

.....

Apache Continuum

.....

The Apache Continuum Project

Documentation

1	Index (category)	1
2	Getting Started	2
3	Installation/Upgrade Guides	4
4	System Requirements	5
5	Installation	6
6	Standalone	7
7	Tomcat	11
8	JBoss	16
9	Jetty	18
10	Geronimo	20
11	Glassfish	22
12	Release Notes	
13	Upgrade	23
14	User's Guides	25
15	Managing Projects	26
16	Add a Project	27
17	Edit a Project	30
18	Remove a Project	32
19	Managing Build Definitions	33
20	Project Build Definition	34
21	Project Group Build Definition	35
22	Managing Notification	36
23	Mail Notification	38
24	IRC Notification	39
25	Jabber Notification	40
26	MSN Notification	41
27	Wagon Notification	42
28	Building a project	43
29	Release Management	44
30	Administrator's Guides	45
31	Managing Users and Security	46
32	Security Configuration	47
33	LDAP Configuration	48
34	Adding Project Group	52
35	Managing Builders	53
36	Managing JDKs	54
37	Managing Build Environments	55
38	Managing Schedules	57
39	Managing General Configuration	58
40	Managing Local Repositories	59
41	Managing Purge Configuration	60
42	Managing Parallel Builds	62
43	External databases	64
44	Monitoring Continuum	66
45	Appearance Configuration	67
46	Build Definition Template	69
47	Shutdown Continuum	70

1 Index (category)

1.1 Welcome to Apache Continuum

Apache Continuum is an enterprise-ready continuous integration server with features such as automated builds, release management, role-based security, and integration with popular build tools and source control management systems. Whether you have a centralized build team or want to put control of releases in the hands of developers, Continuum can help you improve quality and maintain a consistent build environment.

1.2 Documentation

1.2.1 Installation/Upgrade Guides

1.2.1.1 [Installation](#)

1.Standalone version

[Standard \(Linux, Mac OSX, Solaris, Windows, others\) as a service or not](#)

1.Webapp

[Tomcat, Jetty, JBoss, Geronimo, GlassFish, ...](#)

1.2.1.2 [Release Notes](#)

[Release Notes](#)

1.2.1.3 [Upgrade](#)

[How to upgrade from a previous version](#)

1.2.2 User's Guide

[User's Guide](#)

1.2.3 Administrator's Guide

[Administrator's Guide](#)

1.2.4 Developer's Guide

[Guide to building Continuum](#)

1.2.5 Knowledge Base

[Frequently Asked Questions](#)

[Wiki](#)

[Old Wiki](#)

[Blog](#)

2 Getting Started

2.1 Getting Started

When you start Continuum for the first time (without an existing database), the first thing you will do is create the admin account and perform the [General Configuration](#).

Create Admin User

Username:	admin
Full Name*:	<input type="text"/>
Email Address*:	<input type="text"/>
Password*:	<input type="password"/>
Confirm Password*:	<input type="password"/>

Admin account creation

After admin account creation, you can log as the admin. The next thing you will see is the General Configuration page.

ation

Path:	<input type="text" value="/path/to/continuum/data/working-directory"/>
	Enter the working directory of the Continuum web application
Path:	<input type="text" value="/path/to/continuum/data/build-output-directory"/>
	Enter the build output directory of the Continuum web application
Path:	<input type="text" value="/path/to/continuum/data/release-output-directory"/>
	Enter the release output directory of the Continuum web application
Path:	<input type="text"/>
	Enter the deployment repository directory of the Continuum web application
URL:	<input type="text" value="http://localhost:8080/continuum"/>
	Enter the base URL for the Continuum web application
Allowed Builds in Parallel:	<input type="text" value="1"/>
	Enter the number of Allowed Builds in Parallel
	<input checked="" type="checkbox"/> Enable Distributed Builds

General Configuration

You may also create users, [add projects](#), etc.

3 Installation/Upgrade Guides

3.1 Installation/Upgrade Guides

In this section, you'll find informations about Continuum installations:

- [System Requirements](#)
- [Installation in different environment \(standalone, webapp, service\)](#)
- [Release Notes](#)
- [How to upgrade from a previous version?](#)
- [Installation of Build Agent for Distributed Builds](#)

4 System Requirements

4.1 System Requirements

JDK	1.5 or above
Memory	No minimum requirement
Disk	The Continuum application package is less than 30MB but will use more disk space when it's checking out and building sources
Operating System	No minimum requirement. Tested on Windows XP, Debian, Fedora Core, Solaris and Mac OS X

5 Installation

5.1 Continuum Installation

In this section, you'll find all information about Continuum installation:

- [Standalone](#)
- [Tomcat](#)
- [JBoss](#)
- [Jetty](#)
- [Geronimo](#)
- [GlassFish](#)

6 Standalone

6.1 Installing Continuum Standalone

6.1.1 Basics & Fundamentals

- Download the standalone version from the [Download page](#)
- Extract the file
- Set a JAVA_HOME environment variable which use a jdk >= 1.5

6.1.1.1 Defining JNDI Resources

6. Mail server configuration

Before to start Continuum, you must configure your SMTP configuration for mail notification. The configuration to do is in `$(CONTINUUM_HOME)/conf/jetty.xml`:

```
<New id="validation_mail" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>mail/Session</Arg>
  <Arg>
    <New class="org.mortbay.naming.factories.MailSessionReference">
      <Set name="user"></Set>
      <Set name="password"></Set>
      <Set name="properties">
        <New class="java.util.Properties">
          <Put name="mail.smtp.host">localhost</Put>
        </New>
      </Set>
    </New>
  </Arg>
</New>
```

6. Databases configuration

By default, Continuum use an embedded [Derby](#) database. If you want to use an other database, you can modify the JNDI configuration in `$(CONTINUUM_HOME)/conf/jetty.xml`:

```
<!-- continuum database -->

<New id="continuum" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/continuum</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/continuum</Set>
      <Set name="user">sa</Set>
      <Set name="createDatabase">create</Set>
    </New>
  </Arg>
</New>
<New id="continuumShutdown" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/continuumShutdown</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
```

```

        <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/continuum</Set>
        <Set name="user">sa</Set>
        <Set name="shutdownDatabase">shutdown</Set>
    </New>
</Arg>
</New>

<!-- Users / Security Database -->

<New id="users" class="org.mortbay.jetty.plus.naming.Resource">
    <Arg>jdbc/users</Arg>
    <Arg>
        <New class="org.apache.derby.jdbc.EmbeddedDataSource">
            <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/users</Set>
            <Set name="user">sa</Set>
            <Set name="createDatabase">create</Set>
        </New>
    </Arg>
</New>
<New id="usersShutdown" class="org.mortbay.jetty.plus.naming.Resource">
    <Arg>jdbc/usersShutdown</Arg>
    <Arg>
        <New class="org.apache.derby.jdbc.EmbeddedDataSource">
            <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/users</Set>
            <Set name="user">sa</Set>
            <Set name="shutdownDatabase">shutdown</Set>
        </New>
    </Arg>
</New>

```

6.1.2 Installing as a Windows Service

- Go to `CONTINUUM_HOME/bin/` and run the following command:
`continuum.bat install`
- Edit the 'Apache Continuum' service
 - To see the services that are on your computer go to Start|Run and enter '`services.msc`'.
- Select the 'Startup Type'
- Go to the 'Log On' tab and select a real user. A real user is required because you'll need a home directory for maven repository and some other things
- Validate your changes

6.1.3 Installing as a Linux Service

Since the Continuum linux script `bin/linux/run.sh` understands the same arguments as linux boot scripts, there is no need to write a particular startup script to add Continuum to the linux boot process. All you need to do, as root, is:

6.1.3.1 Basic script in /etc/init.d

- Create a 'continuum' file under /etc/init.d/ with the following content (replacing continuum_user with the name of an account you have already created):


```
#!/bin/sh
CONTINUUM_HOME=/opt/continuum-1.2
su - continuum_user -c "$CONTINUUM_HOME/bin/continuum console $@ &"
```

6.1.3.2 In a Debian-based system

```
ln -s /usr/local/continuum-[VERSION]/bin/linux/run.sh /etc/init.d/continuum
```

At this point you have Continuum ready to be symlinked from different runlevels. This might sound a bit esoteric, but it is not, you will find these words very fast as soon as you start reading about the init process. Fortunately, Debian GNU/Linux comes with a very handy utility to create this links, just run as root:

```
update-rc.d -n continuum defaults 80
```

If you run this command, you will see something like this:

```
Adding system startup for /etc/init.d/continuum ...
/etc/rc0.d/K80continuum -> ../init.d/continuum
/etc/rc1.d/K80continuum -> ../init.d/continuum
/etc/rc6.d/K80continuum -> ../init.d/continuum
/etc/rc2.d/S80continuum -> ../init.d/continuum
/etc/rc3.d/S80continuum -> ../init.d/continuum
/etc/rc4.d/S80continuum -> ../init.d/continuum
/etc/rc5.d/S80continuum -> ../init.d/continuum
```

What you see is the symlinks that would be created. The above command didn't do anything because of the -n switch, remove it to get the real links created.

