

Installation Guide

Look here for information on installation steps, including for each of the components you'll need.

Important Note: At this time Clearspace is not supported in a virtualized environment. A virtualized environment can slow the application's performance by severely decreasing disk I/O and network I/O under load.

To make evaluating Clearspace easier, it includes a Windows installer. You'll find more information about it in the *readme file*.

This document includes the following sections:

[Installation Overview](#) (page 1) — A high-level view of the installation steps.

[System Requirements](#) (page 2) — Describes the technologies that Clearspace needs and supports.

[Files in the Distributions](#) (page 3) — A list of what's included in the distribution.

[Java Setup](#) (page 4) — Setting up Java if you're using the standalone application server.

[Database Setup](#) (page 6) — Setting up a database to support Clearspace.

[Application Server Setup](#) (page 12) — How to set up your application server to support Clearspace.

[Setting the jiveHome Directory](#) (page 17) — Finding a good location for the jiveHome directory.

Installation Overview

Here's a high level view of installing Clearspace.

1. Review the [system requirements](#) (page 2) to make sure your system supports Clearspace.
2. Copy the Clearspace distribution to the server you'll be using. You'll find the distribution at the [Jive Software web site](#).
3. Decide what database technology you'll be using with Clearspace. Data about people, content, and the means for organizing them is stored in a database that you specify while using the Clearspace setup tool. You should use a separate database for a production deployment, but the Clearspace distribution includes its own database you can use for evaluation purposes.
4. If you're not using the evaluation database, [prepare your database](#) (page 6) for Clearspace.
5. Decide what application server technology you'll be using. The standalone distributions of Clearspace includes an application server. If you're installing with the WAR File, [set up your application server](#) (page 12) by installing the application files and JDBC driver.
6. Deploy Clearspace using the database and application server decisions you made in the preceding steps.
7. [Set the jiveHome directory](#) (page 17) .
8. It is recommended that you allocate extra memory for Clearspace when starting your application server. For example, `-Xmx512M`
9. This process is finished when you complete the setup tool that is displayed when you launch Clearspace for the first time.

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>. You'll find more information about the setup tool in the *ClearSpace Administrators' Guide*.

After you complete the setup tool, use the admin console to begin configuring your ClearSpace installation.

System Requirements

ClearSpace requires a Java application server and a database, as described in the following recommendations. The following lists supported technologies and recommended system configuration. The following table lists the platform components supported by Jive Software for ClearSpace. The components are divided into three configurations indicating level of reliability, stability, and support:

- Recommended -- If you're starting from scratch, use these components.

Note: Be sure also to read *Fine-Tuning ClearSpace Performance* for tips on making the most of your configuration.

	OS	JDK	Application Server	Database	AAA
Recommended	Linux (2.6 Kernel)	Sun Java 6 (Latest JDK 1.6) with latest patches	Apache Tomcat 6.0.10 (or later in 6.0.x series) (page 12)	Postgres 8.x or better (page 9)	LDAP or AD
Certified	Linux (2.6 Kernel) Windows Server 2003 Solaris 10	IBM Java 5 (JDK 1.5) with latest patches	Apache Tomcat 6.0.10 (or later in 6.0.x series) (page 12) WebLogic Server 9.2 (page 16) JBoss 4 or better (page 13) WebSphere Application Server 6.1.0.17 or better (page 13)	MySQL 5.x or better (page 7) Oracle 10.x or better (page 8) Postgres 8.x or better (page 9) Microsoft SQL Server 2005 (page 10) IBM DB2 v9 or better (page 11)	LDAP or AD

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 better multi-threaded 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. (Be sure to see the LDAP and Active Directory Guide for more information on configuring those resources for use in Clearspace.)

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).

Supported Browsers

Clearspace is supported on the following browsers (browsers on mobile devices aren't supported):

- Internet Explorer versions 6 and 7
- Firefox version 2 and 3
- Safari versions 3 and later

Known Issues

Poor Performance When Citrix Is Used

Clearspace will not work properly when accessed through Citrix. This is because Citrix doesn't support AJAX, a technology Clearspace uses for accessing the server asynchronously in the background.

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 the distributions are:

- A database directory that contains SQL scripts to create new Clearspace databases. See [Database Setup](#) (page 6) for more information.
- A jiveHome directory where Clearspace stores configuration files and additional resources.

Standalone Distribution

These include an application server. With the standalone distribution, included app server and evaluation database, 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 evaluation database), see [Database Setup](#) (page 6) for further installation instructions.

