

# Instalación y explotación del servidor de aplicaciones



**Carlos Álvarez Martín**  
SGBD, 2º ASI, I.E.S. Gonzalo Nazareno

# **Índice de contenidos:**

## **0. Introducción y definiciones previas**

- Oracle WebLogic
- Servidor de aplicaciones
- Dominios WebLogic

## **1. Instalación del servidor de aplicaciones Oracle**

## **2. Tipos de instalación del producto (componentes)**

- Componentes de WebLogic server
- Creación/Modificación de dominios en WebLogic

## **3. Utilidad y funciones básicas de la consola de administración**

- Preferencias de la consola de administración
- Servers
- Deployments
- Security realms
- Log files
- WLDF Console Extension

## **4. Despliegue de una aplicación web que acceda a una BD**

- Introducción
- Fuente de la aplicación
- Exportando proyecto
- Instalando en WebLogic

## **5. Bibliografía**

## **0.- Introducción y definiciones previas:**

*En esta guía tratará de explicarse cómo instalar el servidor de aplicaciones WebLogic de Oracle Corporation. Se realizará en el sistema operativo Windows XP. El SGBD Oracle estará instalado en la misma máquina para realizar la conexión con una BD cuando sea necesario.*

### Oracle WebLogic:

Es el servidor de aplicaciones de Oracle para JavaEE. Está disponible para las plataformas POSIX y Windows entre otras. El desarrollo original del producto corresponde a “BEA Systems” y posteriormente fue absorbido por “Oracle Corporation”.

En cuanto a la posibilidad de conexión con BBDD, soporta entre otras Oracle, MS SQL Server, etc. En definitiva, cualquier BD que sea compatible con el estándar JDBC.

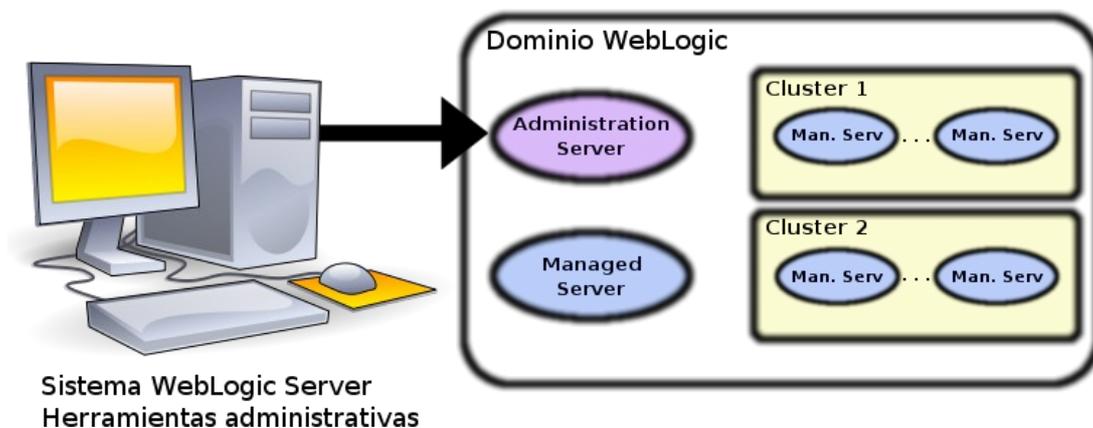
### Servidor de aplicaciones:

Se trata de un software que proporciona servicios de aplicación a máquinas cliente. Se ha extendido la idea de relacionar los servidores de aplicaciones con la plataforma JavaEE debido al peso que tiene en el mercado dicha plataforma.

Dos de los principales beneficios de los servidores de aplicaciones son la centralización del servicio y la disminución de complejidad a la hora de desarrollar aplicaciones.

### Dominios WebLogic:

Un dominio es la unidad básica de administración de un servidor WebLogic. Está formado como mínimo por una instancia, el servidor de administración ('Administration server'). Se pueden encontrar dominios WebLogic complejos en los que existen varias instancias, por ejemplo: un 'Administration server', servidores 'Managed' y clústeres de esos servidores para mejorar la escalabilidad y la disponibilidad de los servicios ofrecidos. Este último ejemplo se puede ver con más claridad con la siguiente figura:



## **1.- Instalar el servidor de aplicaciones de ORACLE:**

Para instalar cualquier producto de Oracle acudimos a su página de descargas en la que tendremos que loguearnos con una cuenta creada previamente a la hora de descargar cualquier producto. Buscamos el producto WebLogic y encontraremos diversos ficheros ([enlace](#)).

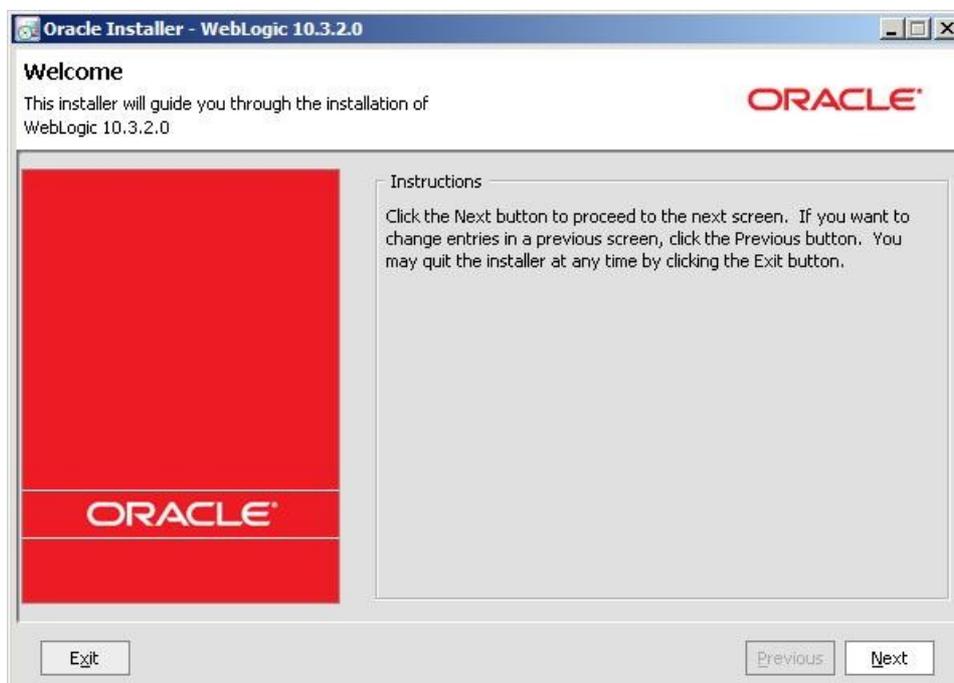
WebLogic forma parte de “Oracle Fusion Middleware” (o “Capa intermedia de Oracle”), que se trata de un conjunto de aplicaciones/utilidades con los que se pueden desarrollar y desplegar aplicaciones sofisticadas. WebLogic es el servidor de aplicaciones, pero existen más herramientas que se pueden acoplar a él, como autenticación centralizada (“Identity Management”), base de datos de Oracle, “WebTier”, etc.

En este caso nos centraremos en descargar el servidor de aplicaciones, existiendo un par de opciones por S.O.

- Oracle WebLogic Server + OEPE - Package Installer (~815MiB)
- Oracle WebLogic Server + OEPE - Net Installer (~39MiB)

La diferencia existente entre ambas descargas, es que la primera de ellas contiene todos los componentes del instalador WebLogic integrados y en cambio el NetInstaller no. Si queremos hacer una instalación personalizada y/o con pocos componentes sería conveniente elegir la instalación por internet.

