<table>
<thead>
<tr>
<th>CONTENTS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Teamwork Concepts</strong> 4</td>
</tr>
<tr>
<td><strong>Teamwork System Layout</strong> 6</td>
</tr>
<tr>
<td><strong>Installing and Running Teamwork Server</strong> 9</td>
</tr>
<tr>
<td>- System Requirements 9</td>
</tr>
<tr>
<td>- Installing 9</td>
</tr>
<tr>
<td>- Upgrading Teamwork Server 10</td>
</tr>
<tr>
<td>- Importing projects and users from earlier versions of Teamwork Server 13</td>
</tr>
<tr>
<td>- Moving Teamwork Server 14</td>
</tr>
<tr>
<td>- Starting Teamwork Server 15</td>
</tr>
<tr>
<td>- Stopping Teamwork Server 19</td>
</tr>
<tr>
<td><strong>Managing Teamwork Server</strong> 20</td>
</tr>
<tr>
<td>- Users Management 20</td>
</tr>
<tr>
<td>- Starting the Administrator's Console 24</td>
</tr>
<tr>
<td>- Administrator's Console Dialog 25</td>
</tr>
<tr>
<td>- Data Migration between Different Repositories 39</td>
</tr>
<tr>
<td>- Changing Teamwork Server Debugging Mode 40</td>
</tr>
<tr>
<td><strong>LDAP Support</strong> 41</td>
</tr>
<tr>
<td>- Enabling LDAP Integration 41</td>
</tr>
<tr>
<td>- Connection Settings 42</td>
</tr>
<tr>
<td>- Authentication Settings 43</td>
</tr>
<tr>
<td>- User Data Retrieval Settings 48</td>
</tr>
<tr>
<td>- Connection Testing 50</td>
</tr>
<tr>
<td>- Subversion and LDAP Integration Working at the Same Time 50</td>
</tr>
<tr>
<td>- Converting Certificates to JKS Format 50</td>
</tr>
<tr>
<td>- Integrating Teamwork Server with SSL-Enabled Active Directory 51</td>
</tr>
<tr>
<td><strong>Connecting Teamwork server via SSH encrypted tunnel</strong> 53</td>
</tr>
<tr>
<td>- Configure the Teamwork Server side 54</td>
</tr>
<tr>
<td>- Client Side Configuration 56</td>
</tr>
</tbody>
</table>

With MagicDraw Teamwork Server you can assign as many developers as needed to work simultaneously on the same project using multiple workstations. The resulting Teamwork project is saved on the server for sharing with other MagicDraw® applications. Users with administrator rights can create new users by creating a name and assigning various permissions to work on projects. The permissions assigned will determine whether the new user can update, commit, edit, create, and delete model elements, diagrams, and projects.

To enable Teamwork support, you should install and run the MagicDraw Teamwork Server software. Each MagicDraw application acts as a client in the Teamwork system.

At www.magicdraw.com/viewlets in the Project management an collaboration section, you will find the Teamwork viewlets to help you understand how to work with the Teamwork Server.

| NOTE | The Teamwork Server functionality is available with MagicDraw client Standard, Professional, Architect, and Enterprise editions only. |

**Teamwork Concepts**

Get acquainted with basic Teamwork concepts.

**Author**

A user who has committed a new project version.

**Version**

A unique number assigned to the committed project. Project version numbers begin at zero (for the initial version) and increase with every new project version.

**Comment**

Optional description of changes in the committed version.

**Tag**

Information about the status of a project (approved, initially tested, etc.), or other important information.

**Administrator Login**

The default Administrator's account in Teamwork System is:

Login name: Administrator
Password: Administrator

To prevent illegal access, it is advisable to change the Administrator's password.

For more information, see Section “Managing Teamwork Server” on page 20.

**Teamwork Server Administrator's Console**

A remote connection for Teamwork Server status observation and administrative control. The server holds information about active users and loaded projects. The Administrator can
shutdown or restart the server, change its properties, and view log files (including debug information) for the server and separate projects.

Repository
A storage place for projects and their versions managed by the Teamwork Server.

Project category
A concept which enables visual grouping of projects in Teamwork Server repository.

Native user
A user whose account data is stored locally, i.e. in the native Teamwork Server repository.

External user
A user whose account data (all except the login name) is stored in an external database, e.g., Subversion, ClearCase, or LDAP.

Teamwork Module
A Teamwork project containing one or more shared packages. Modules are created with a purpose to reuse them or to decompose projects into parts.

Dependency between two elements
A situation where one element (dependent element) refers to the data of the other element (independent element).
Teamwork System Layout

The diagram above describes the layout of the Teamwork system. MagicDraw clients communicate (using Java RMI, over TCP/IP) with the Teamwork Server to retrieve projects stored on the server, edit them, and commit them back to the Teamwork Server for storage.

The Teamwork Server keeps track of projects versions. Additionally, it performs several administrative functions such as user login, authentication, and check permissions to access projects.

The Teamwork Server uses repositories for project version storage. The administrator can select any one of the supported repository types in the Teamwork Administrator's console to configure the server (for more information, see Section “Starting the Administrator’s Console” on page 24). Data can migrate from one to another repository type. This functionality is also accessible from the Teamwork Administrator's console.

3 different types of repositories are supported in MagicDraw:

- Built-In Repository (the default MagicDraw repository type).
- SVN repository (since v12.5).
- ClearCase repository (since v12.5).

Regardless of the repository used, users will not feel the difference because the user workflow remains the same.

For more information about specifying repositories, see the Administrator’s Console dialog description, "Repository tab" on page 34.

For more information about importing or exporting project to the native repository, see the Administrator’s Console dialog description, "Projects tab" on page 26.
Built-In Repository

This is a default repository type. When the Teamwork Server is first installed and started, it is configured to use this type of repository. The Built-In repository is the only type of repository available in the version prior to 12.5 of Teamwork Server.

When the Teamwork Server is configured to use the Built-In repository, a directory is designated for project storage. The Server then uses its internal proprietary code to implement a versioned repository for a collection of projects.

Additionally, a simple user authentication/authorization scheme is implemented in the repository to store a simple list of users and their passwords (securely encrypted using one-way encryption) in a user file. When MagicDraw users log into the server, it uses this user file to verify these users and their passwords. Users’ rights to access different projects are also described using this file.

<table>
<thead>
<tr>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple to configure (the only configuration parameter is a directory where project versions will be stored) and does not dependent on external tools.</td>
<td>Consume a lot of storage capacity, which increases complexity and maintenance costs. The teamwork administrator has to set up the appropriate backup policy and monitor the growth of disk space used when project versions accumulate. Usually, organization have their own control systems to store code, documents etc. and these system determine the appropriate backup and space allocation.</td>
</tr>
</tbody>
</table>

SVN Repository

The Teamwork Server can be configured to use SVN repository as a back-end. In this mode, the Teamwork Server retrieves and commits project versions into the SVN repository.

To use this repository type, SVN client must be correctly installed on the computer where the Server runs. The Teamwork Server must be able to launch svn executable - svn executable must be in PATH and have appropriate permissions. The supported SVN client versions are 1.4, 1.5, and 1.6 (since Teamwork Server version 17.0).

All SVN repository access methods are supported:

- local access through file:// type URLs
- remote repository access through svn://, svn+ssh:// type URLs
- remote repository access through HTTP and WebDAV - http://, https:// type URLs

**NOTE** Only password based logins are supported (public/private key logins for svn+ssh:// and https:// access methods are currently not available).

The Teamwork Server with SVN repository supports pass-through authentication into the SVN. A pass-through authentication is used for all access methods except file:// method. In this case, the Teamwork Server only maintains a list of users, but not their passwords; their passwords are not stored. When a user tries to log into the Teamwork Server, server does not verify the password itself, but it logs into the SVN with the typed user name/password. All the project update/commit actions on the repository are performed by the server on the user’s request. Hence, if you explore the repository with the SVN tools, you will see that all the changes are attributed to the correct user.

For file:// type URLs, pass-through authentication is not possible. The Teamwork Server uses the same built-in authentication method as the Built-In repository type, the users list with their encrypted passwords is maintained in a repository file. The server authenticates users using this file. Actions in a repository are performed by the server on the user’s request. If the server is started as NT service, all actions in the repository will be attributed
to the Local System user (unless a different user is specified in the service settings). If the server is started manually, all actions in the repository will be attributed to the user who started the server. This difference can only be seen when examining the SVN repository with the SVN native tools. When looking at the project versions with MagicDraw client, all commit actions will be attributed to the users who performed them.

**NOTE**

When a project file is committed into the SVN repository, the server stores additional auxiliary information about the project in an additional directory. If you commit the MyProject.mdzip project into the server, the auxiliary information will be stored in the MyProject_files directory nearby. Do not delete this directory from the repository.

For best performance, the Teamwork Server and SVN repository should have a good link between them. Optimally Teamwork Server could run on the machine where the repository is installed.

