

# AccuRev

**AccuRev®**  
**GitCentric™**

## **Installation and Release Notes**

**Version 2013.2**

*Revised 10/17/13*

---

+1. 978. 341. 5367  
sales@accurev.com  
www.accurev.com

AccuRev, Inc.  
300 Baker Avenue, Suite 205  
Concord, MA 01742-2131

**Copyright © AccuRev, Inc. 1995–2013**

**ALL RIGHTS RESERVED**

This product incorporates technology that may be covered by one or more of the following patents: U.S. Patent Numbers: 7,437,722; 7,614,038.

**TimeSafe** and **AccuRev** are registered trademarks of AccuRev, Inc.

**AccuBridge**, **AccuReplica**, **AccuWork**, **AccuWorkflow**, **Kando**, **GitCentric**, and **StreamBrowser** are trademarks of AccuRev, Inc.

All other trade names, trademarks, and service marks used in this document are the property of their respective owners.

# Table of Contents

<b>Introduction.....</b>	<b>1</b>
Highlights of this Release .....	1
<b>GitCentric Installation Notes.....</b>	<b>3</b>
Introducing GitCentric 2013.2 .....	3
Preparing for Installation .....	3
Generic Names Used During Installation .....	3
Summary of Steps .....	4
Linux Installation .....	5
Existing Linux Servers.....	5
To Install Linux.....	5
Git Installation.....	6
AccuRev Installation.....	6
GitCentric Installation.....	8
Initial GitCentric Log In .....	10
SSH Key Generation.....	10
Log In and Self-Register.....	10
Register with GitCentric .....	12
GitCentric AccuRev Trigger and Kando.xml .....	13
Enabling/Disabling Code Review .....	15
Upgrades from Kando 2012.1 .....	15
Test the GitCentric Installation .....	15
Test the GitCentric Bridge Connection.....	15
Proceed to the GUI.....	16
Displaying the User’s Help .....	16
Troubleshooting .....	16
If You Cannot Log In.....	16
General: Testing the Installation .....	17
General: The Tomcat Logs.....	17
General: Servers.....	18
General: Settings.....	18
LD_LIBRARY_PATH: psql shared library error.....	19
ASSIGN_USER_PRIVILEGE and CC_USER.....	19
Debug Settings .....	19
<b>GitCentric™ Release Notes .....</b>	<b>21</b>
Changes in Release 2013.2 .....	21
Changes in Patch Release 2013.1.01 .....	23
Changes in Release 2013.1 .....	23
Known Issues .....	23
Gerrit Issues .....	25

Changes from Kando 2012.1 .....	25
<b>Upgrading from Kando 2012.1.....</b>	<b>27</b>
Changes Since Kando 2012.1 .....	27
Ensure that Repos Have Been Moved.....	27
Update Clone Configs.....	27
Migrating Kando Named-ACLs to GitCentric ACLs .....	28
If You Have Only One 2012.1 Named ACL.....	28
If You Have Unique 2012.1 Named ACLs .....	28
If You Have Multiple 2012.1 Named ACLs .....	28

# Introduction

AccuRev GitCentric provides the flexibility of Git and the security of AccuRev.

## Highlights of this Release

AccuRev GitCentric Release 2013.2 provides new features and enhancements, and it addresses issues reported in previous GitCentric releases. Highlights of GitCentric Release 2013.2 include:

- Support for Gerrit Code Review replication.
- The ability to turn off code review from the GitCentric UI.
- Support for Git hooks.
- Upgrade to Gerrit Code Review 2.6.

For details about this release, including a comprehensive list of changes from previous Kando and GitCentric releases, see [GitCentric™ Release Notes](#) on page 21.

To install GitCentric, see [GitCentric Installation Notes](#) on page 3.



# GitCentric Installation Notes

## Introducing GitCentric 2013.2

This chapter describes how to install or upgrade GitCentric software at your site.

GitCentric Release 2013.2 is intended for new customers and customers currently using a Kando 2012.1 or GitCentric 2013.1 release.

If you are upgrading from Kando 2012.1, you must review [Upgrading from Kando 2012.1](#) on page 27 to determine whether or not you should undertake the upgrade at this time, and if so, what manual steps you must take after installation.

For a high level summary of new features in Release 2013.2, see [Highlights of this Release](#) on page 1. For a detailed list of known issues, see [GitCentric™ Release Notes](#) on page 21.

## Preparing for Installation

GitCentric requires the following for a successful installation. **Note:** Other components such as Git may have their own prerequisites.

Table 1. Prerequisites and Platform Support

Component	Supported Configuration
Operating System	Linux 2.6 or later for GitCentric server. Compatible Ubuntu versions are 10.10 and later, RedHat 6.2 and later. Other versions can be supported, but they have not been certified. Web browsers and Git can be run from any platform.
Browsers	Chrome, Firefox, MSIE 8, MSIE 9, and MSIE 10 are supported. <b>Note:</b> The appearance of some pages in MSIE 8 may differ from other browsers, but all functionality should work correctly.
Git	Version 1.7.4.1 or later. (For Windows operating systems, AccuRev suggests that you install msysGit.)
Java	Java 1.6 or later. (The AccuRev installation installs Java at <code>&lt;ac_home&gt;/j re</code> . You may need to update your \$PATH to point to this.)
AccuRev	Version 5.6 or later. (Note that AccuRev installations have their own prerequisites such as Perl for running triggers.)

## Generic Names Used During Installation

The setup process involves a number of configurable elements. For simplicity and clarity, the instructions consistently use a set of generic names, using *this font*.

*Tip:* Use the **Name at Your Site** column in the following table to record the names you plan to use.

Table 2. Generic names used during installation

Generic Name	Name at Your Site	Description
<code>&lt;gc_installer&gt;</code>		Linux admin OS account for installing AccuRev, GitCentric, Git, etc. If you have an existing AccuRev server on Linux, this should be the account you used to install it, such as “ <code>acserver</code> ”. If you are upgrading from an existing Kando 2012.1 installation, this may be user “ <code>git</code> ”. If you are starting from scratch, you can create a new one such as “ <code>gcadmin</code> ”.
<code>&lt;gc_sync_user&gt;</code>		This is the AccuRev “service account” (typically “ <code>gcSyncUser</code> ”) that GitCentric uses to automatically keep Git and AccuRev in sync. This is a member of the “ <code>scm_bridg_group</code> ” group, which is set to be the value of “ <code>ASSIGN_USER_PRIVILEGE</code> ” in the <code>acserver.cnf</code> configuration file on each AccuRev server controlled by GitCentric. For any single mapped repository, there is an AccuRev user who is the service account. <b>Note:</b> The <code>&lt;gc_sync_user&gt;</code> should NOT be a member of an AccuRev administrator group.
<code>&lt;gc_home&gt;</code>		Location where GitCentric is installed. This is typically: (Linux) <code>~&lt;gc_installer&gt;/AccuRevGitCentric</code>
<code>&lt;ac_home&gt;</code>		Location where AccuRev is installed. For GitCentric, this is typically: (Linux) <code>~&lt;gc_installer&gt;/accurev</code> (but sometimes <code>/opt/accurev</code> or elsewhere)
<code>&lt;tomcat_install&gt;</code>		Location where the Tomcat web server is installed. For GitCentric, this is typically: (Linux) <code>&lt;ac_home&gt;/WebUI/tomcat</code>
<code>&lt;db-port&gt;</code>		Port number that the database server runs on. Defaults to <b>5075</b> .
<code>&lt;db-admin-name&gt;</code>		Name of the database superuser. Defaults to <b>postgres</b> .
<code>&lt;db-admin-pass&gt;</code>		Password for <code>&lt;db-admin-name&gt;</code> .