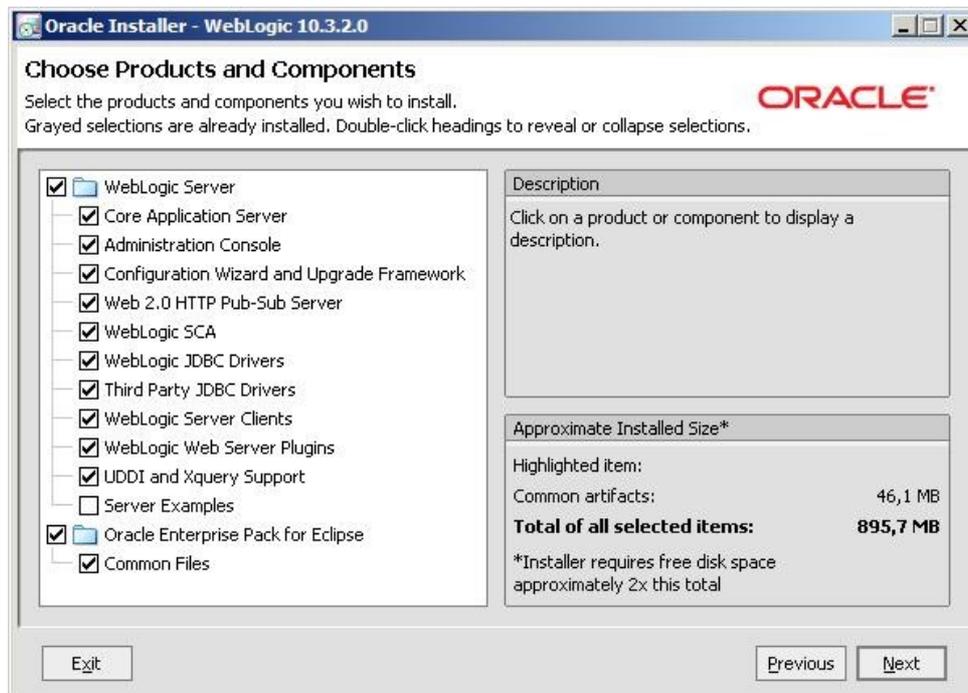
En nuestro caso usaremos la segunda opción, la instalación con “Net Installer”. A continuación adjuntaré las capturas más relevantes de la instalación y una explicación de cada una de ellas.



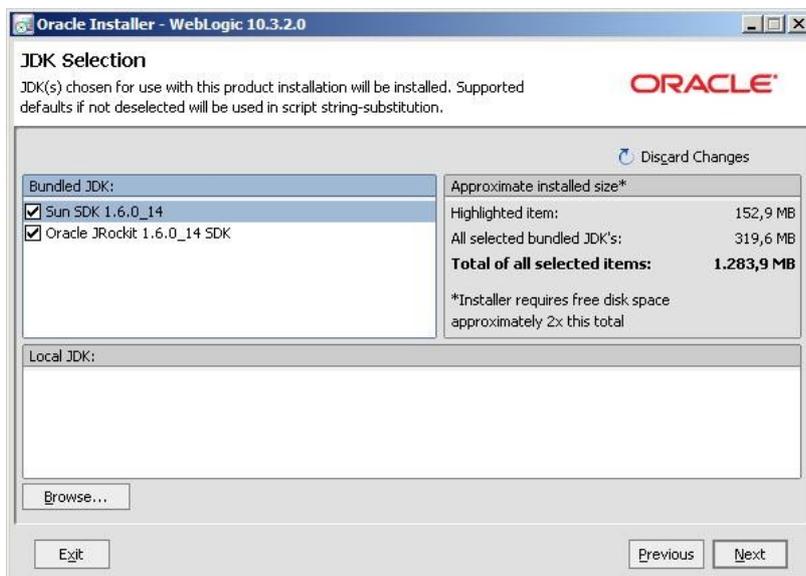
Ésta es la pantalla inicial del producto. Las siguientes pantallas no tienen ninguna complicación en especial. Se elige el 'HOME' para las aplicaciones de 'Middleware' y se especifica

el e-mail a Oracle para que nos notifiquen de actualizaciones de seguridad (opcional). Además, seleccionaremos el directorio temporal en el que se descargarán los ficheros de instalación de WebLogic (sólo si estamos ejecutando el 'Net Installer'). Al pulsar en 'Next' aparecerán dos opciones a escoger. Elegiremos una u otra dependiendo del tipo de instalación que deseemos hacer de WebLogic ('Típica' o 'Personalizada'). En nuestro caso seleccionaremos 'Personalizada'.

Una vez realizado todo lo anterior aparecerá la siguiente pantalla:



Explicaremos cada uno de los componentes en el siguiente punto de la documentación. En nuestro caso seleccionaremos el elemento que no está marcado ('Server Examples') para poder comprobar cómo funciona WebLogic de una manera "rápida". La instalación Típica realiza una instalación del producto con la misma estructura que se muestra en la imagen.



En esta pantalla se puede seleccionar qué JVMs (a través de la SDK) se instalarán para usar con WebLogic.

En este caso marcamos las dos opciones, puesto que para poder ejecutar los ejemplos de WebLogic hace falta la 'JRockit'.

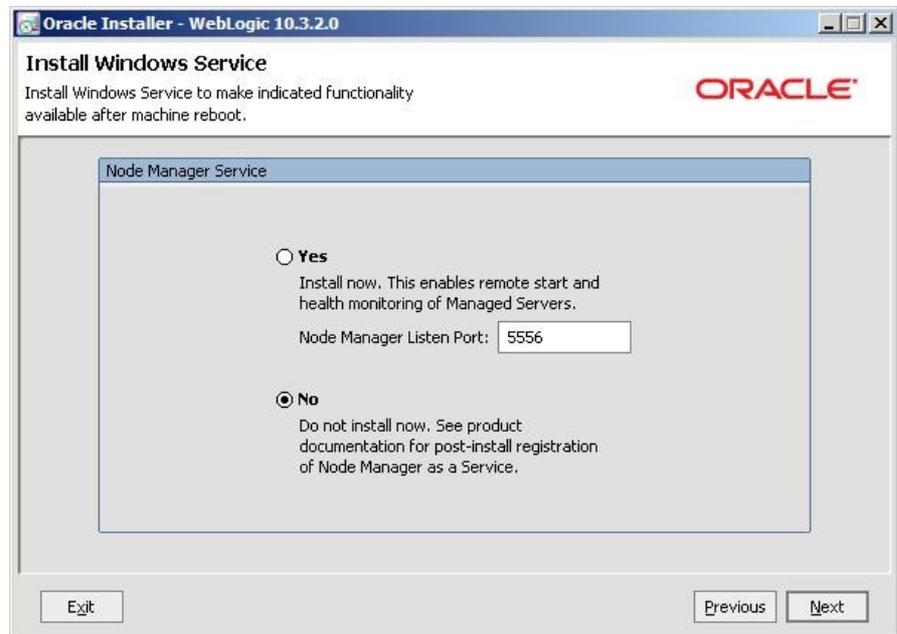
'JRockit' es una JVM propietaria, ahora perteneciente a Oracle, en la que supuestamente se mejora el rendimiento respecto a la de 'Sun Microsystems'.

Una vez que pulsemos sobre “Next” comenzará la descarga de los componentes seleccionados por parte del 'Net Installer'. Al finalizar, tendremos que escoger la ruta en la que se instalarán los productos 'WebLogic' y 'OEPE' (explicado en el siguiente punto). Una vez seleccionados los directorios, pulsamos en “Next” y encontraremos la siguiente pantalla:

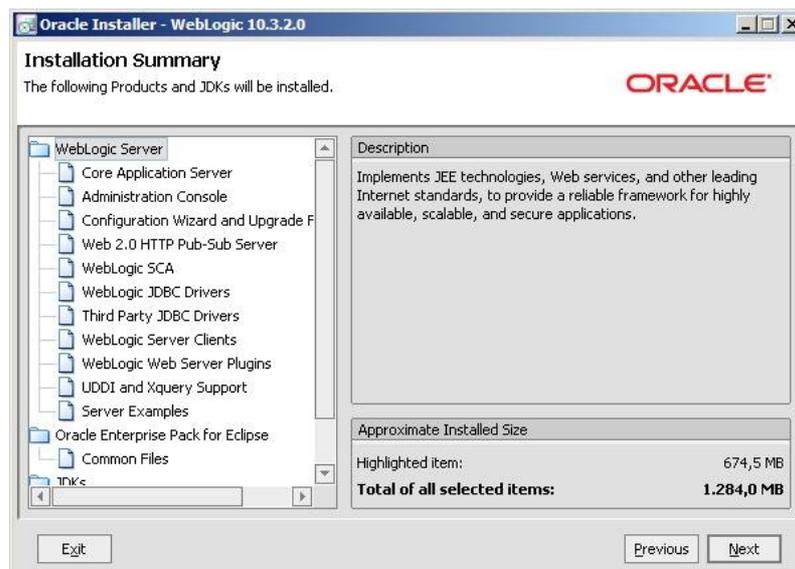
Se nos pregunta si queremos que se active en Windows el servicio 'Node Manager'.