<table>
<thead>
<tr>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Projects with their versioning information can be stored in the same repository alongside the other artifacts, produced by the team: code, documentation, etc. There is no need to maintain a separate repository for MagicDraw projects, schedule backups, and monitor for storage used.</td>
<td>Dependent on external tool, the SVN client.</td>
</tr>
</tbody>
</table>

**ClearCase Repository**

The teamwork Server can be configured to use ClearCase repository as a back-end. In this mode, the Teamwork Server retrieves and commits project versions into the ClearCase repository.

To use this repository type, the ClearCase client must be correctly installed on the computer where the Server runs. It must be able to launch cleartool executable - cleartool executable must be in PATH and have appropriate permissions. The supported ClearCase client versions are v7.0 or later (earlier versions might work, but were not tested).

All actions (update/commit) are performed in the ClearCase repository by the server on user request. If the server is started as NT service, actions in the repository will be attributed to the Local System user (unless a different user is specified in the service settings). If the server is started manually, all actions in the repository will be attributed to the user who started the server. This difference can only be seen when examining the ClearCase repository with the ClearCase native tools. When looking at the project versions with MagicDraw client, all commit actions will be attributed to the users who performed them.

**NOTE**

When a project file is committed into the ClearCase repository, the server stores additional auxiliary information about the project in an additional directory. If you commit the MyProject.mdzip project into the server, the auxiliary information will be stored in the MyProject_files directory nearby. Do not delete this directory from the repository.

For best performance the Teamwork Server should run on the machine where the repository is installed.

<table>
<thead>
<tr>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Projects with their version information can be stored in the same repository alongside the other artifacts, produced by the team: code, documentation, etc. There is no need to maintain a separate repository for MagicDraw projects, schedule backups, and monitor storage used.</td>
<td>Dependent on ClearCase client.</td>
</tr>
</tbody>
</table>
Installing and Running Teamwork Server

System Requirements

The maximum number of projects that can be opened in the Teamwork Server depends on the system capabilities. See the requirements in the following table.

For Teamwork Server with Built-In repository

<table>
<thead>
<tr>
<th></th>
<th>Minimum</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk space</td>
<td>200MB(^1)</td>
<td>200MB or more</td>
</tr>
<tr>
<td>Memory</td>
<td>128MB</td>
<td>256MB or more</td>
</tr>
<tr>
<td>Software</td>
<td>Any platform with JDK version 1.6.0 is suitable. For UNIX platform, X Window server is required.</td>
<td>JDK 1.6.0_24</td>
</tr>
</tbody>
</table>

\(^1\) The size of the project depends on how many versions are committed to the server. Every time someone commits changes to a repository, a new version of the whole project is created. For instance, if the project size is 500kB and there are 10 developers who commit their changes twice a day you will need an extra 400MB disk space in the repository for a two-month run.

For Teamwork Server with SVN repository

<table>
<thead>
<tr>
<th></th>
<th>Minimum</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk space</td>
<td>100MB(^1)</td>
<td>150MB or more</td>
</tr>
<tr>
<td>Memory</td>
<td>128MB</td>
<td>256MB or more</td>
</tr>
<tr>
<td>Software</td>
<td>Any platform with JDK version 1.6.0 is suitable. For UNIX platform, X Window server is required. SVN client version 1.4, 1.5, or 1.6.</td>
<td>JDK 1.6.0_24 SVN client version 1.4, 1.5, or 1.6 (since Teamwork Server version 17.0).</td>
</tr>
</tbody>
</table>

\(^1\) Teamwork Server with SVN or ClearCase repository maintains the checked out project cache. The size of this cache depends only on the number and size of projects stored. It does not grow as new versions check in. The disk space to store these new versions is consumed on the SVN/ClearCase repository itself.

For Teamwork Server with ClearCase repository

<table>
<thead>
<tr>
<th></th>
<th>Minimum</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk space</td>
<td>100MB</td>
<td>150MB or more</td>
</tr>
<tr>
<td>Memory</td>
<td>128MB</td>
<td>256MB or more</td>
</tr>
<tr>
<td>Software</td>
<td>Any platform with JDK version 1.6.0 is suitable. For UNIX platform, X Window server is required. ClearCase client version 7.0 or above</td>
<td>JDK 1.6.0_24 ClearCase client version 7.0</td>
</tr>
</tbody>
</table>

Installing

Windows platform

Run the downloaded file MD_UML_<version>_teamwork_server_win.exe. Follow the instructions in the installation program.
UNIX platform

Confirm that X Window server is running. Run the downloaded file MD_UML_<version>_.teamwork_server_unix.sh. Follow the instructions in the installation program.

MAC OS X platform

Confirm that X Window server is running. Run the downloaded file MD_UML_<version>_.teamwork_server_mac.dmg and install the application by dropping the launcher to the Applications folder (or another preferred location).

Other Platforms

Extract the downloaded file MD_UML_<version>_.teamwork_server_no_install.zip. You can launch the Teamwork Server by opening either teamwork_server.exe or ./teamwork_server in the bin directory.

These files require some modifications prior to launching them.

Confirm you have JVM installed.

Upgrading Teamwork Server

You can use automatic or manual upgrade of Teamwork Server.

In automatic upgrade, a new Teamwork Server version will be installed instead of current one. If you need to keep the current server version without any changes, please upgrade the server manually. The manual upgrade allows for installing a new server version on a new location. You will be able to import your old projects and users, to test a functionality of a new server version, and only after that to remove a previous server version.

Since the version of 16.9 commercial licenses are locked to the particular machine. You will be requested to activate the license and receive the commercial license dedicated for the particular machine after the upgrade. If you have already activated the license before upgrading with a new version, the license will be activated automatically.

NOTE You should have valid Software Assurance to upgrade Teamwork Server. For more about Software Assurance, see at https://www.magicdraw.com/software_assurance.

IMPORTANT! Make sure server and the client versions are the same. Also it is recommended to use the same JVM version for the server and client.

Automatic upgrade of Teamwork Server

The easiest way to renew MagicDraw Teamwork Server is to upgrade it automatically. Using this feature the upgrade with newest versions and service packs is done automatically.

To upgrade Teamwork Server automatically

1. Stop Teamwork Server (see "Stopping Teamwork Server" on page 19) and close the Administrator's Console.
2. Remove Teamwork Server from Windows services if it was added. See the procedure "To remove Teamwork Server from the Windows services" on page 19.
3. Start Teamwork Server using GUI (see "Starting the server using GUI" on page 15).
4. In the Teamwork Server startup dialog, click the Check for Updates button.
5. The **HTTP Proxy Server Connection** dialog opens. Click **Use HTTP proxy server** if you want to use a proxy server. Enter required values and click **OK** when you are done. Checking for updates starts.

6. The **Update Information** dialog opens wherein information about available updates is displayed. Click the **Update to New Version** button to start upgrading the server.

7. The **Import Configuration** dialog opens after the upgrading has finished. Click the **Import** button if you want to import server configuration files used in a previous server version.

8. Receive the commercial license dedicated for the particular machine and activate it, if you did not activate the license before the upgrade. For instructions how to receive Teamwork Server license, see at [https://www.magicdraw.com/installing/?product=teamwork#activating](https://www.magicdraw.com/installing/?product=teamwork#activating).

### Manual upgrade of Teamwork Server

Installation of a newer version does not detect nor remove the current server version. All projects and users can be imported to the new server. For the description of importing procedures, see "[Importing projects and users from earlier versions of Teamwork Server](#)" on page 13. The version of project server and client should be the same and cannot be mixed up. Also it is recommended to use the same JVM version for the server and client.

**To upgrade Teamwork Server manually**

1. Stop Teamwork Server (see "[Stopping Teamwork Server](#)" on page 19).
2. Remove Teamwork Server from Windows services, if it was added. See the procedure "[To remove Teamwork Server from the Windows services](#)" on page 19.
3. Run the installation file. For the successful installation please use the following recommendations:
   - To allow for restoring backup data, install a new Teamwork Server version into the other location than current Teamwork Server is installed. Under **Choose Install**
Folder, add, for example, the version you are installing number to the end of the folder name.

![Choose Install Folder](image1)

- Under Choose Java Virtual Machine, click Use the Java VM installed with this application.

![Choose Java Virtual Machine](image2)

4. Start newly installed Teamwork Server. The Import Configuration dialog opens. For information about using this dialog, see the procedure "To import projects using GUI" on page 13.

**NOTE**
The import time can take more than 1 hour 30 minutes. Importing time depends on quantity and size of projects you are importing.
5. In the Teamwork Server License Manager dialog, enter the license key. For more information about node locked license activation, see at http://www.magicdraw.com/installing/?product=teamwork#unlocking. The Teamwork Server License Configuration dialog with license information opens after you have entered a license key.

Click OK. The Teamwork Server startup dialog opens.

6. Click the Start Server button. The server starts.

7. If it is needed, add Teamwork Server to Window services. See the procedure "To add Teamwork Server to Windows services" on page 17.

8. Remove the old Teamwork Server version.

IMPORTANT! Make sure server and client versions are the same. Also it is recommended to use the same JVM version for the server and client.

Importing projects and users from earlier versions of Teamwork Server

If you have projects stored in earlier versions of Teamwork Server, you can import them automatically using MagicDraw GUI, or to do this manually.