6.1.3.3 In a RedHat-based system

Configuring Continuum in a RedHat-based system (like Fedora Core) is slightly different: Instead of running update-rc.d, you need to add a new service using chkconfig. And in order to add Continuum to chkconfig, it is necessary to add some comments to the /etc/rc.d/init.d/continuum script and run a couple of commands; these tasks are automatically executed by running the chkconfig_install.sh script (note that _continuum_user_ needs to be replaced by the name of an account you have already created):

```
#!/bin/sh
#
# chkconfig_install.sh - install Continuum on a chkconfig-bases system
#
# Author: Felipe Leme <felipeal at apache.org>
#
# figure out what's Continuum's directory
CONTINUUM_HOME=`dirname $0`
cd ${CONTINUUM_HOME}
CONTINUUM_HOME=`pwd`
INITD_SCRIPT=/etc/rc.d/init.d/continuum
if [ -f ${INITD_SCRIPT} ]
then
  echo "File ${INITD_SCRIPT} already exists. Please remove it and try again."
  exit 1
fi
echo "Creating file ${INITD_SCRIPT}"
```

```
cat >> ${INITD_SCRIPT} <<EOF
#!/bin/sh
# chkconfig: 345 90 10
# description: Apache Continuum server
# uncomment to set JAVA_HOME as the value present when Continuum installed
#export JAVA_HOME=${JAVA_HOME}
if [ -z "\${JAVA_HOME}" ]
then
    echo "Cannot manage Continuum without variable JAVA_HOME set"
    echo "  (try to set it on file ${INITD_SCRIPT})"
    exit 1
fi
# run Continuum as root
cd ${CONTINUUM_HOME}
./run.sh \${*}
# run Continuum as user _continuum_user_
#su - _continuum_user_ -c "cd ${CONTINUUM_HOME}; ./run.sh \${*}"
EOF
chmod +x ${INITD_SCRIPT}
echo "Adding Continuum to chkconfig"
chkconfig --add continuum
echo "Enabling Continuum on chkconfig"
chkconfig continuum on
echo "Continuum set to start on run levels 3, 4 and 5."
echo "To start continuum now, run 'service continuum start'"
```

7 Tomcat

7.1 Guide to Install Continuum on Tomcat

Instructions for installing, deploying, configuring Continuum for the Apache Tomcat web container.

Sections:

7.1.1 Basics & Fundamentals

With every Tomcat version you will need a few things before you can deploy Continuum.

- 1 Use at least Java JDK 1.5.0_11 to run Tomcat w/Continuum.
- 2 A defined `<Context>` xml section to define the JNDI resources.
- 3 The JavaMail / Activation JAR files.
- 4 The Apache Derby JAR files.
- 5 Configure `${appserver.base}` java property.

7.1.1.1 Defining JNDI Resources.

Continuum will, on startup, ask the web container for a few JNDI configured resources, two JDBC DataSources, and one JavaMail session.

To configure these JNDI resources in the Tomcat Web Container, you will need to specify a `<Context>` section that Tomcat can utilize for those requests coming from Continuum.

Tomcat has 3 main ways to accomplish this (ordered by most recommended to least recommended)

- 1 Creating a `/$CATALINA_HOME/webapps/continuum/META-INF/context.xml` containing a `<Context>` element.
- 2 Creating a `/$CATALINA_HOME/conf/Catalina/localhost/continuum.xml` containing a `<Context>` element.
- 3 Adding a `<Context>` Section into the `/$CATALINA_HOME/conf/server.xml`

The following are the JNDI names you will need to provide:

- `mail/Session`
- `jdbc/continuum`
- `jdbc/users`

The individual techniques for describing these resources, and the parameters associated with them are specific to the Tomcat version, resource type, and even JDBC implementation type.

For the purposes of this document, the following assumptions are made.

- 1 You are an Apache Tomcat administrator.
- 2 You have an SMTP Server on localhost, port 25, with no login / password.
- 3 You will be using the embedded Apache Derby database. (not an external database, that's another show)
- 4 Details specific to Apache Tomcat, JavaMail, or Apache Derby are left for the reader to research on those projects websites.

7.1.1.2 The JavaMail / Activation JAR files

Note: Continuum requires JavaMail 1.4 (or later)

Apache Tomcat does not typically ship with a copy of the JavaMail or Activation jar files. In your role as the Apache Tomcat administrator of your installation, you will need to obtain these jar files and place it into your preferred lib directory.

The appropriate lib directory to choose is a personal preference, and we do not encourage or enforce a specific location for it, as all installations of Apache Tomcat are different.

For the record, we personally put them in the `$(CATALINA_HOME)/common/lib/` directory.

Direct download links for these jar files.

- JavaMail 1.4 - [mail-1.4.jar](#)
- Java Activation Framework 1.1 - [activation-1.1.jar](#)

7.1.1.3 The Apache Derby JAR files

Note:Continuum 1.2 has been tested with Apache Derby 10.1.3.1

The default installation of Continuum uses the Apache Derby 100% Java database to maintain Continuum-specific information, and also the Users / Security Database.

You will need to obtain the `derby.jar` and `derbytools.jar` and place them into your preferred lib directory.

We put them into the `$(CATALINA_HOME)/common/lib/` directory.

Direct download links for these jar files:

- [derby-10.1.3.1.jar](#)
- [derbytools-10.1.3.1.jar](#)

7.1.1.4 Configure the `appserver.base` java property

The `$(appserver.base)` java property is used by the Continuum internal logging configuration to determine where to output its logs to. It is important to define this property either in the `$(CATALINA_OPTS)` system environment variable (if Tomcat is being launched via the command line) or the service properties (if being launched as a service or daemon).

The format typically expected is `-Dappserver.base=<SOMEWHERE>`

You can utilize the `$(CATALINA_HOME)/bin/setenv.sh` script to set this value in a tomcat specific way.

```
#!/bin/bash
# Keep the appserver.home and appserver.base values the same when running under Tomcat
export CATALINA_OPTS="-Dappserver.home=$(CATALINA_HOME) -Dappserver.base=$(CATALINA_HOME)"
```

7.1.2 Tomcat 5.0.x Specifics

Tested on Tomcat v5.0.28

These instructions explain how to deploy the Continuum 1.2 web application in an existing installation of Tomcat 5.0.x.

Extra Jars:

- You will need the [xalan-2.7.0.jar](#) copied into your `$(CATALINA_HOME)/common/lib/` directory.
- The Xerces XML Implementation provided in Tomcat 5.0.x is old and will cause problems with Continuum and the internal JAXP implementation in JDK 1.5, we recommend that you remove the files in `$(CATALINA_HOME)/common/endorsed/`.
`<Context path="/continuum" docBase="/path/to/continuum-webapp-1.2.war" debug="0">`

```
<!-- JNDI Datasource for User/Security Database (REQUIRED) -->
<Resource name="jdbc/
users" auth="Container" type="javax.sql.DataSource"/>
<ResourceParams name="jdbc/users">
  <parameter>
    <name>driverClassName</name>
    <value>org.apache.derby.jdbc.EmbeddedDriver</value>
  </parameter>
  <parameter>
    <name>factory</name>
    <value>org.apache.commons.dbcp.BasicDataSourceFactory</
value> <!-- Sets up Database Connection Pooling -->
  </parameter>
  <parameter>
    <name>url</name>
    <value>jdbc:derby:database/users;create=true</
value> <!-- Adjust path to suit -->
  </parameter>
  <parameter>
    <name>username</name>
    <value>sa</value>
  </parameter>
  <parameter>
    <name>password</name>
    <value></value>
  </parameter>
</ResourceParams>
<!-- JNDI Datasource for Continuum Database (REQUIRED) -->
<Resource name="jdbc/
continuum" auth="Container" type="javax.sql.DataSource"/>
<ResourceParams name="jdbc/continuum">
  <parameter>
    <name>driverClassName</name>
    <value>org.apache.derby.jdbc.EmbeddedDriver</value>
  </parameter>
  <parameter>
    <name>factory</name>
    <value>org.apache.commons.dbcp.BasicDataSourceFactory</
value> <!-- Sets up Database Connection Pooling -->
  </parameter>
  <parameter>
    <name>url</name>
    <value>jdbc:derby:database/continuum;create=true</
value> <!-- Adjust path to suit -->
  </parameter>
  <parameter>
    <name>username</name>
    <value>sa</value>
  </parameter>
  <parameter>
    <name>password</name>
    <value></value>
  </parameter>
</ResourceParams>
```

```

    </ResourceParams>
    <Resource name="mail/
Session" auth="Container" type="javax.mail.Session"/>
    <ResourceParams name="mail/Session">
      <parameter>
        <name>mail.smtp.host</name>
        <value>localhost</value>
      </parameter>
    </ResourceParams>
  </Context>

```

7.1.3 Tomcat 5.5.x Specifics

Tested on Tomcat v5.5.17 and v5.5.25

This example <Context> assumes technique #2 in the [Define JNDI Resource](#) list. (This example lists out the docBase to the war file itself.)

```

<Context path="/continuum"
  docBase="/path/to/continuum-webapp-1.2.war">
  <Resource name="jdbc/users"
    auth="Container"
    type="javax.sql.DataSource"
    username="sa"
    password=""
    driverClassName="org.apache.derby.jdbc.EmbeddedDriver"
    url="jdbc:derby:database/users;create=true" />
  <Resource name="jdbc/continuum"
    auth="Container"
    type="javax.sql.DataSource"
    username="sa"
    password=""
    driverClassName="org.apache.derby.jdbc.EmbeddedDriver"
    url="jdbc:derby:database/continuum;create=true" />
  <Resource name="mail/Session"
    auth="Container"
    type="javax.mail.Session"
    mail.smtp.host="localhost" />
</Context>

```

Warning: The Tomcat 5.5.20 and 5.5.23 releases are missing MailSessionFactory and a few other classes. JNDI mail sessions will **not** work. Use Tomcat 5.5.17 or see the workaround on [Bug 40668](#).

