

Clearspace Installation Guide

This document describes how to install Clearspace, an application for supporting communities that want to create and share content and collaborate on projects.

If you haven't already, you might be interested in reading the README.html file included at the root of the Clearspace distribution. It provides a set of streamlined installation instructions in addition to listing known issues.

Clearspace Distributions. Clearspace is downloadable in two distributions. For more information, see [Files in the Distributions](#).

If you're administering your Clearspace installation, be sure to check out the Clearspace Administrators' Guide. Setting up Clearspace is pretty easy through its user interface alone, but you might be interested in the information the Guide provides about what can be done (by both users and administrators) in Clearspace.

This document includes the following sections:

[Installation Overview](#) — A high-level view of the installation steps.

[System Requirements](#) — Describes the technologies that Clearspace needs and supports.

[Files in the Distributions](#) — A list of what's included in the distribution.

[Database Setup](#) — Setting up a database to support Clearspace.

[Application Server Setup](#) — How to set up your application server to support Clearspace.

[Setting the jiveHome Directory](#) — Finding a good location for the jiveHome directory.

Installation Overview

To complete the installation of Clearspace, you'll need to perform each of the following steps:

1. Review the [system requirements](#) to make sure your system supports Clearspace.
2. If you're not using the embedded database, [prepare your database](#) for Clearspace.
3. If you're using the WAR File, [set up your application server](#) by installing the application files and JDBC driver.
4. [Set the jiveHome directory](#).
5. It is recommended that you allocate extra memory for Clearspace when starting your application server. E.g. -Xmx512M

After you finish installing Clearspace and start the server, use its web-based setup tool to establish a database connection and to set up the administrator account. Clearspace will display the setup tool the

first time you navigate to Clearspace using a web browser. The URL you use will vary depending on the application server technology you're using. By default for a local installation with the included server (standalone distribution), the URL will be <http://localhost:8080/clearspace>.

After you complete the setup tool, use the admin console to begin setting up your Clearspace installation for use by the community.

System Requirements

Because Clearspace is a pure Java application, it will run on any platform where Java (JDK 1.5 or better) is installed. Clearspace requires a Java application server and a database. The following lists supported technologies and recommended system configuration.

- Operating Systems
 - Windows XP or 2003
 - Linux, Solaris, FreeBSD
 - Intel Mac OS X
- Application Servers (see [Application Server Setup](#) for application server-specific installation instructions)
 - Apache Tomcat (5.5 or better recommended)
 - Caucho Resin (3.x or better recommended)
 - BEA WebLogic
 - JBoss Application Server
 - Jetty Server
 - Any other application server that supports the Servlet 2.3 and JSP 1.2 Spec.
- Databases (see [Database Setup](#) for specific information about databases)
 - MySQL (4.x or better recommended)
 - Oracle (9i or better recommended)
 - Postgres (7.x or better recommended)
 - IBM DB2 (v7 or better recommended)
 - SQL Server (2000 or better recommended)
 - HSQL DB (also used as the embedded database for evaluations)
 - Any other database that has a robust JDBC 2.0 compliant driver
- Environment
 - We recommend a server with at least 1GB of RAM and at least one 1.5 Ghz processor.
 - An optimal deployment for a larger community would have 2GB of RAM and 2 CPUs for additional thread performance.
 - For optimal performance we recommend the application and database servers be hosted separately.
 - Clearspace easily integrates with a LDAP repository or an Active Directory.

Additional Recommendations

When you run a server-side application, you should also have a daily backup solution. At a minimum you should back up your database on a regular basis as well as the configuration files for Clearspace (note: those are stored in one directory).

Files in the Distributions

The files in your distribution will differ depending on whether you downloaded the standalone distribution or WAR distribution. Among the things you'll find in both distributions are:

- A database directory contains SQL scripts to create new Clearspace databases. See [Database Setup](#) for more information.
- A jiveHome directory where Clearspace stores configuration files and additional resources.
- A webservices directory that contains information and files you can use to build clients that access Clearspace web services.