Se trata de un “controlador” de todos los 'Managed Servers' que instalemos en el servidor y queramos administrar desde la consola de WebLogic. Básicamente sirve para aunar todos esos 'Managed Servers' y manejarlos de forma más centralizada.

En nuestro caso marcamos la opción 'No' puesto que sólo usaremos el 'Administration Server' para la práctica.



Al pulsar en “Next” daremos paso a la configuración de las carpetas del menú de inicio de Windows y la disponibilidad para todos los usuarios o para el que ejecuta la instalación. Se puede poner lo que se desee. Después de eso, aparecerá un sumario de todo lo que vamos a instalar:



Pulsamos en 'Next' y comenzará la instalación de WebLogic (paciencia). Una vez que finalice, el instalador nos dará la opción de lanzar el “Asistente de Configuración” de WebLogic para, entre otras cosas, configurar nuestro dominio. Lo obviamos. La instalación ha finalizado.

## **2.- Tipos de instalación del producto (componentes):**

Este apartado estará dividido en dos partes fundamentales. Una destinada a explicar cada uno de los componentes que se instalan junto al producto 'WebLogic server' y la otra a la explicación de las dos formas diferentes de instalar un dominio y de como se lleva a cabo dicha acción.

### **2.1.- Componentes de WebLogic Server:**

#### Core Application Server:

Es la parte básica del servidor WebLogic. Teniendo esta parte, ya se obtiene soporte para la plataforma JavaEE, por lo que podría ser una opción instalar tan solo este componente si deseamos una instalación mínima. Si se desea instalar el servidor WebLogic, obviamente es obligatorio instalar este componente.

#### Administration Console:

Es una aplicación web diseñada para poder administrar, configurar y monitorizar los dominios WebLogic. Su instalación es opcional, pero si no se hace no dispondremos en el arranque del servidor WebLogic de esta potente utilidad.

#### Configuration Wizard and Upgrade Framework:

En realidad esta opción contiene dos componentes. Uno de ellos, el 'Configuration Wizard', es un asistente que nos ayudará a definir los parámetros más básicos para la creación de un dominio WebLogic. Por otra parte, el 'Upgrade Framework' sirve para convertir una configuración de un dominio WebLogic antiguo (desde la versión 9) a uno más moderno.

Si se optara por no instalar este componente, que es opcional, no se podría usar el WLST (WebLogic Scripting Tool) y sólo se recomienda si se quiera poner el servidor en producción y tan sólo haya que importar los dominios WebLogic creados en la fase de pre-producción.

#### Web 2.0 HTTP Pub-Sub Server:

Se trata de un paradigma para la recepción/envío de mensajes de forma asíncrona. Es una característica propia de la web 2.0. Su utilidad es que los usuarios no tienen que pedir explícitamente, mediante una solicitud, el refresco de los mensajes, lo harán de forma automática.

Es un complemento totalmente opcional y si no se instala, tan sólo perderemos esta funcionalidad.

#### WebLogic SCA:

SCA (Service Component Architecture) es una implementación más nueva de SOA (Service Oriented Architecture). Su función es la de interconectar servicios ya programados (no tienen por qué ser del mismo lenguaje) para enlazarlos con una aplicación web.

#### WebLogic y Third-Party JDBC drivers:

Corresponde a los distintos drivers disponibles para crear enlaces con diferentes BBDD. Su instalación no es obligatoria, pero si no están instalados no será posible realizar conexiones con BBDD.

WebLogic Server Clients:

Se trata de los ficheros JAR para los 'Thin-Clients', encargados de conectar a una instancia WebLogic. Incluye los JARs para clientes Web Services, JMS, Store and Forward, WebLogic RMI, JMS .NET y JMX.

Es un paquete opcional.

WebLogic Web Server Plugins:

Son los componentes necesarios para conectar con WebLogic servidores web como pudieran ser Apache, Microsoft IIS, etc.

Este paquete es opcional.

UDDI and Xquery support:

Contiene los componentes necesarios para implementar UDDI (Universal Description, Discovery and Integration). UDDI consiste en un espacio para compartir información entre las empresas siguiendo el estándar de comunicación SOAP (Simple Object Access Protocol).

Este paquete es opcional.

Server Examples:

Permite instalar aplicaciones de ejemplo que trabajan con WebLogic para poder demostrar la potencia y flexibilidad del producto.

Es un paquete opcional que, si se instalase, implicaría la instalación (de forma automática y transparente) de una BD de evaluación.

Oracle Enterprise Pack for Eclipse:

Consiste en un Eclipse modificado para poder trabajar de forma más cómoda con WebLogic y las opciones que ofrece.

Es un paquete opcional, se puede usar cualquier IDE que soporte JavaEE.

**2.2.- Creación/Modificación de dominios en WebLogic:**

Hay que considerar la creación/modificación de los dominios como parte de una instalación en WebLogic. Si no existe un dominio, unidad básica de administración en WebLogic, no podremos arrancarlo y administrar el servidor.

Para realizar tareas relacionadas con los dominios (creación o modificación), podremos usar una herramienta de línea de comandos ('WLSC' o 'WebLogic Scripting Tool') o en su defecto una interfaz gráfica ('Configuration Wizard').

En este caso, por la facilidad de uso y porque el sistema operativo usado lo permite, usaremos la solución gráfica.

Al arrancar la aplicación, se mostrarán dos opciones a elegir. La primera de ellas ('Create a new WebLogic domain') nos permite crear un nuevo dominio en el directorio de proyectos

('user\_projects') de WebLogic. La segunda opción, por su parte, nos permitirá extender la funcionalidad mediante complementos a un dominio ya existente.

El asistente tiene una apariencia como la siguiente:

**Create a new WebLogic domain**  
 Create a WebLogic domain in your projects directory.

**Extend an existing WebLogic domain**  
 Use this option to add new components to an existing domain and modify configuration settings.

Si quisiéramos extender un dominio tan sólo habría que escoger la segunda opción, elegir el dominio existente en el servidor a través de un menú del explorador de ficheros del sistema y personalizar y aplicar los componentes que se deseen a dicho dominio. Los componentes estarán disponibles para seleccionar siempre que estén instalados en el sistema, como es lógico.

En cambio, para crear un nuevo dominio, escogemos la primera opción, pulsamos en 'Next' y dará lugar a una pantalla donde podremos seleccionar si usar una plantilla para crear el dominio o crear un dominio base de acuerdo a los componentes que se seleccionen en esta misma pantalla. Para crear un dominio desde cero y funcional, bastará con dejar marcada la primera opción sin seleccionar ningún componente adicional.

Enter the name and location for the domain:

Domain name:

Domain location:

Tras la pantalla anterior, pasaremos a configurar tanto el nombre del dominio como el directorio en el que estará situado (por defecto en 'user\_projects/domains' dentro del home de MiddleWare).

A continuación, pasaremos a configurar el usuario de administración del dominio WebLogic. En este caso, como se aprecia en la captura, será 'weblogic'.

\*User name:

\*User password:

\*Confirm user password:

Description:

A continuación nos encontramos con una de las pantallas más importantes que tiene el asistente. Se trata de la configuración del modo de arranque del dominio y de la JDK que se utilizará.

