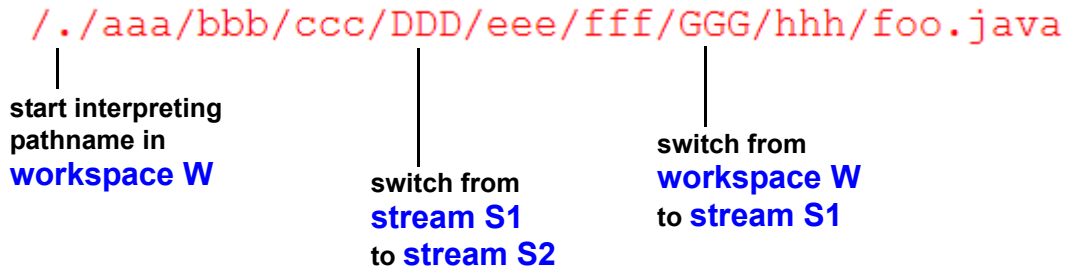
In a double vision situation, the **name** command can list all of an element's pathnames in a workspace or stream:

```
> accurev name -e 28 -v topaz_refact
\\.tools\tools.readme
\\.README-tools.txt
```

Cross-Link Overlaps

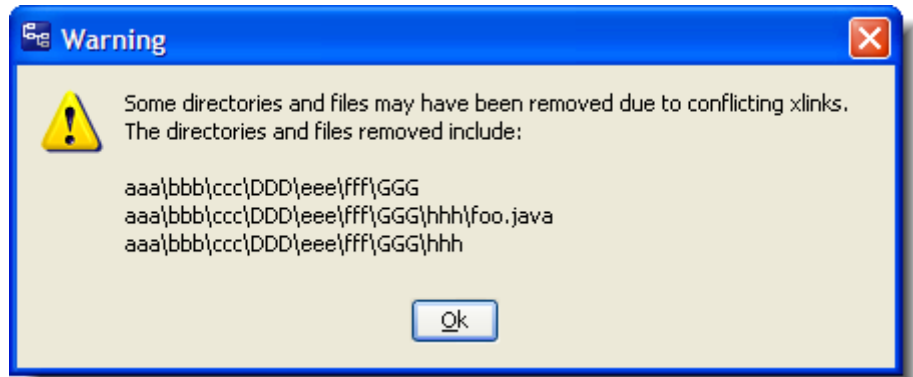
Section *Multiple Cross-Links: Chaining* on page 10 describes how a set of cross-links can define a "chain" of backing streams to be used at different components in a pathname:

```
././aaa/bbb/ccccc/DDD/eee/fff/GGG/hhh/foo.java
```



In this case, the second link in the cross-link chain (**S1** > **S2**) occurs at a *higher* pathname component, **DDD**, than the first link (**W** > **S1**, at component **GGG**). AccuRev recognizes this situation as a cross-link overlap.

When a workspace that has a cross-link overlap gets updated, AccuRev removes the subtree below the component where the first link was created.



Internet Explorer 7 and Context-Sensitive Help

The context-sensitive help system for the AccuRev GUI uses a Web browser to display help topics (your default browser or the browser configured with the `AC_BROWSER` preference). The current release of Microsoft's Internet Explorer 7 browser does not work correctly with the AccuRev GUI's help system: all calls to the help system from the GUI cause the overview "AccuRev Help System" topic to appear.

If you wish to use Internet Explorer 7 as your default Web browser, install another Web browser and configure it to be invoked by the GUI's help system. For example, if you obtain the Opera browser at www.opera.com, you might configure it for use with AccuRev as follows:

```
accurev setpref AC_BROWSER "c:\Program Files\Opera\Opera.exe"
```

This setting is stored in your personal AccuRev preferences file — for example, at `C:\Documents and Settings\derek\accurev\preferences.xml`.

Source Code Availability

In accordance with the GNU open-source policy, the source code for the AccuRev programs **acdifff** and **acdifff3** is available at this Web address:

```
http://www.accurev.com/download/open-source/acdiff\_source.zip
```

These programs implement text-file comparison and merging operations.

XML-Format Output of CLI Commands

Many of the commands in the AccuRev CLI tool, **accurev**, can produce output in XML format (`-fx` option). The documentation for this release does not describe the structure of the commands' XML-format output.

Principal Enhancements in Version 4.5.2

New Form of 'ln' Command

(Issue #11833) The AccuRev CLI command **ln** ("link") now features the option `-i`. The **ln -i** command lists all the link elements that point to a specified element — either directly or indirectly. See the **ln** reference page in the *AccuRev CLI User's Guide* for details.

Principal Enhancements in Version 4.5.1

The following sections summarize the product enhancements in AccuRev Version 4.5.1.

Support for Apple Macintosh

AccuRev Version 4.5.1 client software is supported on the Apple Macintosh computer (either Intel x86 or Motorola PowerPC), running OS X Release 10.0 or later. The AccuRev Server is not supported on the Apple Macintosh.

The AccuRev GUI should be configured to use Version 1.5.x of the Java Virtual Machine (“JVM 1.5.x”). Use the following procedure to establish this configuration:

1. Download the installation package at <http://www.accurev.com/download2.htm>.
2. Install the AccuRev client software to the default location: **Applications/AccuRev**.
3. Install JVM 1.5.x on the computer.
4. Open a Finder window, and navigate to folder **Applications/AccuRev/bin**.
5. Hold down the **Ctrl** key and click the **acgui** icon, then select **Show Package Contents** from the context menu.
6. Open the folder **Contents**.
7. Open the file **Info.plist** in a text editor (for example, TextEdit).
8. Search for the character string `JVMVersion`.
9. Modify the line that follows `<key>JVMVersion</key>`:

```
change:  <string>1.3*</string>
to:      <string>1.5*</string>
```
10. Save the file, and exit the text editor.

