GitCentric™

Installation and Release Notes

Version 2018.1

Table of Contents

Introduction

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

Highlights of the Current GitCentric Release

- You can now see **…/accurev/storage/site_slice/log/trigger.log** output from the **server_preop_trig.pl** when an error occurs.
- Fixed the Gerrit review & settings URL sent in Gerrit review mails.

Note: The current release is the last one to support a 32-bit platform. From the next release onwards, the GitCentric installer will support only 64-bit platforms.

Note: GitCentric was formerly known as Kando.

GitCentric Installation Notes

Introducing GitCentric

This chapter describes how to install or upgrade GitCentric software at your site. It also describes the process for migrating an existing GitCentric installation to a new machine.

GitCentric Release 2018.1 is intended for new customers and customers currently using a previous GitCentric release. For a high level summary of new features in Release 2018.1, see Highlights of the Current GitCentric Release on page 5. For a detailed list of known issues, see GitCentric Release Notes on page 24.

Note: If you are upgrading from Kando 2012.1, contact Customer Care at:

http://supportline.microfocus.com/

Preparing for Installation

This section describes requirements for a successful upgrade or installation.

Required Software

GitCentric requires the following software for a successful installation.

Note: Other components may have their own prerequisites.

Table 1 Prerequisites and Platform Support

Component	Supported Configuration
Operating System	For GitCentric server: ▪ Linux 2.6 or later ▪ SUSE 10 SP2 or later 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 can be run from any platform.
Browsers	GitCentric is supported on these browsers: ▪ Chrome (latest) ▪ Firefox (latest) ▪ Microsoft Internet Explorer 9 and later ▪ Microsoft Edge (the default browser in Windows 10 and Windows 10 Mobile)
Java	Java 1.6 or later. (The AccuRev installation installs Java at `<ac_home>/jre`. You may need to update your $PATH to point to this.)
AccuRev	Version 5.7 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
<gc_installer>		Linux admin OS account for installing AccuRev, GitCentric, etc. If you have an existing AccuRev server on Linux, this should be the account you used to install it, such as "acserver". If you are starting from scratch, you can create a new one such as "gcadmin".
<gc_sync_user>		This is the AccuRev "service account" (typically "gcSyncUser") that GitCentric uses to automatically keep Git and AccuRev in sync. This is a member of the "scm_bridge_group" group, which is set to be the value of "ASSIGN_USER_PRIVILEGE" in the **acserver.cnf** configuration file on each AccuRev server controlled by GitCentric. For any single mapped repository, there is an AccuRev user who is the service account. **Note:** The *<gc_sync_user>* should NOT be a member of an AccuRev administrator group.
<gc_home>		Location where GitCentric is installed. This is typically: (Linux) ~*<gc_installer>*/AccuRevGitCentric
<ac_home>		Location where AccuRev is installed. For GitCentric, this is typically: (Linux) ~*<gc_installer>*/accurev (but sometimes /opt/accurev or elsewhere) **Note**: If you are using AccuRev in a replication environment, this is the location of the AccuRev master server.
<tomcat_install>		Location where the Tomcat web server is installed. For GitCentric, this is typically: (Linux) *<ac_home>*/WebUI/tomcat
<db-port>		Port number that the database server runs on. Defaults to **5075**.
<db-admin-name>		Name of the database superuser. Defaults to **postgres**.
<db-admin-pass>		Password for *<db-admin-name>*.

Summary of Steps

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

Install Linux, if necessary. The GitCentric server needs to run on a Linux host. See Linux Installation.

1. 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. This user should also have a corresponding AccuRev account that is a member of the Admin group.

Download the components you will need to install (AccuRev installer, GitCentric installer, AccuRev and GitCentric license keys, and so on). See [Linux Installation on page 9]Linux Installation, and AccuRev Installation on page 9, for download instructions.

2. As OS user *<gc_installer>*, install the AccuRev (5.7 or higher) Server, Web UI server, and client. See AccuRev Installation on page 9.

    As OS user *<gc_installer>* install GitCentric. See GitCentric Installation on page 12.

3. Restart Tomcat.

4. Configure GitCentric and AccuRev. This consists of the following steps:
    *   Add the "ASSIGN_USER_PRIVILEGE" role in the AccuRev server configuration file (**acserver.cnf**). Restart the AccuRev server for this change to take effect.
    *   Copy **server_master_trig.pl** from **<gc_home>/bin to <ac_home>/storage/site_slice/triggers**
    *   Edit **server_master_trig.pl** (after it is copied) to have the correct path to the AccuRev client on the line beginning **'$::AccuRev ='**
    *   On the AccuRev server, establish a persistent AccuRev login with:
         "**accurev login -n <user> <password>**"

Note: Step 3 and 4 are required for the **server_master_trig.pl** script to run AccuRev commands when the trigger fires

Then configure the **server_master_trig.pl** file in the AccuRev **triggers** directory. See GitCentric Installation on page 12.