To start Clearspace:

- Standalone distribution - 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_<dist>_x_x_x/ |- database/ |- jiveHome/ |- server/ |- webservices/ |- README.html |- start-clear
```

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](#) (page 2) .

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 |- README.
```

Java Setup

Clearspace requires a valid Java installation and a JAVA_HOME environment variable set. If you already have an application server installed then you already have Java installed and working properly. You'll need to go through the following steps if you're using the standalone distribution.

Java Installation Instructions

You can install Java so that it will be found system-wide, or you can install it so that it will be found by the standalone distribution.

Download the Java Distribution

1. Download the Java distribution for your OS:
 - [Linux](#)
 - [Windows](#)
 - [Solaris](#)

Installing Java for Standalone Use

1. Follow the installation instructions for your platform. Record the path to the directory where you installed Java.
2. Set a JAVA_HOME environment variable. An environment variable is a way tell Clearspace where Java is installed.
 - **Unix**

- Open your favorite text editor and type the following text: `export JAVA_HOME="<java_install_path>",` where `<java_install_path>` is the directory where Java is installed. For example: `export JAVA_HOME="/Library/Java/Home/"`.
- Save this to a file called `setJavaHome` in the Clearspace standalone installation directory. For example, if you extracted Clearspace to `/home/clearspace/jive_clearspace_standalone_2_0_1` then you'll save this file as `/home/clearspace/jive_clearspace_standalone_2_0_1/setJavaHome`.

- **Windows**

- On your desktop, right click on 'My Computer', select 'Properties'
- Click the Advanced Tab
- Click the Environment Variables button
- Under System Variable, click New.
- In the 'Variable name' field, enter: `JAVA_HOME`
- In the 'Variable value' field, enter the path to your JDK installation (e.g. `C:\Program Files\Java\jdk1.6.0`).
- Click OK on all open System Properties screens (3)

3. You've now completed the Java installation.

Installing Java for WAR Use

1. Follow the installation instructions for your platform. Record the path to the directory where you installed Java.
2. Set a `JAVA_HOME` environment variable. An environment variable is a way tell Clearspace where Java is installed.

- **Unix/Linux**

- Edit the `".profile"` file in your home directory (or corresponding file for your shell).
- Set the `JAVA_HOME` environment variable by adding the following line to the file:
 - `export JAVA_HOME=/path/to/java/directory`
- Be sure there are no spaces after the end of the directory name.
- Do not add an extra slash after the directory name.
- Save changes to the file and then "source" it: `source .profile`

- **Windows**

- On your desktop, right click on 'My Computer', select 'Properties'
- Click the Advanced Tab
- Click the Environment Variables button
- Under System Variable, click New
- In the 'Variable name' field, enter: `JAVA_HOME`
- In the 'Variable value' field, enter the path to your JDK installation (e.g. `C:\Program Files\Java\jdk1.6.0`).
- Click OK on all of the open System Properties screens.
- Close any open command prompt windows. The next time you open a command prompt, the `"JAVA_HOME"` variable will be set correctly.

Database Setup

Clearspace stores data about users, documents, spaces, and so on in a database.

You can use either your own database or the evaluation 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.

All three distributions include an evaluation database, but you should use a separate database for production deployment. If you're using the evaluation database, you can skip to [Application Server Setup](#) (page 12) .

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](#) (page 6) to learn how to install the schemas and JDBC drivers.

When creating the database user that own the application tables, it is strongly recommended that this user *not* be granted view access to any schema other than the one that it owns. During upgrades the application will read metadata about its tables defined in the database. Since the application does not know what the default schema is, it may inadvertently retrieve information about its tables defined in other schemas. If you have set up application instances using other schemas in the same database, then this can result in erroneous information passed to the application.

If you absolutely must use a database user with access to other schemas, then you should ensure there is only one set of application tables defined across all schemas to prevent this object name conflict.

Note: If you're upgrading to a version whose database schema is different from the previous version, the application will automatically detect the difference after you've upgraded. When you next start the application and navigate to the admin console, you'll be prompted to start the database upgrade. This is a necessary step before you can start using the application after upgrade. You don't need to run any database-related upgrade scripts.

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.

