Chapter 1

Installing Oracle Weblogic Server

In this chapter we will move our first steps with Oracle Weblogic Server by learning how to install it using the guided wizard installation or the compacted zip distribution. Once completed the installation, we will create our first Weblogic domain which will be used in the next chapters to deliver Java EE application and services.

Getting Oracle Weblogic Server

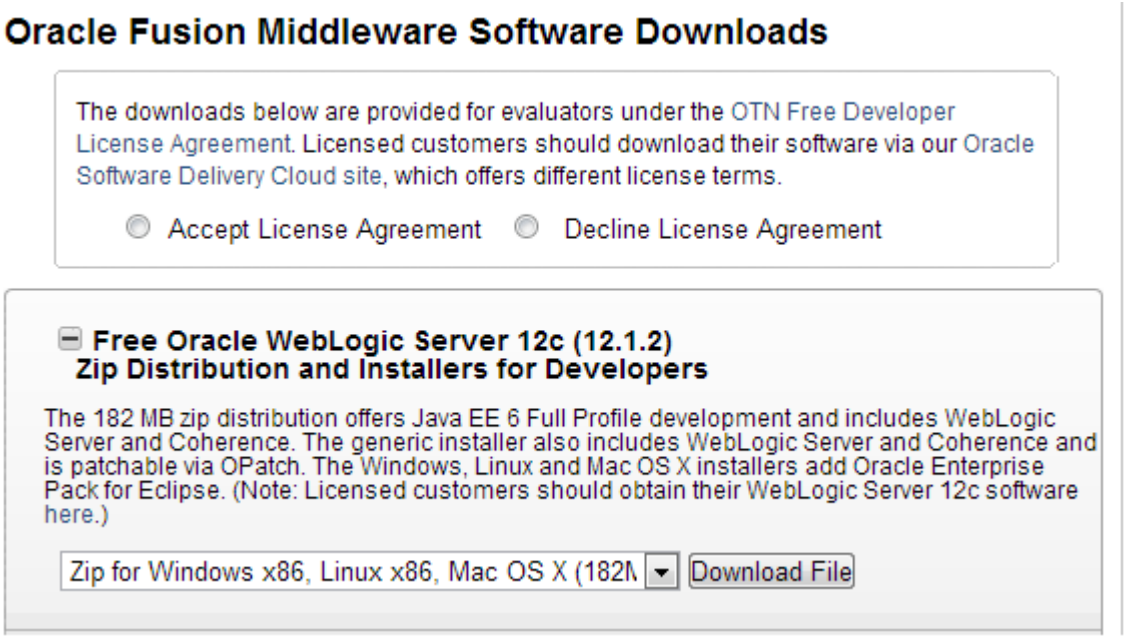
Oracle Weblogic products are available at http://www.oracle.com/technetwork/middleware/fusion-middleware/downloads/index.html
From the download area, you can access to the Oracle Fusion Middleware software.

These distributions are intended to use **for development only**. If you are a licensed customer you can obtain Oracle Weblogic server at Oracle Software Delivery cloud at (https://edelivery.oracle.com/).

As you can see from the Oracle network site, there are basically three types of distributions of Oracle WLS:

- **Full Installer 32 bit:** this distribution is platform specific and includes Oracle WLS, Oracle Coherence and Oracle Enterprise Pack for Eclipse. It is meant for installation with a 32bit JVM only.
- **Generic:** this distribution is platform independent and can be used to install Oracle WLS on any supported 32bit or 64bit platform. It ships as a generic archive and includes Oracle WLS and Oracle Coherence.
- **Zip distribution:** this distribution is also platform independent and contains barely Oracle WLS to be extracted from a compressed archive.

And here's a snapshot from Oracle portal where you are going to download Oracle WLS:



You need at first to accept the license agreement in order to proceed. Next, you will be prompted to enter Oracle Single Sign-on user name and password (former registration required). Once done with login, download will start.

Installing Oracle WLS using the Platform Installer

The simplest way to install the whole platform bundle is the Full Installer which can be carried out with a simple and intuitive wizard. Once downloaded the installer execute (Windows):

```
oepe-12.1.2.1-kepler-installer-win32.exe
```

On Linux:

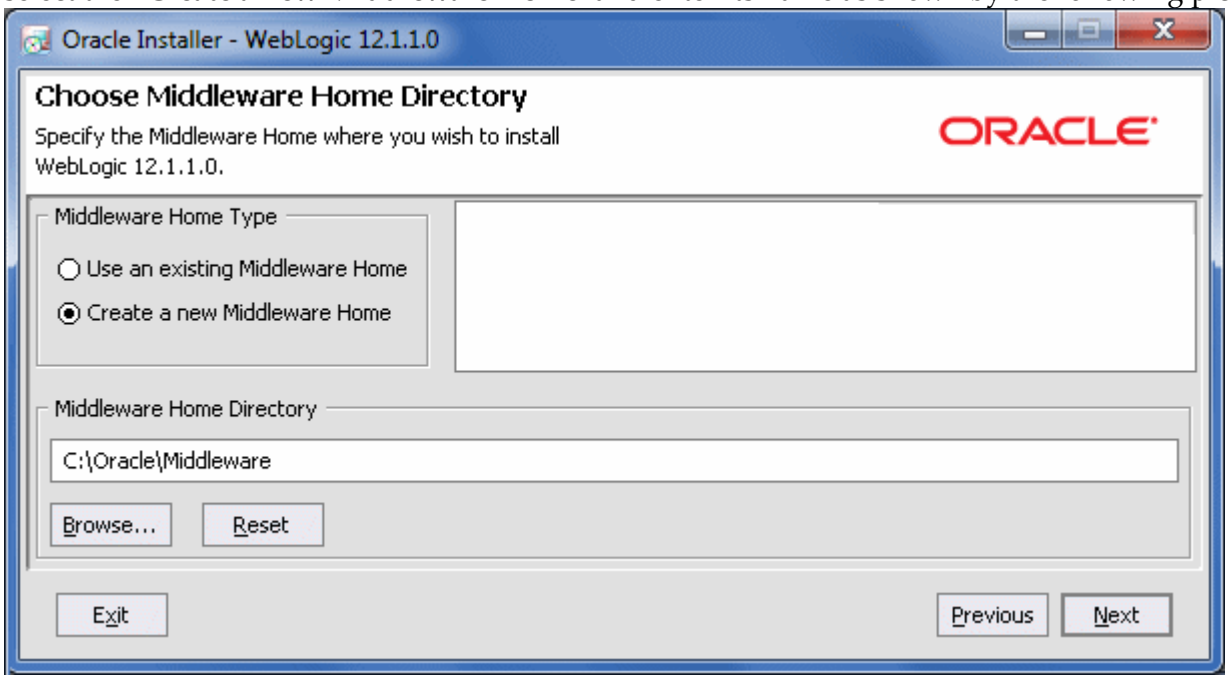
```
chmod a+x oepe-12.1.2.1-kepler-installer-linux-gtk.bin
./oepe-12.1.2.1-kepler-installer-linux-gtk.bin
```

The install wizard will start and you will be asked to complete the following steps:

1. Choose Middleware Home Directory

You need at first to specify the **Middleware Home** directory that will serve as the central support directory for all Fusion Middleware products installed on the target system, including WebLogic Server.
If you already have a Middleware Home directory on your system, it is recommended that you select the "**Use an existing Middleware Home**" and then browse that directory. Otherwise, simply

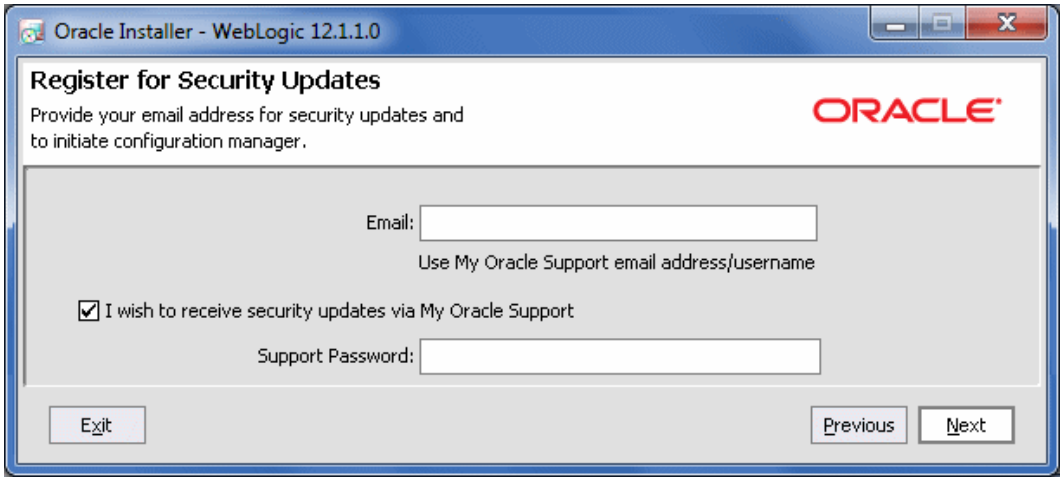
select the "**Create a new Middleware Home**"and enter its name as shown by the following picture:



Click Next to continue.

2. Register for Security Updates

Specify whether you want to register the product installation with My Oracle Support. By registering, Oracle Support notifies you immediately of any security updates that are specific to your installation.