New Command: ‘accurev unmap’

(issue #11105) The **maintain** utility includes the command **unmap**, which applies temporary index maps to the actual index files in the AccuRev Server database. But you must stop the AccuRev Server before invoking **maintain unmap**. Now, you can achieve the same result without stopping the AccuRev Server, using the command **accurev unmap**. See *‘maintain’ Command Reference* on page 95 of the *AccuRev Administrator’s Guide* for more on the **unmap** command.

Principal Enhancements in Version 4.5

The following sections summarize the product enhancements in AccuRev Version 4.5.

New Security Features

AccuRev’s security capability has been strengthened in these areas: user authentication, group management, and the access control list (ACL).

New “AccuRev Login” User Authentication Scheme

Each registered AccuRev user has a username/password combination, recorded in the AccuRev repository. Prior to Version 4.5, a person’s AccuRev username was either the same as his operating-system username (default), or a username established with environment variable `ACCUREV_PRINCIPAL`. This scheme is still supported, and is now called the “traditional” user-authentication scheme.

Version 4.5 also features a new “AccuRev login” user-authentication scheme. In this scheme, a user must run the **login** command before using other AccuRev commands.

A new installation of type Full (client and server) is automatically configured to use the “AccuRev login” scheme. Upgrading an existing Full installation automatically retains the “traditional” scheme. (Note: page 6 of the *AccuRev Technical Notes* manual mistakenly shows the installation wizard prompting the user to select a scheme.) The user-authentication scheme can be changed subsequently, as described in the *AccuRev Administrator’s Guide*.

For details, see the **login** reference page in the *AccuRev CLI User’s Guide*. In the AccuRev GUI, use the **Tools > Login** command.

Group Management

In Version 4.5, a group’s members can include other groups, as well as individual users. See the **addmember** reference page in the *AccuRev CLI User’s Guide*.

More Flexible ACL Facility

Prior to Version 4.5:

- Each “permission” in the repository’s access control list (ACL) could apply only to a group, not to an individual user.
- Each stream or depot could have only one permission at a time; for example, creating a new permission for a stream automatically removed an existing permission for that stream.
- Permissions were not inherited; a permission on a stream affected *only* that stream, not any of the streams below it.

All these restrictions have been relaxed in Version 4.5. For example, access to stream **gizmo_integration** can be granted only to the individual users **tom**, **dick**, and **harry**. And that access control applies to the entire stream hierarchy below **gizmo_integration** (but can be overridden at lower levels).

For details, see the **setacl** reference page in the *AccuRev CLI User’s Guide*. In the AccuRev GUI, use the **Admin > Security** command (ACL subtab).

Version Control of File System Links

Prior to Version 4.5, AccuRev provided version control of files and directories (folders) only. Version 4.5 provides:

- file links, which work somewhat like hard links
- directory links, which work somewhat like symbolic links (Unix/Linux) or junctions (Windows).

For details, see the **ln** reference page in the *AccuRev CLI User’s Guide*. In the AccuRev GUI, link objects are created through a **Copy / Paste Link** command sequence.

Cross-Links: Specifying the Backing Stream on a Per-Element Basis

The set of streams and workspaces in each AccuRev depot is arranged in a stream hierarchy. Each workspace or stream has a parent stream (or backing stream). Data flows in both directions between child and parent:

- A workspace gets versions of elements from its backing stream with the **Update** command. A dynamic stream inherits versions of elements from its backing stream automatically.
- A workspace or dynamic stream sends versions of elements to its backing stream with the **Promote** command.

In Version 4.5, you can “go outside the lines” of the stream hierarchy for specified elements in specified workspaces or streams. For example, stream **widget_dvt** might be the backing stream of workspace **widget_dvt_john**. But you can make stream **exper_481** the backing stream for element **base.java** in workspace **widget_dvt_john**. That is, when the workspace is updated, a new version of **base.java** comes from stream **exper_481**, not from stream **widget_dvt**.

This “outside the lines” relationship between workspace **widget_dvt_john** and stream **exper_481** for a particular element is called a cross-link. It’s indicated by the status flag (**xlinked**). You can create cross-links ...

- from workspaces to streams
- from streams to higher-level streams
- from streams to lower-level streams

You can cross-link directory elements as well as file elements. This changes the backing stream for all the elements in the subtree below that directory.

Similarly, you can create cross-links for file system link objects (file links and directory links, described in *Version Control of File System Links* above.) Be sure to keep in mind that cross-links apply to the depot’s *stream* hierarchy, whereas file links and directory links apply to the depot’s *directory* hierarchy.

Cross-Links Are Read-Only

In Version 4.5, the **Update** command can traverse a cross-link to bring versions into a workspace from an “alternative” stream. But you cannot **Keep** or **Promote** a cross-linked element (or elements under a cross-linked directory).

Given this situation, the principal anticipated use of cross-links is to access alternative versions of build components. For example, suppose your team has been working to improve the performance of an application’s memory-management library. The interface to the library has remained the same — only the implementation has changed. To test the improvements, you could build the application twice ...

- ... in the regular manner, so that the build uses the latest versions of the memory-management library sources (say, in directory subtree **./src/mmgmt**).
- ... in a workspace that cross-links directory **./src/mmgmt** to stream **Widget_2.8**, a snapshot that contains an old release’s versions of the memory-management library sources.

Creating Cross-Links

You create cross-links using the include/exclude facility. Instead of asserting that you wish to “cross-link to” a particular stream, you assert that you wish to “include from” that stream. The element must already be present in both the workspace/stream you’re changing and the “include from” stream.

CLI example: To create a cross-link from workspace **widget_dvt_john** to stream **exper_481** for element **base.java**:

1. In workspace **widget_dvt_john**, go to the directory where element **base.java** resides.
2. Enter this command:

```
accurev incl -b exper_481 base.java
```

GUI example: To create a cross-link from workspace **widget_dvt_john** to stream **exper_481** for element **base.java**:

1. In a File Browser opened for workspace **widget_dvt_john**, navigate to the directory where element **base.java** resides.
2. Check the **Include/Exclude Mode** checkbox below the Folder pane.
3. Choose **Include from Stream** from the context menu of file **base.java** in the Details pane.
4. A popup window appears, listing all of the depot’s streams. Choose **exper_481**.