## Summary of Steps

The procedures in this chapter will cover the details for performing the following steps:

1. The GitCentric server needs to run on a Linux host. See [Linux Installation](#) on page 5.
2. Create (or identify) a `<gc_installer>` OS administrative user, and log in to this account to perform the rest of the installation. You will also continue to use this account to start and stop servers. See [Linux Installation](#) on page 5. This user should also have a corresponding AccuRev account that is a member of the Admin group.
3. Download the components you will need to install (AccuRev installer, Git installer, GitCentric installer, AccuRev and GitCentric license keys, etc.). See [Linux Installation](#) on page 5, [Git Installation](#) on page 6, and [AccuRev Installation](#) on page 6 for download instructions.
4. As OS user `<gc_installer>`, install Git 1.7.4.1 (or higher). See [Git Installation](#) on page 6.
5. As OS user `<gc_installer>`, install the AccuRev (5.6 or higher) Server, Web UI server, and client. See [AccuRev Installation](#) on page 6.

6. As OS user `<gc_installer>` install GitCentric. See [GitCentric Installation](#) on page 8.
7. Restart Tomcat.
8. Configure GitCentric and AccuRev. This consists of at least defining an "ASSIGN\_USER\_PRIVILEGE" role in the AccuRev server configuration file (acserver.cnf) and its corresponding `<gc_sync_user>` user and `scm_bri dge_group` group. Then configure the Kando.xml and server\_master\_trig.pl files in the AccuRev `triggers` directory. See [GitCentric Installation](#) on page 8.
9. Bring up the GitCentric Web UI and log in. See [Initial GitCentric Log In](#) on page 10.

## Linux Installation

Currently the GitCentric server is supported only on Linux operating systems.

### Existing Linux Servers

If you have an existing Linux machine that you wish to use as a host, you can install GitCentric on that system.

1. Create an administrative `<gc_installer>` user and log in to this account. This is the account you will use to install AccuRev and the GitCentric software.
2. Proceed to [Git Installation](#) on page 6.

### To Install Linux

If you need to install a Linux operating system from scratch, check the AccuRev support matrix for the current AccuRev release, such as:

[http://www.accurev.com/accurev\\_support\\_matrix\\_6\\_0\\_1.html](http://www.accurev.com/accurev_support_matrix_6_0_1.html)  
and choose a Linux version.

After installation, you may need to add some support packages and do some configuration as explained later in this document.

1. Go to the download page for the Linux distribution you have decided to use and install the latest stable release. For example, if you decide to use Ubuntu, go to <http://www.ubuntu.com/download>.
2. Once you have installed Linux, you will be prompted to set up a username and password. Create a `<gc_installer>` user and log in to this account. (This assumes that you do not already have an existing Linux AccuRev admin user on your network such as "acserver".) This is the account you will use to install AccuRev and the GitCentric software. This document uses "gcadmi n" as an example installer account.
3. As you use your new system, you will probably find that you need to install extra packages. If you are a Windows user, read documentation about the "apt-get install" command on Ubuntu (or "yum" on Red Hat or CentOS). Basic Linux commands and procedures are beyond the scope of this document, so be prepared to do some web searching. Most basic questions can be answered by searching for whatever task you want to perform, and appending "Ubuntu" or "Red Hat" or "CentOS", etc.

## Git Installation

Before proceeding with a GitCentric installation, you must install Git version 1.7.4.1 or higher.

On an up-to-date Linux machine where you have root access and a connection to the Internet, this can be as simple as:

```
> sudo apt-get install git-core
```

or

```
> sudo yum install git-core
```

However, if you are in a highly secure network environment without root or Internet access, and particularly if you are using an old OS version, getting Git installed can be more challenging. You should attempt to complete this step well in advance of installing GitCentric, so that you can:

- Discover any limitations on access or dependencies.
- Identify a source or pre-built version of Git that will install on your machine (start with <http://www.git-scm.com>).
- Identify any dependencies that you may have difficulty addressing due to access or policy restrictions. Currently, installing Git may encounter the following dependencies:

```
libidn. so. 11  
ld-linux. so. 2  
libm. so. 6  
libpthread. so. 0  
librt. so. 1  
libgcc_s. so. 1  
libstdc++. so. 6  
libc. so. 6  
linux-gate. so. 1
```

- Work with your IT department to address any of the above issues.

## AccuRev Installation

You need to have an AccuRev (5.6 or higher) server (including the AccuRev Web UI tomcat server and the AccuRev PostgreSQL server) on the Linux machine to act as the GitCentric server. If you already have AccuRev 5.6 or higher installed on a Linux host that you can use for GitCentric, just review the fresh-install instructions below to ensure that you have everything that you need.

1. Log in to Linux as the *<gc\_installer>* administrative user (which you should have created above in [Linux Installation](#)).
2. Ensure that you have a "Downloads" directory (for example, "~/Downloads"). If you do not, create one:

```
> cd  
> mkdir Downloads
```

3. Get a copy of the full 5.6 (or higher) AccuRev installer from the AccuRev web site:

<http://www.accurev.com/download.html>

The full installer is for installing the AccuRev server (which also installs the AccuRev database), AccuRev WebUI server (which installs a Tomcat web server), and a local copy of the AccuRev client.

4. Follow the procedures in the AccuRev *Installation and Release Notes* to do a full install as the OS *<gc\_installer>* user to install the AccuRev server, AccuRev Web UI server, database, and AccuRev

client. Although you can install these to wherever you want, the default is to install them under the `<gc_installer>` home directory (for example, `-<gc_installer>/accurev`).

**Note:** If you choose to install in a different directory such as the root-owned `/opt` directory, you will need to be able to perform a `sudo mkdir` command, and then `chmod` and `chown` for `<gc_installer>`. You cannot install AccuRev or GitCentric as root.

5. After installing, make sure that the AccuRev server, AccuRev DB Server, and AccuRevTomcat (Web UI) server are running.

**Note:** Make sure that the AccuRev license file you acquire includes a license for GitCentric as well as for AccuRev.

6. When the installation is complete, let AccuRev guide you through the Quick Setup Wizard and create a user account for the administrative `<gc_installer>` admin user.
7. Use the user and group options under the **Admin->Security** menu in the AccuRev GUI, or the **mkuser**, **mkgroup**, and **addmember** CLI commands, to create AccuRev users and groups to work with GitCentric.

Either now or later, you can also choose to create additional AccuRev users who you want to have accounts on the GitCentric AccuRev server.

The following table summarizes the minimal AccuRev users and groups that you should define when first installing GitCentric. The specific names shown are suggestions; they are not required to be used.