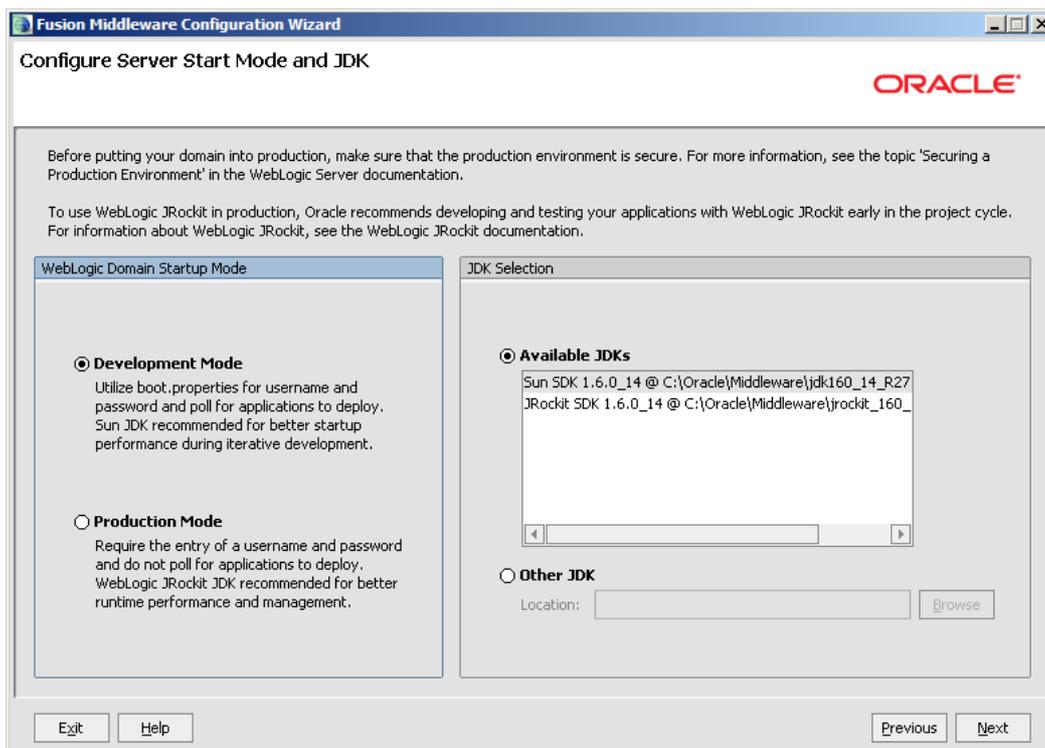
Los dominios WebLogic tienen dos formas de operar. Se trata de 'Modo de desarrollo' y del 'Modo de producción'.

Si estamos en un entorno en el que estamos desarrollando una aplicación, escogeremos el modo de desarrollo. Algunas de las referencias existentes son por ejemplo el despliegue automático de las aplicaciones, menos conexiones simultáneas del servidor a través de los drivers JDBC y sobre todo un manejo un poco más ágil de la consola.

Además si queremos usar SSL para el despliegue y la utilización de nuestras aplicaciones, en el modo de desarrollo se podrá hacer con unos certificados que vienen con la instalación de WebLogic. En cambio si quisiéramos usar SSL en un entorno de producción tendríamos que instalar nuestros propios certificados.

Una opción útil e imprescindible para el servidor en modo de producción es que mantiene los logs de funcionamiento del servidor, cosa que no ocurre cuando WebLogic se encuentra en modo de desarrollo.

La apariencia que tiene la pantalla donde seleccionamos los modos es la siguiente:



En cuanto a la JDK a utilizar, para el 'Modo de desarrollo' se recomienda usar la JDK de Sun Microsystems. Es recomendable porque el arranque del servidor resulta mucho más rápido que si usásemos la JRockit. Debido a los múltiples reinicios que se puede hacer en un servidor en 'Modo de desarrollo', es imprescindible disponer de la rapidez de arranque que brinda la JDK de Sun.

Si pusiéramos nuestro servidor en 'Modo producción' que tuviera mucha demanda, sería recomendable usar la JDK JRockit ya que está optimizada para trabajar con mucha carga y en estos casos es más rápida y estable que la JDK de Sun.

El asistente también permite la elección de una JDK personalizada y/o que no se haya reconocido en el sistema.

Una vez que pulsemos sobre 'Next' accederemos a una pantalla de configuración adicional, donde podremos escoger las siguientes opciones: 'Administration Server' (para modificar las opciones del servidor principal de WebLogic), 'Managed Servers Clusters and Machines' (podremos añadir, borrar o modificar las opciones de cada uno de los componentes) y por último 'RDBMS

Security Store' (se podrán modificar las opciones de dicha característica).

**Administration Server**  
Modify Settings

**Managed Servers, Clusters and Machines**  
Add or Delete  
Modify Settings

**RDBMS Security Store**  
Modify Settings

En este caso, seleccionaremos las tres opciones y pulsaremos sobre 'Next' para observar qué se puede configurar en cada uno de los puntos.

En la sección 'Administration Server' se podrán configurar las siguientes opciones:

\*Name:

\*Listen address:

Listen port:

SSL listen port:

SSL enabled:

Cambiamos el nombre del 'Admin Server'.

Cambiamos las direcciones en las que escuchará el servidor en caso de tener más de una interfaz de red o una sola con alias.

Cambiamos el puerto de escucha por defecto (por si tuviéramos más de un dominio en el mismo servidor).

Por último se pueden cambiar las opciones para SSL.

En las secciones 'Configure Managed Servers', 'Configure Clusters' y 'Configure Machines' podremos configurar los 'Managed Servers' los 'Clusters' y las 'Máquinas' de nuestro dominio WebLogic respectivamente.

Por último, se podrá configurar la opción 'RDBMS Security Store'.

Create RDBMS tables in your data store before booting your domain using scripts in your WebLogic Server's server/lib directory

Database Type:

Driver:

Class Name:

DBMS SID:  User Name:

DBMS Host:  User Password:

DBMS Port:  Confirm User Password:

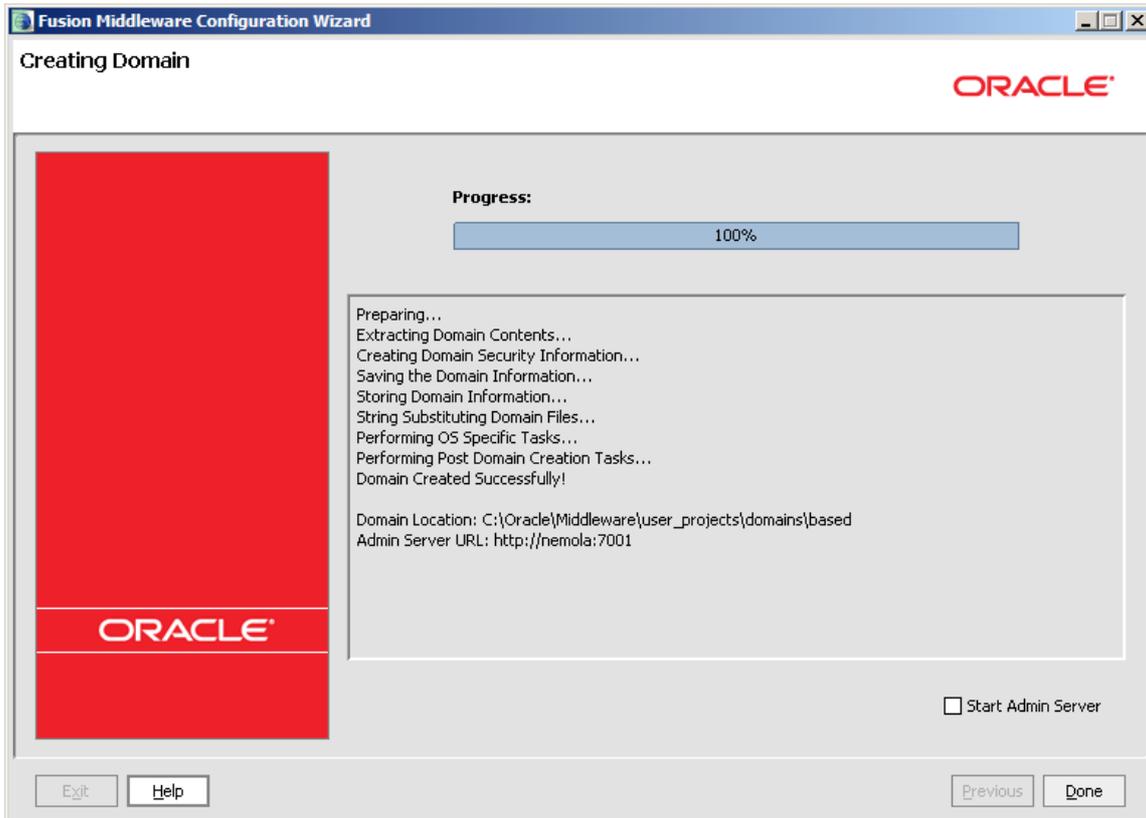
URL:

Known Properties:

Additional Properties:

Se trata de configurar el dominio para que sus usuarios se almacenen en un sistema de base de datos relacional (RDBMS) en lugar de hacerlo en el servidor LDAP que tiene integrado el servidor WebLogic.