7.1.4 Tomcat 6.0.x Specifics

Tested on Tomcat v6.0.14

```

<Context path="/continuum"
  docBase="/path/to/continuum-webapp-1.2.war">
  <Resource name="jdbc/users"
    auth="Container"
    type="javax.sql.DataSource"
    username="sa"
    password=""
    driverClassName="org.apache.derby.jdbc.EmbeddedDriver"
    url="jdbc:derby:database/users;create=true" />

```

```
<Resource name="jdbc/continuum"
  auth="Container"
  type="javax.sql.DataSource"
  username="sa"
  password=""
  driverClassName="org.apache.derby.jdbc.EmbeddedDriver"
  url="jdbc:derby:database/continuum;create=true" />
<Resource name="mail/Session"
  auth="Container"
  type="javax.mail.Session"
  mail.smtp.host="localhost" />
</Context>
```

8 JBoss

8.1 Guide to Install Continuum on JBoss

Instructions for installing, deploying, configuring Continuum in JBoss. Tested with JBoss 4.2.2.GA and 4.0.5.GA.

Sections:

8.1.1 DataSource Configuration

- Copy `derby-10.1.3.1.jar` into `$(JBOSS_HOME)/server/default/lib/`
- Create a JDBC deployment configuration file named `derby-continuum-ds.xml` in `$(JBOSS_HOME)/server/default/deploy` with the following contents:


```
<?xml version="1.0" encoding="UTF-8"?>
<datasources>
  <local-tx-datasource>
    <!-- The jndi name of the DataSource, it is prefixed with java:/ --
  >
    <!-- Datasources are not available outside the virtual machine -->
    <jndi-name>continuum</jndi-name>
    <!-- for in-process persistent db, saved when jboss stops. The
    org.jboss.jdbc.DerbyDatabase mbean is necessary for properly db shutdown --
  >
    <connection-url>jdbc:derby:../database/continuum;create=true</
connection-url>
    <!-- The driver class -->
    <driver-class>org.apache.derby.jdbc.EmbeddedDriver</driver-class>
    <!-- The login and password -->
    <user-name>sa</user-name>
    <password></password>
    <!-- The minimum connections in a pool/sub-
pool. Pools are lazily constructed on first use -->
    <min-pool-size>5</min-pool-size>
    <!-- The maximum connections in a pool/sub-pool -->
    <max-pool-size>20</max-pool-size>
    <!-- The time before an unused connection is destroyed -->
    <idle-timeout-minutes>5</idle-timeout-minutes>
    <!-- Whether to check all statements are closed when the connection is returned
        this is a debugging feature that should be turned off in production --
  >
    <track-statements/>
  </local-tx-datasource>
</datasources>
```
- Create a JDBC deployment configuration file named `derby-users-ds.xml` in `$(JBOSS_HOME)/server/default/deploy` with the following contents:


```
<?xml version="1.0" encoding="UTF-8"?>
<datasources>
  <local-tx-datasource>
    <!-- The jndi name of the DataSource, it is prefixed with java:/ --
  >
    <!-- Datasources are not available outside the virtual machine -->
    <jndi-name>users</jndi-name>
```

```

    <!-- for in-process persistent db, saved when jboss stops. The
    org.jboss.jdbc.DerbyDatabase mbean is necessary for properly db shutdown --
  >
    <connection-url>jdbc:derby:../database/users;create=true</
connection-url>
    <!-- The driver class -->
    <driver-class>org.apache.derby.jdbc.EmbeddedDriver</driver-class>
    <!-- The login and password -->
    <user-name>sa</user-name>
    <password></password>
    <!-- The minimum connections in a pool/sub-
pool. Pools are lazily constructed on first use -->
    <min-pool-size>5</min-pool-size>
    <!-- The maximum connections in a pool/sub-pool -->
    <max-pool-size>20</max-pool-size>
    <!-- The time before an unused connection is destroyed -->
    <idle-timeout-minutes>5</idle-timeout-minutes>
    <!-- Whether to check all statements are closed when the connection is returned
        this is a debugging feature that should be turned off in production --
  >
    <track-statements/>
  </local-tx-datasource>
</datasources>

```

With ../database used for the derby db path, databases will be created into \$JBOSS_HOME

8.1.2 Mail Configuration

Edit \$JBOSS_HOME/server/default/deploy/mail-service.xml

8.1.3 Install Continuum

To Install Continuum, you'll need the Continuum war.

Extract the Continuum war into \$JBOSS_HOME/server/default/deploy/continuum.war

8.1.4 Continuum Configuration

By default, the 'working directory' and the 'build output directory' are stored under the WEB-INF directory. If you want to change them (necessary on Windows due to the path length limitation), you can configure them in the [Configuration page](#).

By default, Continuum logs are stored into \$ appserver.base/logs/. appserver.base is a system property. If you don't define it in the JBoss startup script, it will be empty, so the Continuum logs directory will be at the root of your disk. If you want to use another location, you should modify \$JBOSS_HOME/server/default/deploy/continuum.war/WEB-INF/classes/log4j.xml

9 Jetty

9.1 Guide to Install Continuum on Jetty

Instructions for installing, deploying, configuring Continuum for Jetty. Tested with Jetty 6.1.5.

Sections:

9.1.1 Datasource Configuration

- Create a JDBC resource named '**jdbc/continuum**'
- Create a JDBC resource named '**jdbc/users**'

For derby you can copy the following configuration in your \$JETTY_HOME/etc/jetty.xml

For other databases see [PostgreSQL](#) and [MySQL](#)

```

<!-- ===== --
>
<!-- Continuum datasources --
>
<!-- ===== --
>
<New id="continuum" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/continuum</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName">your_directory/continuum_database</
Set>
      <Set name="user">sa</Set>
      <Set name="createDatabase">create</Set>
    </New>
  </Arg>
</New>
<New id="continuum_shutdown" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/continuum_shutdown</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName">your_directory/continuum_database</
Set>
      <Set name="user">sa</Set>
      <Set name="shutdownDatabase">shutdown</Set>
    </New>
  </Arg>
</New>
<New id="users" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/users</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName">your_directory/users_database</Set>
      <Set name="user">sa</Set>
      <Set name="createDatabase">create</Set>
    </New>
  </Arg>
</New>

```

```

<New id="users_shutdown" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/users_shutdown</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName">your_directory/users_database</Set>
      <Set name="user">sa</Set>
      <Set name="shutdownDatabase">shutdown</Set>
    </New>
  </Arg>
</New>

```

9.1.2 Mail Configuration

The following xml can be used for the jetty mail configuration:

```

<!-- ===== -->
<!-- Mail configuration -->
<!-- ===== -->
<New id="validation_mail" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>mail/Session</Arg>
  <Arg>
    <New class="org.mortbay.naming.factories.MailSessionReference">
      <Set name="user"></Set>
      <Set name="password"></Set>
      <Set name="properties">
        <New class="java.util.Properties">
          <Put name="mail.smtp.host">localhost</Put>
          <Put name="mail.from">continuum@localhost</Put>
          <Put name="mail.senderName">Continuum</Put>
          <Put name="mail.debug">>true</Put>
        </New>
      </Set>
    </New>
  </Arg>
</New>

```

9.1.3 Install Continuum

To Install Continuum, you'll need the Continuum war.

Copy or extract the Continuum war into \$JETTY_HOME/webapps-plus/

9.1.4 Start Jetty

To start Jetty, you'll need to use \$JETTY_HOME/etc/jetty.xml **and** \$JETTY_HOME/etc/jetty-plus.xml due to the datasources and mail configuration.

9.1.5 Continuum Configuration

By default, the 'working directory' and the 'build output directory' are stored under the WEB-INF directory. If you want to change them (necessary on Windows due to the path length limitation), you can configure them in the [Configuration page](#).

10 Geronimo

10.1 Guide to Install Continuum on Geronimo

Instructions for installing, deploying, configuring Continuum for Apache Geronimo.

Sections:

10.1.1 Datasource Configuration

Continuum uses two datasources (jdbc/continuum and jdbc/users). You can create one or two databases for them.

Here, we create only one database 'continuum' used by both datasources. Look at [Configuring database pools](#) in Geronimo User's Guide.

10.1.2 Deployment Plan

When your datasource is created, before you install Continuum, you must create a **Geronimo Deployment Plan** like this one (Geronimo 2.0.2):

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app
  xmlns="http://geronimo.apache.org/xml/ns/j2ee/web-1.1">
  <environment>
    <moduleId>
      <groupId>org.apache.continuum</artifactId>
      <artifactId>continuum</artifactId>
      <version>1.2</version>
    </moduleId>
    <dependencies>
      <dependency>
        <groupId>console.dbpool</groupId>
        <artifactId>continuum</artifactId>
      </dependency>
      <dependency>
        <groupId>org.apache.geronimo.configs</groupId>
        <artifactId>javamail</artifactId>
        <version>2.0.2</version>
        <type>car</type>
      </dependency>
      <dependency>
        <groupId>commons-lang</groupId>
        <artifactId>commons-lang</artifactId>
        <version>2.2</version>
        <type>jar</type>
      </dependency>
    </dependencies>
    <hidden-classes>
      <filter>org.apache.commons.lang.</filter>
      <filter>org.jaxen.</filter>
    </hidden-classes>
  </environment>
</context-root>/continuum</context-root>
```