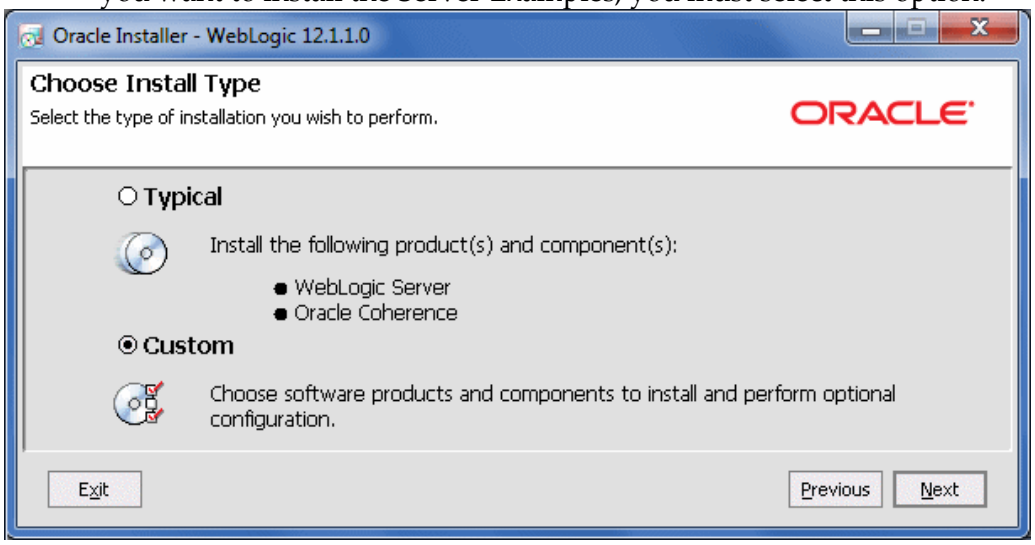


Click Next to continue.

3. Installing Oracle WLS

Now the server installation will start. There are two types of installation you can perform:

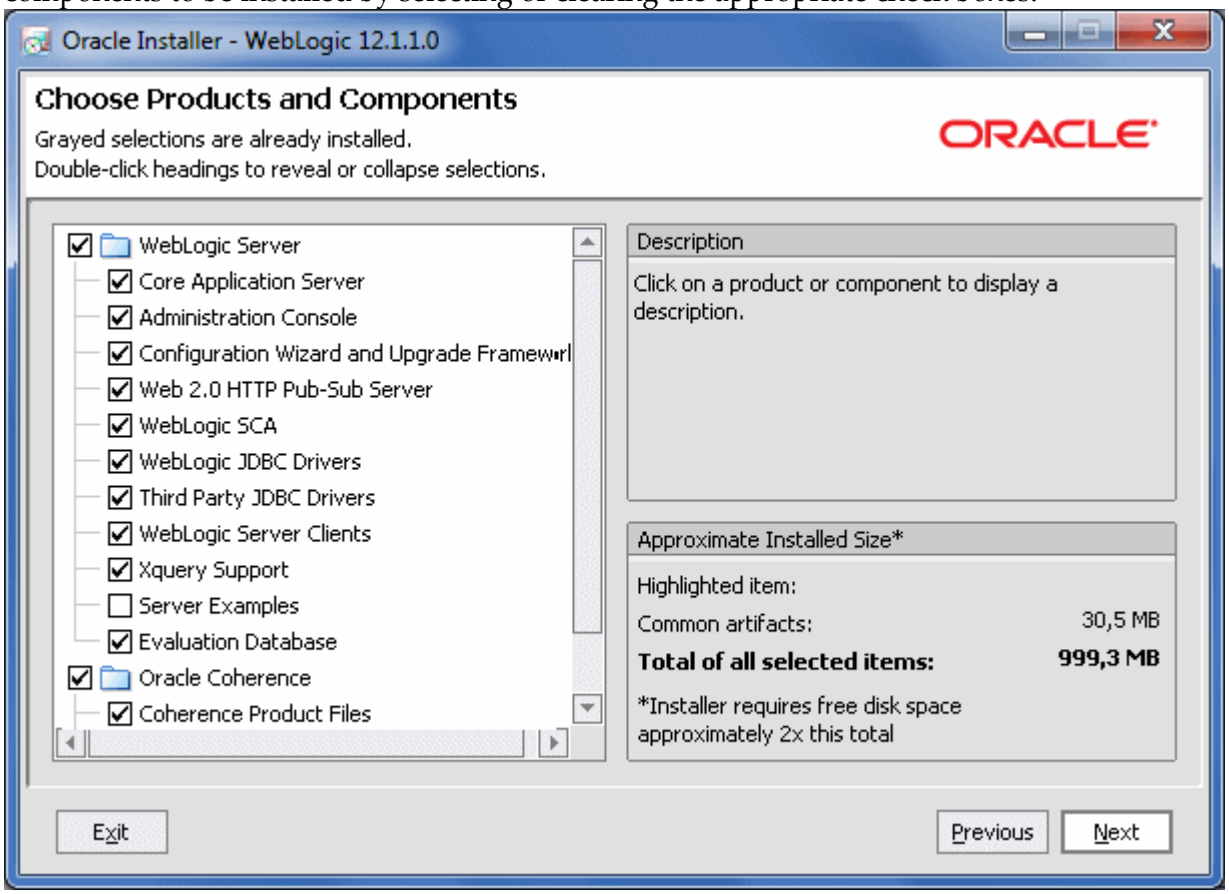
- **Typical:** All the software components included in your distribution are installed on your system. Typical installation does not include the Server Examples.
- **Custom:** You select the software components to be installed. On Windows systems, you also have the option to install Oracle WebLogic Server Node Manager as a Windows service. If you want to install the Server Examples, you must select this option.



Click Next to continue.

4. Choose products and Components

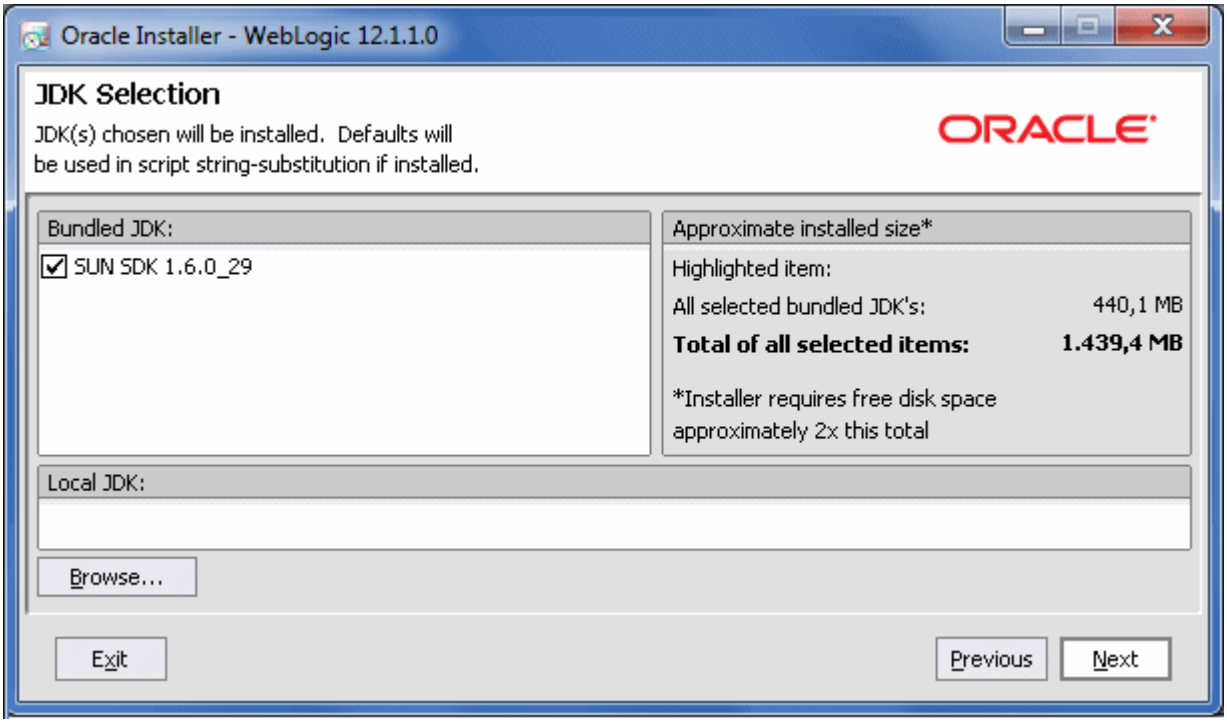
This screen displays a tree view of all the components available for installation. Specify the components to be installed by selecting or clearing the appropriate check boxes.



The suggested product installation requires that you install at least the Core Application Server, the Administration Console and Weblogic JDBC Drivers. Click Next to continue.

