Post-Installation Modifications

Post-installation modifications may be required, particularly the database connection properties for installations that used the Standard PostgreSQL Server or an existing database.

Also, instructions are provided below for increasing the amount of memory allocated to Servoy, which can enhance overall performance.

Modifying Database Connection Settings

Users can access the `servoy.properties` file found in the `../application_server/` folder (found in the root installation folder) to modify database connection settings.

- This file contains the database connection settings that were created during installation. Any post-installation modifications can be made here.
- If the full-featured PostgreSQL is installed, in order to use it with the sample database files, the sample database information needs to be entered manually in this file.

Servoy also provides additional database connection configuration via the Resources project while working on a solution. For details see Working with Database Servers.

Creating the Servoy Repository

The Servoy Repository, a set of tables in the database underlying the nameserver connection 'repository_server' is required to run Servoy. If the bundled PostgreSQL installation option is used, the repository is already installed. When using the full-featured PostgreSQL installation or when choosing to connect to an existing database, the repository will need to be manually installed.

This is achieved via the command line process that can also be used to upgrade the repository:

1. Shut down the Servoy Application Server if it is running
2. Open a command line or terminal window and execute the following command in the `../application_server/` directory:

Macintosh/Linux/Unix

```
./servoy_server.sh -upgradeRepository
```

Windows

```
servoy_server.bat -upgradeRepository
```

Executing the above command starts the Servoy Application Server and will create or upgrade the table structure required for the Servoy Repository to the version that the Servoy Application Server requires.

Increase Memory Allocations

The startup configuration file can be modified to increase the memory allocated to Servoy, which can improve overall performance.

1. Open the `../developer/Servoy.ini` file in a text editor. (For Mac users, this file is exposed in the `MacOS` folder, in the Servoy application package contents)
2. Change the `-Xmx` argument, for example, from `-Xmx512m` to `Xmx1024m`.
3. Restart Servoy to effect the changes.

Install Mozilla XulRunner As Internal Browser

Servoy Developer uses a browser component in several locations, like the Developer Start Page and the updated Form Editor for Servoy Mobile in Servoy 7.4.

By default the browser component used is the default browser of the operating system, so Internet Explorer on Windows or Safari on OSX.

In case of issues with the default browser (especially with the Form Editor), it is possible to install a FireFox based browser component called XulRunner into Servoy Developer, which will then be used instead of the default browser.

The XulRunner browser is available as plugin and can be installed using the following steps:

- Go to **Help** > **Install New Software...**
- Select the Xulrunner Plugin update site from the available sites or use https://www.servoy.com/developer/xulrunner if the Xulrunner Plugin update site is not available
- Follow the steps of the installation wizard to install the plugin

- After the plugin is installed and before Servoy Developer is restarted the servoy.ini file in the `{servoyInstall}/developer` directory needs to be updated. Add the following line to the end of this file:

```
-Dorg.eclipse.swt.browser.DefaultType=mozilla
```

After restart of Servoy Developer the XulRunner browser component will be used as browser component inside Servoy Developer.

Running Servoy Developer on Java 7 on MAC OSX

If running Servoy Developer on OSX and there is a need to run Java 7 (for example for JavaFX integration), some extra steps are required, as due to a bug in Eclipse (on which Servoy Developer is based), the process of running Servoy Developer on Java 7 is not as straight forward.

To setup Servoy Developer on OSX to work with Java 7 (assuming Java 7 is already installed) the following steps need to be performed:

1. Navigate to `/path/to/servoy/developer/servoy.app/Contents/` and edit the `Info.plist` file:
    a. go to the part where it says something like 'to use a specific Java version (instead of the platform's default) uncomment one of the following options:'
    b. here, add a line as (with the path to the Java 1.7 install):

```
<string>-vm</string><string>/Library/Java/JavaVirtualMachines/jdk1.7.0_25.jdk/Contents/Home/bin
/java</string>
```

2. Navigate to `/path/to/servoy/developer/servoy.app/Contents/MacOS/` and edit the `servoy.ini` file:
    a. above the '-vmargs' line, add two lines, as follows, but corresponding to the Java 1.7 install:

```
-vm
 /Library/Java/JavaVirtualMachines/jdk1.7.0_25.jdk/Contents/Home/bin/java
```

3. In this same directory (`.../developer/servoy.app/Contents/MacOS/`) create a script called `servoy.sh`, which contains the following text:

```
#\!/bin/sh
\#
export JAVA_HOME=/Library/Java/JavaVirtualMachines/1.7.0.jdk/Contents/Home
LAUNCHER_JAR=../../../plugins/org.eclipse.equinox.launcher_1.2.0.v20110502.jar
LAUNCHER_LIB=../../../plugins/org.eclipse.equinox.launcher.cocoa.macosx.x86_64_1.1.101.v20120109-1504
java \
\-showversion \
\-XX:MaxPermSize=256m \
\-Xms40m \
\-Xmx512m \
\-Xdock:icon=../Resources/servoy.icns \
\-XX:+UseParallelGC \
\-XstartOnFirstThread \
\-Dorg.eclipse.swt.internal.carbon.smallFonts \
\-Dnativeswing.interface.inprocess.useExternalSWTDisplay=true \
\-Dnativeswing.dependencies.checkVersions=false \
\-Dsun.awt.disableMixing=true \
\-Dosgi.requiredJavaVersion=1.6 \
\-jar $LAUNCHER_JAR \
\--launcher.library $LAUNCHER_LIB
```

    Make sure the script can be executed (maybe execute a `chmod a+x servoy.sh`)
4. Make sure the `JAVA_HOME` points to the Java 1.7 install
5. Launch Servoy using the created `servoy.sh` script

To double check that Servoy Developer actually runs with 1.7, in the Servoy Developer go to **Help** > **About Servoy Developer** > **Installation Details** > **Configuration**, and see `java.runtime.version` for example)

> **Note**
>
> The four steps above correspond to advice from https://www.servoy.com/forum/viewtopic.php?f=5&t=19253#p103941