```

    <resource-ref>
      <ref-name>jdbc/continuum</ref-name>
      <resource-link>continuum</resource-link>
    </resource-ref>
    <resource-ref>
      <ref-name>jdbc/users</ref-name>
      <resource-link>continuum</resource-link>
    </resource-ref>
    <resource-ref>
      <ref-name>mail/Session</ref-name>
      <resource-link>mail/MailSession</resource-link>
    </resource-ref>
  </web-app>

```

The 'continuum' string used in `console.dbpool` and `<resource-link>` is the name of the datasource created in the previous part.

`commons-lang 2.2` is required by Continuum and must be installed in the geronimo repository. `org.jaxen.` needs to be added as a `<filter>` in the `<hidden-classes>` section because of [GERONIMO-3894](#)

10.1.3 Install Continuum

To Install Continuum, you'll need the Continuum war and the deployment plan.

Look at [Installing and removing applications](#) in Geronimo.

10.1.4 Continuum Configuration

By default, the 'working directory' and the 'build output directory' are stored under the WEB-INF directory. If you want to change them (necessary on Windows due to the path length limitation), you can configure them in the [Configuration page](#).

11 Glassfish

11.1 Guide to Install Continuum on GlassFish V2

Instructions for installing, deploying, configuring Continuum for GlassFish.

Sections:

11.1.1 Datasource Configuration

- Create a connection pool named '**continuum**' for your database
- Create a JDBC resource named '**jdbc/continuum**' linked to the '**continuum**' connection pool
- Create a JDBC resource named '**jdbc/users**' linked to the '**continuum**' connection pool

11.1.2 Install Continuum

To Install Continuum, you'll need the Continuum war.

Look at [Procedure To Deploy From the Admin Console](#) in GlassFish.

11.1.3 Continuum Configuration

By default, the '`working directory`' and the '`build output directory`' are stored under the WEB-INF directory. If you want to change them (necessary on Windows due to the path length limitation), you can configure them in the [Configuration page](#).

12 Upgrade

12.1 Upgrading Continuum

This document will help you upgrade Continuum from 1.2.x to 1.3.2.

When upgrading Continuum, it could have some database model changes. The Data Management tool exports data from the old database model and imports the data into the new database model.

12.1.1 Upgrading from 1.2.x to 1.3.2

There are 2 databases that need to be considered: one for the builds and one for the users.

There were no changes in the users database from 1.2.x to 1.3.2, so you can simply point Continuum 1.3.2 at your existing user database.

The builds database has had model changes, and will need to be exported and imported.

First, download the Data Management tools you will need. The tool is a standalone jar that you can download from the central repo.

You will need to download two versions of the tool, one for the export out of the old version and one for the import into the new version:

Note: The 1.2, 1.2.2 and 1.2.3 released versions of this tool have a bug. To export databases from 1.2.2 or 1.2.3, you will need to use version 1.2.3.1 of the tool. To export databases from 1.2, you may use the 1.1 version of the tool.

- <http://repo1.maven.org/maven2/org/apache/continuum/data-management-cli/1.3.2/data-management-cli-1.3.2-app.jar>
- <http://repo1.maven.org/maven2/org/apache/continuum/data-management-cli/1.2.3.1/data-management-cli-1.2.3.1-app.jar>
- <http://repo1.maven.org/maven2/org/apache/maven/continuum/data-management-cli/1.1/data-management-cli-1.1-app.jar>

Next, follow these steps to export data from the old version

- Stop the old version of Continuum
- Execute this command to create the builds.xml export file


```
java -Xmx512m -jar data-management-cli-1.2.x-app.jar -
buildsJdbcUrl jdbc:derby:${old.continuum.home}/data/databases/
continuum -mode EXPORT -directory backups
```

Then, follow these steps to import the data to the new version

- Start the new version of Continuum to create the new data model, but do not configure it.
- Stop Continuum
- Execute this command to import the builds data from the xml file you created earlier:


```
java -Xmx512m -jar data-management-cli-1.3.2-app.jar -
buildsJdbcUrl jdbc:derby:${new.continuum.home}/data/databases/
continuum -mode IMPORT -directory backups
```

Finally, be aware that sometimes the NEXT_VAL values in the SEQUENCE_TABLE need to be adjusted.

- Before starting Continuum for the first time after the import, connect to the db with a client like [Squirrel SQL](#) and check the values in the NEXT_VAL column of the SEQUENCE_TABLE.
- Values must be greater than the max id value in each table.

- For example, the next value of "org.apache.maven.continuum.model.Project" must be greater than the greatest id in Project table.
- Here are some example SQL statements. You may need to add or remove lines depending on the contents of your database.

```
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(systemconfiguration_id)+1 from SYS
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(id)+1 from BUILDQUEUE) WHERE SEQUE
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(id)+1 from SCHEDULE) WHERE SEQUENC
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(id)+1 from BUILDDEFINITION) WHERE S
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(id)+1 from LOCALREPOSITORY) WHERE S
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(id)+1 from PROJECTGROUP) WHERE SEQ
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(scmresult_id)+1 from SCMRESULT) WHI
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(projectdependency_id)+1 from PROJE
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(id)+1 from BUILDDEFINITIONTEMPLATE
UPDATE SEQUENCE_TABLE set NEXT_VAL = (select max(id)+1 from ABSTRACTPURGECONFIGURAT
```

Now you can start your new version of Continuum.

13 User's Guides

13.1 User's Guides

- [Building Projects](#)
- [Cancelling Builds](#)
- [Managing Build Definitions](#)
- [Managing Projects](#)
- [Notifications](#)
- [Releasing Projects](#)

14 Managing Projects

14.1 Managing Projects

- [How to add a project](#)
- [How to edit a project](#)
- [How to remove a project](#)

15 Add a Project

15.1 Add a Project

15.1.1 Maven 2 project

From the menu, choose the 'Maven 2.0.x Project' entry

Add a maven2 project

The pom.xml file must be available through one of the following protocols: http, https, or ftp (The file protocol is also supported but is disabled by default for security and must be enabled manually).

Or it can be uploaded from a local file (Note this doesn't support multi modules project).

Add a maven2 project

You can define username/password if the POM URL requires authentication.

If your SCM stores credentials like CVS or SVN and you want to use the SCM Credentials cache, check the "Use SCM Credentials Cache" field.

You can define the group you want to use or "Defined by POM" in this case project.name will be used as Project Group.

By default, Continuum adds each sub-module POM as an individual Continuum Project. If you want to add only the root POM without sub-modules, check the "load only root as recursive build" field.

You can choose a Build Definition Template which will be applied to your project. If you choose a Build Definition Template, the build definitions from the template will be added in addition to any build definitions that are defined at the project group level.

15.1.2 Maven 1 project

From the menu, choose the 'Maven 1.x Project' entry

Add a maven1 project

The project.xml file must be available through one of the following format : http, https and ftp (file protocol is off by default for security and must be enabled manually).

Or it can be uploaded (Note this doesn't support multi modules project).

The project you're adding can't use a parent.

Add a maven1 project

You can define username/password if the POM URL require an authentication.

If your SCM store credentials like CVS or SVN and you want to use the SCM Credentials cache, check the "Use SCM Credentials Cache" field.

You can define the group you want to use or "Defined by POM." In this case project.name will be used as the Project Group.

You can choose a Build Definition Template which will be applied to your project.

15.1.3 ANT Project

From the menu, choose the 'Ant Project' entry

Add a Ant project

TO WRITE

Add a Ant project

15.1.4 Shell Project

From the menu, choose the 'Shell Project' entry

Add a shell project

TO WRITE

Add a shell script project

15.1.5 Add a project from the Project Group

From the Project Group, you can add a project without using the menu. With this operation, the Project Group will be set to the current group.

Add a project from a project group

15.1.6 Scm hints

15.1.6.1 ClearCase

With ClearCase, you can configure SCM things in few ways. For example, you can use, in the scm URL in your POM, the absolute path of your config spec file like this:

```
<scm>
  <connection>scm:clearcase:absolute_path_of_config_spec</connection>
</scm>
```

The SCM URL format used for ClearCase is defined [here](#)

and you can create a `clearcase-settings.xml` file under `$ user.home/.scm/` with the following content:

```
<clearcase-settings>
  <viewstore>\\mymachine\myvwstore</viewstore>
  <useVWSPParameter>true</useVWSPParameter>
</clearcase-settings>
```

This configuration won't work with each ClearCase installation because each ClearCase configuration is different, so we recommend reading the [ClearCase page](#) on the Maven-SCM site.

16 Edit a Project

16.1 Edit a Project

16.1.1 Project View

When you edit a project from the Group Summary page, you will see basic information. The first part of the page contains information regarding:

- project name
- project version
- project scm url
- project Last Build Date
- all build definitions attached to the project

Project View Top

With the 'Edit' button, you can change some project informations like the scm url if your project was moved. With the 'Build Now' button, you build the project manually with the default build definition.

Second part contains information regarding:

- project notifiers
- project dependencies
- project developpers

Project View Bottom

16.1.2 Build Results

You can view all project's Build Results with the link Builds You will see all build results summary : start/end Time, result. It's possible to delete some build results with checkboxes

Build Results

Now you can edit a build result

16.1.3 Build Result

Editing a build result will display :

- scm changes (author and file path)

- project dependencies (project recorded in this continuum instance)
- detail of the build definition used
- the full build command line output

Build Result

17 Remove a Project

17.1 Remove a Project

To remove a project, go to the Project Group Summary and click the **delete** icon.

Delete icon

You can also use the 'Delete group' button in the Project Group Summary to remove the whole project group.

18 Managing Build Definitions

18.1 Managing Build Definitions

Each project or project group must have at least one attached build definition in order to build.

Depending on the project, you can define different values which will be used in order to build your project.

Continuum includes some default build definitions which can be changed in the [Build Definition Templates screen](#).

You can add or edit build definitions at the [Project Group level](#) or at the [Project level](#).

19 Project Build Definition

19.1 Project Build Definition

The project level build definition is executed to a specific project where the build definition is defined.

Project Build Definitions

To add build definition to your project:

- 1 Select a project within the group.
- 2 Under the **Build Definitions** section in the **Project Information** screen, click **Add**.
- 3 Supply the values for the necessary fields.
 - Enter your **POM filename**, which is very likely the same as previous builds, `pom.xml`.
 - In the **Goals** text box, enter your new goals. For example, `clean site deploy` or `release:branch`.
 - You may have arguments for Maven, such as `--batch-mode --non-recursive`. Add your arguments in the Arguments text box or leave it empty.
 - Check **Build Fresh** to do a fresh checkout from CVS before the build process. Check **Always Build** if you want to execute the goal every time you build the project. If you want this build definition to be your default for building your project, check the **Is it default?** check box.
 - Select your build schedule from the pull-down menu. There will only be the `DEFAULT_SCHEDULE` if you have not defined any others.
 - Select from the pull-down menu the **Build Environment** for the build goal.
 - Select the **Type** from the pull-down menu.
 - You can supply a short description of the build definition (**Definition**).
- 4 Click **Save**.

20 Project Group Build Definition

20.1 Project Group Build Definition

The project group level build definition is executed on all of the projects in the group.

Project Group Build Definitions

To add build definition to your project group:

- 1 Under the **Build Definitions** tab in the **Project Information** screen, click **Add**.
- 2 Supply the values for the necessary fields.
 - Enter your **POM filename**, which is very likely the same as previous builds, `pom.xml`.
 - In the **Goals** text box, enter your new goals. For example, `clean site deploy or release:branch`.
 - You may have arguments for Maven, such as `--batch-mode --non-recursive`. Add your arguments in the Arguments text box or leave it empty.
 - Check **Build Fresh** to do a fresh checkout from CVS before the build process. Check **Always Build** if you want to execute the goal every time you build the project. If you want this build definition to be your default for building your project, check the **Is it default?** check box.
 - Select your build schedule from the pull-down menu. There will only be the `DEFAULT_SCHEDULE` if you have not defined any others.
 - Select from the pull-down menu the **Build Environment** for the build goal.
 - Select the **Type** from the pull-down menu.
 - You can supply a short description of the build definition (**Definition**).
- 3 Click **Save**.

21 Managing Notification

21.1 Managing Notification

Notifiers can be attached to a Maven project by adding them to the pom in the [ciManagement section](#).