Standalone Distribution

The standalone distribution includes an application server. This distribution requires the least amount of installation effort and is a great choice for evaluating Clearspace. For the quickest installation (such as for evaluation), use the included application server and embedded database. If you're doing this, installation is complete when you extract the distribution's contents as shown in the hierarchy below.

If you're going to be using a separate database technology (rather than the embedded database), see [Database Setup](#) for further installation instructions.

From a command prompt, use the BAT (for Windows) and SH (for Unix-based OSes) files to start and stop Clearspace on the included app server.

```
jive_clearspace_standalone_x_x_x/  
| - database/  
| - jiveHome/  
| - server/  
| - webservices/  
| - README.html  
| - start-clearspace.bat  
| - start-clearspace.sh  
| - stop-clearspace.bat  
| - stop-clearspace.sh
```

WAR Distribution

The WAR distribution does not include an application server. This distribution is intended for deployment on the application server of your choice. Clearspace supports several app server technologies as describes in the [System Requirements](#).

The clearspace.war file is a standard Web Application Archive (WAR) that contains the Clearspace application.

```
jive_clearspace_war_x_x_x/  
| - database/  
| - jiveHome/  
| - webservices/  
| - clearspace.war  
| - EditWAR.jar
```

Database Setup

Clearspace stores data about users, documents, spaces, and so on in a database. You can use either your own database (Clearspace supports several of the most common) or the embedded database that's included with the Clearspace distribution. When you're setting up Clearspace with its setup tool, you'll be prompted for information about the data source you want to connect to — in other words, it's a good idea to make your database decision *before* using the Clearspace setup tool.

Both distributions include an embedded database, but you should use a separate database for production deployment. If you're using the embedded database, you can skip to [Application Server Setup](#).

If you're using a separate database technology, you should set it up before you set up your application server. You will need to create a database from schema and ensure that the application server you're using includes JDBC drivers for the database. You can find schemas in the distribution's database directory. View the [Database Setup](#) to learn how to install the schemas and JDBC drivers.

JNDI Settings: The Clearspace setup tool provides a mechanism for configuring your data source connection, however, you can also configure this connection via your application server's JNDI settings. See [Resin JNDI example](#).

Note: The embedded database Clearspace provides is suitable for evaluation and testing, but you shouldn't use it for deployment. If you want to use the embedded database, simply select the "Embedded Database" option when running through the Clearspace setup tool. For more information on the setup tool, see the Clearspace Administrators' Guide.

This guide includes setup guidelines for the following database technologies:

[MySQL](#)
[Oracle](#)
[SQL Server](#)
[PostgreSQL](#)
[DB2](#)
[HSQLDB](#)

Clearspace Schemas. If you use your own database, you'll need to create a new database using the database schema appropriate to your database technology before using the Clearspace setup tool. You'll find schemas for nine database technologies in the database directory just beneath the root of the Clearspace distribution.

Required JDBC Drivers. You'll need to put a JDBC driver for your database technology in the application server's classpath. Your database's setup documentation should include information on how to install the schemas and required JDBC drivers. The standalone distribution of Clearspace includes a standalone application server; if you use this server the path is for installing the JDBC driver is `<installation_directory>/server/shared/lib`. You'll need to restart your application server for the driver to be registered. For more information about setup instructions specific to certain application servers, see [Application Server Setup](#).

Note: If you need to re-run the setup tool, open the `jiveHome/setup.xml` file, find the line `<setup>true</setup>`, and change the `true` value to `false`. Restart your application server, point your browser at <http://localhost:8080/clearspace>

MySQL

Setup Instructions

1. Make sure that you are using MySQL 4.1 or later and the MyISAM table type (default).
2. Create a database for the Clearspace tables using the MySQL command line tool:
`mysql> CREATE DATABASE <database_name>;`
3. Next import the schema file from the database directory of the installation:
 - Unix/Linux: `mysql <database_name> < jive_clearspace_mysql.sql`
 - Windows: `mysql <database_name> < jive_clearspace_mysql.sql`
4. Start the Clearspace setup tool, and use the appropriate JDBC connection settings.