*Table 3. AccuRev User and Group Accounts Required by GitCentric*

Name	User or Group	Member of Group	Description
<code>admin</code>	group	N/A	Standard AccuRev group for administrative users.
<code>scm_bridge_group</code>	group	N/A	Group of users on each AccuRev server that the GitCentric bridge uses to keep Git branches in sync with AccuRev streams. Typically consists of at least user <code>gcSyncUser</code> , and is assigned to <code>ASSIGN_USER_PRIVILEGE</code> in each AccuRev server's <code>acserver.cnf</code> file. Different users with different access rights can be assigned to this group for security reasons. Value does not necessarily need to be <code>"scm_bridge_group"</code> .
<code>gadmin</code>	user	<code>admin</code>	Administrative login account on the GitCentric AccuRev server. Value does not necessarily need to be <code>"gadmin"</code> .

Name (cont'd)	User or Group	Member of Group (cont'd)	Description (cont'd)
gcSyncUser	user	scm_bri dge_group	The “service account” user on each AccuRev server that is mapped to branches in a Git repository. Member of the <code>scm_bri dge_group</code> group, which is assigned to the <code>ASSIGN_USER_PRIVILEGE</code> setting in each AccuRev server’s <code>acserver.cnf</code> file. (See <a href="#">Step 8</a> below.) Should NOT be a member of the <code>admin</code> group. You need to have at least one service account user, and you can have multiple such users with different privileges for different streams. This user's AccuRev Element ACL (EACL) permissions determine which files can be edited in the GitCentric repository. Files that cannot be edited do not appear at all in the GitCentric repository. You will set up ACLs after the installation is complete. This process is referenced in the GitCentric <i>User’s Help</i> , and is documented in the AccuRev user documentation.

- On each AccuRev server that will have streams mapped to Git branches (which may or may not include the GitCentric AccuRev server, depending on your configuration), open the AccuRev server file `<ac_home>/bin/acserver.cnf` and add the following line:

```
ASSIGN_USER_PRIVILEGE = scm_bri dge_group
```

Save `acserver.cnf` and close the editor.

**Note:** If you have an existing AccuRev installation and the `acserver.cnf` file contains a line defining `CC_USER`, see [ASSIGN USER PRIVILEGE and CC USER](#) on page 19 in the Troubleshooting section.

- Restart the AccuRev server:

```
> acserverctl stop
> acserverctl start
```

## GitCentric Installation

Except where noted below, first time GitCentric installations and upgrades from previous Kando and GitCentric releases are the same. However, if you are upgrading from a Kando 2012.1 installation, you must perform some manual configuration and migrations steps after the installation completes, as described in [Upgrades from Kando 2012.1](#) on page 15

Refer to [Generic names used during installation, Table 2](#) on page 4 during this procedure.

- Log in as the `<gc_installer>` administrative user (which you should have created above in [Linux Installation](#)). `<gc_installer>` should be the same user that installed AccuRev.

**Note:** If you have just installed AccuRev for the first time as described in the previous section, you should log out and log back into the OS to ensure that environment variables which the AccuRev installer appended to your `.profile` (`JRE_HOME` and `LD_LIBRARY_PATH`) have been set and are available to the GitCentric installer.

- Get a copy of the latest GitCentric installer and save it in your Downloads directory. You can get this from the AccuRev web site:

<http://www.accurev.com/download.html>

3. The GitCentric installation assumes that the Tomcat web server is running, and that “hot deploy” mode is enabled (this is the default mode when Tomcat is installed). If your site has turned off this mode, AccuRev recommends that you turn it back on for the duration of the installation process by setting `autoDeploy` to `true` in the Tomcat `server.xml` configuration file.

4. Make sure that the AccuRev database server is running.

You can use the following commands to verify that the database server is running:

```
> <ac_home>/postgresql/bin/psql -h localhost -U <db-admin-name> -p <db-port> -l
```

**Note:** The final option for the `psql` command is “hyphen lower-case el”.

5. From the `Downloads` directory, run the installer, where `<AccuRevGitCentricInstaller>` is the name of the current GitCentric installation package, such as `AccuRevGitCentric_2013_2_00_Linux_x86.bin`:

```
> sh ./<AccuRevGitCentricInstaller>.bin
```

**Note:** You can run the installer in a non-graphical console mode with the `-i` option:

```
> sh ./<AccuRevGitCentricInstaller>.bin -i console
```

The installer will automatically run in console mode if it determines that it is unable to run in graphical mode.

6. The installer procedure is fairly self-explanatory. Note that:

- You should ensure that the default path to Git is accurate, or adjust it as necessary. For example, if the default is `/usr/bin`, ensure that the actual location is not elsewhere, such as `/usr/local/bin`.
- Adjust the path of the AccuRev server as necessary, and ensure that the path is accurate. For example, if the default assumes `/home/<gc_installer>/accurev` and the server is actually located at `/opt/accurev`, adjust the path accordingly.
- Adjust the path of the GitCentric installation as necessary. Again, if the default is `/home/<gc_installer>/AccuRevGitCentric` and you want to install in `/opt/AccuRevGitCentric`, adjust the path accordingly.

**Note:** If you install in `/opt` or some other root-owned directory, you must use `sudo` commands to create the installation directory and ensure that ownership and permissions are correct.

**UPGRADES ONLY:** If you are upgrading an existing Kando or GitCentric installation, you must specify the existing installation path as the location for the new GitCentric installation.

- On the “Code Review email settings” page, it is not required to enter a value for **SMTP Access Username**. Leave this blank if you are not sure what to enter.
- If you are upgrading from Kando 2012.1, you must perform the manual configuration and migration steps described in [Upgrades from Kando 2012.1](#) on page 15.

7. Restart Tomcat: `<tomcat_home>/bin/shutdown.sh`, followed by `<tomcat_home>/bin/startup.sh`.

# Initial GitCentric Log In

Before you log into GitCentric for the first time, ensure that:

- You have an AccuRev login account that belongs to the Admin group (see [Table 3](#) above).
- You have an SSH public key.

## SSH Key Generation

If you do not already have an SSH key, generate one now from the OS command line:

```
> ssh-keygen -t rsa -C "<your_email>@<your_domain>"
```

This creates a populated subdirectory at `~/.ssh`. You will need the contents of `~/.ssh/id_rsa.pub` in upcoming steps. The topic of SSH keys is beyond the scope of this document, but there is a great deal of information on the web that you can locate with your favorite search engine. One site that has relevant information is <https://help.github.com/articles/generating-ssh-keys>.

## Log In and Self-Register

Point your browser to the GitCentric web browser. If you chose defaults during installation, the address should be similar to:

```
http://<gitcentric_server_name>:8080/gitcentric
```

Adjust the “8080” port if necessary.

Since you are the first user to log into GitCentric after installation, you are presented with a one-time “bootstrap” screen which asks you to specify the AccuRev server and some additional information. No GitCentric menus are visible in the GUI at this point.

AccuRev GitCentric >

Welcome to AccuRev GitCentric! You are the first user and so you are seeing this one-time log-in and configuration screen. Specify the AccuRev host and port to use, and an Accurev user account on that machine which will automatically be added to GitCentric as an administrator account. The CLI path points to the AccuRev client executable to use with that server. Example: /home/accurev/bin/accurev

AccuRev Server:  Port:

AccuRev CLI Path:

Username:

Password:

**Note:** When you complete this screen, you are automatically added to the Administrators group, so you should make sure that an administrator is, in fact, the first user to log into a new GitCentric installation.