Note on Creating a Cross-Link

When you create a cross-link to a file using the CLI command **incl -b** in a workspace that has never been updated, the file might be deleted automatically:

```
Including /myfile .  
Removing "myfile" .  
Operation complete.
```

Updating the workspace will cause the cross-linked file to reappear in the workspace. This behavior does not occur if you use the GUI command **Include from Stream**.

Relationships between AccuWork Issue Records

AccuWork supports this kind of relationship between a pair of issue records:

- **Duplicate**: You can specify that issue record B duplicates issue record A, so that no work need be done on B. (Perhaps the same bug was reported twice.) Any number of records (B, C, D, ...) can duplicate a given record (A). Your issue database might be set up to require establishing a Duplicate relationship when an issue record's State field is set to the value **Duplicate**.

You cannot create “relationship chains”: if B duplicates A, you cannot make C a duplicate of B.

Issue record relationships can be viewed only in the AccuWork GUI. The edit-form tables that display issue relationships are not included when you export an issue record (for viewing or

printing). These tables are not included when you use the AccuWork command-line interface to dump the contents of an issue record.

When you create or delete a relationship between issue records, the change to both issue records is stored in the issue database immediately. There is no need to invoke the Save command on the issue record.(The Save button is not enabled in the edit-form toolbar.)

Viewing and Maintaining an Issue Record's “Duplicate” Relationships

The “duplicate” relationship is not symmetrical: “A duplicates B” is not the same as “B duplicates A”.

Accordingly, an edit-form field displaying a duplicate-type relationship consists of two tables: the top table shows issue records that are duplicated by the current record; the bottom table shows issue records that the current record duplicates.

Because of the “relationship chains” restriction, issue records can appear in only one of these tables, not both.

On the other hand, “A duplicates B” is the same as “B is duplicated by A”. This means that when you add an entry to the top table of one issue record, AccuWork automatically adds it to the bottom table of the other issue record (and vice-versa).

A Duplicate relationship field has its own toolbar, with these buttons:

Link Issue

Create a relationship link with another issue record. AccuWork prompts you to enter an issue record number.

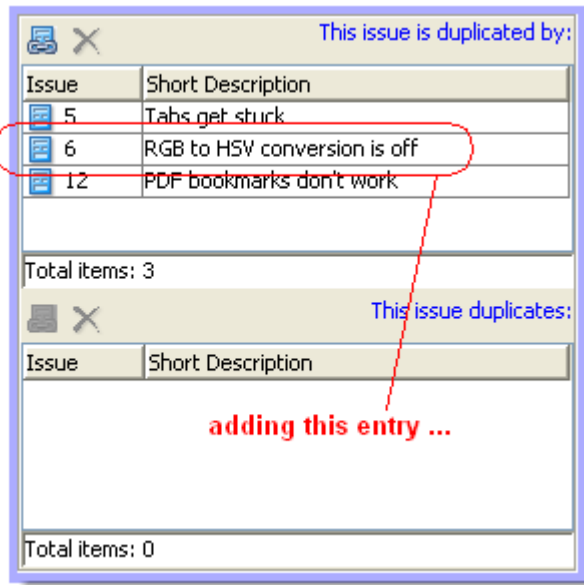
Remove Link

Remove the selected relationship link.

Note: In AccuWork's default issue database schema:

- There is a State field in the header section, one of whose values is **Duplicate**. (The Status value must be **Closed** to enable the setting of the State value to **Duplicate**.)

in issue record #1



Issue	Short Description
5	Tabs get stuck
6	RGB to HSV conversion is off
12	PDF bookmarks don't work

Total items: 3

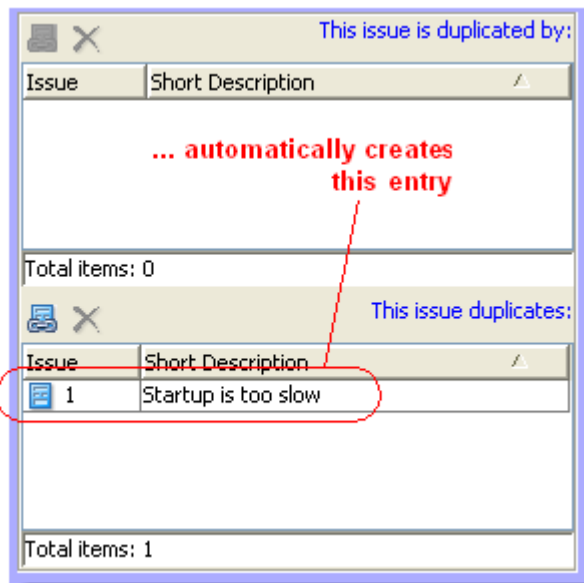
This issue duplicates:

Issue	Short Description
-------	-------------------

Total items: 0

adding this entry ...

in issue record #6



This issue is duplicated by:

Issue	Short Description
-------	-------------------

Total items: 0

This issue duplicates:

Issue	Short Description
1	Startup is too slow

Total items: 1

... automatically creates this entry

- There is a Duplicates field in the Relationships subtab.
- An edit-form validation specifies that when you set the State field's value to **Duplicate**, you must indicate which issue record is duplicated, by creating an entry in the Duplicates field.

Terminology: issues “in” a stream

Strictly speaking, the only objects that are “in” a stream are versions. It makes sense to describe a set of versions as being “in” a stream. And so, it makes sense to describe the set of versions in a change package as being “in” a stream. From there, we make the leap to describing the AccuWork issue record containing the change package as being “in” a stream.