To import projects using GUI

1. After the installation is complete and Teamwork Server is launched for the first time, the Import Configuration dialog opens.

2. In the Location box, specify the path to the previous Teamwork Server installation that was found in your computer. Click the ... button to browse for the desired location.
3. Select the **Import all projects and users** check box to transfer projects and users to the latest version of Teamwork Server.

4. Click **Import**.

To import projects manually:

- Copy the **Projects** folder with all its files from the previous Teamwork Server installation folder to the new `<Teamwork Server home>/Projects` folder.

**NOTE**

Follow the instructions in dialogs to update profiles during project import.

Migrating from the SVN/ClearCase repository to the Built-In repository type:

1. Start the server on the SVN/ClearCase repository.

2. Open the **Administrator’s Console**, trigger project export. Select a directory to dump the permanently store the projects. This will be the directory in which the server will subsequently operate.

3. In the **Administrator’s Console**, reconfigure the server for the Built-In repository. Specify the directory where you stored the projects as a directory to work with.

4. Restart server in order to use this new repository.

5. Projects are now in a new Built-In repository.

### Moving Teamwork Server

You can move your Teamwork Server from one computer to another. During this transfer, server configurations, projects, and user information will be moved.

Let’s say, we are moving Teamwork Server from the computer A to the computer B. In order to do that, follow the instructions below.

To move Teamwork Server:

1. Stop Teamwork Server in the computer A (see "**Stopping Teamwork Server**" on page 19).

2. In a computer A, back up all data in case unsuccessful data transfer.

3. Install Teamwork Server in the computer B (see "**Installing**" on page 9).

4. In the computer A, copy Project” and Data folders from the Teamwork Server installation directory and paste them in the same location in the computer B.

   **NOTE**

   If your projects were stored otherwhere than in the Project folder, copy that folder wherein Teamwork Server projects were stored and the path to that repository. In the computer B, restore the path to the same repository and paste the folder with projects to that location. For more information about how to change the path to a repository, see the procedure "**To change a repository path**" on page 15.

5. Start Teamwork Server (See "**Starting Teamwork Server**" on page 15).

   **NOTE**

   Starting Teamwork Server for the first time, you will be required to enter a license key. For more information about license activation, see at http://www.magicdraw.com/installing/?product=teamwork#unlocking.
6. You can remove back up data in the computer A (optional).

**IMPORTANT!** Before removing back up data from the computer A, make sure that all data and server configurations are restored in the computer B after moving.

To change a repository path

1. In the **Teamwork Administrator's Console**, select the **Repository** tab.
2. In the **Repository Location** box, type or select the repository path.
3. Click **Apply Changes**.

For more information about repository, see "**Teamwork System Layout**" on page 6

**Starting Teamwork Server**

To start the server for the first time, run the file `teamwork_server.exe` from the bin folder and browse the license key when prompted. The key is no longer required the next time the server is run.

For more information about node locked license activation, see at [http://www.magicdraw.com/installing/?product=teamwork#unlocking](http://www.magicdraw.com/installing/?product=teamwork#unlocking).

**Starting the server using GUI**

To start Teamwork Server

1. Do one of the following:
   - Start Teamwork Server the same way you would start any application on your operating system.
   - Run `teamwork_server.exe` in the server bin folder. The **Teamwork Server** startup dialog opens.
2. Click the **Start Server** button.

To change the server license key

1. Run `teamwork_server.exe` in the server bin folder. The **Teamwork Server** startup dialog opens.
2. Click the **Licence Manager** button. The **Teamwork Server Licence Manager** dialog opens.
3. On the opened dialog, click the **Select License Key Files** button to browse for a file with the Teamwork Server license key.
4. In a browser, open a new licence key file.
5. Click **OK** after you have changed the server license key.

**NOTE**: Restart the server to apply changes.

To change the server port for the current launch

1. Run `teamwork_server.exe` in the server bin folder. The **Teamwork Server** startup dialog opens.
2. Click the **Change Server Port** button and enter the new server port. This port is used when launching the server, or when adding the Windows service.

The Teamwork Server exports remote objects through one port: RMI registry port.

**Starting the server without using GUI**

To start Teamwork Server from the command line

Do one of the following:
• Add the parameter NOGUI when starting the server from the command line.
• Run teamwork_server_nogui.exe in the server bin folder.

To add Teamwork Server to Windows services

1. Run teamwork_server.exe in the server bin folder. The Teamwork Server startup dialog appears.
2. Click the Add Windows Service button.
3. After the service is added, select one of the following:
   • Start this service from the Windows Services list.
   • Reboot the computer and the Service will start automatically.
   • To run the server, click the Start Server button.

NOTE
This feature is available only on Windows operating systems.

IMPORTANT!
Windows 7 OS and Windows Vista OS Firewall do not allow remote connections. Hence after adding Teamwork Server to Windows 7 or Windows Vista services, you have to add the Teamwork Server port number 1100 in Windows Firewall Exceptions list. Only then all remote connections to Teamwork Server will be allowed.

To start Teamwork Server as a service on Red Hat Linux

1. Create a new service script file named “teamwork”.
2. Copy the following script code and paste it into the file.

```bash
#!/bin/bash
#
# chkconfig: - 91 60
# description: MagicDraw TeamWork Server
### BEGIN INIT INFO
# Provides: teamwork
# Required-Start: $local_fs $network $named $remote_fs $syslog
# Required-Stop: $local_fs $network $named $remote_fs $syslog
# Short-Description: MagicDraw TeamWork Server
# Description: This script is used to start MagicDraw TeamWork Server
### END INIT INFO

RETVAL=0
TEAMWORK_HOME="/var/MagicDraw_Teamwork_Server/bin"
prog="teamwork_server_nogui"
prog_stop="stop_teamwork_server"
desc="MagicDraw TeamWork Server"
args="SERVICE"

check() {
    if [ -f /var/lock/$prog ]; then
        if ps -p $(cat /var/lock/$prog 2>/dev/null) >/dev/null; then
            return 0
        fi
    fi
    return 3
}

status() {
    check
    if [ $? -eq 0 ]; then
```
TEAMWORK SYSTEM
Installing and Running Teamwork Server

```
TEAMWORK_HOME=

case "$1" in
  start)
    start
    RETVAL=$?
    ;;
  stop)
    stop
    ;;
  restart)
    stop
    start
    RETVAL=$?
    ;;
  status)
    status teamwork
    RETVAL=$?
    ;;
  *)
    echo "$Usage: $0 {start|stop|restart|status}"
    exit 3
esac
exit $RETVAL
```

NOTE
This script can also be used in non-RedHat based GNU/Linux distributions.
3. Change the value of the TEAMWORK_HOME variable according to the path of the Teamwork Server installation bin folder.
4. Save the file and move it into the system directory “/etc/init.d”.
5. In the command line, type the following commands:

   cd /etc/rc3.d
   ln -s ../init.d/teamwork S99teamwork

   **TIP** You can also configure the service for runlevel using the following command:
   chkconfig --level 3 teamwork on

6. In the command line, type the following command:

   service teamwork start

To change the server license key from the command line

1. Stop the Teamwork Server (see "Stopping Teamwork Server" on page 19).
2. Start the Teamwork Server from the command line. Add the following argument: “-changeKey -key:<path to the key file location>”.

   This is the sample for the Windows OS:
   teamwork_server_nogui.exe -changeKey -key:C:\MagicDraw_16_0_TeamworkServer_key.xml

   **NOTE** The path to the key file should be fully qualified and without spaces.

### Stopping Teamwork Server

To stop Teamwork Server

- In the <Teamwork Server home>\bin directory, run the stop_teamwork_server.exe.

To stop Teamwork Server on Red Hat Linux

- In the command line, type the following command:

   service teamwork stop

To remove Teamwork Server from the Windows services

1. Run teamwork_server.exe in the server bin folder. The **Teamwork Server** startup dialog opens.
2. Click the **Remove Windows Service** button.

   **NOTE** This feature is available only on Windows operating systems.
Managing Teamwork Server

Users Management

You can create two types of users in Teamwork Server: normal users that may have different permissions and users with administrator rights. Users with administrator rights can do the following actions in Teamwork Server:

- Manage users.
- Create projects and assign users to them.
- Manage project versions and branches.
- Set user permissions for the system and projects (the read and edit modes are set by default).
- Remove users and projects from Teamwork.

The Teamwork Server users have their own user accounts (including login names and passwords given by the administrator) and various types of permissions. According to the storage place of the user accounts, users can be either:

- **Native** - the user’s account data is stored locally.
- **External** - the user’s account data is stored in the external database (Subversion/ClearCase and/or LDAP). Only the login name of an external user is stored locally.

You can create, edit, or remove both types of Teamwork users regardless of whether the integration with any external database (Subversion, ClearCase, LDAP) is enabled or disabled. The names of native and external users are unique per single server.

You can convert an external user to a native one and vice versa. Users with administrator rights can change a specific user's type by editing the user's account information, or convert a whole list of active Teamwork users by using the Teamwork Administrator's console.

You will be connected to Teamwork Server once the authorization process, which will prompt for your user ID (login name and password), has been completed. Upon verification, you can work with the system.