Table 4. Initial Connection Values

Field	Example	Comments
<b>AccuRev Server:</b>	<code>localhost</code>	This is the host of the AccuRev server used with GitCentric. You may add other AccuRev servers later, but this is the AccuRev server where the GitCentric database and Tomcat server reside, and is typically on the same machine as the GitCentric installation. The value can be the name of the server, an IP address (not recommended), or "localhost". Note that if you are connecting to GitCentric from a remote machine via your web browser, <code>localhost</code> indicates the AccuRev server on the host you are connecting to, not your local machine where you started your browser.
<b>Port:</b>	<code>5050</code>	This is the AccuRev port number, and is typically "5050".
<b>CLI Path:</b>	<code>/home/gcadmin/accurev/bin/accurev</code>	This is the path to the AccuRev client executable on the GitCentric server that should be used to communicate with the AccuRev server. For example: <code>&lt;ac_home&gt;/bin/accurev</code> Where <code>&lt;ac_home&gt;</code> is the actual installation directory, such as <code>/home/&lt;gc_installer&gt;/accurev</code> . When you are connecting to the GitCentric AccuRev server on a fresh installation, where the client and AccuRev server typically reside on the same machine, this may seem redundant. But if you later add AccuRev servers on other machines, you must have compatible clients installed on the GitCentric server machine to communicate with the remote server. This becomes exceptionally important if you have multiple AccuRev servers and they are at different release levels. For example, if you need to work with two AccuRev servers -- one running version 5.6 and one running version 5.7 -- you would need two AccuRev clients installed on the GitCentric server: one 5.6 client to work with the 5.6 server, and a separate 5.7 client to work with the 5.7 server.
<b>User Name:</b>	<code>&lt;gc_installer&gt;</code>	Specify the AccuRev credentials for <code>&lt;gc_installer&gt;</code> (such as "gcadmin") that you created during " <a href="#">AccuRev Installation</a> " on page 6.
<b>Password:</b>	<code>&lt;gc_installer_password&gt;</code>	

Upon completion of the bootstrap screen, you are taken to the self-registration page, which is described in the following section, [Register with GitCentric](#) on page 12.

# Register with GitCentric

When any user logs into GitCentric for the first time, GitCentric prompts that user through a self-registration process. This is provided so that users with valid AccuRev accounts do not need to involve an administrator in getting set up on GitCentric:

My Account	Code Review	Administration	Documentation
------------	-------------	----------------	---------------

### Register

Register Contact Information

Full Name

Preferred Email

Register SSH Keys

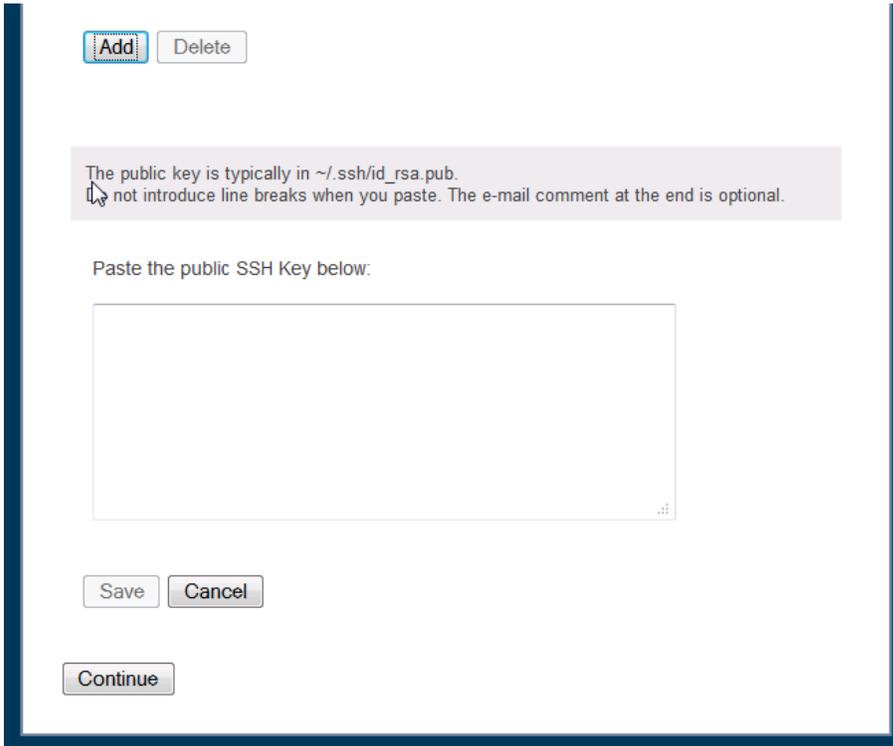
To register your SSH public key with GitCentric, click "Add". See [GitHub's Guide to SSH Keys](#).

Algorithm	SSH key	Comments
-----------	---------	----------

Enter your information as follows:

1. **Full Name:** Your real name (unless this is an account for a role such as "Guest User").
2. **Preferred Email:** Enter the same email address here that you specify for the Git `user.email` configuration for your clones. If these do not match, pushes from your clone will fail. If you need to maintain multiple email addresses, you can add them later at **My Account -> Contact Information**. (See [Contact Information](#) on page 51.)
3. **SSH key:** Click the **Add** button and copy the contents of your SSH public key file here (see [SSH Key Generation](#) on page 10) and **Save** it. (**Note:** If your AccuRev user account is already associated with an SSH public key, that key will be automatically inserted in this field.) Your public SSH key file is typically `~/.ssh/id_rsa.pub`. You can choose to do this later through **My Account -> Public Keys**

(see the GitCentric on-line help), but until this is set, you will be able to use only the code review features of GitCentric.



4. Click **Continue** when done.

## GitCentric AccuRev Trigger and Kando.xml

Once the installer is complete and you have completed the initial “bootstrap” page and the self-registration page, configure AccuRev as follows:

1. Check to see if a `triggers` directory exists under `<ac_home>/storage/site/site/triggers`. If not, create it.
2. Copy the `Kando.xml` and `server_master_trigger.pl` files from `<gc_home>/bin` to the AccuRev server `triggers` directory:

```
> cd <gc_home>/bin
> cp Kando.xml server_master_trigger.pl <ac_home>/storage/site/site/triggers
> cd <ac_home>/storage/site/site/triggers
> chmod +x server_master_trigger.pl
```

**Note:** If you are upgrading from a Kando 2012.1 installation, you will need to overwrite the existing versions of these files.

3. Read the documentation comment lines at the end of `server_master_trigger.pl` and make any adjustments if necessary.

- Log in to the GitCentric UI and navigate to **Administration-> Servers**.

AccuRev GitCentric > GitCentric Administrator <gcadmin@accurev.com> | My Accounts | Logout

My Account Code Review Administration Documentation

Repositories People AccuRev Servers

### Servers

Add or remove AccuRev servers from GitCentric. (You need at least one AccuRev server even if you do not map branches to streams. See [GitCentric overview doc](#).)