MySQL Character Encoding Issues

MySQL does not have proper Unicode support, which makes supporting postings in non-Western languages difficult. However, the MySQL JDBC driver has a workaround which you can enable by adding `<mysql><useUnicode>true</useUnicode></mysql>` to the `<database>` section of your `jive_startup.xml` file. When using this setting, you should also set the Jive character encoding to `utf-8` in the Admin Console.

MySQL 4.1 introduces better support for character encodings than previous versions. This new functionality will assign a default character encoding to the database and its tables and columns. It's best to set the default character encoding for your database before installing the Jive schema so that you can be sure that you will not have encoding problems in the future. After creating your database, execute the following line in the MySQL console:

```
ALTER DATABASE <database name> DEFAULT CHARACTER SET <character set>;
```

For example, if you plan on using UTF-8 in Clearspace, you should enable the JDBC driver workaround mentioned above and then execute this line in the MySQL console:

```
ALTER DATABASE <database name> DEFAULT CHARACTER SET utf8;
```

MySQL Max Attachment Size Issues

You can fix the max attachment size problem on a MySQL server by following the directions here:

<http://dev.mysql.com/doc/refman/5.0/en/packet-too-large.html>

The easiest way to change the MySQL server setting on Windows is to add a line to your `my.ini` file (you'll find this file in a location such as `C:\database\mysql\5.0.19`); on Linux, look for the `my.cnf` file instead.

Add the following line after the [mysqld] section heading:

```
max_allowed_packet = 500M
```

After you add the line, restart MySQL.

JDBC Drivers

Use the following values in the Clearspace setup tool:

- Driver: `com.mysql.jdbc.Driver`
- URL: `jdbc:mysql://[host-name]:3306/[database-name]`

Oracle

Setup Instructions

1. If you're connecting to the Oracle database via the command line using SQLPLUS, you can connect to the database to execute the commands above by doing the following:

```
sqlplus system/XXXX
```

where XXXX is the "system" user's password.

2. Create a "clearspace" user or some other user who will "own" the Clearspace tables. This isn't necessary, but doing so will allow your tables to exist in a separate tablespace. Typical commands are as follows:

```
CREATE USER clearspace;  
IDENTIFIED BY clearspace;  
GRANT CONNECT, RESOURCES TO clearspace;
```

3. Next, import the schema from the database directory of the Clearspace installation using sqlplus (or your favorite Oracle tool such as Toad or DbVisualizer). If you have sqlplus on your PATH, you can execute the following command in the directory containing the Oracle schema. The following assumes you've set up a user account called "clearspace" with password "clearspace":

```
sqlplus clearspace/clearspace @ jive_clearspace_oracle.sql
```

JDBC Drivers

For most users, the best drivers for Oracle are the ones written by Oracle (which come with the database or can be downloaded from Oracle's website). Jive Software recommends using the "thin" drivers.

Please consult the Oracle documentation to decide which version of the JDBC thin driver is best for you. Typically, the drivers are in "classes12.zip" or "classes12.jar". Do not use the classes11.zip JDBC driver. If you use the ZIP file you may wish to rename the classes12.zip file to classes12.jar or oracle.jar because some application servers will not work correctly with ZIP files.