```
<ciManagement>
  <system>continuum</system>
  <url>http://127.0.0.1:8080/continuum</url>
  <notifiers>
    <notifier>
      <type>mail</type>
      <sendOnError>true</sendOnError>
      <sendOnFailure>true</sendOnFailure>
      <sendOnSuccess>false</sendOnSuccess>
      <sendOnWarning>false</sendOnWarning>
      <configuration>
        <address>continuum@127.0.0.1</address>
      </configuration>
    </notifier>
  </notifiers>
</ciManagement>
```

Or they can be added to a Project Group with the Notifiers link:

Notifiers Link

Or to a project with the Add button on the Edit Project page:

Notifier Button

Here you select between Notifier implementations provided with Continuum:

Notifier Type Choice

You have the following choices:

- [Mail](#)
- [IRC](#)
- [Jabber](#)

- [MSN](#)
- [Wagon](#)

22 Mail Notification

22.1 Mail Notification

If you are using Maven 2, email notifiers configured in pom.xml will be automatically added, and you may have as many as you need.

```
<ciManagement>
  <system>continuum</system>
  <notifiers>
    <notifier>
      <configuration>
        <address>email address</address>
      </configuration>
    </notifier>
  </notifiers>
</ciManagement>
```

Or you can manually add mail notifiers at the project group or individual project level.

On the **Add Notifier** form select **Mail** from the pull-down menu and click Submit.

Add Email Notifier

You must either provide a single email address, or select to notify the latest committers.

In addition you may select one or more "Send on..." options.

Then, click **Save**.

23 IRC Notification

23.1 IRC Notification

- 1 Under Notifiers from the Project Information page, click Add.
- 2 On the Add Notifier form select IRC from the pull-down menu and click Submit.
- 3 On the next page, fill out the form with the following information:
 - IRC Host
 - IRC Port
 - IRC Channel
 - Nick Name
 - Alternate Nick Name
 - User Name
 - Full Name
 - Password which can be left blank if you did not specify one in your IRC account

Add IRC Notifier

- 4 Select the 'Send on ...' options you want.
- 5 Click Save. The IRC notifier you just added will be listed under Notifiers on the Project Information page along with the other notifiers you may have set up prior to this.

24 Jabber Notification

24.1 Jabber Notification

- 1 Under Notifiers from the Project Information page, click Add.
- 2 Select Jabber from the pull-down menu and click Submit.
- 3 On the Add/Edit Jabber Notifier page, fill out the form with the following:
 - Jabber Host
 - Jabber Port is typically 5222
 - Jabber Login
 - Jabber Password
 - Jabber Domain Name is the fully qualified domain name of your Jabber login
 - Jabber Recipient Address

Add Jabber Notifier

The Jabber login and Jabber recipient address must be different. The Jabber recipient must also be added in the contact list of the sender (at least for Google talk).

- 4 Select the "Send on ..." options you want.
- 5 Click Save. The Jabber notifier you just added will be listed under Notifiers on the Project Information page along with the other notifiers you may have set up prior to this.

25 MSN Notification

25.1 MSN Notification

- 1 Under Notifiers from the Project Information page, click Add.
- 2 On the Add Notifier form select, MSN from the pull-down menu and click Submit.
- 3 On the next page, fill out the form with the following:
 - MSN Login
 - MSN Password
 - MSN Recipient Address

Add MSN Notifier

- 4 Select the 'Send on ...' options you want.
- 5 Click Save. The MSN notifier you just added will be listed under Notifiers on the Project Information page along with the other notifiers you may have set up prior to this.

26 Wagon Notification

26.1 Wagon Notification

- 1 Under Notifiers from the Project Information page, click Add.
- 2 On the Add Notifier form select Wagon from the pull-down menu and click Submit.
- 3 On the next page, enter the Project Site URL.

It must be a Wagon URL like the URL used in `distributionManagement`. For the Webdav protocol, you can use `dav:[http_url]` or `file:///path_to_the_directory` if you want to copy it locally with the file protocol.

- 4 Enter the `ServerId`. This should match the value of the `<id>` in the server configuration found in `settings.xml`. Sample configuration follows:

```
<server>
  <username>admin</username>
  <password>admin123</password>
  <id>continuum.site</id>
</server>
```

Using the server configuration in `[user_home]/.m2/settings.xml` is a workaround for user authentication using the wagon notifier.

The `<username>` and `<password>` should be set to the user account that has the **Site Producer** and **Site Observer** roles.

Add Wagon Notifier

- 5 Select the "Send on ..." options you want.
- 6 Click Save. The Wagon notifier you just added will be listed under Notifiers on the Project Information page along with the other notifiers you may have set up prior to this.

The file protocol will create `buildresult.txt` file locally in the location specified in the Wagon Notifier configuration. If you used the Webdav protocol, a `buildresult.txt` file will be copied to your disk, relative to where you told it to be placed when you set up the Wagon notifier.

You can also view the `buildresult.txt` file from a browser at the Project Site URL you specified.

27 Building a project

27.1 Building Projects

For Ant and maven builds the following system properties will be added via -D command line:

- `continuum.project.group.name`
- `continuum.project.lastBuild.state` : last build state int flag
- `continuum.project.lastBuild.number`
- `continuum.project.nextBuild.number`
- `continuum.project.id`
- `continuum.project.name`
- `continuum.project.version`

Projects can be built in two ways:

- 1 [Scheduled Builds](#)
- 2 [Forced Builds](#)

28 Release Management

28.1 Releasing Projects

The release process has two steps, [Prepare](#) and [Perform](#).

- 1 First, it prepares your project for release by doing the following:
 - Checks that your project is in a "releasable" state.
 - Updates the POMs for the new development version.
 - Runs a test build.
 - Generates a tag for the release.
 - Generates the artifact to be deployed, such as a jar file.
 - Generates the site and deploys it.
 - Commits the new POMs to the new development version. It will do the commit to the SCM connection URL you specify (explained later).
- 2 Then the release management system performs the release by checking out a clean copy of the tagged release and deploying the artifacts and site.

28.1.1 Modifying POM for Release

The source code control system you use must be specified in your POM file in an `scm` entry similar to the one shown here:

```
<scm>
  <connection>
    scm:svn:http://svn.apache.org/repos/asf/maven/components/trunk
  </connection>
  <developerConnection>
    scm:svn:https://svn.apache.org/repos/asf/maven/components/trunk
  </developerConnection>
  <url>http://svn.apache.org/viewcvs.cgi/maven/components/trunk</
url>
</scm>
```

To determine what to enter for your connection URL (the " `<connection>` " tag) go to the link <http://maven.apache.org/scm/scms-overview.html> for a list of supported SCMs and click on your source code control system to see the appropriate syntax for your particular software.

In general the format for an SCM URL is:

```
scm:<scm_provider><delimiter><provider_specific_part>
```

In the example above, the `scm_provider` is "svn" (Subversion). The `delimiter` is a colon. And the rest of the line is the " `provider_specific_part`".

It is important to prepare the release just before performing the release. The reason these processes have been separated into two steps is to allow the user to confirm that everything has been set correctly before the actual release. If between the prepare and performing of the release there is a change to the SCM tree, an error will occur causing the prepare process to restart.

29 Administrator's Guides

29.1 Administrator's Guides

- [Managing Users and Security](#)
- [Adding a Project Group](#)
- [Managing Builders](#)
- [Managing JDKs](#)
- [Managing Build Environments](#)
- [Managing Build Agents](#)
- [Managing Schedules](#)
- [Managing General Configuration](#)
- [Managing Local Repositories](#)
- [Managing Purge Configuration](#)
- [Managing Parallel Builds](#)
- [External Databases](#)
- [Monitoring Continuum](#)
- [Appearance Configuration](#)
- [Build Definition Templates](#)
- [Shutting Down Continuum](#)
- [Understanding Distributed Builds](#)

30 **Managing Users and Security**

30.1 Managing Users and Security

[Security Configuration](#)

[LDAP Configuration](#)

31 Security Configuration

31.1 Continuum Security Configuration

Security properties and password rules can be configured in the `security.properties` file, which by default is searched for in:

- `~/m2/security.properties`
- `$CONTINUUM_HOME/conf/security.properties`

(In the list above, `~` is the home directory of the user who is running Continuum, and `$CONTINUUM_HOME` is the directory where Continuum is installed, such as `/opt/continuum-1.2.`)

Following are some of the properties you can modify. For a complete list, consult the default properties file in Redback's svn repo: [config-defaults.properties](#)