Server ID	UUID	AccuRev Server:Port	AccuRev CLI Path
1	2388eccf-29cb-41e6-97c3-8ec64d13a710	localhost:5050	/home/gcadmin/accurev/bin/accurev

Add Delete

2013 AccuRev ([www.accurev.com](http://www.accurev.com))

- Copy the value from the **UUID** column (you can use the clipboard widget if you have a Flash-enabled browser), and paste it into the `Kando.xml` server UUID line. For example (using the UUID from the screenshot above):

```
<configurati on>
  <bridge_uri >http://localhost:8080/kandoBridge/v1</bridge_uri >
  <server_uuid>2388eccf-29cb-41e6-97c3-8ec64d13a710</server_uuid>
</configurati on>
```

**Note:** Make sure that the tag is “server\_uuid” and not “server\_id”. The latter was used in older Kando 2012.1 `Kando.xml` files.

- If you expand to multiple AccuRev servers, or if you put the Web UI server on a different host, you will need to edit the `<server_uuid>` and `<bridge_uri >` entries to reflect the new configuration.
- Test that `server_master_trigger.pl` can compile:

- Look at the first line of `server_master_trigger.pl` and note the Perl executable that it expects to find (for example, “#!/usr/bin/perl”).
- Using this path, use the “perl -c” command to test the trigger:

```
> /usr/bin/perl -c ./server_master_trigger.pl
```

You know that your installation is complete and functioning when you see the message:

```
“server_master_trig syntax ok”
```

If any errors appear, address them before proceeding. For example, if Perl is not installed in the expected location, modify the path in `server_master_trigger.pl` (or install it where the script expects to find it). If any optional Perl modules are missing, such as `XML::Simple`, install them (for example, as `root` try installing `libxml-sax-expat-perl` followed by `cpan install XML::Simple`).

**Note:** Whenever you add any additional AccuRev servers to GitCentric, you will need to repeat these steps of copying and configuring the `Kando.xml` and `server_master_trigger.pl` files.

*Windows only:* If you add a Windows-based AccuRev server to your GitCentric installation, you need to convert the .pl file to a batch file using the PL2BAT utility (PL2BAT should be part of your Perl distribution):

```
>PL2BAT server_master_trig.pl
```

## Enabling/Disabling Code Review

GitCentric's Gerrit Code Review feature is optional. You and your users *must* perform some configuration steps to reflect whether or not code review is part of your workflow. See the “Enable/Disable Code Review” section of the *GitCentric User's Help* for details.

## Upgrades from Kando 2012.1

If you are upgrading from a Kando 2012.1 installation, you must perform the following steps:

- During installation, specify your existing Kando installation directory when prompted by the installer about where to install GitCentric.
- After installation, ensure that any Kando repositories have been moved to the new GitCentric storage area (typically `<gc_home>/site/git`). If not, troubleshoot and use the **kandoMaintain mvrepos** command to manually move the repos to the correct location (see the “The kandoMaintain Utility” appendix in the *GitCentric User's Help*).
- Modify the `[remote "origin"]` entry of the `config` files in any existing clones to point to the migrated repos. See [Upgrading from Kando 2012.1](#) on page 27.
- After installation, migrate any existing Kando “named” ACLs to GitCentric “group” ACLs. See [Upgrading from Kando 2012.1](#) on page 27.
- Update any Kando backup/restore scripts to reflect the following changes:
  - The `backupdb` and `restoredb` options to **kandoMaintain** have been changed to `backupdbs` and `restoredb` to reflect that they now operate on multiple databases in parallel
  - **kandoMaintain** options take a GitCentric home path argument where it expects to find a `dbsettings.conf` file and a Gerrit site directory including an `etc/gerrit.config` file that contains the connect string for Gerrit Code Review.

## Test the GitCentric Installation

Perform the following tests to ensure that various GitCentric modules are in place and that the operating system meets basic requirements to run GitCentric successfully.

### Test the GitCentric Bridge Connection

Test the connection to the GitCentric bridge and ensure that there are no firewall issues:

1. From the GitCentric server command line:

```
wget http://localhost:8080/kandoBridge/v1
```

You should see a “HTTP Request sent” message, followed by additional information, and ending with “v1 saved [nn/nn]”

2. Cat the file "v1":

```
cat ./v1
```

You should see the following message:

```
<error>
<text>No operation allowed at this level</text>
</error>
```

Any other result indicates a problem that should be corrected.

## Proceed to the GUI

At this point you can point your browser to the GitCentric installation:

```
http://<gitcentric_server_name>:8080/gitcentric
```

Adjust the "8080" port if necessary.

Bring up the documentation as described in the next section. After reading the "Concepts and Overview" chapter, proceed to the "How to..." chapter for task-based procedures for both end users and administrators.

## Displaying the User's Help

The GitCentric *User's Help* is available from the GitCentric GUI as both browsable and searchable HTML, and as PDF.

Once you display the GitCentric GUI in a browser, you can display the help by clicking the **Documentation** menu at the right of the GitCentric menu bar. To access the PDF version of the Help, click the PDF icon () in the upper-right corner of the Help display.

When starting the GitCentric GUI for the first time, we recommend that you bring up the Help and navigate to the "Concepts and Overview" and "How to..." chapters, which will guide you through a number of "getting started" procedures.

## Troubleshooting

This section describes some troubleshooting tools and techniques you can use to help resolve issues you might encounter when starting and running GitCentric.

### If You Cannot Log In

If your first attempt to connect to the GitCentric Web UI server fails with an error such as "Unable to Connect" (Firefox), "The webpage cannot be displayed" (Internet Explorer), "This webpage is not available" (Chrome), or something similar, check that:

- Your syntax is correct (for example, "<http://localhost:8080/gitcentric>")
- You are actually pointing to the GitCentric server (for example, are you specifying "localhost:8080" when the web server is actually installed on a remote machine?).
- Your network connection is working.

- The Tomcat server needs to be restarted. (There is a known issue with re-installations requiring that you stop and start the Tomcat server when the installation is done: `<tomcat_home>/bin/shutdown.sh`, followed by `<tomcat_home>/bin/startup.sh`.)
- The Tomcat deployment config file: (`conf/Catalina/localHost/gitcentric.xml`) is correct.

If you can see the initial connection dialog (described in [Initial GitCentric Log In](#) on page 10), but it is rejecting the values you enter, double-check that you are specifying the correct server and port values, and particularly that you are specifying the complete path to the AccuRev client executable on that server.

If you can see the login dialog, but get “Invalid User name or password” errors, check the usual credential details like correct spelling and proper password, and if these look correct, extend your troubleshooting to see if the AccuRev server is actually up and running, and that its license server is also up and running. (“Username and password” errors can sometimes be caused by factors beyond the login credentials.) Test that you can actually log directly into the AccuRev server with the same credentials.

If you still have problems, proceed through the following troubleshooting procedures and suggestions.

## General: Testing the Installation

Make sure that you performed the tests in [Test the GitCentric Installation](#) on page 15 as part of the installation process. You can also perform these tests whenever you suspect that something has stopped working.

## General: The Tomcat Logs

The best troubleshooting tool -- once you get past the basic OS installation and get the GitCentric installation running -- is the Tomcat server logs.