Use the following values in the Clearspace setup tool:

- Driver: `oracle.jdbc.driver.OracleDriver`
- Server: `jdbc:oracle:thin:@<host>:<port>:<oracle_sid>` (the default port is 1521)

SQL Server

Setup Instructions

1. When installing SQL Server you must select "Mixed Mode" authentication.
2. First create a user using SQL Server auth; the default database for user should be master. (Use these settings for now; you'll change them later.)
3. Create a new Database, make the owner the user you just created.
4. Run a new query on the database (use a select DB script if you have one), selecting your database from the drop down and selecting Execute!
5. Go back to your user properties and set the following:
 - Server Roles: check all
 - User Mapping: db_owner and public
6. Click Tools > Database Engine Tuning Advisor and enter the following values:
 - ServerName = HOST
 - Auth = SQL Server
 - Authentication Login = your user

JDBC Driver

For SQL Server, Jive Software recommends using the open source jTDS Driver found here:

<http://jtds.sourceforge.net/>

Use the following values in the Clearspace setup tool:

- Driver: `net.sourceforge.jtds.jdbc.Driver`
- URL: `jdbc:jtds:sqlserver://<host>:<port>;DatabaseName=<db_name>` (the default port is 1433)

PostgreSQL

Setup Instructions

1. Create a new database user using the PostgreSQL createuser utility:

```
createuser \-W
```

2. Create the database using the "createdb" utility:

```
createdb \--encoding=UNICODE dbname
```

3. Note, the --encoding switch is optional, but is a good idea so your database will support Unicode

content.

- Grant the user permission to the newly created database. You'll need to locate the Postgres data directory — such as `/var/lib/postgresql`. Consult your Postgres installation or administrator — as this value might be different between installations.
- Once you locate the directory it should contain a file called `pg_hba.conf`.
- Open the file and go to the bottom of it — you should see an access control section. You'll need to edit access permissions for this database. The following is a sample entry:

```
local   all         all         trust
host    all         all         127.0.0.1      255.255.255.255  trust
host    all         all         your.ip.address 255.255.255.255  trust
```

Note, this is a pretty open configuration. Please consult your system administrator for the best access configuration.

- Log in to the `psql` application using the user you made above:

```
psql \-U user_you_created \-W \-d dbname
```

- Once there, run the following command to import the Clearspace database schema from the database directory of the installation:

```
\i /path/to/jive/schema/file.sql
```

- Proceed to the Clearspace setup tool and use the appropriate JDBC settings when prompted.

JDBC Driver

JDBC Driver for PostgreSQL can be found in various distribution packages at PostgreSQL's Home Page. Refer to the JDBC documentation at that site for the latest driver and for more information.

Use the following values in the Clearspace setup tool:

- Driver: `org.postgresql.Driver`
- URL: `jdbc:postgresql://<host_name>:<port>/<database_name>` (default port is 5432)

DB2

Setup Instructions

You'll need to create a new database with a name limit of eight characters, or use an existing database. Connect to the database using any valid user name allowed to create tables and indexes. Keep in mind that DB2 UDB applies the user name as the table schema. Use the same user when you configure Clearspace with the administration interface.

- Using the DB2 Control Center, open Object View and expand the tree so you can view the list of databases.
 - Right-click the "All Databases" folder, then select Create Database (either Standard or With Automatic Maintenance); follow the steps in the wizard to create your new Clearspace database.

- b. Once the database is created, single-click on the Clearspace database and verify that you are connected, if not, click the "Connect" link in the bottom window of the Control Center.
 - c. Once connected, right-click on your Clearspace database and select "Query".
2. Using the command line, execute the following:

```
DB2 CREATE DB [DBNAME] ALIAS [DBALIAS]
DB2 CONNECT TO jive USER [DBUSER] USING [DBPASSWORD]
```

3. Start the DB2 command window (in Windows), or the user's shell (in Unix) and import the schema (found in the database directory of the installation) with:

```
db2 \-tvf jive_clearspace_db2.sql
```

JDBC Drivers

Use the JDBC 1.2 compliant driver, db2jcc.jar and valid db2jcc_license_cu.jar, located in the bin directory of your DB2 installation directory:

<DB2_INSTALL_DIR>\SQLLIB\BIN\db2jcc.jar, db2jcc_license_cu.jar

Use the following values in the Clearspace setup tool:

- Driver: com.ibm.db2.jcc.DB2Driver
- URL: jdbc:db2://<HOST>:<PORT>/<DBNAME> (the default port is 50000)

HSQldb

Special Note

Clearspace bundles HSQL as its embedded database. If you choose to use the embedded database, you can configure it with the Clearspace Setup Tool. If you have a standalone installation of HSQL, follow the instructions below to connect to it.