```
# Security Policies
#security.policy.password.encoder=
security.policy.password.previous.count=6
security.policy.password.expiration.days=90
security.policy.allowed.login.attempt=3
# Password Rules
security.policy.password.rule.alphanumeric.enabled=false
security.policy.password.rule.alphacount.enabled=true
security.policy.password.rule.alphacount.minimum=1
security.policy.password.rule.characterlength.enabled=true
security.policy.password.rule.characterlength.minimum=1
security.policy.password.rule.characterlength.maximum=8
security.policy.password.rule.musthave.enabled=true
security.policy.password.rule.numericalcount.enabled=true
security.policy.password.rule.numericalcount.minimum=1
security.policy.password.rule.reuse.enabled=true
security.policy.password.rule.nowhitespace.enabled=true
```

Note: If installed standalone, Continuum's list of configuration files is *itself* configurable, and can be found in: `$CONTINUUM_HOME/apps/continuum/webapp/WEB-INF/classes/META-INF/plexus/application.xml`

32 LDAP Configuration

32.1 LDAP Configuration

Continuum supports LDAP for authentication. To configure it, you should follow these steps:

- Shutdown Continuum
- Open `apps/continuum/webapp/WEB-INF/classes/META-INF/plexus/application.xml` and uncomment the following xml

```
<!--
  Ldap Authentication can be enabled by setting enabling these components and sett
  =====
  user.manager.impl=ldap
  ldap.bind.authenticator.enabled=true
  redback.default.admin=admin
  security.policy.password.expiration.enabled=false
  ldap.config.hostname=ldap.hostname
  ldap.config.port=389
  ldap.config.base.dn=o=com
  ldap.config.context.factory=com.sun.jndi.ldap.LdapCtxFactory
  ldap.config.bind.dn=uid=myusername,o=com
  ldap.config.password=s3cr3t
  #ldap.config.authentication.method=
  =====
  * ldap.config.hostname - The hostname of the ldap server
  * ldap.config.port - The port of the ldap server
  * ldap.config.base.dn - The baseDn of the ldap system
  * ldap.config.context.factory - context factory for ldap connections
  * ldap.config.bind.dn - the core user used for authentication the ldap server, m
  * ldap.config.password - password for the bindDn for the root ldap connection
  until this process is better documented, the following is the document for confi

  http://redback.codehaus.org/integration/ldap.html
-->
<!--

this component manages the connection to the ldap server
-->
<!-- component>
  <role>org.codehaus.plexus.redback.common.ldap.connection.LdapConnectionFactory</
role>
  <role-hint>configurable</role-hint>
  <implementation>org.codehaus.plexus.redback.common.ldap.connection.ConfigurableLd
implementation>
  <requirements>
    <requirement>
      <role>org.codehaus.plexus.redback.configuration.UserConfiguration</
role>
    </requirement>
  </requirements>
</component-->

<!--
```

```

this component manages the mapping of attributes in ldap to user information in red
=====
ldap.config.mapper.attribute.email=mail
ldap.config.mapper.attribute.fullname=givenName
ldap.config.mapper.attribute.password=userPassword
ldap.config.mapper.attribute.user.id=cn
ldap.config.mapper.attribute.user.base.dn=
ldap.config.mapper.attribute.user.object.class=inetOrgPerson
ldap.config.mapper.attribute.user.filter=(attributeName=value)
=====
* email-
attribute - The name of the attribute on a user that contains the email address
* full-name-
attribute - The name of the attribute on a user that contains the users fullName
* password-
attribute - The name of the attribute containing the users password, used for the authen
* user-id-
attribute - The name of the attribute containing the users userId, most commonly cn or
* user-base-dn - The base dn that will be subtree searched for users.
* user-object-
class - the objectClass used in the ldap server for indentifying users, most commonly i
-->

<!-- component>
<role>org.codehaus.plexus.redback.common.ldap.UserMapper</role>
<role-hint>ldap</role-hint>
<implementation>org.codehaus.plexus.redback.common.ldap.LdapUserMapper</
implementation>
<configuration>
  <email-attribute>email</email-attribute>
  <full-name-attribute>givenName</full-name-attribute>
  <password-attribute>userPassword</password-attribute>
  <user-id-attribute>cn</user-id-attribute>
  <user-base-dn>o=com</user-base-dn>
  <user-object-class>inetOrgPerson</user-object-class>
</configuration>
<requirements>
  <requirement>
    <role>org.codehaus.plexus.redback.configuration.UserConfiguration</
role>
  </requirement>
</requirements>
</component-->

<!--

If caching is desired to improve performance then make uncomment this and make sure

user.manager.impl=cached
-->

<!-- component>

```

```

    <role>org.codehaus.plexus.redback.users.UserManager</role>
    <role-hint>cached</role-hint>
    <implementation>org.codehaus.plexus.redback.users.cached.CachedUserManager</
implementation>
    <description>CachedUserManager</description>
    <requirements>
      <requirement>
        <role>org.codehaus.plexus.redback.users.UserManager</role>
        <role-hint>ldap</role-hint>
        <field-name>userImpl</field-name>
      </requirement>
      <requirement>
        <role>org.codehaus.plexus.cache.Cache</role>
        <role-hint>users</role-hint>
        <field-name>usersCache</field-name>
      </requirement>
    </requirements>
  </component-->

<!--
if using the user manager authenticator to authenticate the user and not the ldap b
this definition has the correct password encoder

Note: you should probably just use the ldap bind authenticator which is enabled by

ldap.bind.authenticator.enabled=true

in the security.properties
-->

<!-- component>
  <role>org.codehaus.plexus.redback.policy.UserSecurityPolicy</role>
  <role-hint>default</role-hint>
  <implementation>org.codehaus.plexus.redback.policy.DefaultUserSecurityPolicy</
implementation>
  <description>User Security Policy.</description>
  <requirements>
    <requirement>
      <role>org.codehaus.plexus.redback.configuration.UserConfiguration</
role>
      <field-name>config</field-name>
    </requirement>
    <requirement>
      <role>org.codehaus.plexus.redback.policy.PasswordEncoder</role>
      <role-hint>sha1</role-hint>
      <field-name>passwordEncoder</field-name>
    </requirement>
    <requirement>
      <role>org.codehaus.plexus.redback.policy.UserValidationSettings</
role>
      <field-name>userValidationSettings</field-name>
    </requirement>
  </requirements>

```

```

    <requirement>
      <role>org.codehaus.plexus.redback.policy.CookieSettings</role>
      <role-hint>rememberMe</role-hint>
      <field-name>rememberMeCookieSettings</field-name>
    </requirement>
    <requirement>
      <role>org.codehaus.plexus.redback.policy.CookieSettings</role>
      <role-hint>signon</role-hint>
      <field-name>signonCookieSettings</field-name>
    </requirement>
    <requirement>
      <role>org.codehaus.plexus.redback.policy.PasswordRule</role>
      <field-name>rules</field-name>
    </requirement>
  </requirements>
</component-->

```

All you need to configure is the `ConfigurableLdapConnectionFactory` and `LdapUserMapper` components

- Add a `security.properties` file under `$CONTINUUM_HOME/conf/` with the following content:


```

user.manager.impl=ldap
ldap.bind.authenticator.enabled=true
redback.default.admin=adminuser
security.policy.password.expiration.enabled=false
user.manager.impl=cached
ldap.bind.authenticator.enabled=true

```

adminuser is a LDAP user and will be the default Continuum admin. **guestuser** is a LDAP user and will be used for the `guest` role, generally, it is an utility LDAP account.
- Restart Continuum

32.1.1 Other resources

[Redback LDAP page](#)

33 Adding Project Group

33.1 Adding Project Group

From the home page (Group Summary) Use the the 'Add Project Group' button

Add Project Group

Then You will have the project Group detail.

Edit Project Group detail

Some fields are mandatory :

- Project Group Name
- Project Group Id

34 Managing Builders

34.1 Managing Building Tool

From the menu, choose the 'Installations' entry

Installations

Here you must choose the Installation Type you want to add (here a Tool)

Installation Type Choice

You must configure the tool you want to add

Tool Setup

You can use the checkbox if you want to add a Profile with the same name as your tool name.

The value 'Value/Path' field must contains the path to the tool :

- For maven2 : it must be similar to your M2_HOME
- For maven1 : it must be similar to your MAVEN_HOME
- For ant : it must be similar to your ANT_HOME

A control will be made to validate the path value :

- For maven2 : path/bin/mvn -v will be tested
- For maven1 : path/bin/maven -v will be tested
- For ant : path/bin/ant -v will be tested

If the test, the following error will be displayed

Installation validation failed

35 Managing JDKs

35.1 Managing JDKs

From the menu, choose the 'Installations' entry

Installations

Here you must choose the Installation Type you want to add (here a Tool)

Installation Type Choice

You must configure the tool you want to add

Tool Setup

You can use the checkbox if you want to add a Profile with the same name as your jdk name.

The value 'Value/Path' field must contain the jdk path (as a `JAVA_HOME` value).

The value will be validated by testing path `/bin/java -version`

If the test fails, the following error will be displayed

Jdk validation failed

36 Managing Build Environments

36.1 Build Environment

36.1.1 Goal

The main goal is to compile/test sources against the targeted jvm, mvn/maven version, ant version and to add defined envvar (MAVEN_OPTS, ANT_OPTS ...). This is now possible at the build definition level (in 1.1).

36.1.2 Setup Build Environments/Installations

36.1.2.1 Creating a Build Environment

From the menu, choose the 'Build Environments' entry

Build Environment Menu

Enter a name

Add Build Environment

36.1.2.2 Attach JDK to a Build Environment

The list box contains all available installations (jdk, mvn etc..). The installation's type is automatically recognized. In the modification mode of a build environment, adding an already attached type will override the existing one.

Add JDK to Build Environment

36.1.2.3 Link Build Environment to Build Definition

The list box contains all available installations (jdk, mvn etc..). The installation's type is automatically recognized. In the modification mode of a build environment, adding an already attached type will override the existing one.

Setup Build Environment

36.1.2.4 Installations used in Notification mail

The notification mail contains the following informations :

- **Java Home version** : coming from \$JAVA_HOME/bin/java -version.
- **Builder version** : coming from builder used version arg (mvn used -v, ant use -version).