Note: If you are using GitCentric in a replication environment, perform all AccuRev server configuration on the AccuRev master server.

Bring up the GitCentric Web UI and log in. See [Initial GitCentric Log In on page 13].

About Uninstalling GitCentric

Generally speaking, it is not necessary to uninstall your current GitCentric version before upgrading. However, if you wish to uninstall, be sure to back up your current GitCentric database. See [Uninstalling GitCentric on page 23].

About Moving GitCentric to a New Machine

One variation of the GitCentric installation process involves moving GitCentric repositories to a new machine on which a new version of GitCentric is installed -- as might be required after you have upgraded   a server, for example. The following steps outline the procedure to follow.

Note: Refer to Chapter 8,"Moving the AccuRev Server and Repository to Another Machine" in the *AccuRev Administrator's Guide* for information related to moving AccuRev.

To move GitCentric to a new machine:

On the Existing Machine:

1. Using tar or a similar copy utility, make a copy of all Git repositories that are mapped to AccuRev. Git repositories are saved in your *<gc_home>*/site/git directory.

On the New Machine:

1. Install GitCentric as described in GitCentric Installation on page 12.

2. Drop the GitCentric repositories you copied in [Step 1] to *<gc_home>*/site/git in the new GitCentric installation.

3. Stop and restart Tomcat:

```
<tomcat_home>/bin/shutdown.sh
<tomcat_home>/bin/startup.sh
```

4. Configure your new GitCentric installation. At a minimum, this requires creating new branch-stream mappings to associate your Git repositories with your AccuRev depots. In addition, users will need to re-register with GitCentric.

   For more information, see "Map a Git Branch to an AccuRev Stream" in Chapter 2, How To..., in the *GitCentric User's Help* for information on this and other configuration topics.

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 AccuRev Installation below.

To Install Linux

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

http://supportline.microfocus.com/prodavail.aspx

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 "gcadmin" 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", and so on.

AccuRev Installation

You need to have an AccuRev (5.7 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 5.7 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 5.7 (or higher) AccuRev installer from the AccuRev web site:

   http://supportline.microfocus.com/websync/productupdatessearch.aspx

   The full installer is for installing the AccuRev server (which also installs the AccuRev database), AccuRev Web UI 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 *<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
admin	group	N/A	Standard AccuRev group for administrative users.
scm_bridge_group	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 `gcSyncUser`, and is assigned to `ASSIGN_USER_PRIVILEGE` in each AccuRev server's **acserver.cnf** file. Different users with different access rights can be assigned to this group for security reasons. Value does not necessarily need to be "scm_bridge_group".
gcadmin	user	admin	Administrative login account on the GitCentric AccuRev server. Value does not necessarily need to be "gcadmin".
gcSyncUser	user	scm_bridge_group	The "service account" user on each AccuRev server that is mapped to branches in a Git repository. Member of the scm_bridge_group group, which is assigned to the `ASSIGN_USER_PRIVILEGE` setting in each AccuRev server's **acserver.cnf** file. (See Step 8 below.) Should NOT be a member of the admin 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 User's Help*, and is documented in the AccuRev user documentation.