Setup Instructions

Using the Clearspace Setup Tool, on the Datasource Setting page, choose "Embedded Database". Jive Software recommends only choosing this option if you are using Clearspace for development or for evaluation purposes.

JDBC Drivers

Because HSQLDB is embedded in Clearspace, so there is no need to download the JDBC driver separately.

JNDI

Resin JNDI Example:

1. Edit \$RESIN_HOME/conf/resin.conf
2. Add a JDBC pool configuration as in the following example.

In this example, the database is hosted on MySQL, the database name is clearspace with a user and password of clearspace.

```
<database>
  <jndi-name>jdbc/clearspace</jndi-name>
  <driver type="com.mysql.jdbc.Driver">
    <url>jdbc:mysql://localhost:3306/clearspace</url>
    <user>clearspace</user>
    <password>clearspace</password>
  </driver>
  <prepared-statement-cache-size>8</prepared-statement-cache-size>
  <max-connections>20</max-connections>
  <max-idle-time>30s</max-idle-time>
</database>
```

Application Server Setup

While Clearspace is a pure Java application, the application servers on which it can be deployed vary in how they support such applications. This document describes the application server-specific steps you might need to take in order to get your Clearspace installation full deployed and running.

These instructions are part of the installation process described in the [ClearspaceInstallGuide](#). See that document for a more complete view of the installation process.

Note: If you're using the application server that's included in the standalone distribution of Clearspace, you don't need this guide.

The server technologies covered here include:

[Apache Tomcat](#)

[BEA WebLogic Server](#)

[Caucho Resin Server](#)

[JBoss Application Server](#)

[Jetty Server](#)

[Oracle Application Server](#)

[Orion Application Server](#)

[SAP Web Application Server](#)

[Sun GlassFish Application Server](#)

Apache Tomcat

1. Place the war file in the /tomcat/webapps directory.
2. Restart Tomcat.
3. Navigate to the now-running Clearspace instance and use the Clearspace setup tool to finish setting up Clearspace. Using the default port on Tomcat, navigate to the following URL:

<http://localhost:8080/clearspace>

BEA WebLogic Server

You can deploy Clearspace via EAR or WAR (recommended) through WebLogic's autodeploy feature or its Administration Console.

Issues and Workarounds

Issue: WebLogic and the XFire framework are not compatible out-of-the-box.

Workaround: You must set your EXT_PRE_CLASSPATH to point to the XFire jar. This should not interfere with any other web applications you are hosting.

On Linux

- Edit /etc/profile and add: export
EXT_PRE_CLASSPATH="/your/path/to/WEB-INF/lib/xfire-jsr181-api.jar"
- Save and source your /etc/profile. For example: source /etc/profile

On Windows

- Right click on **My Computer**, then click **Properties**. Select the **Advanced** tab.
- Click **Environment Variables**.
- Under **System variables**, click "New".
- In the **New System Variable** dialog, enter the following values:
 - Variable name: EXT_PRE_CLASSPATH
 - Variable value: C:\path\to\WEB-INF\lib\xfirejsr181-api.jar

Issue: Weblogic may not correctly find the context-root.