**NOTE** If there are two MagicDraw clients with the same login name, only one client is allowed to log into Teamwork Server at a time.

You can manage users in Teamwork Administrator’s Console or in MagicDraw UML when connection to Teamwork Server is established.

To add a new native user

1. From the **Collaborate** menu, select **Users**. The **Edit Users** dialog opens.
2. Click the **Add** button. The **Add User** dialog opens.
3. Enter the user's login name, full name for better identification, and password.
4. Click **OK**.
5. Select the types of system permissions for the user in the **Permissions** list.

To add a new external user

1. From the **Collaborate** menu, select **Users**. The **Edit Users** dialog opens.
2. Click the **Add** button. The **Add User** dialog opens.
3. Enter the user’s login name and full name for better identification.
4. Select the **External User** check box.
5. Click **OK**.
6. Select the types of system permissions for the user in the **Permissions** list.

**NOTE**  
As you cannot set a password for an external user in MagicDraw’s Teamwork Server, use an appropriate tool for managing the external database (Subversion, ClearCase, or LDAP) wherein the user's account is stored.

### To convert a native user to an external one by editing the user's account information

1. From the **Collaborate** menu, select **Users**. The **Edit Users** dialog opens.
2. Click the **Edit** button. The **Edit User** dialog opens.
3. Enter the user’s full name.
4. Select the **External User** check box.
5. Click **OK**.

**NOTE**

- The password of a native user, who has been converted to an external user, will be retained. However, it will not be used in the user authentication.
- The user's native password will be enabled again only if the user is converted back to a native user.

### To convert all native users to external

**NOTE**  
All converted users will be able to log into Teamwork Server only if they are available in the external user sources (LDAP, Subversion, or ClearCase server to which your server is integrated).

1. Start **Teamwork Administrator’s Console** (see "Starting Teamwork Administrator’s Console" on page 24).
2. In the **Active Users** tab, click the **Convert Native Users to External** button (see "Active Users tab" on page 25).
3. Click **Yes** to confirm your decision.

You will be informed once the conversion has been completed. A Teamwork Server's user conversion can be:

- **Successful** - when all the users are converted from native to external. In this case the informational message is displayed, and you can check the list of all converted users in the server log.
- **Unsuccessful** - when the conversion failed. In this case an error message is displayed, and you can see the server log for more details.
- **Non-applicable** - when there are no users to convert from native to external. In this case an informational message is displayed.

For the information about the server log file, see "Log File tab" on page 30.

### To convert an external user to native by editing the user's account information

1. From the **Collaborate** menu, select **Users**. The **Edit Users** dialog opens.
2. Click the **Edit** button. The **Edit User** dialog opens.
3. Enter the user’s full name.
4. Clear the **External User** check box.
5. Type and retype the password.

**NOTE**  
If the converted user used to be a native user, the password will be the same one used when he or she was a native.
6. Click OK.

To convert all external users to native

NOTE

- All converted users will not be able to log into Teamwork Server as they do not have passwords; therefore, the administrator has to set up a password for each user after the conversion.
- If the converted user used to be a native user, the password will be reset to the same one used when he or she was a native.

1. Start Teamwork Administrator’s Console (see "Starting Teamwork Administrator’s Console" on page 24).
2. On the Active Users tab, click the Convert External Users to Native button (see "Active Users tab" on page 25).
3. Click Yes to confirm your decision.

You will be informed once the conversion has been completed. A Teamwork Server’s user conversion can be:

- **Successful** - when all the users are converted from external to native. In this case an informational message is displayed, and you can check the list of all converted users in the server log.
- **Unsuccessful** - when the conversion failed. In this case an error message is displayed, and you can see the server log for more details.
- **Non-applicable** - when there are no users to convert from external to native. In this case an informational message is displayed.

For the information about the server log file, see "Log File tab" on page 30.

To remove a user from Teamwork

1. From the Collaborate menu, select Users. The Edit Users dialog opens.
2. In the Users area, select the user and click Remove.

To assign a project to a user

1. From the Collaborate menu, select Users. The Edit Users dialog opens.
2. Click More, if you do not see the Teamwork projects list. The list of available Teamwork projects is displayed in the Available Projects area.
3. Select a project you want to assign to the selected user.
4. Click the << button to move the selected project to the Assigned Projects list.
5. Click OK when you are done.

NOTES

- Once a user has been added to a project, the default user rights will be created allowing the user to access the project only according to the rights given.
- The system permissions have a higher priority over the project permissions. For example, a user whose system permissions allow model editing can edit all projects, even if the user does not have rights to edit the projects.

User permissions

You can give several types of permissions to the Teamwork users to coordinate the work of the whole team. You can specify the types of user permissions in the Edit Users dialog.

There are two categories of permissions:

- System access – user’s administrative permissions to access and manage users and projects.
• Project access – user’s permissions to work on specific projects.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Users can</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit model</td>
<td>Modify a Teamwork project.</td>
</tr>
<tr>
<td>Read model</td>
<td>Open the content of any Teamwork project.</td>
</tr>
<tr>
<td><strong>NOTE:</strong> If the List not assigned projects permission is not selected, you will be able to open the project using its URL.</td>
<td></td>
</tr>
<tr>
<td>Read with modules (Legacy)</td>
<td>Enables to access modules data from the project that is using it.</td>
</tr>
<tr>
<td>Assign user to project</td>
<td>Assign any user to any Teamwork project.</td>
</tr>
<tr>
<td>Edit project properties</td>
<td>Edit Teamwork project names and tags.</td>
</tr>
<tr>
<td>List not assigned projects</td>
<td>See all (assigned and not assigned) teamwork projects. If not selected, only projects that are assigned to the user will be listed.</td>
</tr>
<tr>
<td>Create project / category</td>
<td>Create a new Teamwork project or a category.</td>
</tr>
<tr>
<td>Rename category</td>
<td>Edit a category name.</td>
</tr>
<tr>
<td>Create user</td>
<td>Create a new user.</td>
</tr>
<tr>
<td>Remove project / category</td>
<td>Remove a project or a category from Teamwork Server.</td>
</tr>
<tr>
<td>Edit user properties</td>
<td>Edit user names and passwords.</td>
</tr>
<tr>
<td>Remove user</td>
<td>Delete user accounts from Teamwork Server. This permission will unlock all model elements locked by the user in all projects.</td>
</tr>
<tr>
<td>Access user list</td>
<td>Allows user to see other Teamwork Server users.</td>
</tr>
</tbody>
</table>

To view the users’ permissions

1. From the Collaborate menu, select Users. The Edit Users dialog opens.
2. A list of users and their permissions is presented in the Permissions area.

To edit user permissions

1. From the Collaborate menu, select Users. The Edit Users dialog opens.
2. In the Users area, select a user that permissions you want to edit.
3. Select the check box to give or clear it to remove the selected permission in the Permissions area.

To assign a user to a project

1. From the Collaborate menu, select Projects. The Edit Projects dialog opens.
2. Click More, if you do not see the unassigned users list. The list of available users is displayed in the Available Users area.
3. Select the user you want to assign to the selected project.
4. Click the << button to move the selected user to the Assigned Users list.
5. Click OK when you are done.

**NOTE** When a user is added to a project, default user rights are created, allowing the user to access the project according to the rights given.
Starting the Administrator’s Console

The MagicDraw UML Teamwork Server Administrator's Console is used for Teamwork Server status observation and Administrative control. The server holds information about active users connecting to the server and loaded projects. The Administrator can shutdown or restart the server, change its properties, and view log files (including debug information) for the server and separate projects.

Requirements

<table>
<thead>
<tr>
<th></th>
<th>Minimum</th>
<th>Recommended</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disk space</td>
<td>8MB</td>
<td>30MB* or more</td>
</tr>
<tr>
<td>RAM</td>
<td>64MB</td>
<td>128MB or more</td>
</tr>
</tbody>
</table>

1. Without JVM. Additional 22 MB are needed for JVM.

Installing Administrator’s Console

Windows platform

1. Run the downloaded file MD_UML_<version number>_teamwork_admin_win.exe.
2. Follow the instructions in the installation program.

Unix platform

1. Ensure that X Windows server is running.
2. After downloading, open a shell and go to the directory where you downloaded the installer. At the prompt type: sh ./MD_UML_<version number>_teamwork_admin_unix.sh.
3. Follow the instructions in the installation program.

MAC OS X platform

After downloading, double-click MD_UML_<version number>_teamwork_admin_mac.dmg and install the application by dropping the launcher into the Applications folder (or another preferred location).

Requires Mac OS X Leopard or Mac OS X Snow Leopard and Java 1.5.0_19 for 32-bit Mac OS X, and Java 1.6.0_10 for 62-bit Mac OS X.

Other platforms

1. Extract the downloaded file MD_UML_<version number>_teamwork_admin_no_install.zip.
2. Launch teamwork_administrator.exe or ./teamwork_administrator from the bin folder.