8. 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 configuration file *<ac_home>*/bin/acserver.cnf and add the following line:

```
ASSIGN_USER_PRIVILEGE = scm_bridge_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 22 under Troubleshooting.

9. Restart the AccuRev server:

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

IMPORTANT: Be aware that if both AccuRev and GitCentric are installed on the same machine, and the AccuRev software has been upgraded to version 7.x, you must take extra steps regarding GitCentric.

The reason is AccuRev 7.x uses PostgreSQL 9.5, where prior versions of AccuRev used PostgreSQL 8.4, and the AccuRev database has been migrated from 8.4 to 9.5 during the move to

AccuRev 7.x. GitCentric uses the same database as AccuRev, but a different set of tables. See [GitCentric Database Migration After Upgrading AccuRev to 7.x on page 13](#).

In case of a clean install of AccuRev 7.x. you **do not** have to follow the procedure.

GitCentric Installation

Except where noted below, first-time GitCentric installations and upgrades from previous GitCentric releases are the same.

Refer to Table 2 Generic names used during installation on page 7 while performing this procedure.

1. 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.

2. Get a copy of the latest GitCentric installer and save it in your Downloads directory. You can get the GitCentric installer from the Micro Focus SupportLine web site:

   http://supportline.microfocus.com/websync/productupdatessearch.aspx

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, Micro Focus 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 *<AccuRevGitCentricInstall>* is the name of the current GitCentric installation package, such as `AccuRevGitCentric_2017_1_00_Linux_x86.bin`:

   > sh ./*<AccuRevGitCentricInstall>*.bin

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

   > sh ./*<AccuRevGitCentricInstall>*.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:

   • 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.

     Note: If you are using AccuRev in a replication environment, this is the location of the AccuRev master server.

   • 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 GitCentric installation, you must specify the existing installation path as the location for the new GitCentric installation.

- On the "Code Review email settings" page, leave the **SMTP Access Username** field blank if you are not sure what to enter.

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

GitCentric Database Migration After Upgrading AccuRev to 7.x

Here are instructions for database migration after upgrading AccuRev from older versions (5.x or 6.x) to version 7.x.

In the following Unix shell commands, *<ac_home>* is the root directory of the AccuRev installation.

GitCentric is only supported on Unix.

1. Start the old postgres 8.4 database running

   ```
   > <ac_home>/postgresql/8.4/bin/pg_ctl start -o "-p 5076" -D
   <ac_home>/postgresql/8.4/db
   ```

2. Back up the kando database:

   ```
   > <ac_home>/postgresql/8.4/bin/pg_dump -U postgres -h localhost -p 5076 -Fc -f
   /tmp/kando.dump kando
   ```

3. Enable the use of the 9.5 binaries and get the correct library:

   ```
   > export LD_LIBRARY_PATH=<ac_home>/postgresql/9.5/lib
   ```

4. Back up the **gcreviewdb** database using 9.5 pg_dump to avoid an error from "CREATE PROCEDURAL LANGUAGE plpgsql;"

   ```
   <ac_home>/postgresql/9.5/bin/pg_dump -U postgres -h localhost -p 5076 -Fc -f
   /tmp/gcreviewdb.dump gcreviewdb
   ```

5. Restore the Kando database into 9.5.

   ```
   <ac_home>/postgresql/9.5/bin/pg_restore -U postgres -h localhost -p 5075 -C -d
   postgres /tmp/kando.dump
   ```

6. Restore the **gcreviewdb** database into 9.5.

   ```
   <ac_home>/postgresql/9.5/bin/pg_restore -U postgres -h localhost -p 5075 -C -d
   postgres /tmp/gcreviewdb.dump
   ```

7. Restart Tomcat to reload the **webapps gitcentric.war** and **kandoBridge.war**.

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).
- 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 installation. 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.

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
AccuRev Server	localhost	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, localhost indicates the AccuRev server on the host you are connecting to, not your local machine where you started your browser.
Port	5050	This is the AccuRev port number, and is typically "5050".

Field	Example	Comments
CLI Path	/home/gcadmin/accurev/bin/ accurev	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: *<ac_home>*/bin/accurev Where *<ac_home>* is the actual installation directory, such as /home/*<gc_installer>*/accurev. The GitCentric server requires either a master AccuRev server or a compatible AccuRev client installed locally. In the case where you are using AccuRev servers of different revision levels, you must have a compatible client installed on the GitCentric server machine for each version of the AccuRev server you are using, and each must be configured with GitCentric using its specific path. For example, if you need to work with two AccuRev servers -- one running version 5.7 and one running version 6.2 -- you would need two AccuRev clients installed on the GitCentric server: one 5.7 client to work with the 5.7 server, and a separate 6.2 client to work with the 6.2 server.
User Name	*<gc_installer>*	Specify the AccuRev credentials for *<gc_installer>* (such as "gcadmin") that you created during "*AccuRev Installation* on page 7.
Password	*<gc_installer_password>*	

Upon completion of the bootstrap screen, you are taken to the self-registration page, which is described in

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 to get set up on GitCentric:

Register

Register Contact Information

Full Name [   ]

Preferred Email [   ]

Save

Register SSH Keys

To register your SSH public key with GitCentric, click "Add". See *GitHub's Guide to SSH Keys* .

Algorithm	SSH key	Comments

Add   Delete

Continue

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" in the *GitCentric User's Help*.)

3. **SSH key:** Click the **Add** button and copy the contents of your SSH public key file here (see SSH Key Generation on page 14) 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 not be able to use ssh protocol for git clone, pull, and push operations.

4. Click **Continue** when done.

Install the AccuRev Trigger

Once the installer is complete and you have completed the initial "bootstrap" page and the self-registration page, you need to install the AccuRev trigger **server_master_trig.pl** on the AccuRev server:

Note: If you are using GitCentric in a replication environment, you need to perform this procedure on the AccuRev master server. In addition, if you later add other AccuRev servers to your GitCentric configuration, you must install **server_master_trig.pl** on those servers as well.

1. Check to see if a `triggers` directory exists under `<ac_home>/storage/site_slice`. If not, create it.

2. Copy the **server_master_trig.pl** file from `<gc_home>/bin` to the AccuRev server `triggers` directory:

   ```
   > cd <gc_home>/bin
   > cp server_master_trig.pl <ac_home>/storage/site_slice/triggers
   > cd <ac_home>/storage/site_slice/triggers
   > chmod +x server_master_trig.pl
   ```

3. Locate the `############### ACCUREV CLIENT PATH` section in the **server_master_trig.pl** file and specify the path for your AccuRev client. For example:

   ```
   $::AccuRev = "/usr/accurev/bin/accurev";
   ```

4. Establish a persistent AccuRev Login on the AccuRev server with:

   ```
   accurev login-n <user><password>
   ```

5. Test that **server_master_trig.pl** can compile:

- Look at the first line of `server_master_trig.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_trig.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_trig.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 **server_master_trig.pl** file.

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
```

How to Enable Code Review?