New “Revert” Functionality

Prior to Version 4.5, the CLI **revert** command simulated an “undo promote” or “undo purge” operation on one or more elements in a stream by reinstating older versions of the elements (using the **co** command). Version 4.5 provides a more sophisticated approximation to “undo” — it uses a “reverse patch” algorithm to subtract the precise set of changes that the specified **promote** transaction had added to the element(s). Note that a **purge** transaction can no longer be reverted.

Version 4.5 also extends the “undo” capability to change packages. Each entry in a change package is similar to a patch to a particular element. When you revert a change package, AccuRev removes those patch-like changes from your workspace’s version of each element.

For details, see the **revert** reference page in the *AccuRev CLI User’s Guide*. In the AccuRev GUI, use **Revert** command in a History Browser tab or Stream Issues tab.

Per-Element Exclusive File Locking


Prior to Version 4.5, the exclusive file locking feature could be enabled at the workspace level or at the depot level. In Version 4.5, you can enable this feature on a per-element basis, through the dialog box for the **Add to Depot** or **Keep** command. In the CLI, the **anchor**, **add**, and **keep** commands support the **-E serial** option to enable exclusive file locking for the specified element(s).

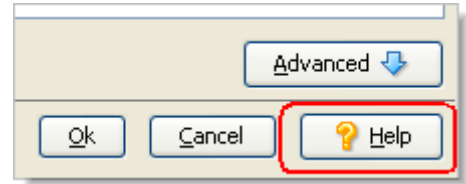
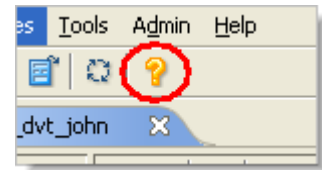
See also *Exclusive File Locking and Anchor-Required Workspaces* on page 3 of the *AccuRev CLI User’s Guide* and *Parallel and Serial Development* on page 27 of the *AccuRev Concepts Manual*.

Note: In this release, **Keep** fails if an element itself has been locked, but succeeds if locking applies at the workspace or depot level. This behavior may change in future releases.

Context-Sensitive Help for the AccuRev GUI

The AccuRev GUI now features a context-sensitive help system, delivered in HTML format. Users can invoke help in these ways:

- In a Browser or other data display, click the  button in the GUI's main toolbar to invoke help on that Browser or data display. Equivalently, press function key **F1**.
- In a dialog box, click the **Help** button.



Configuring a Web Browser to View Context-Sensitive Help Screens

The context-sensitive help system is implemented in HTML and JavaScript. Invoking help automatically launches a Web browser (or opens a new tab in an existing Web browser window). Any JavaScript-enabled Web browser can display AccuRev GUI help topics. By default, the operating system's default Web browser is invoked.

You can specify the complete pathname of the Web browser to be invoked with the AccuRev preference `AC_BROWSER`. Each user can use the `setpref` command to set this preference. For example:

```
accurev setpref AC_BROWSER "C:\Program Files\Opera\Opera.exe"
```

The preference is stored in file `preferences.xml` in the `.accurev` subdirectory of the user's home directory.

Note: On Windows machines, you *must* set this preference to configure Microsoft Internet Explorer as the Web browser to be used for context-sensitive help, even if it is your default browser. Example:

```
accurev setpref AC_BROWSER "C:\Program Files\Internet Explorer\IEXPLORE.EXE"
```

Additional Enhancements in Version 4.5.1

10734

During a **Promote** command, when an AccuRev-AccuWork integration prompts for an issue record, the user can click the **New Issue** button. The edit form that appears now includes scroll bars, if appropriate.

11160

The trigger parameters file passed to a `server_admin_trig` trigger script for a `mkstream` command now includes an XML element that specifies the type of stream being created. For example:

```
<streamType>regular</streamType>
```

The contents of the `<streamType>` element can be any of these strings: **regular**, **passthru**, **snapshot**.

11174

The command **accurev history** is now an alias for **accurev hist**.

Additional Enhancements in Version 4.5

2638

Each depot or stream can now have multiple ACL entries (“permissions”). A permission can now apply to an individual user, not just to a group. These changes makes it simpler to provide (or deny) access to, for example, four individual users or two separate groups.

2737

You can now invoke the Diff Against Most Recent Version, Diff Against Backed Version, or Diff Against Basis Version command on a multiple-file selection. A separate Diff tab opens for each file.

2758

You can now require that certain file types (for example ***.doc** files) use a serial development model instead of a parallel development model. Exclusive file locking (serial development) can now be set for individual file elements. When a new file is placed under version control, a **pre-create-trig** trigger script can determine the file type and, if appropriate, enable exclusive file locking for the newly created element.

3411

The CLI commands **defunct**, **undefunct**, **move**, **purge**, and **co** now support the **-c** option for specifying a comment string.

3426

The progress boxes for several GUI commands — **Update**, **Populate**, include/exclude mode commands, and File Browser searches — now include a **Run in Background** button. Only one task can run in the background at a given time. The progress of a background task is indicated by a status bar in the lower-right corner of the GUI window.



3486

You can now maintain relationships between pairs of AccuWork issue records. Such relationships are tracked in a new type of data field, “relationship”. An issue record's relationships are displayed in its edit form as a pair of tables (“child” and “parent”). See [Relationships between AccuWork Issue Records](#) on page 18 of these notes.

3495

In a Diff or Merge tab, the contributor panes for the current difference section contain red character highlights, indicating the first location in that line where the two contributors differ:

```
1 def get_topdir():
2     if 'ACUREV_BIN' in os.environ:
3         if = os.popen(os.environ['ACCUR:
4     else:
5         f = os.system('accurev info')
```

```
1 def get_topdir():
2     if 'AC_BINDIR' in os.environ:
3         if = os.popen(os.environ['AC_BINDIR
4     elif:
5         f = os.system('""accurev info""')
```

4197, 8878

A progress window now appears during execution of the **Populate** command and the several Include/Exclude mode commands, during the time that files are being copied from the repository to the workspace.