Starting Teamwork Administrator’s Console

To start Teamwork Administrator’s Console

1. Double-click teamwork_administrator.exe in <MagicDraw Teamwork Server installation directory>\bin. The Login to Teamwork Administrator’s Console dialog opens.
2. Type administrator’s login name and password, server name, and click OK. Teamwork Administrator’s Console starts.
Administrator’s Console Dialog

The Administrator’s Console dialog is constructed of six tabs: **Active Users**, **Projects**, **Log File**, **Properties**, **Repository**, and **LDAP Integration**. All these tabs are described in the following sections.

### Active Users tab

In the **Active Users** tab, the administrator is able to observe all the users who are currently connected to the Teamwork Server.

#### The GUI elements of the **Active Users** tab are described in the following table.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Columns of the Active Users list</strong></td>
<td></td>
</tr>
<tr>
<td>Login Name</td>
<td>The Teamwork user’s login name.</td>
</tr>
<tr>
<td>Real Name</td>
<td>The user’s full name.</td>
</tr>
<tr>
<td>Host Name</td>
<td>The host name.</td>
</tr>
<tr>
<td>Host IP</td>
<td>The host IP address.</td>
</tr>
<tr>
<td>Connection Time</td>
<td>The date and time the user connects to Teamwork Server.</td>
</tr>
<tr>
<td><strong>Buttons of the Active Users list</strong></td>
<td></td>
</tr>
<tr>
<td>Convert Native Users to External</td>
<td>Starts the conversion of all connected native users to external users.</td>
</tr>
<tr>
<td>Convert External Users to Native</td>
<td>Starts the conversion of all connected external users to native users.</td>
</tr>
<tr>
<td>Send Message</td>
<td>Opens the Send Message to Remote User dialog for sending message to a specific user. The button is available, when a user is selected.</td>
</tr>
</tbody>
</table>
Projects tab

The Projects tab displays, names, authors, and the status of all projects that are stored on the Teamwork Server.

![Teamwork Administrator's Console. Projects tab](image)

Import and Export from and to repository can also be triggered from this Projects tab. Import and export are conceptually described in the Section “Starting the Administrator’s Console” on page 24. Here only the GUI workflow is presented.

To import projects into the current server

1. In the Teamwork Administrator’s Console Projects tab, click the Import from Native Repository button. The Projects Import Wizard opens.
2. Select the directory where the projects to import are located. This must be a directory where project versions are stored in a Teamwork Server’s Built-In repository format, for example a directory on which the Teamwork Server has previously been configured to operate, or a directory where you have exported the projects using export functionality (see the procedure “To export projects from the current server” on page 28). Click Next.

**NOTE** To restore files from the backup that was made while upgrading the server, in the Path to directory box, type the path to the backup folder.
3. In the **Projects** list, all the projects that are stored in the directory are displayed. Select any projects to import from the list or click on the **Select All** button to import all projects. Click **Next**.

4. The **Resolve Naming Conflicts** tab appears. The import process will check if the selected projects/modules/profiles already exist in the selected destination directory server (comparison is done by name). There are two ways to resolve naming conflicts:

   - **Use destination**. The module will not be imported and all other projects/modules, which are being imported, will be modified to use the already existing module/profile in the current server.
• **Overwrite destination.** Import the selected profiles or modules and commit them by overwriting the existing ones in the current server. Click **Finish**.

To export projects from the current server

1. In the Teamwork Administrator's Console **Projects** tab, press the **Export to Native Repository** button. The **Project Export Wizard** opens.
2. Select projects/modules/profiles to export from the list, or click on the **Select All** button to import all projects/modules/profiles. Click **Next**.
3. Select the directory to export the selected project. This can be either a new directory or a directory where the built-in-type repository is stored.

4. The resolve naming conflicts in the destination directory opens. If you select a new directory to export the project, there will be no naming conflicts. The projects/modules/profiles, to be export already exist in the selected destination directory (comparison is done by name), the naming conflicts will occur.

5. There are two actions to resolve the naming conflicts:
   - **Overwrite Destination.** Overwrite the projects/modules/profiles that already exist in the destination directory with the exported projects/modules/profiles.
   - **Use Destination.** Do not replace the projects/modules/profiles that already exist in the destination directory.
To view the details of a project

Select a project and from the Teamwork Administrative’s Console dialog and click Details. The Project dialog opens.

Log File tab

If errors occur on the Teamwork Server, use the Log File tab to view the error message.
Properties tab

The Properties tab displays the following Teamwork Server properties:

- `muserver.ping.time` records how often the Teamwork Server pings a client. After sending a ping signal, the server waits for the client to answer. If the client does not respond during the interval of time specified in the `muserver.ping.timeout.time` property, the user is logged out from the Server.

- `muserver.ping.timeout.time`. Displays the number of seconds the Teamwork Server will wait for a client's response once the server sent its ping. If no answer is received during this interval, the user is logged out from the Server.

- `muserver.rmiregistry.port`. Specifies the port on which communication between MagicDraw clients and Teamwork Server occurs (RMI traffic over TCP/IP).

**NOTE**

Restart the server to apply changes.

Secured Connection tab

Since version 17.0, you can transfer data in a more secure way using the secured connection (SSL).

**NOTE**

If the SSL connection is established in the server side, you should also use the SSL connection in the client side when connecting to the server.

In order to use the SSL connection, two types of certificates are needed: one for the server and one for the client. Certificates should be in a Java Key Store format.

The server certificate is automatically placed in the `<Teamwork Server installation directory>\cert` folder after the SSL configuration is done.

The client certificate should be located manually. You should create a folder named “certs” and place into it these two files:

1. A client certificate named `cert.jks`.
2. A file named `cert.pass` wherein the certificate password is typed.
In this case applications MagicDraw and Teamwork Administrator’s Console are Teamwork server clients. Both of it should have the client certificate. Hence the certs folder should be placed in two locations:

- `<Teamwork Administrator’s Console installation directory>` for the Teamwork Administrator’s Console application
- `<user home\>`\`.magicdraw\<version number>` for the MagicDraw application (It can be located in the folder `<MagicDraw installation directory>` either but the user home folder is the default one)

**NOTE** If the Teamwork Administrator’s Console is not installed in the separate location, its installation directory is the same as Teamwork Server installation directory or the MagicDraw installation directory (if installed together).

You can get certificates from your system administrator or to generate it by yourself. For more information about generating certificates, see procedure "To generate certificates" on page 33.

To enable the secured connection (SSL)

**IMPORTANT!** Only the user with administrator rights can configure Teamwork Server options. The user should be disconnected from Teamwork in MagicDraw while using the Teamwork Administrator’s Console.

1. Run the Teamwork Server Administrator’s Console.
2. In the Secured Connection tab, select the Enable Secure Connection check box.
3. Click Browse to add the server certificate.
4. Enter the JKS password.
5. Click the Apply Changes button.

**NOTE** If you want to change the password for the certificate, you need to regenerate the certificate with a different password.

To remove a certificate, click the Remove button.

---

1. To find out what is your home folder you may open Help > About MagicDraw and select the Environment tab in the MagicDraw application. Configuration Files links to your home folder.
To generate certificates

TIP! We recommend you to use the KeyTool IUI application for generating certificates. This is a free tool that can be downloaded from the Internet.

1. Run the KeyTool IUI application.
2. Create empty files for storing certificates:
   2.1. Select Create > Keystore.
   2.2. Create an empty keystore file for the server. Do the following:
       2.2.1 In the Keystore file dialog, set the location of the file and type a file name.
       2.2.2 In the Keystore password dialog, type the password for the server keystore file and click OK.
   2.3. Create an empty keystore file for the client. Do the following:
       2.3.1 In the Keystore file dialog, set the location of the file and type a file name.
       2.3.2 In the Keystore password dialog, type the password for the client keystore file and click OK.
3. Create a RSA keypair for the server:
   3.1. Select Create > Keystore’s entry > Private key, with vers. #3 > RSA.
   3.2. In the Keystore file dialog, the Source section, open the created server keystore file and type a password.
   3.3. Provide the required information in the Target section and click OK.
   3.4. The dialog for creating a new alias will open. Type a new private key entry’s alias name and a password for it. Click OK.
   3.5. You will see the created alias. Close the dialog.
4. Exclude a public key from the keypair to provide it to the client:

TIP! For easier certificate transfer in next steps create a new folder “certs” and save the file named cert.jks in it.
4.1. Select Export > Private's key first certificate in chain > As simple certificate file.
4.2. In the Keystore file dialog, the Source section, open the server keystore file and type its password.
4.3. Create a file, whereto the key will be exported. In the Certificate file dialog, the Target section, set a location and type a file name for the client certificate. Click OK.
4.4. The dialog for selecting an alias will open. Select from the list the alias that has been created in step 3.4 and type its password. You will be able to see the created certificate.

5. Import a public key into the client certificate:
5.1. Select Import > Keystore's entry > Trusted certificate > Regular certificate.
5.2. In the Source section set the certificate file, which has been created in step 4, as a regular certificate file.
5.3. In the Target section set the client keystore file client.jks as a keystore file and click OK.
5.4. The dialog will open asking to enter a new alias name. Enter the alias name created in step 3.4 and click OK.
5.5. Some pop-up windows will open informing about the generation process. Close all of them after reviewing.