```
Java Home version :  
    java version "1.4.2_06"  
    Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.2_06-  
b03)  
    Java HotSpot(TM) Client VM (build 1.4.2_06-b03, mixed mode)
```

```
Builder version :  
    Maven version: 2.0.5
```

37 Managing Schedules

37.1 Managing Schedules

Schedules initiates the building of projects continuously. In this section, adding, modifying, and deleting of schedules are discussed.

Clicking the **Schedules** link from the menu on the left will display the following page:

List of schedules page

37.1.1 Adding Schedules

To add a schedule, follow these steps:

- 1 Click the **Add** button below the list of schedules.
- 2 Fill in the needed information in the **Edit Schedules** page shown below.

Adding a build schedule

- **Name** The unique identifier of the schedule. This is a required field.
 - **Description** Brief description of the configured schedule. This is a required field.
 - **Cron Expression** Determines the exact time interval that the build schedule executes.
 - **Maximum job execution time (seconds)** This sets the maximum time for a job to be executed. This is a required field.
 - **Quiet Period (seconds)** This a setting that delays the build if there has been a commit in the defined number of seconds prior. This is useful when using CVS, since commits are not atomic and a developer might be committing midway through an update. It is not typically needed if using Subversion.
 - **Enabled** When selected, scheduled build will be executed.
- 3 Click **Save**.

37.1.2 Editing Schedules

To edit a schedule, follow these steps:

- 1 Click the edit icon which can be found to the right of the schedule.
- 2 Modify the fields in the **Edit Schedules** page.
- 3 Click **Save**.

37.1.3 Deleting Schedules

To delete a schedule, click the delete icon, the rightmost icon inline with the schedule to be deleted.

38 Managing General Configuration

38.1 Managing General Configuration

The first time you start Continuum, you will see the 'General Configuration' page. This page is preconfigured with default values.

General Configuration

Working Directory	The directory where all projects will be checked out
Build Output Directory	The directory where all build output will be stored
Release Output Directory	The directory where all release output will be stored
Deployment Repository Directory	The directory where generated maven2 artifacts will be stored. This directory will be respect a repository structure. It is independant of the maven deploy phase
Base URL	The base Continuum URL. This URL is used in notifications
Number of Allowed Builds in Parallel	The number of build queues that will be utilized to build projects. This allows concurrent checkouts and builds of multiple projects. Available since version 1.3.1.
Enable Distributed Builds	When checked, this Continuum instance will delegate all builds to configured agents.

You can modify values or keep defaults then save.

Although Continuum defaults to storing its data within the installation, it is a best practice to separate the installation and data directories.

These values can be pre-configured in `continuum.xml`. Make sure that Continuum is not running when editing this file.

You can modify values at a later time with the following steps:

- 1 Login to Continuum with an Administrator account
- 2 Click the **Configuration** link under the **Administration** section

General Configuration Menu

- 3 Click the **Edit** button

39 Managing Local Repositories

39.1 Local Repository

Click the **Local Repositories** link under the **Administration** section

Local Repositories

You will see all the local repositories that have been created.

To display:

In this screen you can add/edit/delete/purge a local repository.

You can purge a repository if it has a default [Purge Configuration](#), otherwise the purge icon is disabled.

disabled purge icon

39.1.1 Adding / Editing a Local Repository

All fields are mandatory.

- Name: must be a unique repository name.
- Location: must be the absolute path of a unique repository location.
- Layout: "default" or "legacy". Default layout is for maven2, while legacy layout is for maven1.

Adding a local repository will automatically create a default [Purge Configuration](#) for that repository.

39.1.2 Default

The default local repository location comes from the settings of the user running Continuum. The location cannot be edited within Continuum, but can be changed by specifying it in `~/ .m2/ settings.xml`.

40 Managing Purge Configuration

40.1 Purge Configuration

Click the **Purge Configurations** link under the **Administration** section.

Purge Configurations

You will see all repository purge configurations and all directory purge configurations.

Here you can add/edit/delete the purge configuration.

You can also purge the repository or directory by clicking the corresponding purge icon of the purge configuration.

purge now icon

40.1.1 Adding / Editing Repository Purge Configuration

Specify the repository to purge.

40.1.2 Adding / Editing Directory Purge Configuration

There are two types of directory purge configuration:

- releases - this will purge the working copies associated with releases.
- buildOutput - this will purge the build output of each project.

40.1.3 Purging Repository

- Delete all - will delete the entire contents of the repository
- Days Old - will delete the artifact if it's older than the number of days old but still satisfies the number of retention count.

Example:

Days_Old = 30 days

Retention_Count = 2

If I have 5 artifacts that are older than 30 days, only 3 of them will be deleted because I have to retain any 2 artifacts (Retention count).

- Retention Count - the number of artifacts to retain; all in excess will be deleted
- Delete Released Snapshots - will delete all released snapshots

40.1.4 Purging Directory

- Delete all

- Releases Directory - will delete all releases-* directories
- Build Output Directory - will delete all build output of each project
- Days Old - will delete the directory if it's older than the number of days old
- Retention Count - number of directories to retain

40.1.5 Steps in purging:

- Perform Delete All purging if it is checked.
- If delete all is not checked, then is days old has a value greater than 0? If it has then perform days old purging.
- If days old is 0 then perform retention count purging.
- Delete released snapshots purging will always be performed if it is checked, unless the Delete All is also checked or it's a directory purging.

41 Managing Parallel Builds

41.1 Managing Parallel Builds

Since 1.3.1, Continuum can be configured to checkout and build projects in parallel or concurrently.

41.1.1 Configuring Parallel Builds

The number of build queues to be used when checking out or building projects can be configured in the Configuration page, via the Number of Allowed Builds in Parallel field (see [Managing General Configuration](#)). The value should be greater than one (1) to be able to create a build queue aside from the default.

Another thing to note is that if Distributed Builds is enabled, Parallel Builds is automatically disabled. To make it short, you're only allowed to use one of these functionalities at a time in the current implementation.

41.1.2 Adding a Build Queue

By default, there is already a DEFAULT_BUILD_QUEUE that cannot be deleted configured in Continuum. To add more build queues..

Click the **Build Queue** link under the **Administration** section.

Build Queue

You will see all the build queues.

Build Queues list

Here you can add/delete the build queue. Take note that you would only be allowed to create N number of build queues, where N = the Number of Allowed Builds in Parallel set in the Configuration page.

Input the name for the build queue and save.

Add Parallel Build Queue

41.1.3 Attaching Build Queues to a Schedule

Build queues can be attached to a schedule. When a build is triggered, Continuum gets the build queues attached to the build definition's associated schedule. The project build will then be queued to the build queue with the least number of tasks.

To attach build queues to a schedule:

- 1 Click the **Schedules** link under the **Administration** section. From the schedules list page, create a new schedule or edit an existing one.
- 2 In the Add Build Queue field, move build queues from the left hand list box to the right hand list box by clicking the appropriate buttons in between the two list boxes. All build queues on the right hand list box will be attached to the schedule when you click Save.

Edit Schedule

41.1.4 Viewing Checkouts and Builds in Queues

Current checkouts and builds, and all other projects queued in the checkout and build queues can be viewed in the **Queues** page. These executing and queued tasks can also be cancelled by clicking the corresponding Cancel icons for each task. Multiple-cancel is also supported.

Queues

41.1.5 Limitations

Below are some feature limitations for the current implementation:

- 1 Project groups cannot be built multiple times simultaneously.
- 2 Concurrent build of inter-dependent projects in a group is not yet supported.

42 External databases

42.1 How to use an external database?

42.1.1 Supported databases

- Derby
- MySQL
- MS SQL Server
- Oracle (not yet, but coming soon)

42.1.2 Configure Continuum

42.1.2.1 Standalone version

To use an external database with Continuum standalone, you must configure DataSources in \$CONTINUUM_HOME/conf/jetty.xml

```
<!-- continuum database -->

<New id="continuum" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/continuum</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/continuum</Set>
      <Set name="user">sa</Set>
      <Set name="createDatabase">create</Set>
    </New>
  </Arg>
</New>

<New id="continuumShutdown" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/continuumShutdown</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/continuum</Set>
      <Set name="user">sa</Set>
      <Set name="shutdownDatabase">shutdown</Set>
    </New>
  </Arg>
</New>