Whenever something is not working right, these logs should be your first stop, in order of usefulness:

```
<tomcat_install>/logs/kandoBridge.log
<tomcat_install>/logs/catalina.out
<tomcat_install>/logs/kandoGerrett.log
<tomcat_install>/logs/localhost.<date>.log
```

You should **cd** to the tomcat logs directory, and dump out the last several lines from the most recent log files. For example:

```
> tail -75 kandoBridge.log | more
```

Scroll through and look for error messages that can help point you to problems with ports or database names, or missing directories, and so on.

Example: One common error is “You are not authorized to use the -U option.”, which appears when files fail to appear in the AccuRev stream after being pushed from Git. This indicates that there is a problem with the `ASSIGN_USER_PRIVILEGE` setting in `acserver.cnf`.

Note that you can modify the settings in the following files to control the level of details logged:

```
<tomcat_home>/webapps/[kandoBridge | gitcentric]/WEB-INF/classes/log4j.properties
```

You can also check the AccuRev server logs:

```
<ac_home>/storage/site/site/logs/acserver.log
<ac_home>/storage/site/site/logs/trigger.log
```

## General: Servers

To confirm if the Tomcat web server is running at all, from a web browser running on the server, enter:  
`http://localhost:8080`

If you get the generic Apache Software Foundation page, the Tomcat server at least is running, and your problem is further downstream.

**Note:** If you have multiple Tomcat installations, this could cause the GitCentric installation to hang. Set `$CATALINA_HOME` to point to the Tomcat which is supposed to work with GitCentric.

To confirm if the AccuRev server is running and configured, try logging in from the CLI:

```
> accurev login
```

If you get a "Connection refused"..."AccuRev Error: 1" message, try starting the server with:

```
> <ac_home>/bin/acserverctl start
```

To confirm if the AccuRev PostgreSQL database is running and recognizes the GitCentric database:

```
> <ac_home>/postgresql/bin/psql -h localhost -U postgres -p 5075 -l
```

(That last argument is "hyphen-lower-case-el".)

You should get a display showing all DBs and their owners. If "accurev" appears in this list but "kando" does not, you need to re-examine the installation steps (or your manual steps) executing the **kandoMaintain** command. If you cannot bring up this list at all, you need to re-examine your steps installing and starting the AccuRev server (which also install and start the PostgreSQL server).

## General: Settings

There are several parts that need to work together in a successful GitCentric installation. Here is a list of things to double-check. Make sure that:

- `ASSIGN_USER_PRIVILEGE` is defined in `acserver.cnf` for the AccuRev server, and that you restart the AccuRev server after defining or changing it.
- the two `.war` files have expanded in the Tomcat webapps directory, creating directories named `gitcentric` and `kandoBridge`.
- `<gc_home>/dbsettings.conf` exists and contains valid entries for the DB port (typically `localhost:5075`), and DB username/password (typically `postgres/postgres` unless you change these for security reasons).
- your AccuRev server contains properly installed and configured triggers files (see [GitCentric AccuRev Trigger and Kando.xml](#) on page 13).
- you clear your browser cache after updating to the latest `.war` files.

Also, remember that unlike AccuRev, Git does not track empty directories.

Finally, experienced Gerrit users may recognize that GitCentric includes a Gerrit Code Review installation, although some details may be modified or ignored in the GitCentric environment. For example, certain code review installation values (such as the `httpd jetty` web server port setting) are not used in the GitCentric environment.

## LD\_LIBRARY\_PATH: psql shared library error

This should be done automatically by the GitCentric installer, but if you try to run the `psql` command and get an error regarding `libpq.so.5`, you need to update your `LD_LIBRARY_PATH` to point to the `lib` subdirectory under the AccuRev version of PostgreSQL:

```
> export LD_LIBRARY_PATH=<ac_home>/postgresql/lib:$LD_LIBRARY_PATH
```

where `<ac_home>` is the directory where AccuRev is installed.

## ASSIGN\_USER\_PRIVILEGE and CC\_USER

GitCentric requires that the `acserver.cnf` file on each AccuRev server that has streams mapped to Git branches via GitCentric contains a setting called `ASSIGN_USER_PRIVILEGE`. Existing AccuRev installations that included early releases of GitCentric, or which made use of the AccuRev “ClearCase Adapter” product, may have a similar setting in this file named `CC_USER`. If this setting is in use, the following rule applies:

If you specify both `CC_USER` and `ASSIGN_USER_PRIVILEGE`, the `CC_USER` takes precedence.

## Debug Settings

See the GitCentric Configuration Settings section of the “The kandoMaintain Utility” chapter of the *GitCentric User’s Help* for settings that might be helpful for debugging purposes.

For example, to troubleshoot errors during import and export operations you can use **kandoMaintain** to enable a logging file and preserve temporary files:

```
kandoMaintain addconfig -u <dbadmin> -n ACCUREV_COMMAND_LOGFILE -v <fullpath>  
kandoMaintain addconfig -u <dbadmin> -n PRESERVER_TEMPFILES -v y
```



# GitCentric™ Release Notes

This chapter provides detailed information about the changes and known issues in the current GitCentric release, GitCentric 2013.2, as well as in previous GitCentric releases.

## Changes in Release 2013.2

This section documents changes in Release 2013.2.

### 27876 -- Support for Gerrit Code Review replication

You can now configure GitCentric to notify Gerrit Code Review to push changes from Git repositories to one or more remote hosts after exporting an AccuRev stream to a Git branch. See “Enable Gerrit Code Review Replication” in Chapter 2, How To... in the *GitCentric User’s Help* for more information.

### 27955 -- Ability to turn Gerrit Code Review on/off

The General tab under Repositories (**Administration** > **Repositories** > **General**) has a new field that allows users to turn Gerrit Code Review on and off from the GitCentric UI. This field, **Enable Code Review**, is on by default. Users and the GitCentric administrator must perform other configuration tasks in order to fully enforce Gerrit Code Review for individual clones. See “Enable/Disable Code Review” in Chapter 2, How To... in the *GitCentric User’s Help* for more information.

### 28254, 30418 -- Support for custom push hooks

GitCentric 2013.1 ignored any custom push hooks (pre-receive, update, post-receive, and post-update) that were created in Kando 2012.1. The current GitCentric release, 2013.2, uses Gerrit hooks to access and execute any Git hooks you might have created in the `<gc_home>/site/git/<repoName>.git/hooks` directory. These Gerrit hooks, ref-update and ref-updated, are installed to `<gc_home>/site/hooks`; their scripts do not require any modification. See the Gerrit Code Review documentation for more information on hooks: <https://gerrit-review.googlesource.com/Documentation/config-hooks.html>.

### 28920 -- Upgrades leave hooks directory

In the previous release, when upgrading from Kando 2012.1 to GitCentric 2013.1, the upgrade did not remove the Git hooks directory (`<gc_home>/site/git/<repoName>.git/hooks`), which was no longer used in GitCentric 2013.1. This did not cause a problem, but it was a possible source of confusion.

In GitCentric 2013.2, the Git hooks directory will remain, but the GitCentric installer removes any Kando.xml files from the Git hooks directory for each repo. Any hooks in those directories are executed by new Gerrit hooks, ref-update and ref-updated.