Generated certificates are ready to use now. Paste them into the right location. For the information on how to configure SSL in a server side please refer to procedure “To enable the secured connection (SSL)” on page 32.

Repository tab

Repository tab is a very important tab. It contains information about the repository of the server. This tab determines where and how the Teamwork Server stores projects and their version information, user lists, etc.

For more information about repositories, see “Teamwork System Layout” on page 6.

NOTE Restart the Server to apply changes.

There are 3 different types of repository to choose in the Repository type combo box. General information about the types is described in the Section “Teamwork System Layout” on page 6. This section, presents the configuration fields in detail.

The Repository tab layout changes with the type of repository selected.
Built-In repository

The Built-In repository type is the simplest one to configure. There is only one editable parameter, that is the Repository Location.

![Figure 10 -- Teamwork Administrator's Console. Repository tab for Built-In repository type](image)

The GUI elements of the Repository tab for the Built-In repository type are described in the following table.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields</td>
<td></td>
</tr>
<tr>
<td>Repository Location</td>
<td>Designate the directory where the Teamwork Server stores information about: project list, users list, project versions, etc.</td>
</tr>
<tr>
<td>Authentication Type</td>
<td>The Built-In repository type can only use built-in authentication type. This field is not editable and for information purposes only.</td>
</tr>
<tr>
<td>Buttons</td>
<td></td>
</tr>
<tr>
<td>Test Connection</td>
<td>Test new parameters before applying them to the Teamwork Server. If the server cannot start with these new parameters, you will not be able to connect it with the Administrators console and change the parameters back to the old values. You will need to manually edit the muserver.properties file on the computer where Teamwork Server is installed (this file is located in &lt;install.root&gt;/data directory). The Built-In repository type is simple because it requires at least a repository location directory that can be created and edited.</td>
</tr>
</tbody>
</table>
SVN repository

When configuring the SVN repository, SVN must be installed on the computer where Teamwork Server runs. The `svn` executable must be accessible on the system's PATH and have appropriate permissions to execute.

<table>
<thead>
<tr>
<th>Element name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fields</strong></td>
<td></td>
</tr>
<tr>
<td>Repository URL</td>
<td>An URL field, which points to the SVN repository that will be used. The path can direct to a repository root or any other inner repository folders. All other paths (configurations location, profile location, also project paths, when adding project to the repository) will relate to this location. All repository access types, which are supported by standard SVN installation, are supported here. In particular, file:// (local repositories), svn:// and svn+ssh:// (remote repositories accessible through remote login and ssvnservr), http:// and https:// (remote repositories, accessible through HTTP and WebDAV) URLs can be used here.</td>
</tr>
<tr>
<td>Project Cache Directory</td>
<td>The directory where the Teamwork Server checkout files to work with the repository. The Teamwork Server uses this directory to launch all necessary SVN operations to implement project storage and create project version. This is a “scratchpad” for the Teamwork server to work.</td>
</tr>
</tbody>
</table>
Server Configuration Location

The directory in SVN repository where the Teamwork Server keeps its configuration files (project and user lists).

System Profile Location

The directory in SVN repository where the MagicDraw's system profiles are stored.

Authentication Type

Currently not editable. This is an informational-only field, which indicates what authentication type will be used depends on the repository access method (repository URL). If the repository URL is:

- `svn://`, `svn+ssh://`. The Authentication Type will be **Remote Login**. This means that when performing SVN actions, the Teamwork Server will log in remotely, as the user, who performs the action. A pass-through authentication of MagicDraw clients will be used.
- `http://`, `https://`. The Authentication Type will be **Basic/Digest HTTP**. This means that when performing SVN actions, the Teamwork Server will authenticate with the web server using HTTP Basic Auth or HTTP Digest Auth (see RFC2617), as the user who performs the action. A pass-through authentication of MagicDraw clients will be used.
- `file://`. The Authentication Type will be **Built-In**. A pass-through authentication is impossible for this type of repository access, hence the Teamwork Server refers back to the same authentication type as that of built-in repositories. The Server will keep its own users list with their passwords, and authenticate users itself.

The pass through authentication means that the Teamwork Server will only keep users list, but not their passwords. When a MagicDraw client logs into the Teamwork Server, the server will not verify the password, but will log into SVN instead. It stores the user's password in memory and use it for all subsequent communications with the SVN prompted by the user's actions. This feature decreases the burden on both the administrator and users because all passwords are kept in one place.

Certificate based authentication method for `svn+ssh://` and `https://` is not yet supported.

For pass-through authentication methods (all types of repository URLs except `file://`) the following fields are available:

**Administrator login name**

Specify the administrator of the Teamwork Server. This user will have full permissions to the projects and various administrative tasks, such as updating user list file, project list file, etc. in the SVN repository.

**Administrator password**

The Administrator's password. The Teamwork Server needs this password to perform various administrative tasks in the SVN repository.

**NOTE:** this password will be stored and encrypted in a form that is difficult to decrypt. This encrypted password is stored in the muserver.properties file (located in `<install.root>/data` directory of the Teamwork Server). You can set restricted permissions on this file for example, accessible only to user in the Teamwork Server.

**Auto-add unknown users to Teamwork if they log in successfully**

Decreases the administrator's burden to manage users list. If this check box is cleared, all MagicDraw users who try to log in, but are not in the users list of the Teamwork Server will be refused an access if they supply correct usernames/passwords to login the SVN repository. If the check box is checked, MagicDraw users who are not in the users list of the Teamwork Server, but supply correct usernames/passwords to login to the SVN repository will be automatically added to the list and access the server.

The default Built-In authentication method (`file://` type URLs) login name is "Administrator" without a password field (not necessary). The "Auto-add unknown users to the Teamwork if they log in successfully" check box is also unavailable (meaningless) in this case.
The ClearCase type of repository has a few parameters. When configuring this repository, the ClearCase client part must be installed on the computer where the Teamwork Server runs. The Teamwork Server will use cleartool executable to access and manage files in the ClearCase repository, so cleartool must be accessible - the PATH and have appropriate permissions to run. Also, the ClearCase repository must have the right permissions for the user, the Teamwork Server.

<table>
<thead>
<tr>
<th>Property name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Project Cache Directory</strong></td>
<td>The directory where the Teamwork Server checkouts files to work with the</td>
</tr>
<tr>
<td></td>
<td>repository. The Teamwork Server will create a snapshot view (in ClearCase</td>
</tr>
<tr>
<td></td>
<td>parlance) in this directory and place any necessary VOBs there project and</td>
</tr>
<tr>
<td></td>
<td>perform all operations necessary to implement project storage and project</td>
</tr>
<tr>
<td></td>
<td>versions. This is a “scratchpad” for the Teamwork Server to work.</td>
</tr>
</tbody>
</table>
### Property name | Function
---|---
**Server Configuration Location**<br> The directory in the ClearCase repository, where the Teamwork server keeps its configuration files (project and user lists).<br> **NOTE:** Must begin with some VOB name and cannot be left empty.

**System Profile Location**<br> A directory in the ClearCase repository where the MagicDraw's system profiles are stored.<br> **NOTE:** Must begin with some VOB name and it cannot be left empty.

**Authentication Type**<br> The ClearCase repository type can only use built-in authentication type. This field is not editable and for information purposes only.<br> The Built-In authentication means that the Teamwork Server manages its own users list and their passwords. When MagicDraw clients connect to the server, it will authenticate them using this list.<br> When the Teamwork Server works with the ClearCase repository, it uses the standard system authority to identify itself to the repository. This means that the Server run as authorized by the user. All actions in the ClearCase repository will be attributed to that user (irrelevant to users in the connecting MagicDraw client). If a user starts the server, it will run as authorized by that user. If the server is started as NT service the service will run under a local system account. This can be a source of cryptic errors, for example if the local system account has no rights to access the ClearCase repository, the Teamwork Server will fail when started as a service, while everything works all right when started from the command line.

**Administrator Login Name**<br> When the Built-In authentication type is used, the administrators name is “Administrator”. This field is not editable and for informational purpose only. The default Administrator's password is (for a new set-up repository) “Administrator”. For security reason it is highly recommended to change the default password after the server starts.

**Buttons**

**Test**<br> Test new parameters before applying them to the Server. If the server cannot start with these new parameters, you will not be able to connect it with the Administrators console and change the parameters back to the old values. You will need to manually edit the muserver.properties file on the computer where Teamwork Server is installed (this file is located in <install.root>/data directory). This test ensures that Teamwork Server is able to create the ClearCase view in the specified directory and to access the two configured VOBs (for configuration storage and profile storage).

### LDAP Integration tab

For more information about LDAP Integration, see "LDAP Support" on page 41.

### Data Migration between Different Repositories

3 different types of repositories are supported in MagicDraw v12.5 and above. By default (when first installed) the server uses Built-In repository type. When switching to a different repository type, the task of project migration is inevitably increased.

There are many different but similar use cases, such as migrating the data from the previous built-in-type repository into a new SVN/ClearCase based repository.
It is possible for the Teamwork Server to import and export projects. This function is triggered from the Administrators console (see section “Administrator’s Console Dialog” on page 25). Import & Export is only possible in the Built-In repository type format. The Built-in repository type is a kind of intermediate form for information interchange.