5. JDK Selection

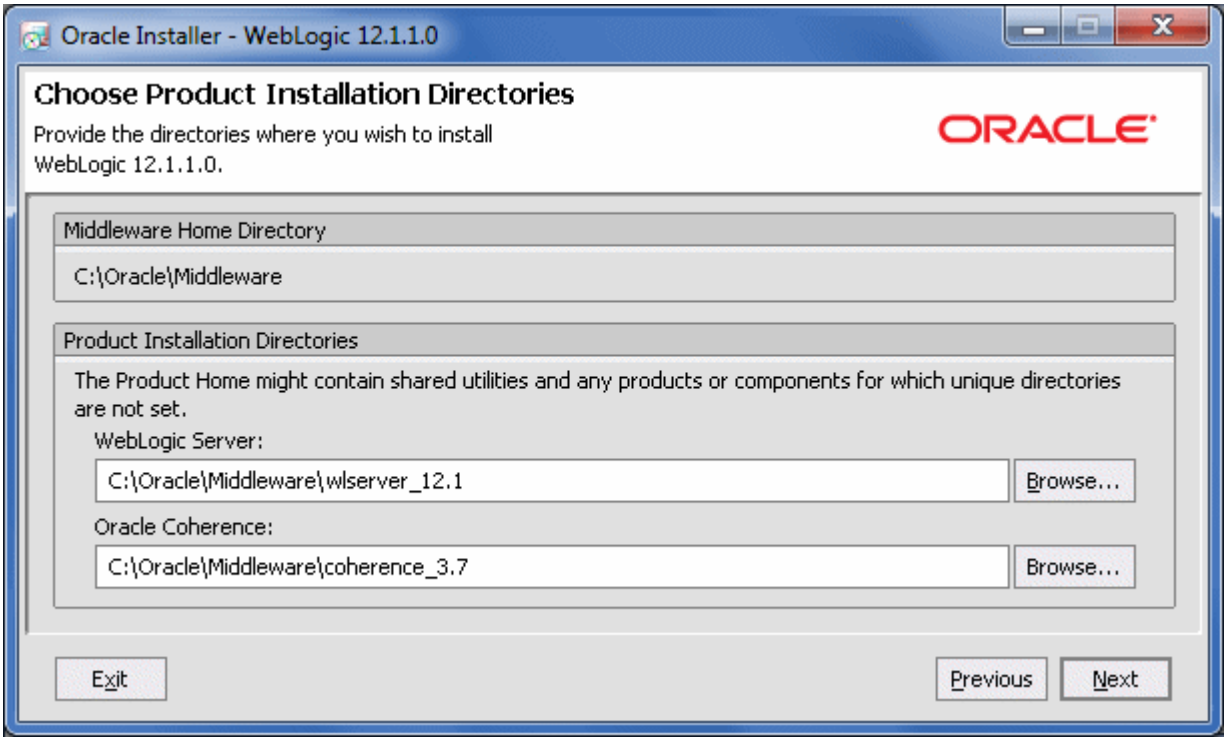
This screen displays a list of JDKs. Select the JDK that you want to install with the product. You can also browse and select a local JDK (at least 1.6.0_05) installed on your machine.



Click Next to continue.

6. Choose product installation directories

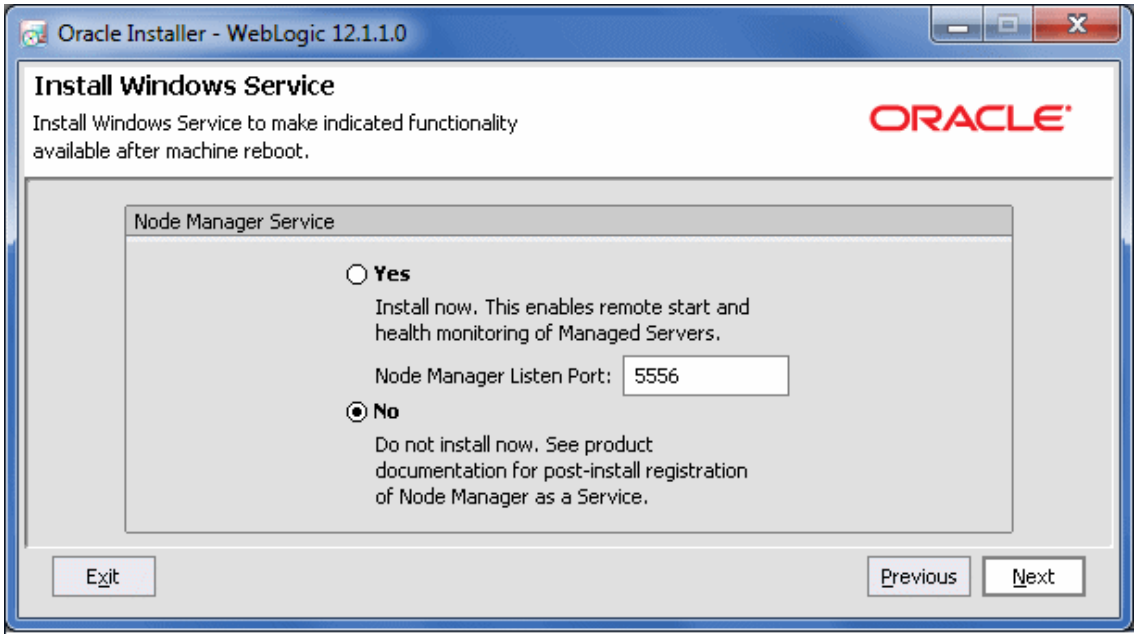
Specify the directories in which you want to install the products (WebLogic Server and Coherence, if applicable).



Click Next to continue and confirm your product installation directories in the next screen.

7. Install WLS as Service

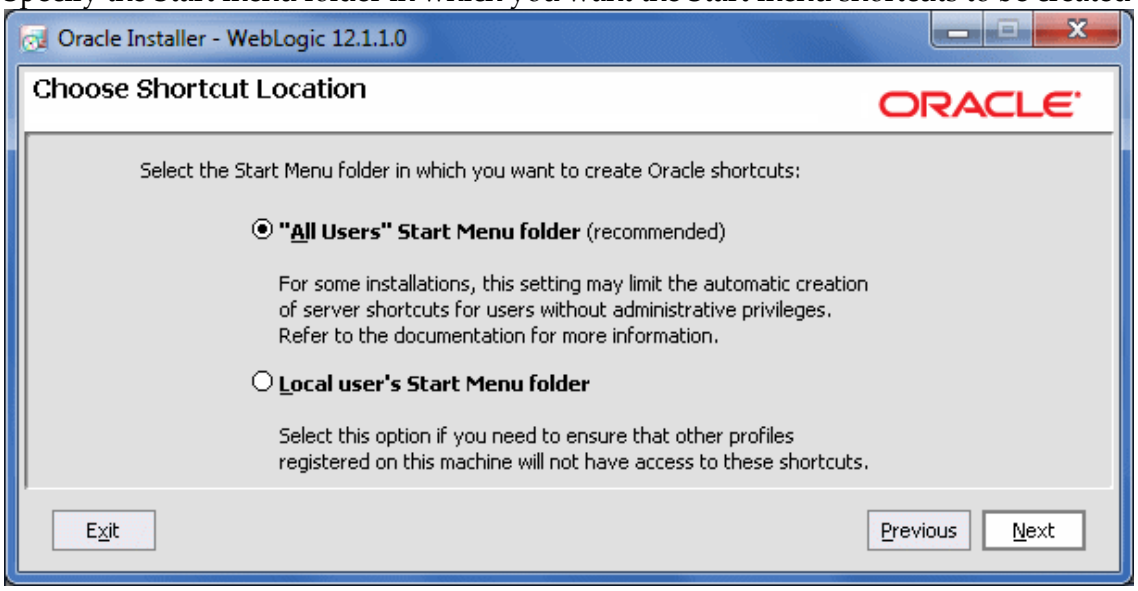
Choose whether you want to install Oracle WLS as service. For example, if you are running the Installer on Windows, in this screen you will be able to configure the WLS Node Manager service as System boot service. The Node Manager is used to monitor, start and stop server instances in a WebLogic domain. If you choose to defer the installation of the Node Manager as a service, you will need to install it manually (See Chapter 2 "*Starting the Node Manager*" for more info about it).



Click Next to continue.

8. Choose Shortcut Location

Specify the Start menu folder in which you want the Start menu shortcuts to be created



When the installation program has finished copying the specified files to your system, click Next to continue. Once completed, you can move to the next step, that is creating a new WLS domain and start the application server (See recipe "Creating a Weblogic Domain" contained in this chapter).

Running the Installer Program in Console Mode

If you got as little as a terminal you can still execute the Platform Installer in **Console Mode**. To start the installation program in Console Mode, open a terminal window and go to the directory that contains the installation program.
Launch the installation by entering the name of the installation program passing **-mode=console**. For example, to start the WebLogic Server Package installer for Windows in console mode, enter:

```
oepe-12.1.2.1-kepler-installer-win32.exe -mode=console
```

On Linux:

```
chmod a+x oepe-12.1.2.1-kepler-installer-linux-gtk.bin
./oepe-12.1.2.1-kepler-installer-linux-gtk.bin -mode=console
```

Where *file_name.bin* is the name of your installation program

Installing the Generic Installer distribution

The generic installer distribution can be used to install Oracle WLS on any supported 32 or 64 bit platform. The installation process is essentially the same as the full installer but you need to be aware of a few things before getting started:

- This distribution does not include a JDK therefore it's your duty to check that you have a correctly installed JDK. You should set as well the JAVA_HOME pointing to the folder where Java is installed and update the System PATH accordingly.
- This distribution can be installed by using a terminal Window.

Here's how to proceed with the installation:
Linux users:

```
$ export JAVA_HOME=/home/myhome/myjavahome
$ export PATH=$JAVA_HOME/bin:$PATH
```

Windows users:

```
set JAVA_HOME=yourJavaInstallPath
set PATH=%JAVA_HOME%/bin;%PATH%
```

Now you can move on with the installation by executing:

```
java –jar wls_121200.jar
```

Once uncompressed the archive, you can follow the same steps depicted in the Full Installer distribution.

Installing the WLS ZIP distribution

Installing Oracle WLS using the zip distribution is not carried out using an Installer Wizard; however, if you have no prejudices about using operating system's shells, this installation can be a real time saver (and disk space saver as well!). As a matter of fact, this distribution is supported on Windows, Linux and Mac OS X systems and contains the necessary artifacts required to setup a basic installation of Oracle WebLogic Server.