## 28980 -- Gerrit Code Review upgraded to 2.6

GitCentric 2013.2 incorporates Gerrit Code Review Version 2.6. This upgrade has resulted in a number of changes, including:

- The Gerrit Code Review UI menus have changed.
- The Gerrit Code Review documentation is now accessed directly from Gerrit. For example: <http://code.google.com/p/gerrit/>.
- The **Require Change-Id in commit message** field (**Administration > Repositories > General**) is now set to True by default.

## 29586 -- Date/Time Format settings reflected throughout GitCentric UI

In previous releases, changes to the **Date/Time Format** on the **Preferences** page (**My Account > Preferences**) were not reflected in the **Registered** date on the **Profile** page (**My Account > Profile**). This problem has been corrected in the current release.

## 30111 -- Branches table now links to Gitweb

The Branches table in the GitCentric UI (**Administration > Repositories > Branches**) now provides links to the Git web interface, Gitweb.

## 30217 -- Files renamed during export causes export to fail

AccuRev renames files with spaces in their path during export. In previous releases, renaming the file caused the export (and subsequent exports that included the renamed file) to fail. This problem has been corrected in the current release.

## 30308 -- Self-reviews disallowed by default

In new, first-time GitCentric installations, the default Gerrit Code Review prolog rules have been modified to disallow self-review of changes. Changes can now be submitted only if they are approved by someone other than the changes' author. If you are *upgrading* to GitCentric 2013.2, self-review is still permitted unless the administrator manually installs the Gerrit Code Review prolog rules file, rules.pl. See "Allowing Self-Reviews" in Chapter 2, How To... in the *GitCentric User's Help* for more information.

## 30319 -- Merge fails if the AccuRev Connector is configured after code review

In previous releases, if you configured the AccuRev Connector (**Administration > Repositories > AccuRev Connector**) *after* pushing the change from your clone to code review, reviewing, publishing, and submitting the change would result in the change remaining in a "Submitted, Merge Pending" state. This problem has been corrected in the current release.

## 31659 -- "Ignore symbolic links" checkbox

After upgrading from Kando 2012.1, the **Ignore symbolic links** checkbox was reversed from its Kando state. However, the GitCentric status was correct. This problem has been corrected in the current release, and the setting of the **Ignore symbolic links** checkbox now correctly reflects GitCentric status.

## Changes in Patch Release 2013.1.01

This patch release addressed some issues that were reported after the general release of GitCentric 2013.1:

### **29672, 29873 -- AccuRev promote hangs if issue number required but not provided**

A problem has been corrected so that if you push a change without specifying an issue number in the commit message when one is required, GitCentric now returns an error message. As part of this fix, users running AccuRev 5.7 or earlier, and who require that issue numbers be included in commit messages, **MUST** enable “Issue 0” in their AccuRev Change Package Schema. This is an administrative procedure that must be done with the assistance of AccuRev Customer Support. Please contact Support at <http://support.accurev.com>, or via e-mail at [support@accurev.com](mailto:support@accurev.com) for help with this configuration step.

### **30217 -- Pathnames with spaces can cause failures**

A problem has been corrected which could cause failures when exporting from AccuRev to Gitcentric, if AccuRev had renamed a file with a space in its path.

### **30509 -- GitCentric installer fails to determine the correct Tomcat web port**

A problem with the GitCentric installer has been corrected which sometimes prevented it from determining the correct Tomcat port, and resulting in a bad configuration file which prevented GitCentric from starting.

## Changes in Release 2013.1

This release is a major redesign of the original Kando 2012.1 product, and provides a completely new graphical user interface (GUI), a new model for administering Git repositories and their related ACLs (Access Control Lists), and introduces a new, integrated code review feature.

## Known Issues

AccuRev is aware of the following issues in the current GitCentric release.

### **24313 -- Import complete Git commit history**

This release provides **BETA evaluation functionality** to import complete Git commit history from an existing repository to your AccuRev depot. If you wish to try this functionality, contact AccuRev Technical Support.

### **24861 -- Change package gaps**

If you use change packages (CPKs) with GitCentric in AccuRev 5.7 or earlier, be aware that GitCentric does not currently span CPK gaps automatically. This is not an issue in AccuRev 6.0 owing to changes in CPK functionality.

### **27473 -- email confirmation**

GitCentric currently does not provide confirmation email for account verification.

## 27790 -- Authorization

GitCentric currently supports only AccuRev authentication (that is, GitCentric users must have a valid AccuRev account). GitCentric does not currently support other authorization types such as LDAP, OpenID, etc.

## 28337 -- Login license manager errors

If the license manager is not running when you try to log into GitCentric, the error(s) displayed by the system may not clearly indicate the cause of the problem.

## 28925 -- Stream properties

If you upgrade from Kando 2012.1 to GitCentric 2013.2, Kando stream and branch information is not migrated. You can manually run the **set properties** command, or re-save a branch-to-stream mapping.

## 28936 -- kandoMaintain backup files

The **kandoMaintain backupdbs** command creates backup files in the same directory as the kandoMaintain.jar, not where **kandoMaintain** is run from.

## 29070 -- Re-installs require Tomcat reboot

If you re-install GitCentric over an existing installation, sometimes you cannot log into GitCentric when the installation is done. To fix this, restart Tomcat (`<tomcat_home>/bin/shutdown.sh`, followed by `<tomcat_home>/bin/startup.sh`).

## 29564 -- Register all IP addresses for a server

For security reasons, the GitCentric bridge accepts commands only from IP addresses that it recognizes. If you have registered an AccuRev server that has multiple IP addresses (for example, if the server has multiple Ethernet cards) you must add server entries for each IP address. In this case, you should perform GitCentric operations only against one of these servers entries -- typically one where you have specified a host name rather than an IP address. If you have a situation where IP addresses can change over time (such as dynamic IP addresses assigned via DNS), you may need to consider setting `AC_BRIDGEAPI_SECURITY_POLICY` to `AllowAnyHost`. Note, however, that this workaround has a security impact that you need to carefully consider before implementing it. Procedures for adding server entries and GitCentric configuration settings are documented in the GitCentric On-line Help.

## 29089 -- Errors in catalina.out log file

You may see instances of “SEVERE” errors in catalina.out, relating to “failure to unregister JDBC driver” and “remove threads on exit”. These errors can be safely ignored in this release.

### 30954 -- Some strings with special characters require quote marks

A known issue with Gerrit requires that you enclose certain values in `gerrit.config` and `secure.config` files in double quotes if they contain special characters, as summarized in the following table.

Config File	Section	Value
gerrit.config	[gerrit]	basepath
	[httpd]	listenURL
	[kando]	KANDOHOME
		bridgeURL
secure.config	[database]	password

For example, if the password in your `secure.config` file is `my! pass#ord`, you would need to enclose it in double quotes, like this: `password = "my! pass#ord"`. The `gerrit.config` and `secure.config` files are located in `<gc_home>/site/etc`.

### 31660 -- Code Review

GitCentric Code Review submit/merge operations do not return issues that have been closed in change comments.