Note that words “import” and “export” here are used relatively to the currently running server (the one to which administrator’s console is attached):

- “import” means importing data into the current server from the designated directory;
- “export” means exporting data from the current server into the designated directory.

**Migrating the server from the Built-In repository to the SVN/ClearCase repository**

1. The Server is started on the Built-In repository (projects/versions stored in a directory).
2. Open the Administrator's Console. Reconfigure the server to work with the SVN/ClearCase repository.
3. In the **Administrator's Console**, reconfigure the server to work with the ClearCase/SVN repository.
4. Restart server to use the new repository. The Server starts. There is nothing in the repository but the profiles needed to work with MagicDraw.
5. Login again to the **Administrator’s Console** and trigger project import. Select the directory to import from, the same directory from where the export was performed.
6. Projects are now in a new SVN/ClearCase repository.

**Migrating from SVN repository to ClearCase repository**

1. Server is started on SVN/ClearCase repository.
2. Open Administrator's Console, trigger project export. Choose some temporary directory to dump the data to.
3. In **Administrator’s Console**, reconfigure server for work with some ClearCase/SVN repository.
4. Restart the server, to use new repository. Server will be started, repository will be empty - no projects - only the profiles, needed for work with MagicDraw will be added to repository.
5. Login again in the **Administrator’s Console**, and trigger project import. Choose the directory to import from - the same directory where you previously did export.
6. Projects are now in a new, ClearCase/SVN repository.

**Changing Teamwork Server Debugging Mode**

You can change Teamwork Server debugging mode without restart.

To enable Teamwork Server debugging:

1. Start Teamwork Server Administrator’s Console.
2. Select **Enable Debugging** from the Menu (Figure 13 on page 41).

**NOTE** **Enable Debugging** action is disabled in the Offline Administrator's Console mode.
LDAP Support

LDAP Integration allows Teamwork Server to authenticate its users against LDAP servers. LDAP Integration enables pass-through MagicDraw authentication against LDAP servers, by passing client’s authentication information to LDAP servers.

LDAP Integration supports Simple User+Password and SASL authentication, SSL/TLS protocols, and several LDAP servers configured for a single integration.

Enabling LDAP Integration

To enable LDAP Integration

1. Start Teamwork Administrator’s Console (see “Starting Teamwork Administrator’s Console” on page 24).
2. Click the LDAP Integration tab.
3. Click Enable LDAP Integration. LDAP integration settings become active (see the following figure).
4. After enabling the integration, specify all mandatory setting values. Mandatory settings are marked with the star sign at the end of name. There are four groups of settings in the LDAP Integration tab:
   - Connection Settings (see on page 42)
   - Authentication Settings (see on page 43)
   - User Data Retrieval Settings (see on page 48).
5. Click Apply Changes when you are done. You will be required to test the connection against the LDAP server. For more information about the connection testing, see “Connection Testing” on page 50.
Connection Settings

Connection Settings specifies network and security settings for connecting to LDAP servers.

![Teamwork Administrator's Console, LDAP Integration tab. Connection Settings](image)

Connection Settings are described in the following table.

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server Address(es)</td>
<td>A list of servers separated by spaces where each entry holds server address and server port. If unspecified, the 389 port is used. At least one server address must be specified. Usually a master server and its slaves (replicas) are specified for round-robin authentication. Teamwork Server authenticates against servers in the order they are listed in the Server Adress(es). If authentication does not succeed to the first server in the list, the second server is used, and so on until authentication is successful. Authentication failure for the whole integration is considered as an inability to authenticate against any of the specified servers. A single server in the specified list is queried within the period of time specified in the Server Timeout setting.</td>
</tr>
<tr>
<td>Server Timeout</td>
<td>A time duration that specifies maximum period of time in milliseconds to successfully authenticate to a single server. If authentication is unsuccessful within this period of time, the next server in the server list is queried. The default value for this option is 500 milliseconds.</td>
</tr>
</tbody>
</table>
Authentication Settings

LDAP Integration supports two most popular LDAP authentication methods. They are as follows:

- Simple User+Password (see "Authentication settings for the Simple User+Password authentication type" on page 44).
- SASL (see "Authentication settings for the SASL authentication type" on page 47)

Teamwork Server transforms user credentials entered in MagicDraw to LDAP authentication credentials by using the templates in authentication settings. After successful authentication to LDAP, a special user for each authenticated LDAP user is created in Teamwork Server. They differ from ordinary users as they have no passwords (in order to complete authentication, authentication to LDAP server(s) is used). You can perform various actions for these users, otherwise. It is possible to setup permissions, remove users, and do other common actions with users.

NEW! You can automatically create a proxy user account for Teamwork Server. Select Auto-add unknown users if they login successfully checkbox and authenticated user credentials without password will be stored in Teamwork Server on successful login.

To select an authentication type for LDAP integration

- In the Authentication Type list, select a desired protocol.

![Figure 15 -- Teamwork Administrator’s Console, LDAP Integration tab. Selecting Authentication Type](image-url)
Authentication settings for the Simple User+Password authentication type

Using the Simple User+Password authentication type, you can select the following options:

- Use User DN template
- Retrieve User DN by using an LDAP query

Authentication using a user DN template has the following characteristics:

- A hard-coded template is filled-in with the user login supplied on the login process to Teamwork Server.
- User DN is used to login to LDAP server.

Authentication using retrieved user DN performs in the following order:

1. A query template is filled-in with the login name entered by the user.
2. An anonymous bind or specific User DN and password is used to connect to the LDAP server.
3. The LDAP server is queried for User DN by using the query produced in the step #1, Search Base, and Search Scope settings’ values.
4. The LDAP server returns User DN by the query.
5. Teamwork disconnects from the LDAP server.
6. Teamwork tries to login to LDAP using returned User DN and password supplied by the user during the login process.
Figure 16 -- Teamwork Administrator's Console, LDAP Integration tab. Authentication Settings (Simple User+Password)

Authentication Settings are described in the following table.

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use User DN template</td>
<td></td>
</tr>
<tr>
<td>Search Base</td>
<td></td>
</tr>
<tr>
<td>Search Scope</td>
<td>Subtree</td>
</tr>
<tr>
<td>Anonymous Bind</td>
<td></td>
</tr>
<tr>
<td>Bind DN</td>
<td></td>
</tr>
<tr>
<td>Bind Password</td>
<td></td>
</tr>
</tbody>
</table>

Settings that are active when the Use User DN template is selected
User DN stores a template, which is used for mapping the user’s authenticating against Teamwork Server to LDAP distinguished names when authenticating. The template recognizes a single keyword $(login). An example of the template:

\[ cn=$(login), dc=example, dc=com \]

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>User DN</td>
<td>User DN stores a template, which is used for mapping the user’s authenticating against Teamwork Server to LDAP distinguished names when authenticating. The template recognizes a single keyword $(login). An example of the template: cn=$(login), dc=example, dc=com</td>
</tr>
</tbody>
</table>

**Settings that are active when the Retrieve User DN by using an LDAP query is selected**

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Query</strong></td>
<td>The LDAP query for retrieving User DN, for example: ( uid=$(login) )</td>
</tr>
<tr>
<td><strong>Search Base</strong></td>
<td>DN, from which a search should begin, for example: dc=example,dc=com</td>
</tr>
</tbody>
</table>
| **Search Scope** | Search scope specifies whether the search must be restricted to the directly owned DNs only or it must be performed in the whole subtree. Choose one of the following:  
  - One level  
  - Subtree |
| **Anonymous Bind** | A mode of bind, specifying, whether the user connects to LDAP server with a specific user or anonymously for being able to find the User DN which corresponds to the user who is trying to login to Teamwork.  
  **IMPORTANT!** You must have such a user if you do not have anonymous access. |
| **Bind DN**      | Specific User DN for connecting to the LDAP server and perform queries.  
  **NOTE:** This element is active, when **Anonymous Bind** is not selected. |
| **Bind Password** | A specific password to connect to the LDAP server and perform queries (you must to have such a user if you do not have anonymous access!)  
  **NOTE:** This element is active, when the **Anonymous Bind** is not selected. |
Authentication settings for the SASL authentication type

Figure 17 -- Teamwork Administrator's Console, LDAP Integration tab. Authentication Settings (SASL)

Authentication Settings are described in the following table.

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication Identity</td>
<td>Login name supplied by the user is transformed to an authentication identity when authenticating. The authentication Identity is a mandatory template. The template recognizes a single keyword $(login). An example of a template: $(login)</td>
</tr>
<tr>
<td>Authorization Identity</td>
<td>Login name supplied by the user is transformed to an authorization identity when authenticating if the Authorization Identity template is specified. The template recognizes a single keyword $(login). An example of a template: $(login) or $(login)@example</td>
</tr>
<tr>
<td>Realm</td>
<td>Specifies the realm of an authentication identity for the SASL bind.</td>
</tr>
</tbody>
</table>
User Data Retrieval Settings

Authenticated users usually have an access to User DN attributes in the LDAP database. If the user information retrieval is enabled, and User DN attributes are accessible to the authenticated user, Teamwork Server retrieves their values and sets them for the corresponding external users.