An optional supplemental zip (*wls1212_dev_supplemental.zip*) is available as a separate download. The supplemental zip contains samples, evaluation database (Derby) and L10N console help files.

Here are the steps to perform installation of the Zip distribution:

1. Unzip the file

Extract the contents of the zip to a directory. This directory is referred to as MW_HOME (eg: */home/myhome/mywls*).

2. Setup OS variables

Setup **JAVA_HOME**, **MW_HOME** and **JAVA_VENDOR** (Windows users only) variables in the current shell as required for the target platform.
Linux:

```
$ export JAVA_HOME=/home/myhome/myjavahome
$ export MW_HOME=/home/myhome/mywls
```

Windows:

```
set JAVA_HOME=C:\home\myhome\myjavahome
set MW_HOME=C:\home\myhome\mywls
set JAVA_VENDOR=[Sun|Oracle]
```

3. Run the installation script

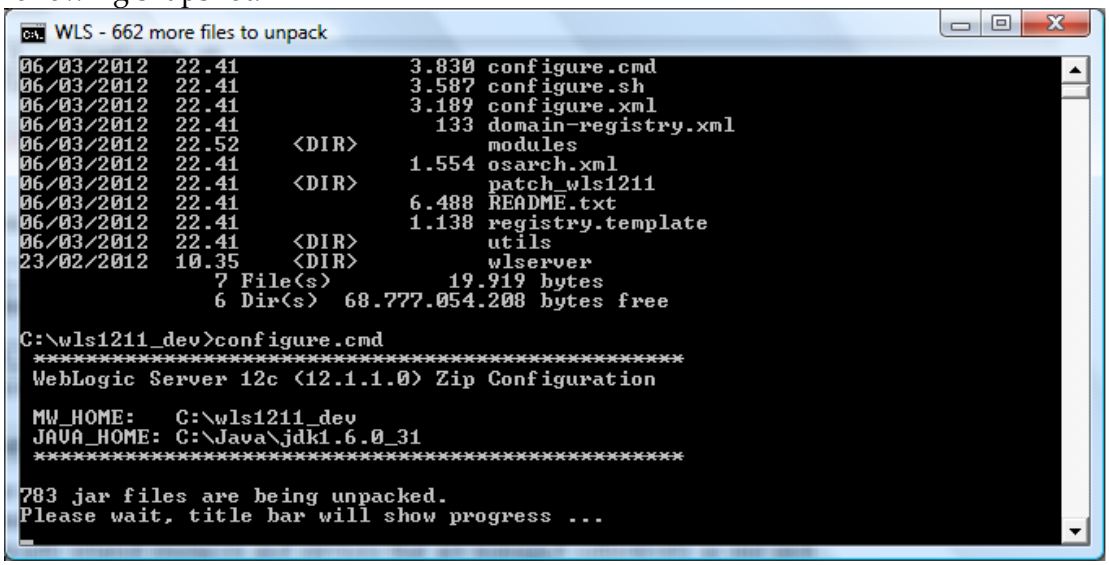
Launch the installation configuration script in the *MW_HOME*. This step is required to be run only once. If you move the installation to another location/machine, you need to rerun this step. On a Linux machine execute:

```
$ . ./configure.sh
```

On a Windows platform execute instead:

```
  configure.cmd
```

Now the configure command will start unpacking the archived distribution as you can see from the following snapshot:



It might take a while to complete but then you are done with the application server installation.

4. Set the WLS environment variables

Now execute the *setWLSEnv* script which is located in the *bin* folder of your server distribution. This will set a proper environment configuration for executing Oracle WLS shell commands:
Linux

```
$ . $MW_HOME/wlserver/server/bin/setWLSEnv.sh
```

Windows

```
%MW_HOME%\wlserver\server\bin\setWLSEnv.cmd
```

Once completed, you can move to the next step, that is creating a new WLS domain and start WLS.

Creating a Weblogic domain

The first thing we will learn is how to create a WLS domain.

A **Weblogic domain** is the basic administrative unit of WebLogic Server. It consists of one or more WebLogic Server instances and logically related resources and services that are managed collectively as one unit.

The **Administration Server** provides a central point for managing the domain and providing access to the WebLogic Server administration tools. A domain primarily includes **Managed Servers** which host application components and resources that are managed as part of the domain.

A domain may also include Weblogic Server **Clusters**, which are groups of Managed Server instances that work together to provide scalability and high availability for applications. (Clustering is discussed in detail in Chapter 9 of this book).

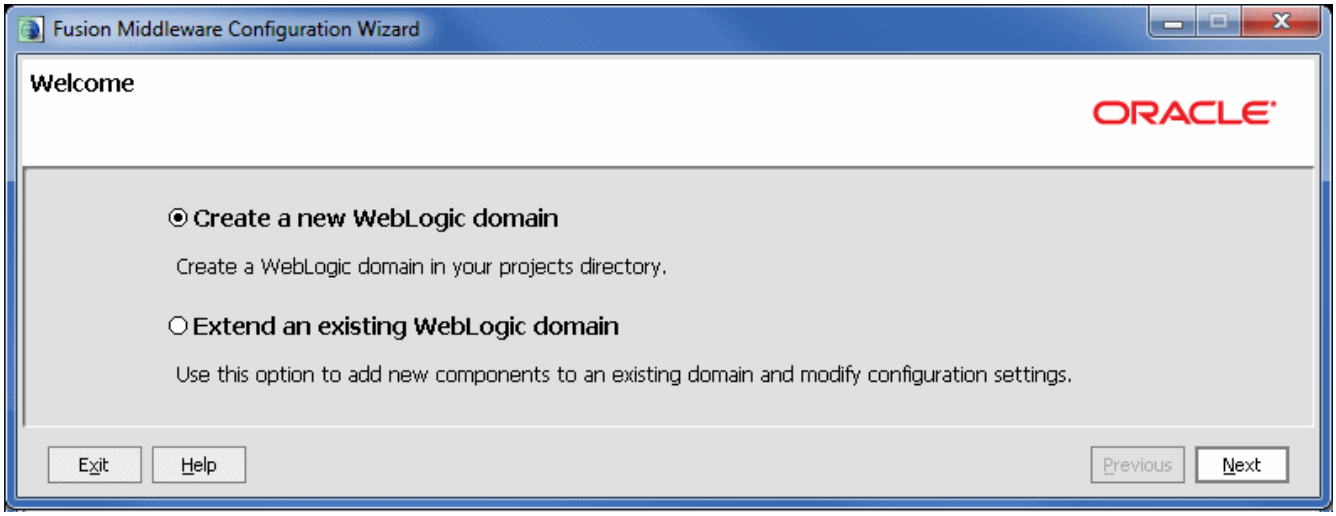
There are several ways to create a WebLogic domain: the recommended way is using the **Fusion Middleware Configuration Wizard** to create a domain based on or more domain templates.
The first step to take is to run the domain configuration script which can be found in the *$MW_HOME/wlserver/common/bin*
On Linux:

```
$ config.sh
```

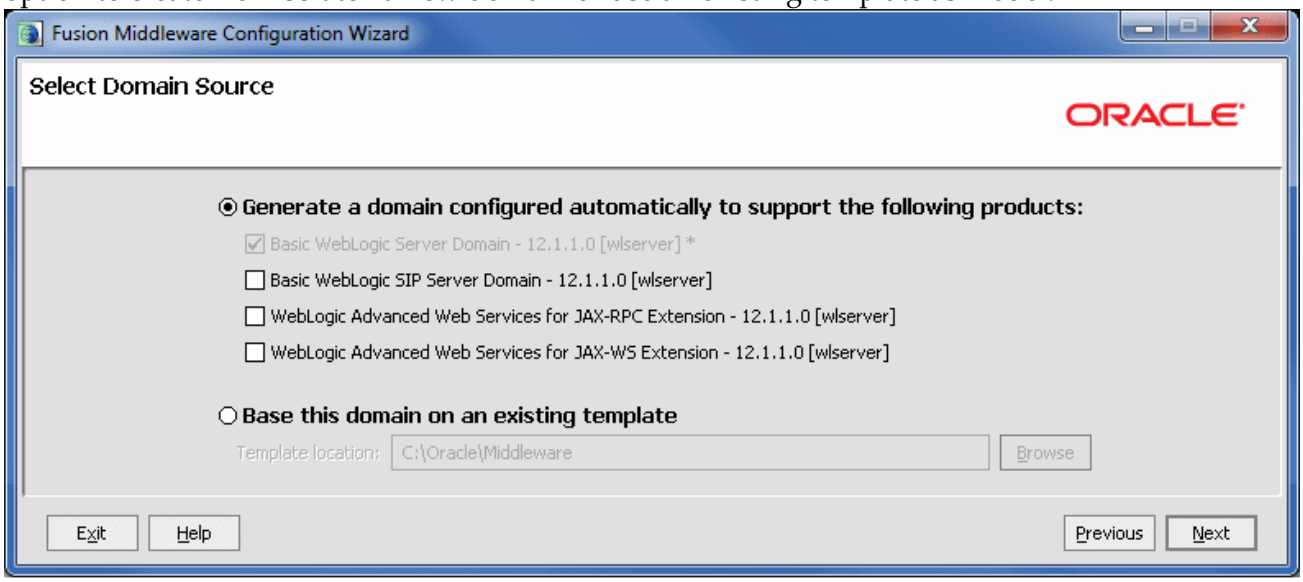
On Windows:

```
config.cmd
```

After executing the configuration script, the following screen will be your first step to complete:

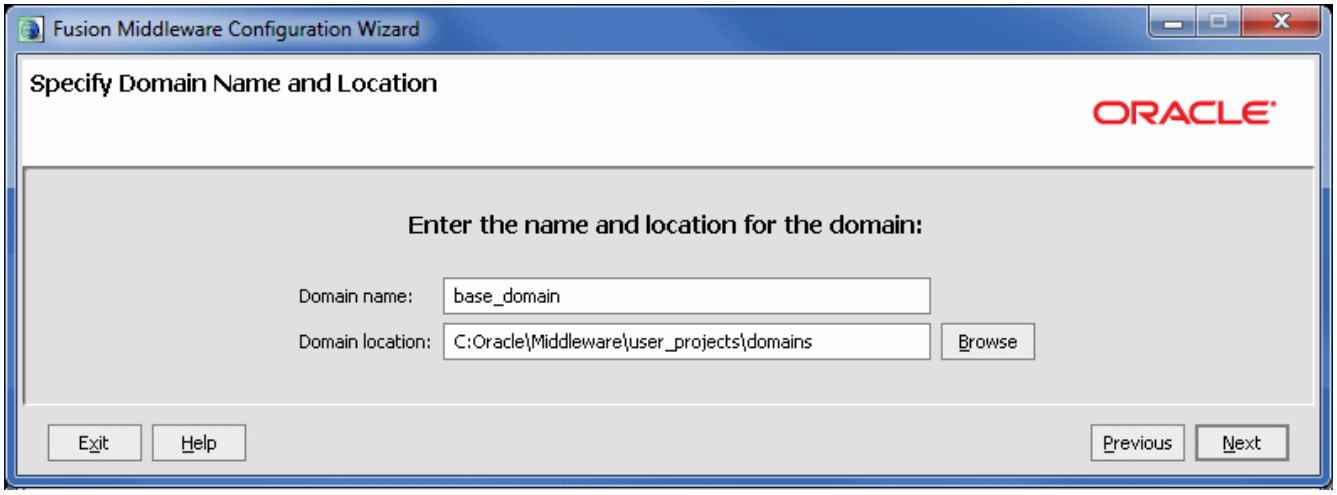


Select "**Create a new WebLogic domain**" and click **Next** to continue. In the next step you have the option to create from scratch a new domain or use an existing template as model.



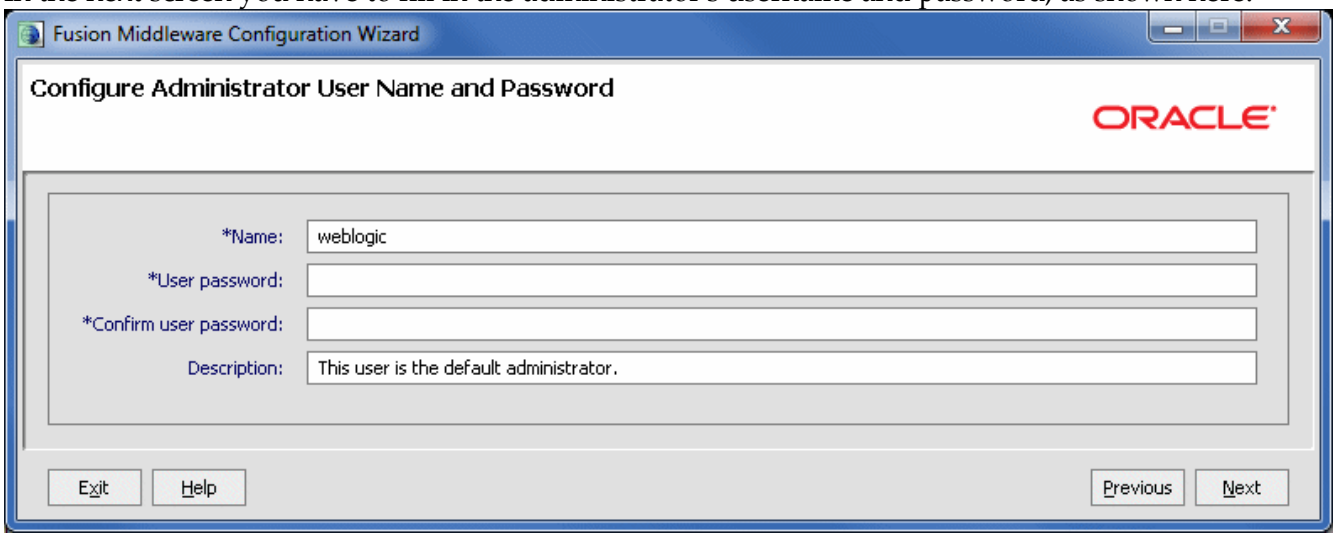
Actually also a brand new domain uses, behind the scenes, a template. (The base template, *wls.jar* that ships with your WebLogic software, is located in the folder *$MW_HOME/wlserver/common/templates/domains*).

Since we still have not created any template click Next to continue with defaults. In the next screen, you can fill in the name and location for your domain, as shown by the following screenshot:

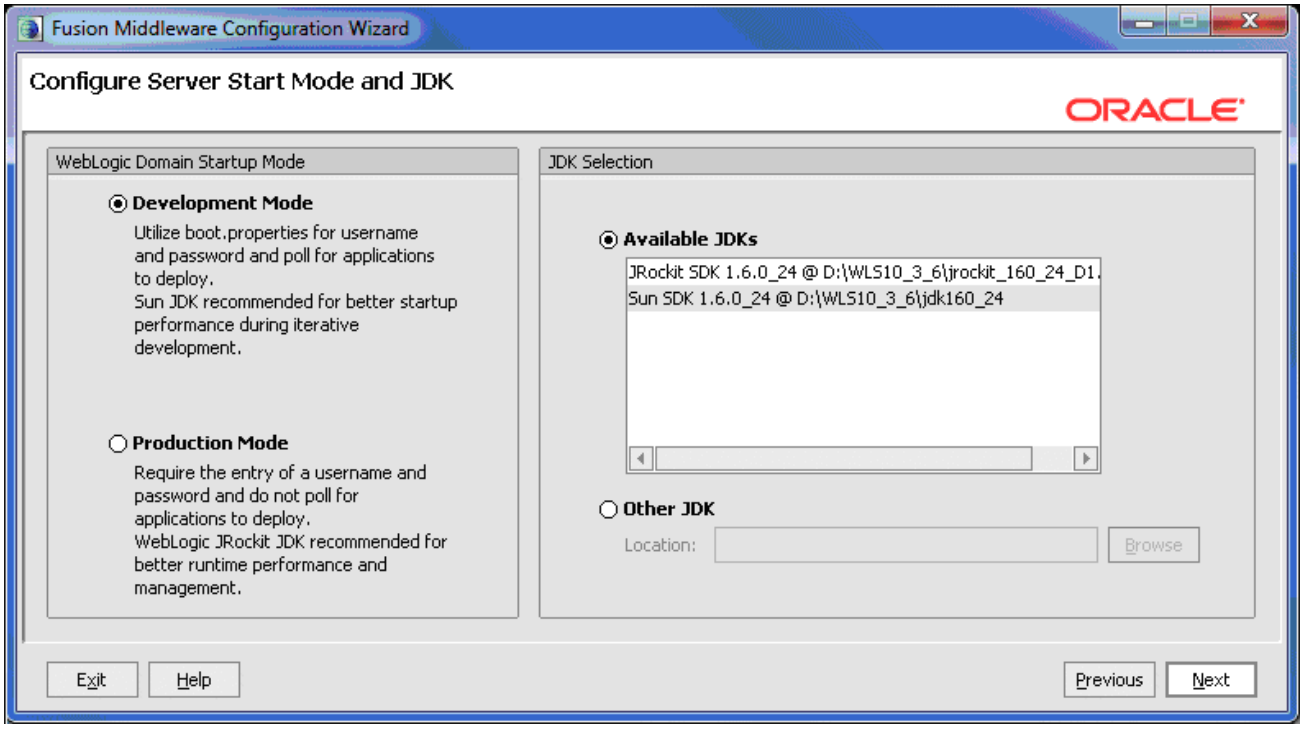


The default installation will suggest a *base_domain* folder under the Middleware home folder. For production environment it is recommended that you create domains outside the *MW_HOME*.

In the next screen you have to fill in the administrator's username and password, as shown here:



Click Next to continue. In the following screen you have to configure the **WLS Domain startup mode** and select a **JDK** among the available ones (Oracle JDK or Weblogic JRockit)



The choice between Development mode and Production mode appears obvious: you will select Development mode for faster application development and Production mode for higher performances in production.

Production Mode Vs Development Mode

Besides the evident different purpose of the two server modes there are also some important differences between them that you must be aware of:

Automatic deployment: When running Development mode the automatic deployment of an application is enabled. So it's enough to drop an application in the domain's autodeploy folder to get it deployed by Oracle WLS. In Production Mode, by default, this is switched off.