Note: The evaluation database Clearspace provides is suitable for evaluation and testing, but you shouldn't use it for deployment. If you want to use the evaluation database, simply select the "Evaluation 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](#) (page 7)
- [Oracle](#) (page 8)
- [PostgreSQL](#) (page 9)
- [Microsoft SQL Server](#) (page 10)
- [IBM DB2](#) (page 11)

Clearspace Schemas. The setup tool is designed to install the schema for the database you specify. When you need to set up the schema manually, you'll find schemas for several database technologies in the database directory just beneath the root of the 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 distributions of Clearspace includes a standalone application server; if you use this server the path is for installing the JDBC driver is `<installation_directory>/server/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](#) (page 12) .

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 5.x or later and the MyISAM table type (default).
2. Create a database for the Clearspace tables using the MySQL command line tool. (First, be sure to read the note below about character encoding issues.)

```
mysql> CREATE DATABASE <database_name>;
```
3. The application setup tool should install the schema you need. If you find you need to install manually, 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 introduced better support for character encodings than previous versions. This functionality assigns 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.

Adjust the InnoDB Buffer Pool Size

When you have the database running on a dedicated server, you should increase the InnoDB buffer pool size from the default (8 MB) to up to 80 percent of the computer's available memory. If the computer has 2 GB of RAM or less, you should think about setting the buffer to something less than 80 percent to ensure that the operating system has enough memory to avoid swapping.

See the [MySQL documentation](#) for more on configuration.

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, RESOURCE TO clearspace;
```

3. The application setup tool should install the schema you need. If you find you need to install manually, import the schema file from the database directory of the installation. For example, you can 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

Jive Software requires that you use the Oracle Call Interface (OCI) driver rather than the thin driver. For more information on installing the OCI driver, which requires that you install the Oracle OCI Instant Client, see the Oracle [JDBC FAQ](#) and [Instant Client page](#).

Please consult the Oracle documentation to decide which version of the JDBC driver is best for you.

Use the following values in the Clearspace setup tool:

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

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.
4. 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.
5. Once you locate the directory it should contain a file called `pg_hba.conf`.
6. 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 255.255.255.2
```

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

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

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

8. The application setup tool should install the schema you need. If you find you need to install manually, import the schema file from the database directory of the installation. For example, run the following command to import the Clearspace database schema from the database directory of the installation:

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

9. 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. When choosing a driver version, be sure to use version 8.2 or later (although the latest version of the driver that supports your version of the database is generally best). 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)

Microsoft 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

Case Sensitivity Issues

SQL Server is case insensitive by default, creating special case sensitivity issues for Jive SBS, which requires a DBMS that matches strings with case sensitivity. The SQL Server issues differ between versions 2008 and 2005.

SQL Server 2008: You can set SQL Server database and/or server collation (the character set used to store data in text fields) to support case sensitivity. If database collation is not specified, server collation will be used. The collation value you'll need to use is `SQL_Latin1_General_CP1_CS_AI`.

You'll find more information on adjusting SQL Server settings in the SQL Server documentation:

[Setting and Changing the Server Collation](#)

[Setting and Changing the Database Collation](#)

SQL Server 2005: Switching to case sensitivity in SQL Server 2005 causes Jive SBS to fail because the database also applies the case sensitivity rules to the queries themselves. Therefore, the application can only support SQL Server 2005 with case insensitivity set. However, this also creates a limitation in which the

application does not support multiple usernames in which the only different is case -- such as "Gladys" and "gladys".

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)

IBM 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.

1. Using the DB2 Control Center, open Object View and expand the tree so you can view the list of databases.
 - a. 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. The application setup tool should install the schema you need. If you find you need to install manually, import the schema file from the database directory of the installation. For example, 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
```

Default Max Attachment Size Limited to 1MB

By default, the maximum size for a BLOB in DB2 is 1MB; this limits the maximum attachment size in Clearspace to 1MB. The issue will be fixed in an upcoming release, but for now, the work around is to issue the following command against your Clearspace database hosted by DB2. (This fix has been tested on DB2 9.1.)