4252

An AccuWork query results table can now be exported in XML format or CSV (comma-separated values) format.

4304

AccuRev now supports link objects in the file system — both links to files and links to directories. See [Version Control of File System Links](#) on page 16 of these notes.

4463, 8743

The File Browser's Details pane can now be configured to include additional AccuRev-level details (element type, element-ID, development mode) and operating-system-level details (size, last-modified timestamp). The **Format Columns** command on the context menu of any column header changes the column configuration.

5214

The new “AccuRev login” user-authentication method provides improved security by storing a user's session file in encrypted format. See [New Security Features](#) on page 15 of these notes.

5260

The AccuRev GUI now has a context-sensitive help system. See [Context-Sensitive Help for the AccuRev GUI](#) on page 20 of these notes.

5858

The confusing “Update local password only” checkbox has been removed from the Change Password dialog.

5862

The Merge tool now supports four ways of navigating through difference sections: (1) visit all the difference sections, (2) visit all the conflict sections, (3) visit all the sections in which the “from” version differs from the closest common ancestor, (4) visit all the sections in which the file in the workspace differs from the closest common ancestor.

5877

Users can now Revert (undo the effects of) a change package, as well as a transaction. See [New “Revert” Functionality](#) on page 20 of these notes.

6108

In each stream, an element can now have a “customized” backing stream, rather than using the stream’s official parent stream. See [Cross-Links: Specifying the Backing Stream on a Per-Element Basis](#) on page 17 in these notes.

6622

The File Browser’s **Save As** command is now supported for directories.

7068

In the Change Palette, the **Send to Workspace** command can now be invoked on a multiple-file selection.

7107

The **setlocalpasswd** command, used only by the “traditional” user-authentication scheme, forced the user to type the new password on the command line. So did the **chpasswd** command, which is used by both the “traditional” and “AccuRev login” user-authentication schemes. This posed a security risk, since the password would appear in an operating-system process listing. Now, these commands can now prompt the user to enter the password.

7130

An ACL permission on a stream or depot can now be inherited by the entire stream hierarchy below it. By default, a permission is *not* inherited.

7231

The exclusive file locking feature can now be enabled at the element level. See [Per-Element Exclusive File Locking](#) on page 20 of these notes.

7235

The **diff -b -d** command now succeeds if one or more of the selected elements does not appear in the backing stream. Such elements are listed and bypassed by the file-comparison code.

7244

The GUI’s File Browser now includes support for the CLI **annotate** command.

7323

When it creates an HTML document, the AccuWork **Export Table** command now includes a “Total items” count at the end.

7377

The implementation has been improved for the CLI **revert** command and the GUI **Revert Transaction** command (and the new GUI **Revert Change Package** command). See *New “Revert” Functionality* on page 20 of these notes.

7499

Group names can now be much longer: up to 96 characters.

7559

When performing an **Update** on a client that uses a replica server, the order of operations has been changed, eliminating the need for two synchronizations. For details, see *The Update Command* on page 39 of the *AccuRev Administrator’s Guide*.

7876, 9941

Most command now refuse to run if the user is not authenticated. The exceptions are **help**, **login**, **info**, **secinfo**, and **mkuser**.

7882

Update performance was optimized in the case where new versions exist in the backing stream, but no files need to be copied from the repository to the workspace tree.

7930

Groups can now be nested; that is, a group can have both users and groups as its members.

8153

When the GUI exits, more of its state is now saved, to be restored the next time the GUI starts: “show including hidden” setting, which File Browser tabs are open, expanded/collapsed settings of Stream Browser nodes, Stream Browser “zoom” setting.

8219

The progress box now appears earlier in the execution of an Update.

8418

The functionality provided by **promote -N** is now available in the GUI after a change package is **Patch**’ed into workspace: a user can enable a **Promote** command to proceed by asserting that the version(s) to be promoted already contain the changes in that change package.

8540

The GUI’s **Clone** command has been renamed to **Clone Tab**.

8797

The GUI main menu now includes the command **View > Logs**, which enables the user to view logs created by **Update**, Include/Exclude mode commands, and **Populate**.

8807

In the GUI's Annotate tab, clicking a column now highlights all related changes, based on the value in that column — for example, all changes made by the same user, all changes made in the same transaction, or all changes made on the same date.

8851

The **maintain** utility now includes the command **sizes**, which lists the high water mark of each database file in the site slice.

8853

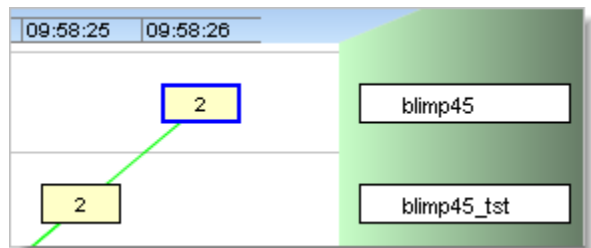
The **maintain** utility now includes the command **unmap**, which applies the temporary index maps to the actual index files.

8884

The **maintain** utility now includes the commands **export**, **export_indices**, and **export_csv**.

9025

The Version Browser now indicates — with a bold blue outline — the version that currently appears in the workspace or stream from which the Browser was launched.



9129

Synchronization of replica databases has been optimized through data compression.

9201

The AccuWork **Export Table** command now defaults to HTML as the output file type (**.html** filename suffix).

9207

When the **Properties** command is invoked on a stream or workspace in the Stream Version Browser, the information box now includes the version-ID of the element's version in that stream or workspace.

9233

The **promote** and **purge** commands both fire a **pre-promote-trig** trigger. For **purge**, the trigger fires only if the version is being purged from a dynamic stream, not from a workspace. The **purge** command now accepts the **-I** option, just like **promote**, with which user specifies one or more issue records.

The **pre-promote-trig** trigger is commonly used as part of the transaction-level integration between AccuRev and AccuWork. In this integration, the **promote** or **purge** transaction

number is recorded in one or more issue records. With either command, the user can now specify the record(s) with **-I**, to bypass the integration's interactive prompt.