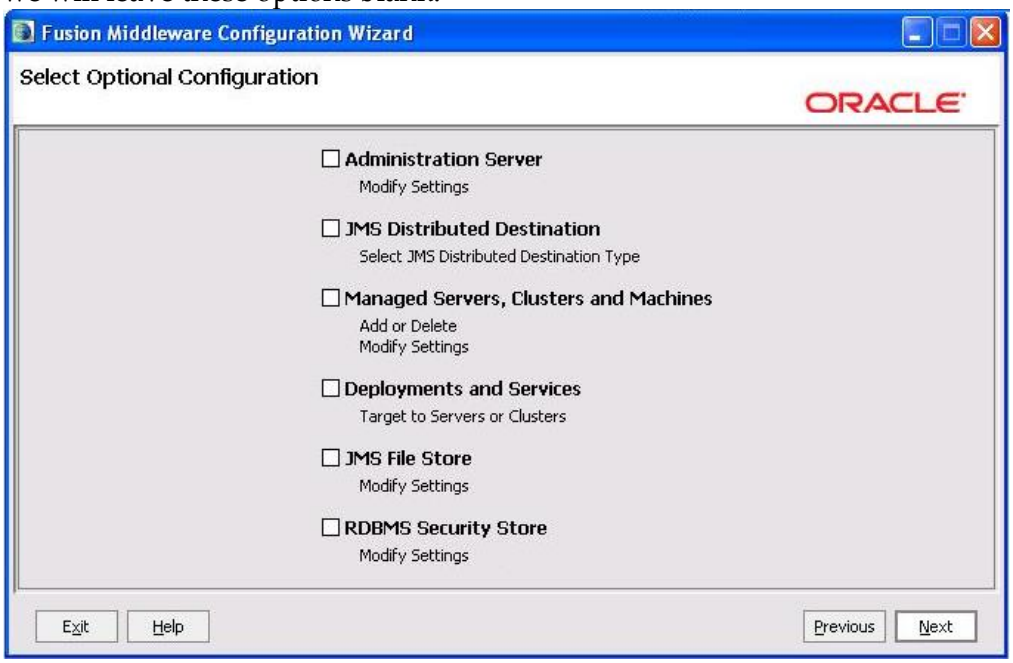
Stricter security policy: In Development mode there is a lower level of security: a file called *boot.properties*, which contains the encrypted administrator username and password, is automatically generated. This does not happen in Production Mode, which uses a stricter level of security.

Debugging: In Development the debugFlag which is used to start the WebLogic Workshop Debugger is enabled, while in Production mode is disabled.

In order to enable/disable the Production mode you can set the variable **PRODUCTION_MODE** in the script ***setDomainEnv.sh/ setDomainEnv.cmd***. For example, on the Windows script:

```
set PRODUCTION_MODE=true.
```

In the next step you can optionally define the configuration of servers and Java EE resources for your servers. Since we will use next chapters' recipes to configure every aspect of the configuration we will leave these options blank:



The final screen will recap the selected configuration options. Click on Create to complete Domain installation.

Creating a domain using the Unix shell

On Unix-like systems, there is usually no graphical environment available. You can still run the domain creation GUI by exporting the GUI to your local desktop with an XClient, such as XVNC; however, if you don't mind using the terminal, just give the option mode=console and you will get a text-based representation of your GUI installation.

```
./config.sh mode=console
```

Creating a domain template

Creating a new domain is a time consuming task which can be automated by creating a domain template.

The term template refers to a **Java Archive (JAR)** file that contains the files and scripts required to create or extend a domain.

There are basically two types of templates that you can create:

- **Domain template**: defines the full set of resources within a domain including infrastructure components, applications, services, security options, general environment and operating system options.
- **Extension template**: defines the applications and services that you can add to an existing domain, including product component functionality and resources such as JDBC or JMS.

In the following section we will show how to create a Domain Template

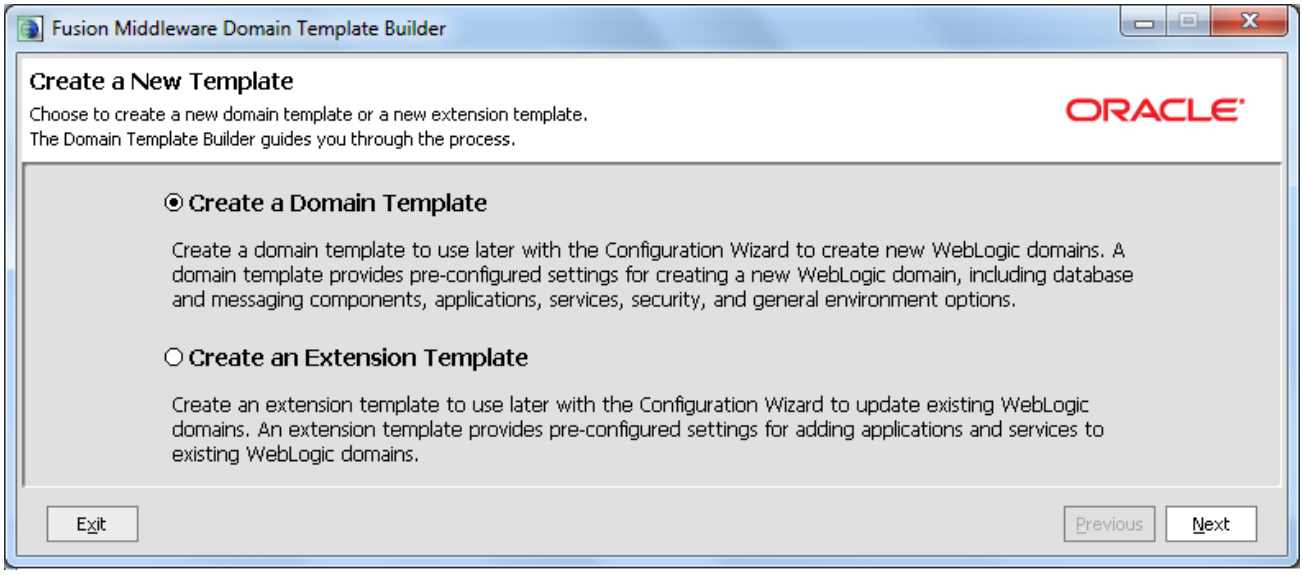
Creating a Domain Template

To create a new Oracle WebLogic Server domain template you can use the **Domain Template Builder**. Follow these steps in order to launch it:

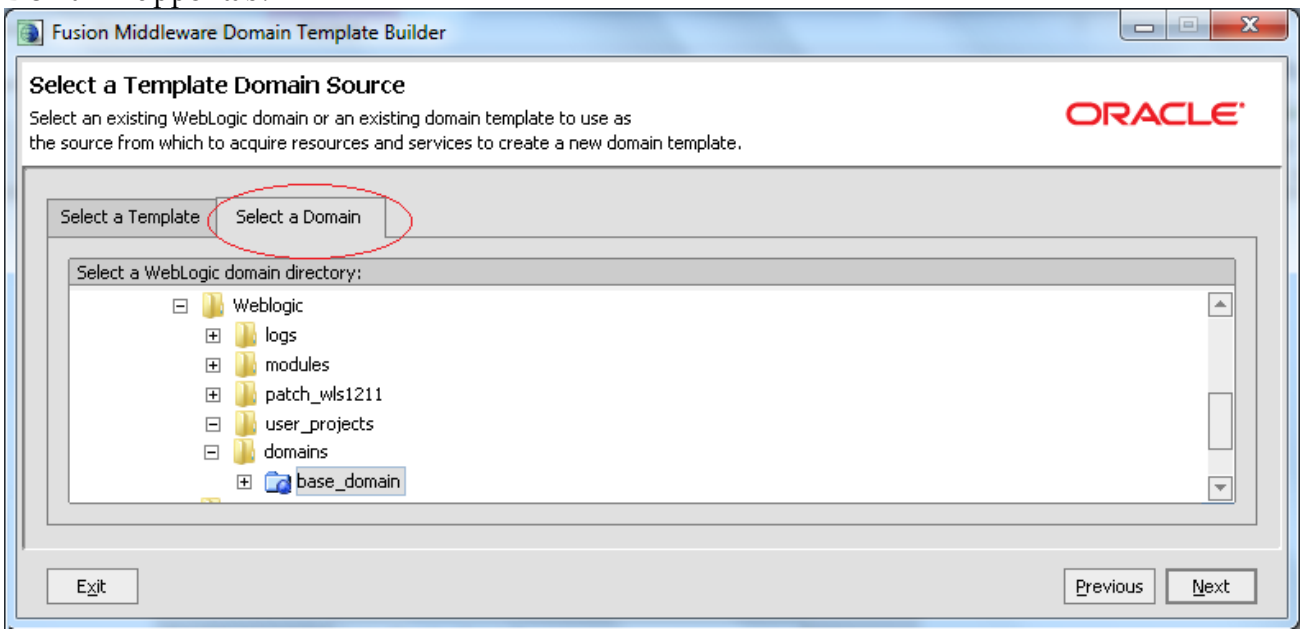
1. From the terminal window, navigate to the location of the **Domain Template Builder tool**. It is found under *<MIDDLEWARE_HOME>/wlserver/common/bin*.
2. Run the tool by entering the following command (config_builder.bat for Windows users):

```
$ ./config_builder.sh
```

The Domain Template wizard will start. In the "Create a New Template" window of the builder, select "**Create a Domain Template"** as shown by the following snapshot:



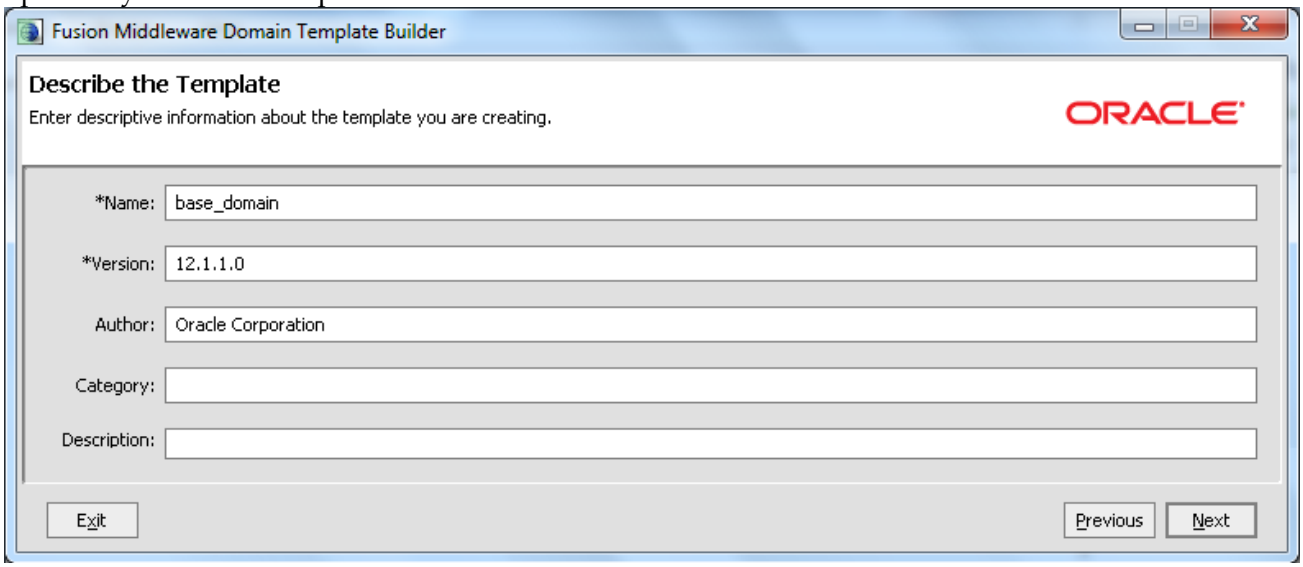
Click Next to continue. In the "Select a Template Domain Source" window, activate the "**Select a Domain**" upper tab.



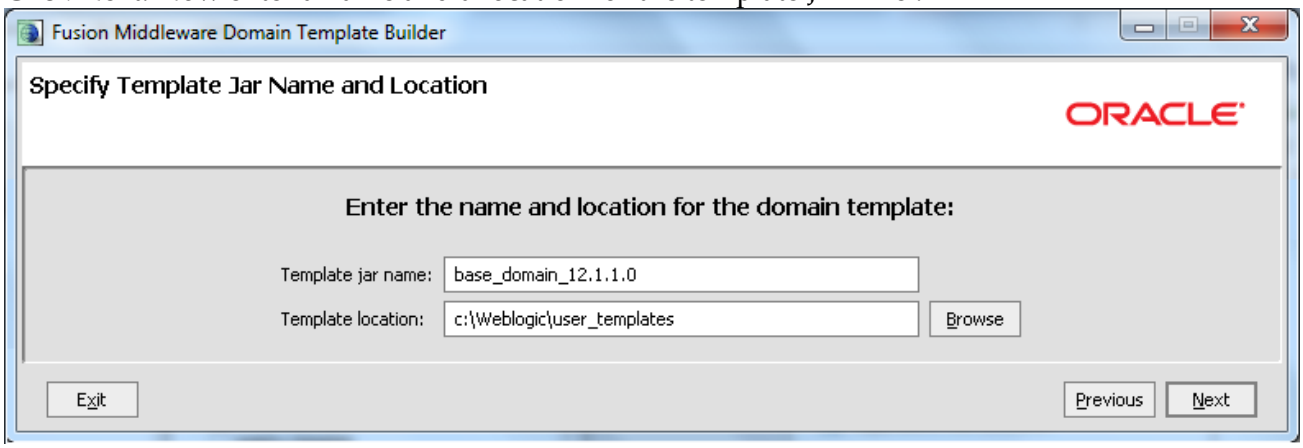
Then, navigate to the domain folder that you want to create a template from. In the above screenshot we are creating a template from the **base_domain** domain that we have formerly created.

Note: By choosing the **Select a Template** tab instead, you can create a template by extending an existing one template which needs to be selected from your file system as well.

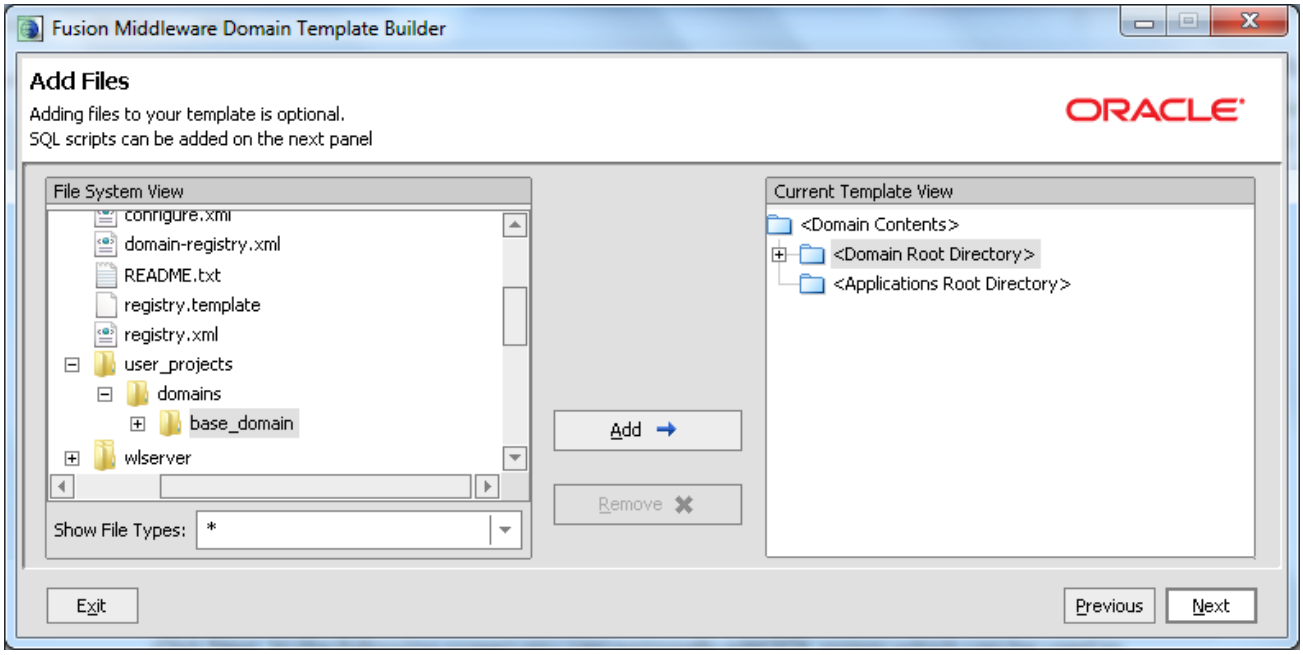
Once selected the domain, click Next. In the following screen, select the name for the domain and optionally enter a description for it:



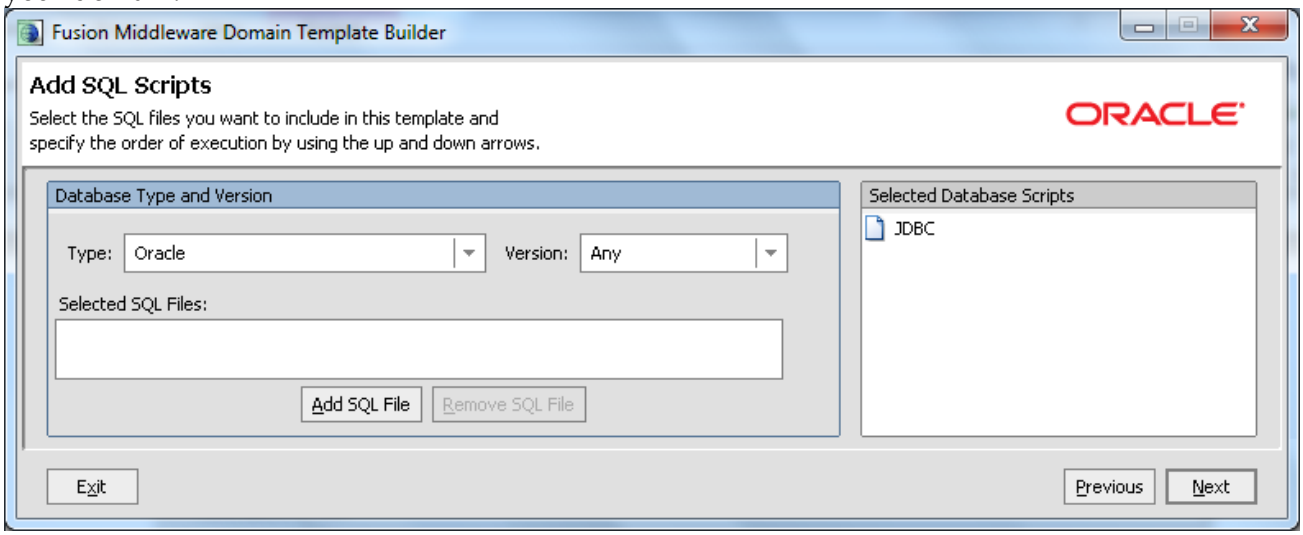
Click Next. Now enter a name and a location for the template JAR file :