Click the link https://youtu.be/HP8bsuJG8PY for a video that explains how to enable 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 -q -O - 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 a message displaying a SHA, like this:

```
{"sha", "<40-hex chars>"}
```

where `<40-hex chars>` is the SHA value. Any other result indicates a problem that should be corrected.

Proceed to the GUI

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 Chapter 1, Concepts and

---

Overview, proceed to Chapter 2, How to... 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 browseable and searchable HTML, and as PDF.

Once you display the GitCentric GUI in a browser, you can display the help by clicking the help button (

) 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.

Tip: When starting the GitCentric GUI for the first time, we recommend that you bring up the Help and navigate to Chapter 1, Concepts and Overview and Chapter 2, How to..., 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 <u>Initial GitCentric Log In on page 13</u>), 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 as described under 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/kandoGerrit.log
<tomcat_install>/logs/catalina.out
<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 file 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_slice/logs/acserver.log
<ac_home>/storage/site_slice/logs/trigger.log
<gc_home>/site/logs/sshd.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", not "hyphen-numeral-one".)

You should get a display showing all databases 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 database port (typically localhost:5075), and database username/password (typically "postgres"/"postgres" unless you change these for security reasons).
- Your AccuRev server contains properly installed and configured triggers files (see Install the AccuRev Trigger on page 18).
- You clear your browser cache after updating to the latest .war files. Also, remember that Git, unlike AccuRev, 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 Appendix A, The kandoMaintain Utility, in 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_TEMP_FILES -v y
```

Uninstalling GitCentric

The GitCentric uninstaller *removes everything* in the GitCentric installation directory, including the GitCentric database. Therefore, before running the uninstaller, you should back up any files you wish to keep, and you should use the `full_backup.sh` command to back up the GitCentric database. See " Appendix B. Backup and Restore" in the *GitCentric User's Help* on the database backup process before running the uninstall command.

Running the uninstall Command

The GitCentric **uninstall** command must be run from the home directory of the user who performed the GitCentric installation, *<gc_installer>*. To uninstall GitCentric, from this location run:

```
./Uninstall\ AccuRevGitCentric
```

The uninstaller displays a warning message. You can cancel the uninstall operation at this time if you choose.

GitCentric Release Notes

This chapter provides detailed information about the changes and known issues in the current GitCentric release, GitCentric 2018.1, as well as in previous GitCentric releases.

Understanding Issue Numbers

In cases where an issue is identified with two numbers, 34987 (2788687), for example, the first represents   the issue number in the AccuWork issue tracking system; the second, in parentheses, is the issue number in the SupportLine system used by Customer Care.

Changes in Release 2018.1

636572 -- Deprecation note for support 32 bit

The current release is the last one to support 32-bit platforms. GitCentric installer will support only 64-bit platforms from the next release onwards.

1114925 -- Bad error message when server preop trig blocks Git push

In earlier releases, if an AccuRev server preop trigger is set and the user doesn't have permission to push the code, GitCentric returned a bad error message. This has been fixed in the current release.

Now the user will get the trigger output as the message's preop trigger error.

1115883 -- Add Gerrit email parameters info

In earlier releases, the Gerrit review & settings URL were sent incorrectly in the mail. This has been fixed in the current release.

620823 -- GitCentric 2015.2 release notes state incorrect minimum AccuRev version required

In the GitCentric 2015.2 release notes, an incorrect minimum AccuRev version was mentioned. This is updated now.

The following fix was delivered as part 2017.1-HF1 release:

1109050 -- Getting odd error when creating new repo

In earlier releases, the user received an ambiguous error while creating a new repository with an existing name. Now, the message for the storage write exception has been updated to "Cannot create repository <Repo Name> : Permission Denied."

The following fix was delivered as part 2017.1-HF2 release:

1109336 -- GitCentric treats AccuRev elinks the same as slinks

In earlier releases, after a user commits a change to a file (which is of type symbolic link), no other committed changes could be pushed to GitCentric. This is fixed now.

Changes in this Release 2017.1

1097613 -- Default settings in log4j.properties cause sync failure

Certain synchronization operations require a larger log file to complete successfully. GitCentric increased the max log size to 100 MB in the log4j file. This issues has been resolved in the current release.

1101108 -- Commits page returns 500 Internal Server Errors when multi-byte characters in messages

When displaying the repository history with commit messages containing multi-byte characters, GitCentric would display a "500 Internal Server Error". This issue has been resolved in this release.

1104114,40990 -- Cloning of a repository via GitCentric is timing out

The gerrit.config sshd.idleTimeout configuration has been restored in this release. It can be used to configure timeout on clone/pull/push operations.

See https://gerrit- review.googlesource.com/Documentation/config-gerrit.html#sshd.idleTimeout for more information.

1104728, 41934 -- Problem pushing changes via GitCentric to AccuRev

Customer reported that a push to AccuRev was failing.This issues has been resolved in the current release.

624310 -- Missing xercesImpl.jar causing product failure with error 'SAXParser not found'

This issue has been resolved in this release.

624311 -- This issue has been described in several ways : Unable to negotiate ... no matching key exchange method found

"ssh gets an error saying unable to agree on KeyAlgorithms" that's just a restatement of the error message. The KeyAlgorithms he refers to are the key exchange method or algorithm.

"sshd needs to support NOT just diffie-hellman-group1-sha1 [for the ssh key exchange]" that is what we do in the workaround to solve the problem on the GC server side

Install GitCentric on any Linux server. Have a modern Linux (e.g. Ubuntu 16.04 or later) client communicate with that server via git clone/push/pull or via ssh commands. He will see the above errors. This issue has been resolved in this issue. (The workaround has been documented in Known Issues and Workarounds on page 31.)

624309 -- Simplify Code - Review

Create a repository. Turn code review on (using the Administration > Repo > Basic tab). Then edit some ACLs for that repository. It is now not possible to turn code review off. Solution is to replace graphic user interface (GUI) check box for code review on/off with two ACL-only repositories: Code- Review, and Direct-Push. By re-parenting the user repository to one of these two, he can turn code review on or off. This issues has been resolved in the current release.

1102562, 2844963 -- Delete file and replace with a directory causes push error

A certain sequence of git changes was causing a push error. This issues has been resolved in the current release.

1106810, 2873527, 43464 -- catalina.out growing without limit and fills the disk.

The information that is written to the KandoBridge and KandoGerrit logs was duplicated in the tomcat, catalina.out file. Since catalina.out doesn't get rotated or limited, the disk would fill up. Changes in this release mean that catalina.out only gets output that isn't available in kandoBridge or kandoGerrit logs."

Changes in Release 2015.2

The changes for GitCentric 2015.2 are enhancement-based, not issue-based. Therefore, there are no tracking issue numbers to list. The **Introduction** chapter explains the enhancement changes for this release.

Changes in Release 2015.1

This section documents changes in Release 2015.1.

33546 -- Files at top level directory not exported to Git if name starts with a period

In previous releases, when using GitCentric with AccuRev 6.1, files at the top level directory of a stream were not exported to Git if their name started with a period (.). This problem has been corrected in the current GitCentric release.

34353, 34786 -- GitCentric displays unpack error when pushing with Git 1.8.4 and later

In previous releases, if you were using Git 1.8.4.3 and later with Gerrit Code Review 2.6 and later, pushes would fail when the client pushed a reference to an object but not the object itself, as when you push an amended commit without changing the object, for example. In this case you might see an error like the following:

```
error: unpack failed: error Missing tree <SHA>
fatal: Unpack error, check server log
```

This problem has been corrected in the current release by setting the **gerrit.config** file parameter `receive.checkRefencedObjectsAreReachable = false` by default.

35018 (1095884)-- Pushing ignored files results in a protocol error

In previous releases, GitCentric displayed the following error if you were pushing to a branch that was mapped to an AccuRev stream that contained a **.acignore** file and one or more of the files being pushed was being ignored by a rule defined in the **.acignore** file:

```
fatal: protocol error: bad line length 2291
```

This problem has been corrected in the current release, and ignored files no longer result in an error when they are pushed.

35326 -- "Transaction Too High" error on export behavior corrected

In previous releases, when export triggers fired out of order, the error "Transaction Too High" would appear in the GitCentric GUI and in the **kandoBridge.log**, sometimes logging at the ERROR level. This behavior, however, is not an error, and the message has been changed to the following in the current release: "ignoring transaction already processed".

35437 -- GitCentric installer does not handle invalid DB Admin passwords

In previous releases, the GitCentric installer did not properly handle invalid DB Admin passwords, requiring the user to quit and restart the installer in order to enter the correct password. This problem has been corrected in the current release, and the installer now allows you to reenter the DB Admin password using the installer's **Back** button, avoiding the need to quit the installer.

35626 -- Changing the GitCentric server host name causes pushes to fail

In the previous release, changing the GitCentric server host name caused subsequent pushes to fail because existing workspaces were not recognized on the new host. This problem has been corrected in the current release.

36069 (1095727) -- Support for GitCentric and AccuRev servers on different hosts

In previous releases, specifying any value other than `localhost` for the GitCentric bridge host name in **gerrit.config** caused pushes to fail. This problem has been corrected in the current release, and you should now specify the machine name or IP address of the GitCentric bridge host.

36260 -- New: SSH command allows you to delete repositories

A new SSH command, `gitcentric delete-repo`, allows you to delete repositories. In addition to deleting the repository, this command also deletes GitCentric database records related to the repository and configuration information like AccuRev server settings and branch-stream mappings.

36559 (1096247)-- New: improvements to backup and restore

GitCentric 2015.1 introduces two new commands to perform backup and restore operations: `full_backup.sh` and `full_restore.sh`. These new commands address difficulties some users experienced using GitCentric's `backupdbs` and `restoredbs` commands run using `kandoMaintain`, especially when using them to migrate GitCentric from one server to another. Among the improvements provided by `full_backup.sh` and `full_restore.sh`:

- `full_backup.sh` backs up all repositories, databases, and configuration files needed to successfully migrate GitCentric from one machine to another. In the previous release, the .tar file created by the `backupdbs` command omitted Git repositories, Gerrit configuration files, and the database connection configuration file.

- On restoring GitCentric, you no longer have to manually copy the **AllProjects.git** file to the new server.

In addition, changes were made to both the `rmdb` and `restoredbs` commands, which now properly drop both databases required to restore GitCentric.

Micro Focus recommends that you use `full_backup.sh` and `full_restore.sh` to backup and restore GitCentric. See "Appendix B. Backup and Restore" in the *GitCentric User's Help* for more information.

Changes in Release 2014.1

This section documents changes in Release 2014.1.

31299 -- Improvements to the Status Monitor

The Status Monitor, which displays synchronization information for branch-stream mappings, has been improved for GitCentric 2014.1:

- The user interface is labeled.
- The **Status** row now shows the task name first, then the phase. For example, FullExport: streamToCommit - Process Elements.
- When a bridge task completes, the Status area now displays *Idle*.

32096 -- Improvements to the AccuRev Connector

The user interface and messages associated with the AccuRev Connector were enhance for GitCentric to improve usability.

32332 -- Installer improvements

The installer has a number of improvements, including:

- In previous releases, if the installer did not locate software required by GitCentric (postgres or AccuRev, for example), it simply displayed "**None**" in the Version column of the table listing required software. This problem has been corrected in the current release, and the

installer now displays "*<software tool name>* **not found**" when it is unable to locate required software. (33178)

- The console mode now allows you to specify alternate locations for Linux binary dependencies. (32332)

Note: The *uninstaller* now completely overwrites the existing GitCentric installation, including the GitCentric database. See GitCentric Installation Notes on page 6 for important information regarding this change. (32392, 32464, 33683)

32630 -- Incorrect syntax used for root paths

In previous releases, the GitCentric bridge did not display the mount point tree correctly. This problem has been corrected in the current release.

32742 -- Server properties checked each time Tomcat is started

An enhancement in GitCentric 2013.3 (31617) automated the specification of the GitCentric server's URL and UUID. An enhancement in GitCentric 2014.1 ensures that these settings are checked (and, if necessary, reset) each time the GitCentric bridge is restarted.

32879 -- GitCentric now uses JGit

Previous versions of GitCentric, including the GitCentric installer itself, relied on Git. GitCentric now uses an internal copy of JGit, removing the dependency on a installed Git.

32889 -- New: Tools for Browsing the Repository

New: Tools for Browsing the Repository. GitCentric 2014.1 introduces several new tools that make it easy to explore the repository:

- The *Commits page* lets you view the commit history of a specific branch or tag. The Commits page is displayed by default when you start GitCentric.
- The *Source Tree page* lets you review the files -- new, deleted, or modified -- associaed with a specific commit. Clicking a text file displays its contents in a file viewer, which can be useful when developers are troubleshooting problems with new code.