Una vez configurado todo y pulsando sobre 'Next', aparecerá un breve resumen. Si estamos de acuerdo con las opciones elegidas pulsaremos sobre 'Create' para crear el dominio.



Podemos marcar la opción 'Start Admin Server' antes de pulsar sobre el botón 'Done' para arrancar el dominio al finalizar el asistente.

Si el dominio estuviera configurado en 'Modo producción', al arrancarlo tendríamos que indicar el nombre de usuario y el password del usuario administrador de dicho dominio. En el caso de un dominio en 'Modo desarrollo' no ocurriría lo mismo y arrancaría de forma automática.

```
Starting WLS with line:
C:\Oracle\MIDDLE~1\JDK160~1.5-3\bin\java -server -Xms256m -Xmx512m -XX:MaxPermSize=128m -Dweblogic.Name=AdminServer -
Djava.security.policy=C:\Oracle\MIDDLE~1\WLSEU~1.3\server\lib\weblogic.policy -Xverify:none -da -Dplatform.home=C:\Or
acle\MIDDLE~1\WLSEU~1.3 -Dwls.home=C:\Oracle\MIDDLE~1\WLSEU~1.3\server -Dweblogic.home=C:\Oracle\MIDDLE~1\WLSEU~1.3\s
erver -Dweblogic.management.discover=true -Dwls.iterativeDev=false -Dwls.testConsole=false -Dwls.logErrorsToConsole=f
alse -Dweblogic.ext.dirs=C:\Oracle\MIDDLE~1\patch_wls1032\profiles\default\sysexm_manifest_classpath;C:\Oracle\MIDDLE~1\
patch_oepe1032\profiles\default\sysexm_manifest_classpath weblogic.Server
<12-feb-2010 20H41' CET> <Info> <WebLogicServer> <BEA-000377> <Starting WebLogic Server with Java HotSpot(TM) Server UM
Version 14.0-b16 from Sun Microsystems Inc.>
<12-feb-2010 20H41' CET> <Info> <Management> <BEA-141107> <Version: WebLogic Server 10.3.2.0 Tue Oct 20 12:16:15 PDT 20
09 1267925 >
<12-feb-2010 20H41' CET> <Info> <Security> <BEA-090065> <Getting boot identity from user.>
Enter username to boot WebLogic server:weblogic
Enter password to boot WebLogic server:
```

En caso de que queramos arrancar de forma manual el AdminServer podríamos hacerlo desde el menú de Inicio de Windows ('Inicio/Programas/Oracle WebLogic/User Projects/dominio/Start Admin Server for WebLogic Server Domain') o bien desde el directorio del dominio usando el script llamado 'startWebLogic'.

### **3.- Utilidad y funciones básicas de la consola de administración:**

La consola de administración de WebLogic consiste en una aplicación web destinada al mantenimiento de un dominio WebLogic. Entre otras muchas, se pueden llevar a cabo las siguientes tareas a través de esta interfaz gráfica:

- ✓ Configurar, arrancar y parar las diferentes instancias del dominio.
- ✓ Configurar clústeres.
- ✓ Configurar servicios WebLogic, como conexiones con bases de datos (JDBC) y servicio de mensajes (JMS o Java Message Service).
- ✓ Configurar parámetros de seguridad, incluyendo la administración de usuarios, grupos y roles.
- ✓ Configurar y desplegar aplicaciones.
- ✓ Monitorizar tanto el servidor como las aplicaciones desplegadas.
- ✓ Visualizar los logs del dominio y/o servidores.
- ✓ Ver los descriptores de despliegue de las aplicaciones.
- ✓ Editar los elementos del descriptor de despliegue de una aplicación.

En nuestro caso, veremos algunas funciones básicas de la consola, ya que sus posibilidades son muchísimas y su complejidad es elevada. Pero antes de empezar, haré un pequeño comentario sobre las zonas en las que se encuentra dividida la consola, incluyendo posteriormente una captura del aspecto que tienen.

#### Breve explicación de cada zona:

- A) Es una zona variable donde, inicialmente, aparecerán todas las acciones que se pueden accionar desde la consola de administración (Cuando se está en este estado, se dice que se está situado en el 'Home'). Esta zona se adaptará dependiendo del contexto en el que estemos. Por ejemplo, si elegimos la opción 'Servers' esta zona se adaptará de acuerdo a las opciones modificables, vistas de registros, etc. de los distintos servidores.
- B) Aquí se muestra la estructura de un dominio (diferentes opciones y características) en forma de árbol. Si se despliegan todas sus entradas observaremos el mismo contenido que el mostrado en la zona central ('A') de la consola cuando accedemos por primera vez y muestra el 'Home'. Esta zona siempre estará disponible.
- C) Se trata de un pequeño espacio con un enlace con el que podremos acceder a los últimos cambios realizados en nuestro dominio. Así como ver qué cambios se han aplicado o por el contrario aún deben ser aplicados.
- D) Son unas pestañas que nos permiten acceder a los complementos instalados en la consola de administración que dispongan de alguna vista. Por defecto, sólo saldrá la pestaña correspondiente a la consola de administración (Oracle WLS Console).

- E) En esta zona se muestran algunos accesos directos de la aplicación. Por ejemplo, el botón 'Log out' para salir de la consola o el botón 'Home' para mostrar la pantalla inicial en la zona variable 'A'.
- F) Muestra las últimas ubicaciones en las que estuvimos en la consola. Es muy útil cuando se accede, por ejemplo, a algún menú muy escondido.
- G) Es una pequeña zona donde se muestran las F.A.Q. relacionadas a las opciones que estemos manejando en ese momento. Por lo tanto, varía según en qué zona estemos.
- H) Se trata de unos gráficos en los que podemos observar a simple vista el estado de los servidores que se estén ejecutando en nuestro dominio WebLogic.

### 3.1- Preferencias de la consola de administración:

Además de controlar el funcionamiento de los componentes de un dominio, la consola de administración tiene una sección para configurar su propio funcionamiento. Se puede acceder desde la zona de accesos directos ('E') pulsando sobre 'Preferences'.

Una vez dentro de las preferencias nos encontramos con cuatro pestañas.



'User Preferences' corresponde a las opciones correspondientes al usuario que está cambiando las preferencias en ese momento. Se pueden cambiar opciones como el separador usado a la hora de usar el navegador de ficheros de la consola, guardar las rutas de los ficheros localizados desde la consola y el número de ellas, alertas a la hora de realizar algún cambio, etc.

'Shared Preferences' afecta a todos los usuarios y por lo tanto, al funcionamiento global de la consola de administración. Las dos opciones que se pueden cambiar son 'Management operation timeout' y 'Follow Configuration Changes'. 'Management operation timeout' establecerá el tiempo en segundos en que está permitido finalizar un cambio en el dominio. Por su parte, al activar la opción 'Follow configuration changes' la consola instentará adaptarse a los cambios que se hayan hecho en el dominio. Por ejemplo, un cambio en el puerto de escucha del 'Administration server'.

'Extensions' permite que podamos elegir qué componentes debe cargar la consola de administración o bien la posibilidad de añadir nuevas extensiones. Nos permitirá añadir o quitar funcionalidades a la consola. Por ejemplo, un componente interesante, desactivado por defecto, es el 'diagnostics-console-extensions' (se comentará más adelante).