9789

When a workspace with **Pending** elements is reparented, a message box appears to inform the user that promoting these elements will affect the new backing stream, not the former one.

9808, 9809

The messages in the **Delete** and **Defunct** dialog boxes have been clarified.

10795

The XML-format trigger parameters file created when a **purge** command fires a **pre-promote-trig** trigger now contains this XML element:

```
<action>purge</action>
```

Note: for a **promote** command, the trigger parameters file does not contain an `<action>` XML element.

Fixes in Version 4.5.3

7814

The GUI's **Diff** command can now compare a specified version with an arbitrary text file. Similarly, the **merge** command can now use an arbitrary text file as the “from” version.

11040

The **Update** command could fail when it attempted to incorporate multiple namespace changes — in particular, a “cycle” of changes that reestablished an element's original name.

11589

AccuRev did not correctly follow a chain of cross-links — for example, element **file.java** in stream C is cross-linked to the version in stream B, which in turn is cross-linked to the version in stream A.

12010

On some versions of Windows, the **ln** command failed to create a hard link.

12050

The installation wizard failed to add the newly installed AccuRev **bin** directory to the user's search path.

12083

Attempting to create a directory link (**ln -s**) failed if the target was a relative pathname (for example, `../my_sibling_dir`).

12104

The AccuRev GUI sometimes hung when the user clicked on a directory link element.

12129

Invoking the **Defunct** command on one or more elements with **(missing)** status caused the AccuRev GUI to hang.

12156

AccuRev did not correctly process a directory cross-link if the target directory was included in its stream with the **Include Directory Only** command (CLI: **incldo**). In the source directory of the cross-link, all subdirectories of the target directory were visible, not just the **Include**'d subdirectories.

12173

Creating an empty directory in the AccuRev GUI caused the GUI to hang if the **Add to Depot** option was selected.

12258

The **Send to Workspace** command sometimes caused an internal error in the AccuRev Server.

12327

The meaning of “read-only” for cross-linked directories required clarification:

In a cross-linked directory, you can create external objects, but you cannot **add** them to the depot.

12664

If an element was cross-linked in a parent stream, it was not possible to exclude the element in lower-level workspaces or substreams.

12694

If a directory was included in a stream by a cross-link, and two snapshots were made of that stream at different times, the **Show Diff by Files** command failed to find any differences between the two snapshots — ignoring the changes that had been made in that directory in the target stream.

12700

The **Update** command sometimes failed if it was incorporating a set of namespace-related changes to higher-level streams that included **Revert to Backed** (CLI: **purge**) operations on those streams.

12703

The performance of the **show streams** command was slow if a recursive search was performed on the depot's root stream.

12706

AccuRev CLI commands using the **-e** (element-ID) option did not always work with cross-links.

12712

The **Show Diff by Files** command sometimes produced garbled output when comparing the contents of directories that were cross-linked to snapshots.

12733

The performance of the code that determines multiple AccuRev user/group relationships has been improved considerably.

12746

The **In** reference page did not point out this fact: a directory link cannot be instantiated in a workspace or reference tree being accessed over the network from a machine whose operating system does not support symbolic links (Unix/Linux) or junction points (Windows).

Fixes in Version 4.5.2

11344

In AccuWork, a SetTabPermission or SetFieldPermission validation whose condition had the form “CUR_USER is a member of <group-name>” did not restrict access to the tab or field.

11455

AccuRev could not be successfully installed on Ubuntu Linux 6.10.

11546

On a Unix/Linux system, the **Add to Depot** command failed when invoked on an existing symbolic link to a directory. (Similarly, using the CLI **add** command on such a link failed if the link was referenced by a depot-relative pathname.)

11573

AccuRev software was not ready for the changes to Daylight Savings Time, which take effect in the year 2007.

11757

The **Populate** or **pop** command sometimes left a file element with **(modified)** status.

11829

On a Fedora Core 6 system, a shared library error occurred when starting the AccuRev GUI with the **acgui** script.

11835

The sample script **server_preop_trig.pl** contained a superfluous comment, which has been removed.

11863

Internet Explorer 7 does not work with the AccuRev GUI's context-sensitive help system. See [*Internet Explorer 7 and Context-Sensitive Help*](#).

11867

An AccuWork query whose condition used the “is a member of” operator performed slowly if there was a large number of groups.

11875

A timed-out login session for a user prevented several operations, including removing that user or changing that user’s username or password.

11900

A **Send to Change Palette** (or CLI **mergelist**) command sometimes failed if one or more elements to be processed was affected by an include/exclude rule. (The rule might be set in a stream other than the ones specified in the command.)

12023

The performance of the **Include** and **Exclude** commands has been improved for cases involving large numbers of elements.

Fixes in Version 4.5.1

5551

The **Promote** command produced a misleading error message, “Multiple versions specified for element”, in certain cases where the promoted version was not visible in the destination stream (for example, because of a basis time setting).

7247

The **patchlist** command failed if an element list was specified on the command line and the current working directory was not within a workspace.

8037

AccuRev could not be installed on a Linux machine running Fedora Core 4 or Fedora Core 5.

8880

After renaming a user with **chuser**, it makes sense to rename the user’s workspace(s) with **chws -s**. The **chws** command failed if the **server_admin_trig** trigger was enabled.

9346

Entering double-quote characters (for example, around a pathname that includes SPACES) in a user-specified Diff or Merge command in the AccuRev Preferences dialog caused a Java exception.

10209

Excluding one or more lower-level subdirectories from a workspace, then including their parent directory caused the contents of the excluded subdirectories to be copied to the workspace tree. AccuRev reported these files as having (**external**) status.

10333

AccuRev could not be installed on a Linux machine running SuSe 10.1.

10822

The **stat** command sometimes performed slowly when determining the default group of a dynamic stream.