Workaround: Unpack clearspace.war (unzip clearspace.war), then create clearspaceWEB-INF/weblogic.xml. The file should contain the following:

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<weblogic-web-app xmlns="http://www.bea.com/ns/weblogic/90">
  <context-root>clearspace</context-root>
</weblogic-web-app>
```

Navigate to the now-running Clearspace instance and use the Clearspace setup tool to finish setting up Clearspace. Using the default port on WebLogic, navigate to the following URL:

<http://localhost:7001/clearspace>

Caucho Resin Server

1. (Optional) Create a user to run Resin (for example, "resin" or another non-root user).
2. (Optional) Link /usr/local/resin to the current Resin version. This is \$RESIN_HOME.

3. Create a deployment root, such as /usr/local/web, owned by the resin user.
4. Copy the resin.conf file into /usr/local/web/conf. In this file you can specify Clearspace-specific settings if you choose, including the server port number, JVM arguments, and so on.
5. Drop your clearspace.war file into /usr/local/web/webapps.
6. Output logs will appear in /usr/local/web/log.
7. Create a startup script (e.g. start.sh) to configure and start the server. Save the script in \$RESIN_HOME/bin. A basic one could look something like this:

```
#!/bin/sh
JAVA_HOME=/usr/java/latest
RESIN_HOME=/usr/local/resin
SERVER_ROOT=/usr/local/web
export JAVA_HOME
export RESIN_HOME
export SERVER_ROOT

$RESIN_HOME/bin/httpd.sh \-server-root $SERVER_ROOT \
  -conf $SERVER_ROOT/conf/resin.conf \
  $*
```

8. Start Clearspace with the following command: \$RESIN_HOME/bin/start.sh
9. Navigate to the now-running Clearspace instance and use the Clearspace setup tool to finish setting up Clearspace. Using the default port of 80 on Resin, navigate to the following URL: <http://localhost/clearspace>

JBoss Application Server

1. Put the WAR in the <jboss-install>/server/deploy/ directory, or another non-default server deploy directory.
2. If desired, remove database settings from server/deploy/. You will have a chance to define this using the Clearspace Setup Tool.
3. JBoss 4 does not use JDK 1.5 to compile JSPs by default, 1.5 is needed for Clearspace. Edit <jboss-install>/server/default/deployjbossweb-tomcat55.sar/conf/web.xml by uncommenting the following:

```
<!--\- Uncomment to use jdk1.5 features in jsp pages-->
<init-param>
  <param-name>compilerSourceVM</param-name>
  <param-value>1.5</param-value>
</init-param>
```

4. Start the JBoss server with the following command:
/bin/run.sh or bin\run.bat
5. Navigate to the now-running Clearspace instance and use the Clearspace setup tool to finish setting up Clearspace. Using the default port on JBoss, navigate to the following URL: <http://localhost:8080/clearspace>

Jetty Server

1. Place the war file in /jetty/webapps.
2. Point your browser at the Clearspace application, usually <http://localhost:8080/clearspace>.

Oracle Application Server

1. NOTE: Not supported in 1.0.0, will be supported
2. Deploy clearspace.war via Oracle's Application Server Control.
3. Navigate to the now-running Clearspace instance and use the Clearspace setup tool to finish setting up Clearspace. Using the default port on Oracle, navigate to the following URL:
<http://localhost:7777/clearspace>

Issues and Workarounds

- **Issue:** OAS 10.1.3.0 and 10.1.3.1 are currently broken and will not parse multi-value parameters.
- **Workaround:** None, wait for the 10.1.3.1.2 release.
- **Issue:** Documented here: <http://jira.codehaus.org/browse/XFIRE-812>; At startup OC4J loads a set of libraries that are not visible as shared libraries. These include the "jws-api.jar" library. This is controlled from the "[OC4J Root]\j2ee\home\oc4j.jar\META-INF\boot.xml". This jar conflicts with the XFire, xfire-jsr181-api.jar
- **Workaround:**
 1. Copy xfire-jsr181-api.jar to <oc4j.home>/j2ee/lib
 2. unjar oc4j.jar
 3. edit META-INF/boot.xml
 4. add <code-source path="lib/xfire-jsr181-api.jar" /> before jaxr-api
 5. rejar oc4j.jar

Orion Application Server

NOTE: Not supported in 1.0.0, will be supported in upcoming releases.

1. Drop clearspace.war into <orion-install>/applications/
2. Edit <orion-install>/config/application.xml
 - Add the following line below the defaultWebApp definition:
<web-module id="clearspace" path="../applications/clearspace.war" />
3. Edit <orion-install>/config/default-web-site.xml
 - Add the following line below the defaultWebApp definition:
<default-web-app application="default" name="clearspace" />
4. Edit <orion-install>/config/server.xml
 - Add the following entry:
<application name="clearspace" path="../applications/clearspace.war" auto-start="true" />
5. Navigate to the now-running Clearspace instance and use the Clearspace setup tool to finish setting up Clearspace. Using the default port on Orion, navigate to the following URL:
<http://localhost/clearspace>

Issues and Workarounds

Issue: <http://www.jivesoftware.com/issues/browse/CS-525>

Workaround: None, this will be supported in an upcoming release.

Issue: Due to bug in Orion's classloader, XFire and Orion are incompatible out of the box. The Orion development team has indicated that this problem will be resolved with version 3.0.

Workaround:

- Remove the following jars from your Orion installation:
 - crimson.jar
 - jaxp.jar
 - xerces.jar
 - xalan.jar

SAP Web Application Server

Using the SAP Deploy Tool:

- Import clearspace.war
- create EAR from WAR
- Load the EAR
- Deploy the Clearspace instance as you would any SAP developed EAR

Sun GlassFish Application Server

Using the GlassFish Admin Console: <http://localhost:4848>

- Navigate to: Web Application -> Deploy
- Browse to and select the clearspace.war
- Choose 'OK'
- In the Admin Console, click the 'Launch' link for the Clearspace application, this will open the setup tool and you can finish setting up Clearspace.

Setting the jiveHome Directory

The jiveHome directory contains configuration files, logs, data (for the embedded database) and other items that Clearspace needs to run correctly. You need to set up your jiveHome directory in a permanent location. The directory must be readable and writable by your application server, but should not be in the document path of your webserver (such that someone could access the directory from a URL such as <http://www.yourserver.com/jiveHome>). Windows users might pick a directory such as c:\jiveHome while Unix users might install jiveHome to /opt/jiveHome.

Now that your jiveHome directory is set up, Clearspace will need to know its location. There are several ways to do this. Choosing one of the following options depends largely on how you want to register your value in your environment.

- **Option 1: Use the setup tool (recommended).** Use the Clearspace setup tool to set the value. After you deploy the web application (detailed below) and start the setup tool you'll be prompted for the jiveHome location.
- **Option 2: Use the EditWAR tool.** The EditWAR tool is a small application which will modify the packaged web application (clearspace.war) to point to the jiveHome directory. The clearspace.war file is also in the base directory of this distribution. To invoke the EditWAR tool, open a command prompt in the base directory of this distribution and execute this command:
java -jar EditWAR.jar clearspace.war
The tool will then lead you through the process of updating the WAR.

- **Option 3: Set a JNDI value (advanced).** Set the location of the jiveHome directory via JNDI. You can set a JNDI value in your app server of `java:comp/env/jiveHome` with a String value of the path to your jiveHome directory.
- **Option 4: Set with a system property (advanced).** Set the location of the jiveHome directory manually by passing in a Java system property to your app server. Set a property with the name of "jiveHome". Most app servers allow you to pass in an environment variable in the startup script. That might look like this:

```
java -DjiveHome=/usr/foo/jiveHome -cp . com.myappserver.Server
```

Please consult your server documentation if you are not familiar with setting up web applications. Most servers give you a choice between deploying a web application by copying a WAR file into a certain directory (typically "webapps"), or by using a deploy tool. Use the method that you are most familiar with.