User DN is retrieved in the same way as it is done if the Simple User+Password authentication type is enabled (either by using a static User DN template or by querying the LDAP server(s) for User DN). When the user logs in to the LDAP server, this connection is further reused for retrieving user information.

If the user information retrieval is disabled or User DN attributes are not accessible to the authenticated user, Teamwork Server creates an external user with the login name that was specified by the user on the authentication.

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mechanism</td>
<td>Specifies the SASL mechanism to be used for authentication. An example: DIGEST-MD5</td>
</tr>
</tbody>
</table>
Figure 18 -- Teamwork Administrator’s Console, LDAP Integration tab. User Data Retrieval Settings

User Data Retrieval Settings are described in the following table:

<table>
<thead>
<tr>
<th>Setting Name</th>
<th>Description</th>
</tr>
</thead>
</table>
| User DN Attribute-to-Full Name Mapping | After a specific User DN is found, the name of a local user which is created on the authentication is created by using the Full Name Mapping template for this User DN. The Full Name Mapping template supports placeholders in the form of $(attribute), where attribute is an attribute of DN. An example: 
  $ (cn) $ (sn) 
  This will form the Name of the created user out of two LDAP attributes - cn and sn. |
Connection Testing

After you have specified the LDAP Integration setting values, you can test the connection to LDAP server.

To test a LDAP Integration configuration

1. In the LDAP Integration tab of Teamwork Administrator’s Console, click Test Connection. The Test Connection dialog opens.
2. Type user’s login name, password, and click OK.
3. The message with connection results appears.

Subversion and LDAP Integration Working at the Same Time

If Teamwork Server uses Subversion for storing projects and LDAP Integration for authentication, Subversion must recognize (or authenticate) the same users and passwords that are used to authenticate against the LDAP server. Administrator is the user specified in the LDAP Integration configuration.

When Teamwork is integrated with Subversion only, client's authentication information is passed to the Subversion server. When Teamwork Server is integrated with Subversion and LDAP, client's authentication information is passed for both Subversion server(s) and LDAP server(s), but only successful authentication to LDAP server(s) successfully logs the user into Teamwork Server.

Converting Certificates to JKS Format

Teamwork Server recognizes certificates stored in Java KeyStore (JKS) format. If certificates are in PEM format, they have to be converted to JKS format. OpenSSL (http://www.openssl.org/) and Sun Java KeyTool (included into every Java distribution) can be used for this purpose. For example, if we a certificate in a file called cert.pem, then the following commands will covert it to JKS format:

```
openssl x509 -in cert.pem -out cert.der -outform der
keytool -importcert -alias mycert -file cert.der -keystore truststore.jks
```
Integrating Teamwork Server with SSL-Enabled Active Directory


To complete this integration successfully, the following requirements should be passed:

- Windows Server Active Directory should have SSL enabled. This includes a valid Certificate Authority (CA) and a valid certificate for Active Directory (AD) server certificate (for more information on installing and configuring Certificate Services for Windows Server, see Microsoft documentation).
- Any SSL-aware LDAP client should be able to connect to your AD server port 636 with SSL enabled and should have access to its contents (for more information on setting SSL-enabled connections to AD, refer to the specific LDAP client documentation).

To create a Java KeyStore with the included CA and AD server certificates

1. Export the CA and AD server certificates to the DER encoded binary.cer files using the Microsoft Management Console, the Certificates Snap-in.

   **IMPORTANT!** Do not include private keys while exporting.

2. Import the CA and AD server certificates (.cer files) to Java KeyStore (.jks file), using the KeyTool IUI. Do the following steps:
   2.1. Run the KeyTool IUI.
   2.2. Double-click **Create** on the tree, and then click **Keystore** to create a new keystore file.
   2.3. Choose the JKS format and save a new keystore file.
2.4. Set the password for the keystore.

![Figure 19 -- Creating a new keystore file with the KeyTool IUI (steps 2.2, 2.3, and 2.4)](image)

2.5. On the tree, double-click **Import**, **Keystore's Entry**, and **Trusted Certificate**, and then click **Regular Certificate** to import the .cer files into Java KeyStore.

2.6. Select the created keystore file as the target.

2.7. Select the exported CA certificate file (.cer) as the source. Enter the keystore password and click **OK**. Enter “CAAlias” as the alias for the CA certificate and click **OK**.
2.8. Select the exported AD server certificate file (.cer) as the source. Enter the keystore password and click OK. Enter the full name of the AD server as the alias for the AD certificate and click OK.

Now you have a Java KeyStore containing both certificates.

The subsequent steps for the Teamwork Server integration with SSL-enabled Active Directory are the same as for the integration with any other LDAP server. This procedure is described in section “Enabling LDAP Integration” on page 41.

Connecting Teamwork server via SSH encrypted tunnel

In this section you will find the description for SSH-encrypted MagicDraw UML and Teamwork Server connectivity.
Requirements on both server and client are as following:

1. Windows 2000/XP/Vista or Windows Server family OS.
2. Local administrative rights to create local user for tunnelling and to run SSH service.
3. OpenSSH server and client binary installation files. You may obtain it from http://sourceforge.net/project/showfiles.php?group_id=103886&package_id=111688#. The OpenSSH software is free of charge.

Configure the Teamwork Server side

1. Login to the Teamwork Server with administrator privileges.
2. Install OpenSSH with default settings. This will install OpenSSH server and client on your Teamwork Server machine. The warning about editing password and group file appears while installing. Click OK.
3. Create a local user for SSH tunnel. To do this right click on My Computer, then select Manage. In the Local Users and Groups section right-click on Users and choose New User. The New User dialog opens (see Figure 21 on page 54).
   3.1. Enter a new username which you will use to login into SSH service to establish tunnelling. For example, tunnel.
   3.2. Enter the user password according to your local system policy.
   3.3. Clear the User must change password at next logon check box.
   3.4. Click Create. The local user will be created.

NOTE: This is not the same as MagicDraw Teamwork Server user used to checkout and commit UML models from/to the server. Use Teamwork Administrator to manage Teamwork users.
4. Create a local group for SSH tunnel users. To do this right-click on My Computer and then select Manage. In the Local Users and Groups section right-click on Group and choose New Group. The New Group dialog opens (see Figure 22 on page 55).
   4.1. Enter a new groupname, for example, SSH.
   4.2. Add the “tunnel” user to the SSH group.

5. Create SSH-aware local password file with 'tunnel' user entry. Any users in this password file will be able to log on with SSH. To create SSH-aware local password file run command prompt (click "Start"-> "Run", then type "cmd" and click Enter) and then type the following commands:

   cd C:\Program Files\OpenSSH\bin
   mkgroup -l >> ..\etc\group
   mkpasswd -l -u tunnel >> ..\etc\passwd

6. Start OpenSSH Server service from your control panel. To do this right-click on My Computer and then select Manage. In the Services and Application section, the Services item right-click on the OpenSSH Server service and choose Start.

7. Test the SSH server.
   7.1. Type "ssh tunnel@localhost" from your command prompt.

   The following warning appears:
   
   The authenticity of host 'localhost (127.0.0.1)' can't be established.
   Are you sure you want to continue connecting (yes/no)?

   7.2. Type yes and press Enter. NOTE: type the full word "yes", not only "y".
   7.3. Enter the password you created at step 3.
   7.4. Warning about nonexisting home directory appears. Please ignore it.
   7.5. Now you are logged in into localhost via SSH service and you can see the shell prompt.
   7.6. After the SSH server testing exit from the server by typing exit.
Connecting Teamwork server via SSH encrypted tunnel

NOTE: Other way to test if the SSH port (port 22) is opened on the server.

In the command prompt go to the C:\Program Files\OpenSSH\bin and type the 'netstat -na' command. You will get the list of all connections. The state of the port should be "LISTENING" while the SSH server is running.

Client Side Configuration

1. Install OpenSSH client from the same installation file you downloaded. Uncheck "Server" option when asked

The SSH client package is installed to "C:\Program Files\OpenSSH" by default and added into your PATH variable.

2. Establish an SSH tunnel by logging into Teamwork server SSH service from the command prompt:

   ssh - L localport:teamworkserver:teamworkserverport username@teamworkserver

   For example, the following command will establish an SSH encrypted tunnel from client port 1100 to server port 1100. When connecting to localhost:1100, the packets get encrypted and sent to twserver:1100, where actual Teamwork server resides:

   ssh -L 1100:twserver:1100 tunnel@twserver

   localport may be any unused port on your workstation.

Teamwork server port is the port the Teamwork server is running (usually 1100)

You are logged as user "tunnel" to the SSH service on Teamwork server machine. Leave the session opened, as killing it also kills the SSH tunnel used for the MagicDraw Client.

3. Open MagicDraw Client. Use localhost:localport when connecting to the Teamwork server. Localport is 1100 in this case as we used it when created the tunnel.

NOTE: Using any other value than "localhost" or "127.0.0.1" will fail, even if connected to the actual machine name, resolved by DNS. This is because tunnel starts on loopback interface of your workstation only for security reason.