'WLST Script Recording' nos permitirá almacenar los cambios que se hagan en el dominio en un script, por ejemplo en python. Se puede usar posteriormente para crear un nuevo dominio de semejantes características.

### 3.2- Servers:

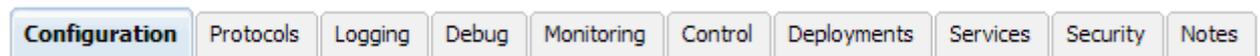
En un dominio WebLogic puede haber más de un servidor, pero sólo uno de ellos es el servidor principal ('Administration server') y es el único que existe en una instalación nueva o básica de un dominio WebLogic. En nuestro caso, la sección 'Servers' tiene una pinta como la que sigue:

<input type="checkbox"/>	Name	Cluster	Machine	State	Health	Listen Port
<input type="checkbox"/>	AdminServer(admin)			RUNNING	✔ OK	80

Desde la sección 'Servers' se pueden hacer diferentes opciones con cada servidor disponible:

- Arrancar/Parar.
- Reiniciar SSL.
- Clonar.
- Borrar.
- Crear uno nuevo.

Por otra parte, se pueden acceder a las propiedades de cada servidor haciendo clic encima del campo que le corresponda en la columna 'Name'. En este caso, hacemos clic sobre 'AdminServer(admin)' y accederemos a un menú de configuración con las siguientes pestañas:



A continuación se explican las acciones más relevantes que se pueden llevar a cabo en cada pestaña.

- Configuration:

Podremos gestionar los puertos de escucha del servidor (no-SSL y SSL); el compilador de java usado; gestión de clústeres; servicios ligados al servidor en concreto (por ejemplo Java Messaging System); almacenes de claves, configuraciones de SSL, uso de CA; modo de despliegue de una aplicación, directorio de subida por defecto para las aplicaciones desplegadas; tuning del servidor permitiendo modificar valores de conexión, puertos, timeouts; comportamiento en casos de sobrecarga, respuestas y peticiones máximas; y parámetros de inicialización del servidor entre otros.
- Protocols:

Se pueden configurar canales de comunicación ya predefinidos o establecer nuevos (por ejemplo para 'ldap', 'https', etc.). Por ejemplo, para el canal predefinido 'http' podemos cambiar cual será el contexto por defecto cuando escribamos la url de nuestro servidor de administración.
- Logging:

Podemos gestionar los logs del servidor. Tanto a nivel global como a nivel del canal que deseemos. Por defecto deja administrar esas dos opciones. Podremos cambiar el modo de rotación de los logs, cada cuanto tiempo deben hacerlo, su nombre, número de archivos de log a almacenar en el directorio de almacenamiento, etc.
- Debug:

Se trata de una sección en la que podremos seleccionar de forma muy detallada mediante un menú en forma árbol, sobre qué componentes queremos hacer logging en modo debug. Por defecto, todos vienen desactivados.
- Monitoring:

Útil para visualizar datos correspondientes al funcionamiento del servidor. Se puede observar el estado de salud del servidor, datos sobre rendimiento (memoria de la JVM), carga del servidor, datos correspondientes a la seguridad del servidor (accesos fallidos por ejemplo) y entre otros, valores correspondientes a las distintas conexiones JDBC que tengamos.
- Control:

Podemos controlar algunos parámetros para el arranque y parada del servidor, así como realizar desde ese misma vista una parada del mismo, por ejemplo.
- Deployments:

Muestra las aplicaciones que han sido desplegadas en el servidor que estemos visualizando. Se puede acceder a las opciones de cada aplicación desplegada clicando sobre su nombre.
- Services:

Sección destinada a la configuración de diferentes servicios para el servidor. Entre otros, se puede configurar JDBC, servidores de correo (con SMTP e IMAP), JMS, etc.
- Security:

Aquí se podrán definir roles y políticas para alterar el comportamiento de nuestro servidor en distintas ocasiones o casos. Trato según el usuario, según el día, etc.
- Notes:

Se trata de una zona de texto destinada a la descripción de la configuración que se haya realizado en el servidor.

### 3.3- Deployments:

Es la sección en la que se muestran todos las aplicaciones desplegadas en nuestro servidor. Se muestra una tabla en la que veremos como información relevante el nombre de la aplicación, el estado, la salud, el tipo de aplicación y la prioridad a la hora de desplegarla (es un valor relativo a los demás despliegues en caso de existir). Un ejemplo de esta tabla sería el siguiente:

Name	State	Health	Type	Deployment Order
TestORCL	Active	OK	Web Application	100

Como en el caso de los servidores, podemos acceder a sus propiedades pulsando sobre el nombre del despliegue deseado. Aparecerá una pantalla de propiedades con las siguientes pestañas:



A continuación se explican las acciones más relevantes que se pueden llevar a cabo en cada pestaña.

- **Overview:**  
Muestra las características básicas de la aplicación desplegada (Nombre, path, modelo de seguridad, etc.). Se puede configurar, por ejemplo, el orden de despliegue de la aplicación.
- **Deployment Plan:**  
En el caso de que la aplicación disponga de plan de despliegue, podremos cambiar los parámetros del mismo desde esta sección.
- **Configuration:**  
Podemos configurar opciones relacionadas con el logueo de la aplicación, al igual que pasaba con los servidores, es decir, nombre de los logs, rotación, tamaño, directorios, etc. Además, podemos configurar otras opciones relacionadas con cookies, tiempo de validez de las sesiones, activar el modo debug, nivel de detalle de la compilación en caso de fallo, etc. También podemos configurar en la sección 'Workload' dentro de 'Configuration' por si queremos que WebLogic gestione el rendimiento de nuestra aplicación con una serie de parámetros que establezcamos.
- **Security:**  
Podremos establecer roles y políticas tanto a nivel de aplicación como a nivel de url. Tiene las mismas posibilidades que en la sección de servidores.
- **Targets:**  
Se podrá seleccionar en qué instancias o clústeres WebLogic, además de VirtualHosts, deseemos que nuestra aplicación se despliegue.
- **Control:**  
Aquí se podrá parar o arrancar la aplicación.
- **Testing:**  
Se muestra una tabla que contiene una serie de URLs para que podamos testar si nuestra aplicación se desplegó de manera satisfactoria.
- **Monitoring:**  
Obtendremos datos relativos a los servlets de nuestra aplicación, sesiones o carga.

- Notes:  
Esta sección, al igual que en los servidores, está destinada a la información adicional sobre la configuración de un servidor.

### **3.4- Security Realms:**

Un reino de seguridad es un contenedor para los mecanismos con los que se protegerán los dominios. Incluye usuarios, grupos, roles, políticas de seguridad y proveedores de seguridad.

Se pueden tener tantos reinos de seguridad como se desee, pero sólo puede existir uno activo en un determinado momento. Por defecto, al crear un dominio WebLogic, ya existe un dominio por defecto y activado llamado 'myrealm'.

Al pulsar sobre la opción 'Security realms' aparecerá una tabla con todos los 'realms' disponibles. Tiene una apariencia como las siguientes:

<input type="checkbox"/>	Name 	Default Realm
<input type="checkbox"/>	myrealm	true

Pulsando sobre el nombre del 'reino' accederemos a la pantalla de modificación y consulta de parámetros. Veremos una serie de pestañas como la siguiente:



- Configuration:  
Podremos establecer el modelo de seguridad por defecto aplicado a las aplicaciones que estén protegidas por el reino de seguridad configurado. Se pueden configurar además orígenes remotos donde se guardarán todas las opciones del reino ('RDMS'). Se pueden establecer los niveles a partir de los cuáles se bloquea un usuario, cuánto tiempo. Y por último, se podrá establecer una caché para no tener que estar consultando constantemente los principales (entradas del reino).
- User and Groups:  
Se podrán manejar los usuarios y grupos del reino. Se pueden crear, borrar y modificar tanto los usuarios como los grupos.
- Roles and Policies:  
Se podrán establecer tanto los roles y las políticas para cada uno de los componentes que conforman el dominio WebLogic (despliegues, servidores, JDBC, JMS, etc.).
- Credential Mappings:  
Tiene la utilidad de mapear las cada parámetro de los principales en el caso de querer usar el reino de forma remota en una aplicación que use los parámetros de forma diferente.
- Providers:  
Permite establecer los mecanismos para cada una de las características del reino (autenticación, validación de passwords, autorización, adjudicación, mapeo de roles, certificados, almacén de claves). Por defecto, todos los mecanismos son los que

incorpora el servidor WebLogic pero, por ejemplo, sería posible establecer como mecanismo de autenticación a un servidor LDAP.