<!-- Users / Security Database -->

<New id="users" class="org.mortbay.jetty.plus.naming.Resource">
  <Arg>jdbc/users</Arg>
  <Arg>
    <New class="org.apache.derby.jdbc.EmbeddedDataSource">
      <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/users</Set>
      <Set name="user">sa</Set>
```

```
        <Set name="createDatabase">create</Set>
    </New>
</Arg>
</New>
<New id="usersShutdown" class="org.mortbay.jetty.plus.naming.Resource">
    <Arg>jdbc/usersShutdown</Arg>
    <Arg>
        <New class="org.apache.derby.jdbc.EmbeddedDataSource">
            <Set name="DatabaseName"><SystemProperty name="appserver.base" default=".."/
>/data/databases/users</Set>
            <Set name="user">sa</Set>
            <Set name="shutdownDatabase">shutdown</Set>
        </New>
    </Arg>
</New>
```

42.1.2.2 Webapp

To use an external database with the Continuum webapp, you should configure the DataSource in your container.

43 Monitoring Continuum

43.1 Monitoring a Continuum instance

43.1.1 JConsole

- Edit `$CONTINUUM_HOME/bin/[platform]/wrapper.conf` to add:
`wrapper.java.additional.9=-Dcom.sun.management.jmxremote`
- Start Continuum
- Start jconsole, (included with Sun JDK 1.5 and later,) select the 'WrapperSimpleApp' line, and click Connect

The following image shows starting Continuum 1.1, adding a project group, then adding and building the trunk of the Shale Framework (14 modules):

The following image shows the same Continuum instance running for several hours:

43.1.2 Garbage Collection and Heap Size

Edit `$CONTINUUM_HOME/bin/[platform]/wrapper.conf` to add:

```
wrapper.java.additional.10=-verbose:gc
or
wrapper.java.additional.10=-Xloggc:gc.txt
```

The first sends to the console (and wrapper log) while the second sends to the filename that you tell it to.

Example output:

```
420.568: [GC 62459K->58474K(65104K), 0.0074933 secs]
420.778: [GC 62634K->58733K(65104K), 0.0060762 secs]
420.919: [GC 62893K->58832K(65104K), 0.0034433 secs]
421.015: [GC 62992K->58723K(65104K), 0.0026420 secs]
421.067: [GC 62883K->60885K(65104K), 0.0107248 secs]
421.140: [GC 65045K->62114K(66384K), 0.0122194 secs]
421.153: [Full GC 62114K->32136K(66384K), 0.3591207 secs]
421.655: [GC 36296K->32242K(65104K), 0.0037377 secs]
421.754: [GC 36402K->33061K(65104K), 0.0037182 secs]
421.835: [GC 37221K->33120K(65104K), 0.0028722 secs]
421.930: [GC 37280K->34582K(65104K), 0.0056601 secs]
422.074: [GC 38742K->35150K(65104K), 0.0072451 secs]
         before->after (total), time
```

This shows the heap size before and after garbage collection as well as the total heap size and the time for the GC run.

43.1.3 References

- [<http://java.sun.com/developer/technicalArticles/J2SE/jconsole.html>]

44 Appearance Configuration

44.1 Configure Appearance

44.1.1 Banner

You can configure the right logo of the banner including title and link on the image. You have to add some information in a pom (like a company pom) (coming from org.apache:apache in this example) :

```
<organization>
  <name>The Apache Software Foundation</name>
  <url>http://www.apache.org/</url>
</organization>
<properties>
  <organization.logo>http://www.apache.org/images/asf_logo_wide.gif</
organization.logo>
</properties>
```

From the menu, choose the 'Appearance' entry :

Appearance

Use the 'Select a Company POM' link :

'Select a Company POM'

The result will be displayed (using org.apache:apache in this example) :

'Selected a Company POM'

Note : the pom is searched in central repository and the repositories available for the active profiles from your \$ user.home/.m2/settings.xml

44.1.2 Footer

You can configure the footer by putting your own html content in the 'Appearance' entry of the menu.

'Configure footer'

The default value is :

```
<div class="xright">Copyright &copy; 2005-  
${currentYear}&nbsp;The Apache Software Foundation</  
div> <div class="clear"><hr/></div>
```

45 Build Definition Template

45.1 Build Definition Template

From the menu, choose the 'Build Definition Templates' entry

Build Definition Templates

You will see all build definition templates and the build definitions which can be added in templates. In this screen you can add/edit/update/delete a build definition (**the Continuum defaults can be deleted**)

Build Definition Templates

In the build definition template edit screen, you can easily one or more build definitions.

Build Definition Template Edit

46 Shutdown Continuum

46.1 Shutdown Continuum

Shutting down Continuum while a project is building or queued is NOT recommended. The build queues page should be used to cancel any currently running and queued builds before shutting down.

46.1.1 Queues view

From the menu, choose the 'Queues' link

Queues

Then you'll get the Queues page

Queues

In this page you can see the current build, projects in the build queue, the current checkout and projects in the checkout queue. For each line, you can cancel the process

46.1.2 Shutdown

To shutdown Continuum, the queue must be empty, so when you want to shutdown, you can either wait until all builds are complete or cancel them.

If you don't want to have new projects added in the build queue before you shutdown Continuum, you can disable all schedules. You'll need to re-enable them on the next startup. (In the future, you'll have an action to automate the disable/enable schedules process.)

47 Developer's Guides

47.1 Developer's Guides

- [Building Continuum](#)
- [XML-RPC](#)

48 Building Continuum

48.1 Guide to building Continuum

48.1.1 Why would I want to build Continuum?

You might want to build Continuum yourself for one of two reasons:

- to try out a bleeding edge feature or bugfix (issues can be found in [JIRA](#)), but you can try our SNAPSHOTs: <http://vmbuild.apache.org/>
- to fix a problem you are having and submit a patch to the development team.

Note that you don't need to build Continuum for day to day use. While we encourage getting involved and fixing bugs that you find, for day to day use we recommend using the latest GA release.

48.1.2 Checking out the sources

All of the source code for Continuum and its related libraries is in a [Subversion](#) repository. You can also [browse the repository](#), or checkout specific modules directly.

All SVN instructions are available on the [Source Repository page](#).

48.1.3 Building the sources

48.1.3.1 Prerequisites

- JDK 5 or greater
- Maven 2

48.1.3.2 Building

To build Continuum, you run this command from the top (trunk) directory:

```
mvn clean install
```

49 XML-RPC

49.1 Guide to use XML-RPC with Continuum

49.1.1 Introduction

In this section, you'll learn how to connect to a Continuum instance and how to do some action on projects.

49.1.2 Requirements

To connect to a Continuum instance, you must use the `continuum-xmlrpc-client` jar.

This library have some others jars as dependencies, so the best way to start the development of a Continuum xmlrpc client is to create a maven2 project with the following dependencies:

```
<dependency>
  <groupId>org.apache.continuum</groupId>
  <artifactId>continuum-xmlrpc-client</artifactId>
  <version>YOUR_CONTINUUM_VERSION</version>
</dependency>
```

49.1.3 Connection to Continuum

To connect to your Continuum with the client API, you must use the `ContinuumXmlRpcClient` class.

The constructor use 3 parameters:

- url, the url of the xmlrpc listener that is `http://host:port/continuum/xmlrpc`
- user, a Continuum user
- password, the user's password

```
URL url = new URL( "http://localhost:8080/continuum/xmlrpc" );
ContinuumXmlRpcClient client = new ContinuumXmlRpcClient( url, username, password )
```

49.1.4 Getting project groups list

You have two ways to get the project groups list. The first is to get only a summary of groups and the second returns groups with details. If you don't need all informations, we recommend to use the first way, so you'll save time to get datas and memory on the server.

```
List<ProjectGroupSummary> pgs = client.getAllProjectGroups();
List<ProjectGroup> pgs = client.getAllProjectGroupsWithProjects();
```

49.1.5 Getting projects in a group

```
List<ProjectSummary> ps = client.getProjects( projectId );
```

If you already have a `ProjectGroup` or `ProjectGroupSummary` object, you can access to the project group id with this:

```
int projectId = pg.getId();
```

49.1.6 Building all projects in a group

- With the default build definition


```
client.buildGroup( projectId );
```

- With a build definition
`client.buildGroup(projectId, buildDefinitionId);`

49.1.7 Building a project

- With the default build definition
`client.buildProject(projectId);`
- With a build definition
`client.buildProject(projectId, buildDefinitionId);`
Note: When you start a build, the project is put in the Continuum queue and will be built when all projects added previously in the queue will be built.

49.1.8 Triggering a build

In some case, users want to use the push build technique with a hook in their SCM, so when a developer will commit some files, a build will be triggered. To do this, you can write a simple xmlrpc client that will use a project id as parameter and you'll use the `buildProject(...)` method described above.

This method will start a forced build.

49.1.9 Removing a project

A project can be removed by supplying the project id.

```
List<ProjectSummary> projects = client.getProjects( projectId );
...
client.removeProject( ps.getId() );
```

49.1.10 Removing build results

Currently, to remove a build result, you need the entire build result (rather than just the summary).

```
List<BuildResultSummary> results = client.getBuildResultsForProject( ps.getId() );
...
BuildResult br = client.getBuildResult( ps.getId(), brs.getId() );
client.removeBuildResult( br );
```

49.1.11 Editing build queue

You can edit all projects which are in the build queue or check if one project is currently in the build queue.

```
List<BuildProjectTask> prjsInBuildQueue = client.getProjectsInBuildQueue();
client.isProjectInBuildingQueue( int projectId );
```

49.1.12 Backup

With the Continuum xmlrpc client, you can backup a full Continuum instance (without users database, for the moment).