10851, 10961

AccuRev displayed a confusing message when an exclusive file lock on an individual element prevented a **Keep** or **Send to Workspace** operation from proceeding.

10926

The context-sensitive help system did not define the context in which the GUI work area is empty (no tabs exist).

10958

When saving the results of a **Merge** operation on an XML file, AccuRev mishandled two-byte UTF-8 characters in a CDATA section.

10961

See #10851.

10974

The **diff -v** command sometimes failed with a “No element named” error message.

10976

Basic version-control operations in a directory failed if a higher-level directory had been renamed and promoted since the last update of the workspace.

11110

AccuRev mishandled some cases in which reparenting a dynamic stream caused elements in that stream to be renamed.

11178

AccuRev sometimes mishandled situations in which include/exclude rules were applied to pass-through streams.

11180

In AccuRev Professional, reparenting a workspace could mistakenly cause a “maximum stream depth exceeded” error.

11181

The AccuRev GUI failed to launch correctly on a display controlled by the Cygwin/X port of the X Window System.

11192

The AccuRev GUI sometimes generated an error when attempting to promote the versions in one or more specified transactions.

11246

If the AccuRev Server was using a **share_map.txt** file to enable access to workspaces from multiple client machines, AccuRev client performance was degraded on a machine that also used ClearCase dynamic views.

11249

Specifying the pathname of an attachment by typing or cut-and-pasting into the text field (instead of using the point-and-click navigation control) caused the AccuRev GUI to hang.

11300

Invoking the **Rename** command on a file in an anchor-required workspace failed to remove the file's read-only attribute.

11314

When a **server_admin_trig** trigger canceled execution of an include/exclude mode command, the AccuRev GUI mistakenly indicated that the command had succeeded.

11316

The **show streams** command executed slowly for a depot with a very deep stream hierarchy.

11352

The AccuRev Server sometimes terminated when contacted by a client from an older, incompatible release.

11443

Attempting to display a stream's set of active transactions would hang if the set was very large.

11445

Changes to the ACL permissions scheme, introduced in AccuRev 4.5, effectively changed the semantics of pre-existing permissions. Now:

- Pre-existing depot permissions will be inheritable by default, so that they work the same way as in prior releases.
- Pre-existing stream permissions will *not* be inheritable by default, so that they work the same way as in prior releases.

11450

ACL permissions were not being applied to time-based streams or snapshots. Now, ACL permissions are correctly applied to data structures, irrespective of time considerations.

11455

AccuRev could not be installed on a Linux machine running Ubuntu 6.10.

11560

If a workspace was not up-to-date, the status of a newly created element (**Add to Depot** command) was (**external**) instead of (**kept**).

11638

Memory leaks, which slowed performance, have been fixed in the Linux version of the AccuRev Server.

Fixes in Version 4.5

Version 4.5 resolves the following issues:

3304

If an element was activated in a workspace, placing an exclusive file lock on the file, and then the workspace was removed, AccuRev did not remove the lock. This prevented the element from being activated in a sibling workspace.

4776

The command **diff -b *** failed if there was a large number of entries in the current directory.

4850

The **merge** command failed for an element whose type in the workspace was **text**, but whose type in the backing stream was **binary**.

4976

Commands that require the user to specify a disk location could fail if a removable disk was removed during command execution.

5277

The built-in support for the Beyond Compare file-comparison tool did not take into account Version 2 of Beyond Compare.

5347

ACL permissions for users with passwords (“authuser”) sometimes conflicted with permissions for users without passwords (“anyuser”). See *More Flexible ACL Facility* on page 16 of these notes.

5475

An error occurred on an attempt to set the element type of a directory.

5536

The **cat -e** command failed when the user specified a non-existent element-ID.

5589

The **maintain vercheck** command failed when the user specified a non-existent depot.

6067

The **revert -t** command failed if, for one or more elements, the stream did not contain any version prior to the reverted version. Now, such elements are bypassed by **revert**; a message appears, informing the user that the elements should be purged or defuncted.

6487, 7647

When a user cancelled an **Update** command, the server task in the AccuRev Server process was not terminated. This could eventually cause the Server process to become unresponsive to client requests.

7111

An error occurred if the **keep** command was invoked on an element of type **text**, but the file in the workspace tree contained non-text characters.

7120

A pick-and-choose workspace was not initially empty if its backing stream was configured with one or more include rules.

7148

Updating a reference tree sometimes failed to bring in namespace-related changes: renamings and defunctings of elements.

7282

The File Browser did not display the status of an element in a workspace correctly after the element was changed with **Revert to Backed** in the parent stream. Now, the correct status is displayed: **(backed)(stale)**.

7301

The **show members** command sometimes listed users who were not actually in the group (as indicated by the **ismember** command).

7358

If an attempt to switch user identities failed (because of incorrect contents of the user's **authn** file), both the CLI and GUI mistakenly reported that the identity had successfully been switched.

7487

When comparing a workspace stream to a dynamic stream or snapshot, the **diff -a -i -v -V** command did not correctly determine the versions in the workspace stream.

7535

In the Version Browser, a version's status information (displayed on mouse-over) was displayed incorrectly if the version's comment string was very long or contained a single-quote character.

7647

See #6487.

8099

The **Diff Against Backed Version** command failed if no version of the element existed in the backing stream.

8175

The **files** and **stat** commands could report different statuses for an element if the backing stream had a time basis.

8250

In the Change Palette, checking the **Use latest** checkbox for some elements, but not for others, could cause the AccuRev Server to fail.

8274

Upgrading an existing AccuRev installation on a Windows machine caused multiple entries to be placed in the Windows **Add or Remove Programs** applet. See *Recommendation: Uninstall First on a Windows Client Machine*.

8452

In a History Browser (or Active Transactions) tab, the **Diff** command could be invoked on binary files.

8454

The **Promote** command, executed in a workspace that had been renamed, caused the AccuRev Server to fail.