- **Migration:**  
Permite importar y exportar toda la base de datos correspondiente al reino (usuarios, passwords, roles, etc.).

### 3.5- Log Files:

Desde esta sección podremos observar, sin salir de la consola de administración, los logs de administración de una forma estructurada.

**Log Files**

View Showing 1 to 7 of 7 Previous | Next

	Name 	Type	Server
<input type="radio"/>	DomainLog	Domain Log	AdminServer
<input type="radio"/>	EventsDataArchive	Instrumentation	AdminServer
<input type="radio"/>	HarvestedDataArchive	Metric Data	AdminServer
<input type="radio"/>	HTTPAccessLog	HTTP Access	AdminServer
<input type="radio"/>	JMSMessageLog/WseeJmsServer	JMS Log	AdminServer
<input type="radio"/>	JMSSAFMessageLog/ReliableWseeSAFAgent	JMS SAF Agent Log	AdminServer
<input type="radio"/>	ServerLog	Server Log	AdminServer

View Showing 1 to 7 of 7 Previous | Next

En esa tabla seleccionaremos el log deseado y pulsaremos sobre 'View'. A su vez, una vez dentro de cada log, podremos ver de forma independiente cada suceso almacenado en él, separando por campos la información.

### 3.6- WLDF Console Extension:

Se trata de una extensión de la consola de administración. Se activa desde las preferencias de la consola y su cometido es el de mostrar en gráficas y en tiempo real el funcionamiento del dominio. Necesita una JVM en el navegador del cliente que la ejecute.

Se accede mediante el botón 'WLDF Console Extension' situado en la zona de las extensiones disponibles de la consola ('D').

Una vez que pulsemos sobre él podremos configurar las diferentes gráficas.

## **4.- Desplegar una aplicación web que acceda a la base de datos:**

### Introducción:

Se ha creado una aplicación que accede a una base de datos Oracle y captura datos de una determinada tabla, con un determinado usuario.

La aplicación está desarrollada en JSP desde el IDE Netbeans. Se podría haber usado la versión personalizada de Eclipse que trae WebLogic. Se trata de OEPE (Oracle Enterprise Pack for Eclipse) que proporciona más facilidades para programar aplicaciones que vayamos a ejecutar en un servidor WebLogic, entre otras cosas.

Los datos para la conexión de la aplicación JSP son los siguientes:

Servidor de la BD:	'localhost'	Usuario de la BD:	'scott'
Puerto de escucha de la BD:	'1521'	Password del usuario:	'tiger'
Nombre de la BD:	'orcl'	Nombre de la tabla:	'dept'

### Código de la aplicación y explicación de los fragmentos más relevantes:

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<%@page import="java.sql.*"%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
    <link rel="stylesheet" href="css/estilo.css">
    <title>Test de conexión con Oracle</title>
  </head>
  <body>
    <center>
      <h2>Test de conexión con ORACLE</h2><br/><br/><br/>
      <%
        //Variables para la conexión...
        String usuario="scott"; //Usuario SCOTT de Oracle
        String password="tiger"; //Password para SCOTT
        String tabla="dept"; //Nombre de la tabla
        String url = "jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP) (HOST=localhost)" +
          "(PORT=1521)) (CONNECT_DATA=(SERVER=DEDICATED) (SERVICE_NAME = orcl)))";
        Connection con = null;
        Statement s = null;
        //-----

        try
        {
          //Cargamos el driver de Oracle
          Class.forName("oracle.jdbc.driver.OracleDriver");
          //Realizamos la conexión con la BD
          con = DriverManager.getConnection(url, usuario, password);

          //Creamos una sentencia SQL para seleccionar las filas
          s = con.createStatement();
          s.execute("SELECT * FROM "+tabla);
          ResultSet rs = s.getResultSet();

          //Mostramos los nombres de las columnas
          out.println("<table>");
          int columnas=rs.getMetaData().getColumnCount();
          out.println("<tr id=columnas>");
          for(int i=1;i<=columnas;i++)
            out.println("<td>"rs.getMetaData().getColumnName(i)+"</td>");
        }
      %>
    </center>
  </body>
</html>
```

```

        out.println("</tr>");

        //Mostramos las filas de la tabla
        while(rs.next()){
            out.println("<tr>");
            for(int i=1;i<=columnas;i++)
                out.println("<td>" +rs.getString(i)+"</td>");
            out.println("</tr>");
        }
        out.println("</table>");