- The *Branches page* helps you determine how divergent branches in a repository have become, enabling you to take appropriate action on "dead" or "stale" branches (by merging or pruning, for example).

As part of this change, the GitCentric user interface has undergone several other changes to improve usability:

- The GitCentric user documentation is now displayed using the help (?) button.
- User account information is accessible using the **My Account** link in the user drop-down menu.
- Repository, group, and server administration functions are available from the **Administration** button on the top right of the GitCentric interface.
- GitCentric notes the repository and branch that are active at the time you log out, and uses that information to return to the selected repository and branch the next time you log in.

33068 -- Initial empty commits now performed automatically

GitCentric now enforces the best practice of making an initial empty commit when creating a repository.

33071 -- Clone command now uses SSH

In previous releases, GitCentric used the HTTP protocol as the default for `git clone` commands copied on the **General** panel of the **Repositories** page. GitCentric now uses the SSH protocol as the default for copied `git clone` commands.

33130 -- ACCUREV_HOME now set to a new directory

In previous releases, the bridge set the ACCUREV_HOME environment variable to `<tomcat_install>`/temp each time an AccuRev command was run. Using the /temp directory had the potential to interfere with the Secure Sockets Layer (SSL) protocol introduced in AccuRev 6.1. To resolve this problem, the GitCentric installation creates a new directory, `<gc_installer>`/ `AccuRevGitCentric/accurev`, and the bridge now sets ACCUREV_HOME to it.

Note that if you have implemented SSL for your AccuRev installation, the SSL certificates must be placed in `<gc_installer>`/AccuRevGitCentric/accurev/.accurev.

33247 -- XML replies to REST API calls are properly escaped

In previous releases, GitCentric did not properly escape special characters in the XML replies to REST API calls initiated by the GitCentric bridge -- typically success and failure messages -- and this sometimes caused an error during GitCentric operations. This problem has been corrected in the current release.

33267 -- Commit authors can now reuse email addresses

In previous releases, pushes would fail if the same email address was associated with multiple commit authors, as could be the case when one user used the same email address across multiple servers. This problem has been corrected in the current release: GitCentric now includes the AccuRev server ID when querying for the email address, which resolves most cases. In the event that a unique email address cannot be determined, GitCentric rejects the push and displays an error message indicating the reason.

33452 -- Initial export operation is retried when the AccuRev session expires

In previous releases, the initial export operation that populates a Git branch mapped to an AccuRev stream would sometimes time-out when the AccuRev stream was the source. Rather than trying to log in the GitCentric user and trying the operation again, GitCentric would behave as if the operation succeeded, but creating multiple 0-length files in the repository as a result.

This problem has been corrected in the current release, and if the GitCentric user session times out during a lengthy operation, it attempts to log in the GitCentric user and try the operation again.

33599 -- Security errors displayed in the GitCentric GUI

In previous releases, when security-related errors occurred (401 Unauthorized, 403 Forbidden, and 405 Method not allowed, for example) they were written only to the `kandoGerrit.log` file. In GitCentric 2014.1, security-related error messages are now displayed in a dialog box allowing for easier troubleshooting and diagnosis.

33731 -- Gerrit Code Review upgraded to 2.7

GitCentric 2014.1 uses Gerrit Code Review Version 2.7.

33760 -- Import History removed from GitCentric 2014.1

The ability to import the complete Git history of the repository into AccuRev, which was in Beta in the previous release, has been temporarily removed from GitCentric. This functionality is expected to be made available in a future release.

Known Issues

Micro Focus is aware of the following issues affecting the current GitCentric release.

636910 -- Git push fails due to presence of pipes (|) in http request

Due to a CVE-2016-6816 fix, when Tomcat is configured with GitCentric, the "|" character is disallowed, leading to HTTP 400 errors while performing push operations.
Workaround:

Enable configuration in *<tomcat_home>*`/conf/catalina.properties`, as follows:
`tomcat.util.http.parser.HttpParser.requestTargetAllow=|`
Restart the Tomcat server.

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.

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.

Table 5 Values that Require Double Quotes for Special Characters

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.

34757 -- Problem displaying Online help in Internet Explorer

Clicking the help button does not open the GitCentric online help in these versions of Internet Explorer:

- IE9, version 9.0.8112.16421
- IE10, version 10.0.9200.16899

For best results, use the latest version of Internet Explorer.

To work around this issue, add the following section to **gerrit.config** (in ***gc_home*/site/etc/**):

[receive]

checkReferencedObjectsAreReachable = false

Gerrit Issues

Since GitCentric (GC) incorporates Gerrit Code Review, you should check the Gerrit Code Review issues page for problems that you might encounter while using Gerrit Code Review in GitCentric.

Known Issues and Workarounds

User sees error "Unable to negotiate with <GC host> port 29418: no matching key exchange method found. Their offer: diffie-hellman-group1-sha1" when trying to clone/pull/push or execute an ssh command against the GC server.

This problem can either be solved with a workaround on each client that experiences the problem, or with one work-around on the GitCentric server.

Solution on client

1. Create a file config

   Run <text editor> `~/.ssh/config`.

2. Ensure that the line below are added in the file config :