```
ALTER TABLE jiveAttachData ALTER attachmentData SET DATA TYPE BLOB(2G)
```

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)

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 fully deployed and running.

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:

- [General Server Information](#) (page 12)
- [Apache Tomcat](#) (page 12)
- [WebLogic Server 9.2](#) (page 16)
- [JBoss](#) (page 13)
- [WebSphere Application Server](#) (page 13)

General Server Information

Headless Servers

When deploying the WAR distribution, if the server you are deploying Clearspace on is headless, you must set the Java system property `java.awt.headless` to "true". If you don't, certain portions of the application that use the Java AWT for generating images will not function properly.

Enable the JVM headless mode by setting the `java.awt.headless` flag to true, as in the following command line example:

```
-Djava.awt.headless=true
```

Apache Tomcat

1. Place the war file in the `<tomcat-install>/webapps` directory.
2. Restart Tomcat. `<tomcat-install>/bin/startup.sh`

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>

Issues and Workarounds

Issue: If you deploy Clearspace to the root of the Tomcat webapps directory and also have Tomcat's default docs web application deployed at the root (in the /docs directory), you'll get a 404 error when you try to view a Clearspace document. That's because Clearspace references its documents at a path such as /docs/DOC-<docid>.

Workaround: If you intend to deploy Clearspace to the root of your Tomcat webapps directory, be sure to remove the web application at /docs if it is there.

JBoss

1. Shut down the application server.
2. Unzip the WAR in the <jboss-install>/server/deploy/ directory, or another non-default server deploy directory.
3. If desired, remove database settings from server/deploy/. You will have a chance to define this using the Clearspace Setup Tool.
4. JBoss 4 does not use JDK 1.6 to compile JSPs by default, 1.6 is needed for Clearspace. Edit <jboss-install>/server/default/deployjbossweb-tomcat55.sar/conf/web.xml by uncommenting the following:

```
<!-- Uncomment to use jdk1.6 features in
      jsp pages--> <init-param> <param-name>compilerSourceVM</param-name> <param-value>1.6</param-value>
```

5. Add a VM parameter:
 - To /bin/run.sh add:
JAVA_OPTS="\$JAVA_OPTS -Djavax.xml.soap.MessageFactory=com.sun.xml.messaging.saa.j.soap.ver1_1.SOAPMessageFactory1_1Impl"
 - Or to bin\run.bat add:
set JAVA_OPTS=%JAVA_OPTS% -Djavax.xml.soap.MessageFactory=com.sun.xml.messaging.saa.j.soap.ver1_1.SOAPMessageFactory1_1Impl
6. Start the JBoss server with the following command:
/bin/run.sh or bin\run.bat
7. 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>

WebSphere Application Server

Note: If you'll be using JNDI to reference the Clearspace database, set the ResultSetHoldability property to 1 (HOLD_CURSORS_OVER_COMMIT). Its default value is 2.

Before Installing Clearspace

Make these settings in the WebSphere console to set a custom property.

1. Expand **Servers > Application Servers > <server>> Web Container Settings > Web Container > Custom Properties**
2. Select **New** and then enter the following:
 - **Property name:** com.ibm.ws.webcontainer.invokefilterscompatibility
 - **Value:** true
3. Click **Apply**.
4. Click **Save to the master configuration**.
5. Restart the application server.

Install Clearspace

Perform the following steps in the WebSphere console.

1. Install the WAR file.
 - a. Expand **Applications > Install New Application**.
 - b. In the **Preparing for the application installation** page, click **Choose file**, then select the clearspace.war
 - c. Enter the context root, such as `"/clearspace"`.
 - d. Click **Show me all installation options and parameters**.
 - e. Click **Next**.
2. In the **Preparing for the application installation** page, accept defaults and click **Next**.
3. In the **Application Security Warnings** page, accept defaults and click **Continue**.
4. In the **Install New Application** page, under **Select installation options**, select the **Precompile JavaServer Pages files** check box.
5. Click **Next**.
6. In the **Install New Application** page, under **Map modules to servers**, accept defaults and click **Next**.
7. In the **Install New Application** page, under **Provide options to compile JSPs**, modify **JDK Source Level** from "13" to "15".
8. Click **Next**.
9. In the **Install New Application** page, under **Provide JSP reloading options...**, accept defaults and click **Next**.
10. Under **Map shared libraries**, accept defaults and click **Next**.
11. Under **Initialize parameters for servlets**, accept defaults and click **Next**.
12. Under **Map virtual hosts...**, accept defaults and click **Next**.
13. Under **Map context roots...**, accept defaults and click **Next**.
14. Under **Summary**, click **Finish**.
15. Click **Save**.

Configure shared libraries to use the Clearspace version of aspectrt.jar

In the WebSphere admin console, make the following changes.

1. Expand **Environment > Shared Libraries**.
2. Choose the most specific scope to your Clearspace installation; by default, this would be "Node=localhostNode01, Server=server1"
3. Click **New**.
4. In the **Shared Libraries** page, set **Name** to "AspectJ"
5. Set **Description** to "Clearspace version of aspectj runtime."
6. Set **Classpath** to the installed version of Clearspace's aspectjrt.jar. For example, this could be: /opt/IBM/WebSphere/AppServer/profiles/AppSrv01/installedApps/localhostNode01Cell/clearspace_war.ear/clearspace.war/WEB-INF/lib/aspectjrt.jar
7. Click **Apply**, then click **Save**.
8. Expand **Servers > Application Servers > <server>**
9. Expand **Java and Process Management**.
10. Click **Class loader**.
11. Click **New**.
12. Under **Class loader order**, select **Classes loaded with Application class loader first**, then click **OK**.
13. Click the newly created class loader.
14. Under **Additional library references**, click **Shared library references**, then click **Add**.
15. Under **Library name**, in the dropdown, select **AspectJ** and click **OK**.
16. Click **Save**.
17. Expand **Applications > Enterprise Applications**.
18. Click the name of you Clearspace installation.
19. Under **Detail Properties**, click **Class loading and update detection**.
20. Under **Class loader order**, select **Classes loaded with application class loader first**
21. Under **WAR class loader policy**, select **Class loader for each WAR file in application**.
22. Click **OK**.
Note: You'll get an error that "Polling interval for updated files should be in the range 0 through 2147483647". This is a bug, enter a random int value and click **OK**.
23. Click **Save**.
24. Restart the server.

Configure shared libraries to use the Clearspace version of wsdl4j.jar

In the WebSphere admin console, make the following changes.

1. Expand **Environment > Shared Libraries**.
2. Choose the most specific scope to your Clearspace installation; by default, this would be "Node=localhostNode01, Server=server1"
3. Click **New**.
4. In the **Shared Libraries** page, set **Name** to "Wsd4j".
5. Set **Description** to "Clearspace version of wsdl4j."

6. Set **Classpath** to the installed version of Clearspace's wsdl4j.jar. For example, this could be:
/opt/IBM/WebSphere/AppServer/profiles/AppSrv01/installedApps/localhostNode01Cell/clearspace_war.ear/clearspace.war/WEB-INF/lib/wsdl4j.jar
7. Click **Apply**, then click **Save**.
8. Expand **Servers > Application Servers > <server>**
9. Expand **Java and Process Management**.
10. Click **Class loader**.
11. Click **New**.
12. Under **Class loader order**, select **Classes loaded with Application class loader first**, then click **OK**.
13. Click the newly created class loader.
14. Under **Additional library references**, click **Shared library references**, then click **Add**.
15. Under **Library name**, in the dropdown, select **WsdI4j** and click **OK**.
16. Click **Save**.

WebLogic Server

1. You can deploy Clearspace through WebLogic's Administration Console.
2. Before deploying, unzip the Clearspace WAR to a directory: `unzip -d clearspace clearspace.war`
3. Deploy and start the Clearspace instance as you would any web application.
4. Navigate to the now-running Clearspace instance and use the Clearspace setup tool to finish setting up Clearspace. Using the default port on WebLogic: <http://localhost:7001/clearspace>

Issues and Workarounds

Issue: WebLogic is incompatible with certain Clearspace libraries out-of-the-box.

Workaround: Use JDK 1.5 and update the EXT_PRE_CLASSPATH with the JAR files described below. (Note: WebLogic Server does not support JDK 1.6.)

1. Configure a WebLogic domain to use the Sun JDK 1.5.
2. Delete weblogic.xml from exploded Clearspace WAR file. You'll find the file in the clearspace/WEB-INF directory.
3. Set your EXT_PRE_CLASSPATH to point to the following five JAR files. The first four are included in the Clearspace WAR file, in the application WEB-INF/lib directory. Explode the WAR, then copy the fifth into the application WEB-INF/lib directory.
 - commons-logging.jar
 - wss4j.jar
 - xmlsec-1.4.1.jar
 - geronimo-ws-metadata.jar
 - velocity-tools.jar — Download from <http://velocity.apache.org/download.cgi> and copy into the WEB-INF/lib directory.

You can update the path with a command such as the following:

```
export EXT_PRE_CLASSPATH="/path/to/commons-logging.jar:/path/to/wss4j.jar:/path/to/xmlsec-1.4.1.jar:/p
```

Setting the jiveHome Directory

The jiveHome directory contains configuration files, logs, data (for the evaluation 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.