8546

If the change-package-level integration between AccuRev and AccuWork was enabled, promoting a version from a workspace to its backing stream did not cause a **pre-promote-trig** trigger to fire.

8625

The ACL facility was inflexible and contained inconsistencies. The enhanced ACL facility addresses these issues. See *New Security Features* on page 15 in these notes.

8687

The GUI mishandled a password that contained the asterisk (*) character.

8700

If **Promote** was invoked on an issue in a Stream Issues tab, with Show Active checked, a problem occurred if an element had previously been removed from the issue's change package: the **Promote** command succeeded, but the issue did not disappear from this listing of the stream's "active" issues.

8701

If a user invoked the **Login** command to change his AccuRev user identity, but then changed his mind and clicked **Cancel**, the user identity would change anyway.

8736

Rotation of the AccuRev Server log file was not working on Windows machines, due to a "file busy" condition. This functionality has been removed: the AccuRev Server log file is not rotated automatically on a Windows server machine.

8748

The **Update** command worked incorrectly after a workspace had been reparented to a snapshot.

8754

The **Keep** command that ends a **Merge** operation always changed a file's EOL setting to "Platform EOL", ignoring the EOL settings of the file and the workspace.

8759

The **New File** and **New Folder** commands allowed an object to be created, and added to the depot, with a name starting and/or ending with one or more SPACE characters. SPACE characters are now trimmed from the start and end of the user-specified name.

8798

The **is member of** and **is not member of** operators in an AccuRev query did not work correctly.

8814

A database-level error occurred if a replica synchronization occurred concurrently with the creation of a new database item: workspace, depot, user, etc.

8819

File status were sometimes incorrect in reference trees that had not been updated recently.

8893

The Anchor Required dialog box was too small to show all data columns.

8920

If a new version of a file of type **ptext** was created with **Keep** on a Unix/Linux client, AccuRev subsequently reported the file's status as **(modified)**.

8979

In a dynamic stream, promoting a stranded element by its element-ID failed if the element had a **(defunct)**-status twin in the stream.

8989

The *AccuRev Administrator's Guide* did not include the fact that the XML-format trigger parameters file for a **pre-promote-trig** trigger includes a **<stream2>** parameter, naming the **Promote** destination stream. This parameter is not present in the text-format trigger parameters file.

9002

The **server_admin_trig.pl** sample script file did not account for the **mkreplica** and **rmreplica** commands, which do fire this trigger.

9020

A deleted user still appeared in user and group listings.

9076

If a depot's name started with a wide capital letter, stream names that also began with that letter were truncated in the Stream Browser graphical display.

9110

An "all" ACL permission did not properly override a "none" permission on the same data structure.

9119

When the user preference "Display of element names in tables" was set to "single column", the items in the column were still sorted on the simple filename, not on the complete pathname.

9121

On a Linux client, the **mergelist** command failed if one of the specified streams was a workspace stream.

9166

On a Windows client, the **start** command placed UNIX-style directory separators in the value of the **ACCUREV_TOPDIR** environment variable.

9180

If any stream in a workspace's backing chain had a time basis, attempting to **Anchor** an element in the workspace failed.

9385

The Stream Browser performed slowly when individual elements were selected with **Ctrl-click** in a stream's Default Group subwindow, especially if the default group contained many elements.

9571

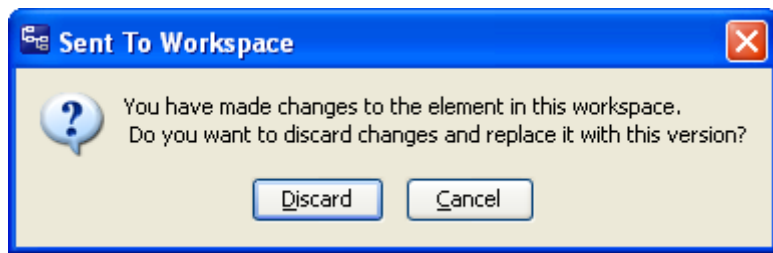
Promoting a version from a workspace that was renamed but not updated could cause the AccuRev Server to fail.

9636

The **diff** command failed when a large number of files were specified on the command line.

9844

The **Send to Workspace** command did not work correctly if the file in the workspace had **(modified)** or **(member)** status. Now, the user is asked whether the file in the workspace tree is to be overwritten with the selected version:



9921

The **diff -v -V -a -I** command, which lists all the differences between two specified streams, did not always provide a complete listing.

9920, 10066

The **mergelist -s -S** command, which lists all files whose changes need to be propagated from one stream to another stream, did not always provide a complete listing.

10023

In the transaction-level integration between AccuRev and AccuWork, specifying multiple issue records when prompted by the **Promote** command (or with the **promote -I** option) was not handled correctly: an issue's **affectedFiles** field might not be updated with the transaction number, or might be updated redundantly.

10184

If a custom TkDiff command line was specified in the AccuRev Preferences dialog, the invocation would fail, due to argument-quoting problems.

10380

(found in V4.5 beta-test) The GUI **Login** command did not work on a Solaris client host.

10499

If a **chstream** command did not run to completion, mutual-exclusion locks were not cleaned up properly.

10568

The help facility for the **accurev** command-line tool omitted **remove wspace** and **reactivate wspace**.

10600

(found in V4.5 beta-test) The presence of a cross-link in a stream caused Change Palette operations involving that stream to fail.

10662

Concurrent execution of the **setacl** and **keep** commands could cause AccuRev database-level problems.

10835

In a File Browser tab open on a dynamic stream or snapshot, a meaningless last-modified time was displayed for directory elements. Now, this field is left blank.

10869

The AccuRev installation wizard was not offering “Full” as the default installation mode.

10907

A workspace with no active files did not have the same contents as a snapshot whose basis time was the same as the workspace’s update level.

10937

(found in V4.5 beta-test) Opening a new Stream Browser tab took considerably more time than in prior versions of AccuRev.