## Gerrit Issues

Since GitCentric incorporates Gerrit Code Review, you should also check the Gerrit issues list at:

<http://code.google.com/p/gerrit/issues/list>

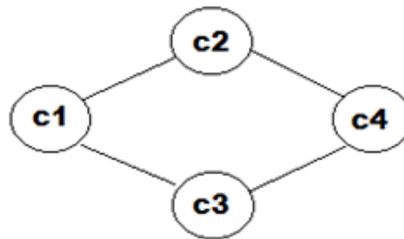
## Changes from Kando 2012.1

Because of the design changes that have gone into GitCentric 2013.2, several concepts and details are different from the original Kando 2012.1 release. This section summarizes those changes for users of Kando 2012.1:

- **Name change** -- The change from “Kando” to “GitCentric” is more than a name change. Kando still refers to the underlying engine and technology, and you will still see this name when you look under the covers. But you may see other AccuRev products in the future in addition to GitCentric that build upon this framework.
- **New UI** -- AccuRev has completely revamped the user interface for GitCentric, and now incorporates the open source Gerrit Code Review tool.
- **Expanded user capabilities** -- In Kando 2012.1, the UI was primarily an Administrator’s tool. While the GitCentric UI includes many administrator capabilities, it also now provides features such as code review and profile management that can be used by all Git users in your organization.
- **Self-registration and management of SSH keys**-- In Kando, an administrator was required to add users and install their SSH keys. In GitCentric, users register themselves with GitCentric and install their own SSH public keys.
- **Groups and ACLs** -- GitCentric has replaced Kando proprietary named ACLs with Gerrit’s powerful group-based ACL security. As part of this implementation, groups are now handled as GitCentric

entities and no longer depend on AccuRev group definitions. However, AccuRev groups are automatically mapped to GitCentric as read-only groups.

- **ACL migration** -- Because of the change in ACLs, Kando 2012.1 customers who implemented named ACLs need to follow the manual ACL migration steps documented elsewhere in this document.
- **kando\_admin** -- In Kando 2012.1, you needed to be a member of an AccuRev group named kando\_admin to log into the UI. In GitCentric 2013.1 the requirement for this group no longer exists. (However, if you are upgrading from Kando, any existing kando\_admin group is automatically imported into a GitCentric group with administrator privileges.
- **Manual operations** -- A few operations that were available through the Kando UI must be performed manually -- or are not required at all -- in GitCentric. For example, there are no “Remove Repo” or “Remove Branch” functions in the GitCentric UI. If you wish to remove a repo or a branch, follow the procedures documented in the “How to...” chapter of the on-line *GitCentric User’s Guide*. If you need to import an existing repository into GitCentric, the on-line help provides extensive directions for doing so. (The complexity of this operation depends on whether or not you wish to use the [BETA evaluation functionality](#) to import the complete commit history of the repository into AccuRev.)
- **Direct push, Gerrit hooks, and ACLs** -- In Kando 2012.1, we made use of Git custom hooks. With the incorporation of Gerrit Code Review, pushes by default go to a special code review branch and not directly into the master repository. Gerrit does not support custom hooks for direct pushes to repositories, and in fact it has its own set of hooks for the code review path. These are documented at <http://gerrit.googlecode.com/svn/documentation/2.1.2/config-hooks.html> If you want to perform direct pushes, you must follow the procedures documented in the on-line help to configure your repository and its ACLs to support this. **Note:** If you are a Kando 2012.1 customer and you require custom direct push hooks, we recommend that you do NOT upgrade to this current release of GitCentric.
- **Transaction commits and history imports** -- AccuRev history now reflects all commits in a merge. For example, given the following graph:



In Kando 2012.1, AccuRev history would only reflect only the changes at C4. With GitCentric, AccuRev history now reflects the commits at C1, C2, and C3, and then shows C4 as merge. This improved history also provides [BETA evaluation functionality](#) that now enables you to import your entire Git history from an existing repository into AccuRev, by first pushing the initial commit, and then pushing the final commit.

# Upgrading from Kando 2012.1

## Changes Since Kando 2012.1

GitCentric 2013.2 has been re-designed since Kando 2012.1. It includes a number of improvements over release 2012.1, and where possible, upgrades from existing installations to 2013.2 handle required changes for you automatically. However, GitCentric 2013.2 uses a fundamentally different strategy for handling repositories and security access settings (Access Control Lists, or “ACLs”), and you will need to perform some manual configuration to adapt your 2012.1 installation for use with 2013.2.

In 2012.1, Kando ACLs were “named” ACLs: you defined ACLs and then applied them to users and groups to control their access to repositories.

In 2013.2, GitCentric has implemented group-based ACLs. You cannot apply them to individual users. However, you can emulate the 2012.1 model by creating inherit-only projects in 2013.2 for each of the named ACLs you created in Kando 2012.1. (See [Migrating Kando Named-ACLs to GitCentric ACLs](#) below.)

## Ensure that Repos Have Been Moved

The GitCentric installer attempts to move any existing Kando repositories to the correct location for GitCentric. You should check `<gc_home>/site/git` after the installation completes to ensure that this step succeeded. If not, you need to troubleshoot why the move failed, and then move them manually using the `kandoMaintain mvrepos` command (see the “kandoMaintain” appendix in the *GitCentric User’s Help*).

## Update Clone Configs

After installation, and after you have confirmed that your Kando repositories have been moved to their new GitCentric location, you must modify the `[remote "origin"]` entry of the `config` files in any existing clones to point to the migrated repos. Open the config file in an editor. In the `[remote "origin"]` section you will see a line looks like this:

```
url = git@repo_host:myrepo
```

You must edit this to remove the user and to use the new GitCentric SSH protocol:

```
url = ssh://repo_host:29418/myrepo
```

# Migrating Kando Named-ACLs to GitCentric ACLs

Determine which of the following scenarios describes your Kando 2012.1 installation and then refer to the appropriate section.

Scenario	Go To
You have only one named ACL for that applies to all of your Kando 2012.1 repositories.	<a href="#">If You Have Only One 2012.1 Named ACL</a> on page 28
You have at least one Kando 2012.1 repository that has a "unique" named ACL (and you do not expect to re-use that ACL).	<a href="#">If You Have Unique 2012.1 Named ACLs</a> on page 28
You have "named ACL's" used on multiple repositories.	<a href="#">If You Have Multiple 2012.1 Named ACLs</a> on page 28

## If You Have Only One 2012.1 Named ACL

This is the easiest scenario to deal with: Edit the GitCentric 2013.2 All-Projects ACL with one that replaces the named ACL you had in Kando 2012.1. This will be inherited by all 2013.2 repositories.

## If You Have Unique 2012.1 Named ACLs

If you have one or more repositories that have a unique 2012.1 named ACL assigned to them, and you do not expect to reuse the ACL: Edit the GitCentric 2013.2 ACL on each repository to replace the named ACL you had in Kando 2012.1.

## If You Have Multiple 2012.1 Named ACLs

If you have "named ACL's" used on multiple repositories, create a "parent only repository", use this to setup the ACL, and make it the parent of all the repositories you need to protect.

This scenario requires additional manual work.

1. For each named ACL that you had in 2012.1, create an inherit-only repository in GitCentric 2013.2.
2. For each named ACL that you had in 2012.1, edit the ACL of each inherit-only repository to replace the named ACL you had in Kando 2012.1.
3. Use the SSH **gerrit set-project-parent** command to make the appropriate repositories children of these inherit-only projects so that the ACLs apply to them. See: <http://gerrit.googlecode.com/svn/documentation/2.2.1/cmd-set-project-parent.html>