```
Host *
KexAlgorithms diffie-hellman-group1-sha1
```

Note: This works for Linux client or Windows client with Git for windows only, does not work for built - in Git, such as TortoiseGit, or Eclipse Git plug-in.

Solution on server

The solution is to add the 'bouncy castle' jars to the project. Bouncy Castle is an open source encryption library written in Australia.

The three jars to be added to the project are :

• bcpg-jdk15on-155.jar

bcpkix-jdk15on-155.jar

• bcprov-ext-jdk15on-155.jar

Note: -155 indicates the version number.

Download the bouncy castle jar files from *https://www.bouncycastle.org/latest_releases.html*

There are two ways to add them to the project.

• One way - Copy the three jar files to `<tomcat install>/lib`

Note: Installing the jar files in this location might require re -configuration of other Web applications supported by the same Tomcat server.

• Alternative way -

1. Copy the three jar files into just the gitcentric.war file using the command line jar command, see:
   http://docs.oracle.com/javase/7/docs/technotes/tools/windows/jar.html

2. Save a copy of `<gc install>/bin/gitcentric.war` for a Roll Back.

3. Create subdirectory `<gc install>/bin/gitcentric.war WEB-INF/lib/` and copy the three jar files in it.

4. Run `jar uvf <gc install>/bin/gitcentric.war WEB-INF/lib/bc*.jar`

5. Delete subdirectory `WEB-INF/lib`.

6. Restart the tomcat server.

The steps taken above affects the **gitcentric.war** application only.

GitCentric Earlier Releases

GitCentric Release 2015.2

GitCentric Release 2015.2 includes:

- **GitCentric Multi-Threaded Bridge.** GitCentric 2015.2 allows the synchronization tasks inside the GitCentric Bridge to be Multi-Threaded. Previously, those sync operations were serialized. Customers had experienced push operations taking a long time - in some case as much as 10 minutes which would cause the push to time out and fail. This fix will eliminate that problem and generally improve performance on a busy bridge.

- **Increase kandoBridge log file size.** To prevent sync errors, the kandoBridge log file size has been increased.

- **Reduce noise in logs.** Changed some logs from level debug to level trace to reduce noise in the logs.

- **Better logging of AccuRev commands.** Improved the logging of AccuRev commands timing.

AccuRev GitCentric Release 2015.1 provides new features and enhancements, and it addresses issues reported in previous GitCentric releases. Highlights of GitCentric Release 2015.1 include:

- **New SSH command to delete repositories.** GitCentric 2015.1 supports deleting repositories using a new SSH command, `gitcentric delete-repo`. See "Appendix C. Command-Line Reference: in the *GitCentric User's Help* for more information.

- **New backup and restore commands.** GitCentric 2015.1 introduces two new commands for backup and restore operations, `full_backup.sh` and `full_restore.sh`. These commands provide broader functionality than the current `backupdbs` and `restoredbs` commands and make it easier to migrate an existing GitCentric installation to a new server. See "Appendix B. Backup and Restore" in the *GitCentric User's Help* for more information.

- **Support for GitCentric and AccuRev Servers on different hosts.** In previous releases, specifying any value other than `localhost` for the GitCentric bridge host name in **gerrit.config** prevented you from hosting GitCentric and AccuRev servers on different machines. This problem has been corrected in the current release, and you can now specify the machine name or IP address of the GitCentric bridge host.

For details about this release, including a comprehensive list of changes from previous GitCentric releases, see .

To install GitCentric, see .

GitCentric Multi-Threaded Bridge

Performance Impact Single Operation

Sync operations are timed one-at-a-time with the old single-threaded bridge and the new multi-threaded bridge. The timings were within one second with the two different bridges.

Performance Impact Multi-threaded operations

The framework was tested with five threads simultaneously causing sync operations and timed those sync   operations. With the multi-threaded bridge the sync operations were 50% slower than one-at-a-time operations. Expect some slowdown because the bridge, AccuRev server, and Gerrit server will be busier.

With the single-threaded bridge the five thread test sync operations were about three times slower than one-at-a-time sync operations. They were not five times slower because some of the activity in the Gerrit server and the AccuRev server is already multi-threaded.

The main realized objective is that push operations not be blocked for the duration of a long running operation.

Limitation: Operations within a Repo are Still Serialized

Synchronizations are run in parallel now except for synchronizations on the same repository. To avoid race   conditions that are repository specific data structures, those operations are serialized. This means that if a long running operation start on one branch of a repository (for example, and initial synchronization with a stream that has 20,000 files) and a push occurs on a second mapped branch in that same repository, that push will still block until the long running operation completes.

Configuration

To achieve best performance from GitCentric, correct configuration is necessary. For this test, modify the

<install>/site/etc/gerrit.config file's sshd section as follows:

```
[sshd]
        listenAddress = *:29418
        threads = 48
        batchThreads = 2
        streamThreads = 50
        maxAuthTries = 12
```

Increasing the number of threads might further improve GitCentric performance, if the server has sufficient cores. Use this web page as a guide: https://git.help.collab.net/entries/24136638-sshd-settings-in-Gerrit-   configuration.

We also modified the tomcat environment script in <accurev install>/WebUI/tomcat/bin/setenv.sh script to the following:

```
export CATALINA_OPTS="$CATALINA_OPTS -server -Xms256m -Xmx1024m"
```

Note that -Xmx1024m means maximum heap size 1024Meg. Given available memory, a larger heap will   further improve performance.