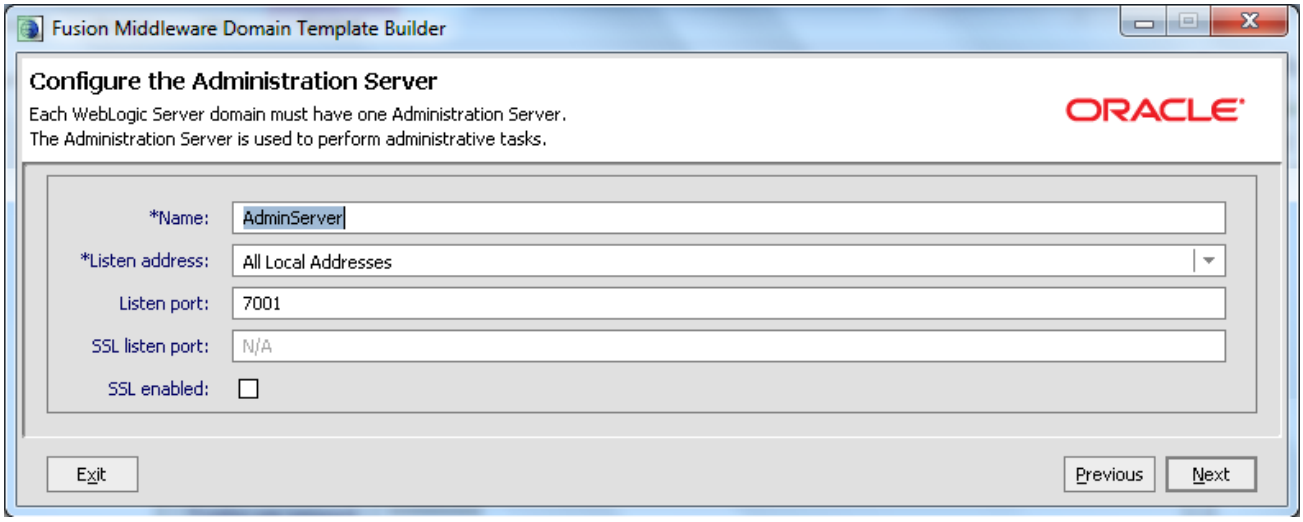
In the next screen you can customize your template by adding files which will be included in the domain when you are creating it from this template:



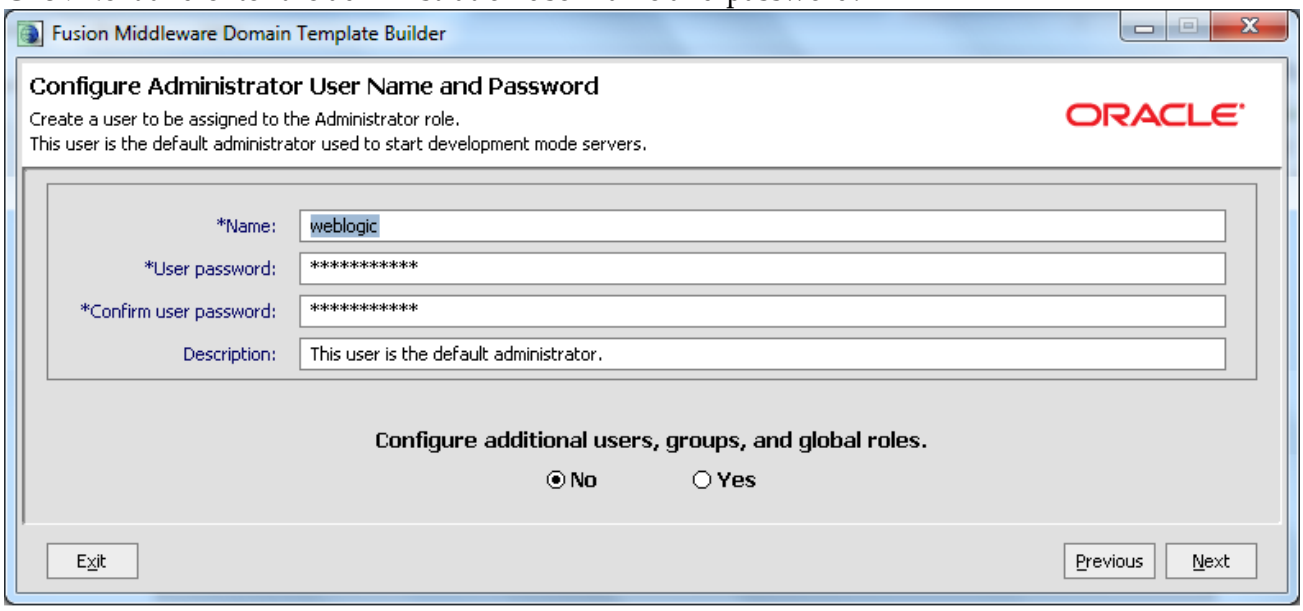
Click Next. In the next GUI you can optionally add SQL scripts which can be used to bootstrap your domain.



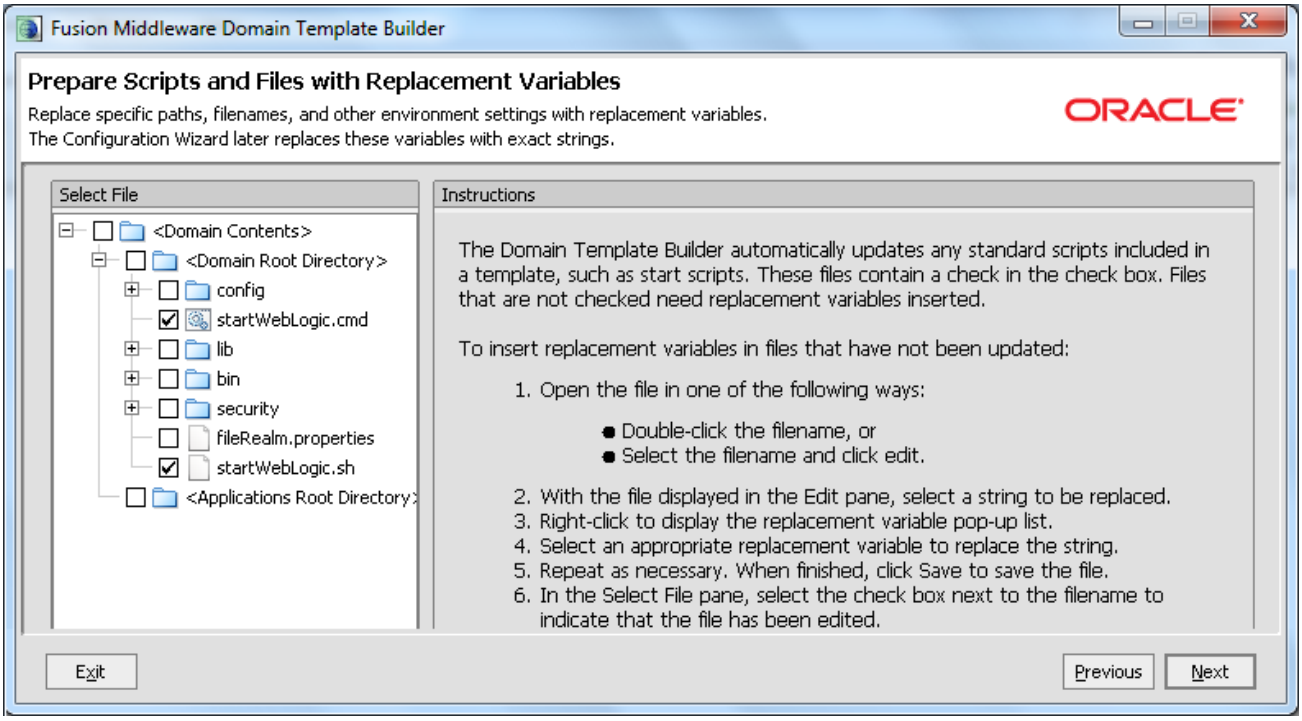
Click Next when done. In the following screen enter the template's built-in server name, its listen address and the listen ports:



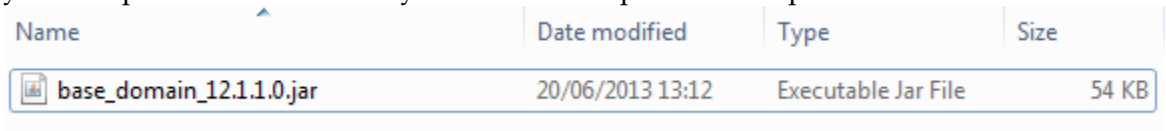
Click Next and enter the administration user name and password.



Click Next. In the following GUI you will be able to replace the standard scripts which are part of the template (e.g. start scripts) with your custom scripts.



Click Next and review your template settings before storing it as a JAR file. As last step, check that your template has been actually written in the specified Template Location folder.



Creating templates from the command line

Oracle WLS contains two utility scripts (named **pack** and **unpack**) which provide a simple one-step solution that can be used to create a template from the command line. These utilities are located in the *MW_HOME/wlserver/common/bin* folder of your distribution. Here's how to use the **pack** shell to create a domain named **mydomain** into a file named ***mydomain.jar***:

```
pack.sh -domain="/export/home/wlsuser/domains/mydomain" -
template="/export/home/wlsuser/templates/mydomain.jar" -template_name="WLSDOMAIN" -
log=/export/home/wlsuser/logs/wlscldompack.log -log_priority=INFO
```

The corresponding command used to expand your domain is **unpack**:

```
unpack.sh -template=mydomain.jar -domain=domains/mydomain -log=wlscldomunpack.log -
log_priority=INFO
```