    }
    //En caso de que algo no funcione...
    catch(Exception error){
        out.println("Error: "+error);
    }
}
%>
</center>
</body>
</html>

```

Se definen las variables con el usuario, password y la tabla y a continuación se crea la cadena (url) de conexión con la BD. Su sintaxis es la misma a la que se puede encontrar en el fichero 'tnsnames.ora' en una instalación de Oracle.

A continuación se carga el driver JDBC para Oracle y se realiza la conexión usando los datos definidos previamente.

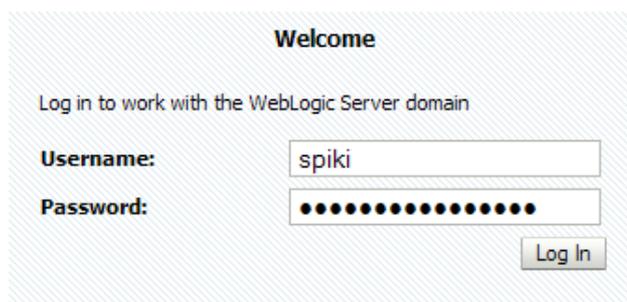
Ahora nuestra aplicación ya podrá capturar datos a los que tenga acceso el usuario 'SCOTT'. En nuestro caso, serán los datos de la tabla 'DEPT'.

### Exportando el proyecto:

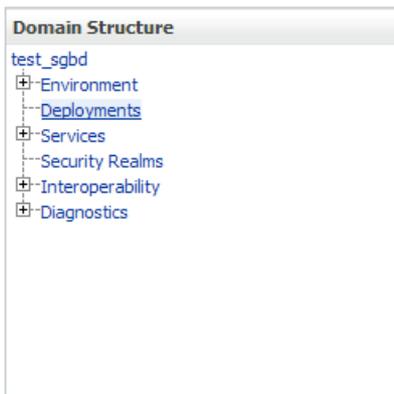
Netbeans es un IDE y nosotros no queremos desplegar nuestra aplicación desde ahí, por lo que tendremos que exportarla para poder usarla desde WebLogic. Existen varias opciones. Entre ellas está la más básica, que consiste en trasladar toda la estructura de directorios de nuestra aplicación a un DocumentRoot que usaremos posteriormente con WebLogic. Pero en este caso optaré por una solución más sencilla y 'limpia'. Se trata de generar un paquete 'war', que contiene todo lo necesario para el despliegue de la aplicación.

Una vez que tengamos exportado el proyecto en formato 'war' lo situamos en el directorio que deseemos. ¡Vamos a instalarlo en nuestro servidor de aplicaciones!

### Instalando la aplicación en WebLogic:



Accedemos a la consola de administración de nuestro dominio, con el usuario de administración.



Una vez que entremos en la consola, clicamos sobre “Deployments” en el menú (“Domain structure”) de la izquierda de consola.

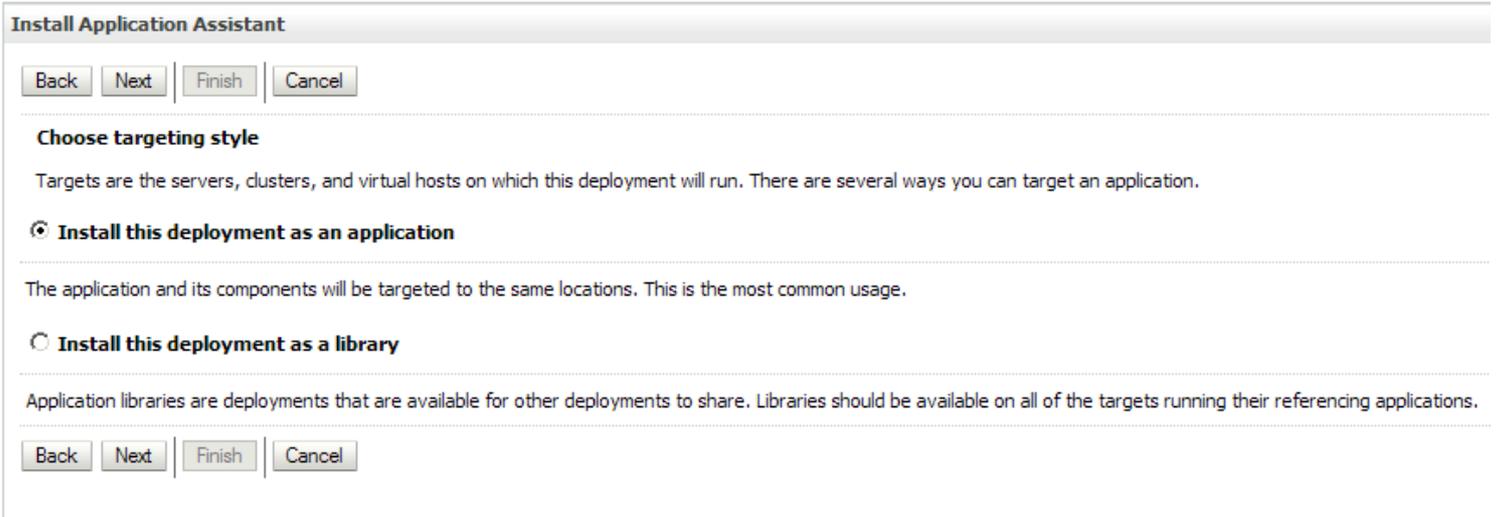
Ahora estamos situados en un sumario de todas las aplicaciones o módulos instalados en nuestro servidor WebLogic. Desde ahí podremos arrancar, parar, rearrancar (redployment) o borrar cualquier entrada que aparezca en ese sumario.

Pulsamos sobre el botón “Install” para comenzar con la instalación de la aplicación. Como antes no pusimos el paquete 'war' en una localización específica, tendremos que usar la opción “Upload your file(s)”. Lo que haremos usando esta opción será colocar nuestra aplicación en una directorio válido para nuestro dominio WebLogic.

Ahora tenemos la opción de seleccionar tanto el archivo de despliegue como el plan de ejecución del mismo (parte opcional). En nuestro caso solo indicaremos el archivo a desplegar.

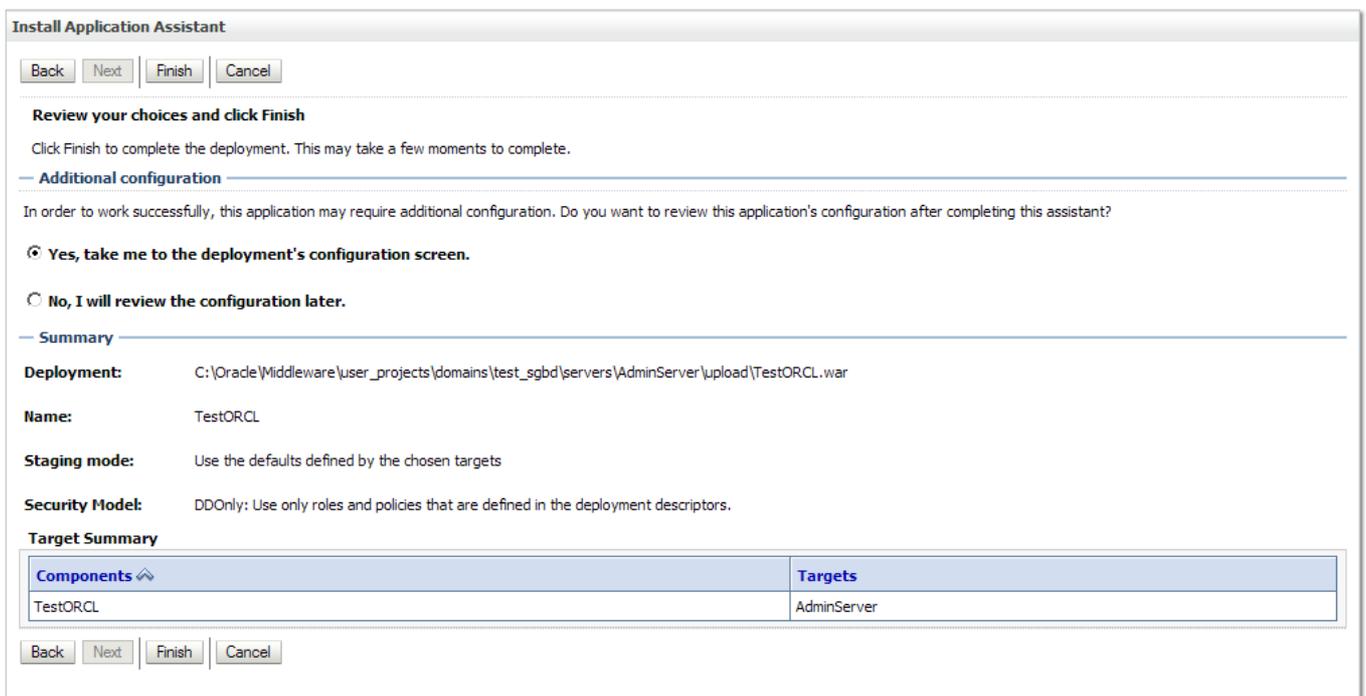
Una vez localizado el paquete 'war' pulsamos sobre “Next”. Seleccionamos nuestro paquete 'war', en este caso 'TestORCL.war' y clicamos sobre “Next”. (Al usar este método, nuestra aplicación estará en el directorio 'upload' del 'AdminServer' del dominio.)

A continuación dejamos marcada la opción “Install this deployment as an application”. Es la opción más común y la usada en situaciones en las que se quiere desplegar una aplicación web, como es el caso. La otra opción (“Install this deployment as a library”) es para cuando se programen librerías que usarán otras aplicaciones y queramos integrarlas con WebLogic.



Al pulsar en “Next” accederemos a un menú de configuración opcional para nuestra aplicación. Se tratan de opciones relacionadas con el acceso a los ficheros de la aplicación, seguridad y el nombre con el que se desplegará la aplicación.

En nuestro caso al tratarse de una prueba, y de una aplicación en absoluto sofisticada, dejamos todas las opciones por defecto. Por nombre mantenemos 'TestORCL', la seguridad de la aplicación la controlará el descriptor de despliegue ('DD') y para la accesibilidad a las fuentes dejamos la primera opción (“Use the defaults defined by the deployment's targets”). Pulsamos nuevamente sobre “Next” y podremos observar un listado con nuestras elecciones a lo largo de la configuración. Además, por si fuera necesario configurar algún parámetro adicional, se puede dejar marcada la opción “Yes, take me to the deployment's configuration screen.” de la sección “Additional configuration”. En nuestro caso esta pantalla muestra un aspecto como el siguiente:



Pulsamos en “Finish” y accederemos a la pantalla de configuración de nuestra aplicación. En nuestro caso lo dejamos todo por defecto, pero pueden personalizarse muchos valores para nuestra aplicación. El orden en que se despliega con respecto a otras aplicaciones del dominio, duración de las cookies en caso de utilizarlas, 'logging' para registrar la actividad de nuestra aplicación, etc.

Pulsamos en “Save” para finalizar la instalación de nuestra aplicación en 'WebLogic'.

Ya podemos acceder a nuestra aplicación desde el navegador web usando como URL:

<http://localhost:7001/TestORCL>

## **5.- Bibliografía:**

- Wikipedia ([en](#), [es](#)).
- Oracle Documentation.
- Foros varios.