

配置 WebLogic Server 11g 集群

单服务器版（Windows）

（本文讲述如何在 WebLogic Server 11g 上配置集群，以及通过 Proxy Server 来访问集群及压力测试）

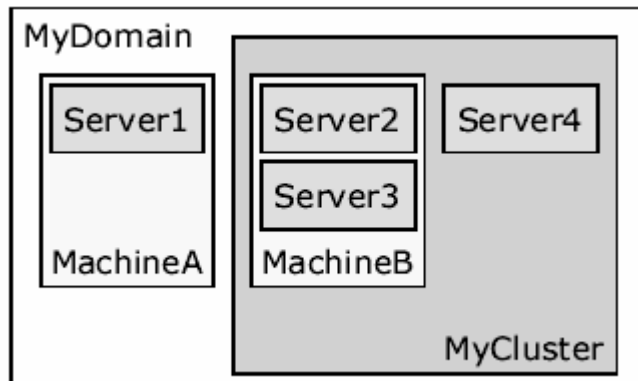
2009-11-1

友情提供：WebLogic 中文博客 www.beansoft.biz
整理：刘长炯 BeanSoft@126.com

友情提示： 本文所需的附件请在 PDF 附件面板中找。

预备知识

什么是 Domain, Server, Machine, Cluster



Domain

Domain 是 WebLogic Server 实例的基本管理单元。所谓 Domain 就是，由配置为 Administrator Server 的 WebLogic Server 实例管理的逻辑单元，这个单元是有所有相关资源的集合。

Server

Server 是一个相对独立的，为实现某些特定功能而结合在一起的单元。

Machine

Machine 是可以对应到服务器所在的物理硬件, 可以是 Unix 或 non-Unix 类型，可以用来远程管理和监控，用于加强 fail over 管理。

Cluster

Cluster 也是一个逻辑概念，用来分组用途相同的服务器实例，一个域中可以有多个集群。集群用来实现负载均衡和容错。

Domain and Server 的关系

一个 Domain 可以包含一个或多个 WebLogic Server 实例，甚至是 Server 集群。一个 Domain 中有一个且只能有一个 Server 担任管理 Server 的功能，其它的 Server 具体实现一个特定的逻辑功能。

本文环境

- 平台：Windows XP SP3, 4G RAM
- 软件：WebLogic Server 11gR1

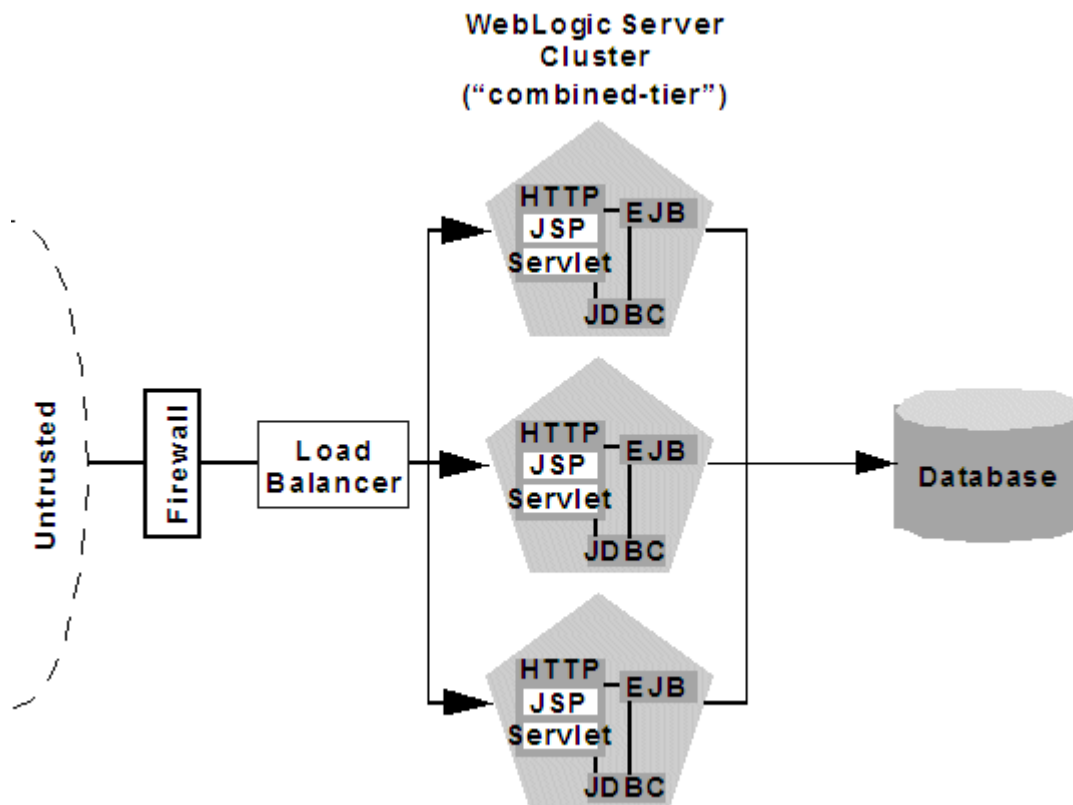
配置 WebLogic Server 集群

WebLogic 集群的体系结构

单层混合型的集群架构

这种架构将所有的 Web 应用以及相关的服务应用全部置于集群中的单一 WLS 实例中，这种架构的优势在于：

- 易于管理
- 灵活的负载均衡机制
- 更强的安全控制

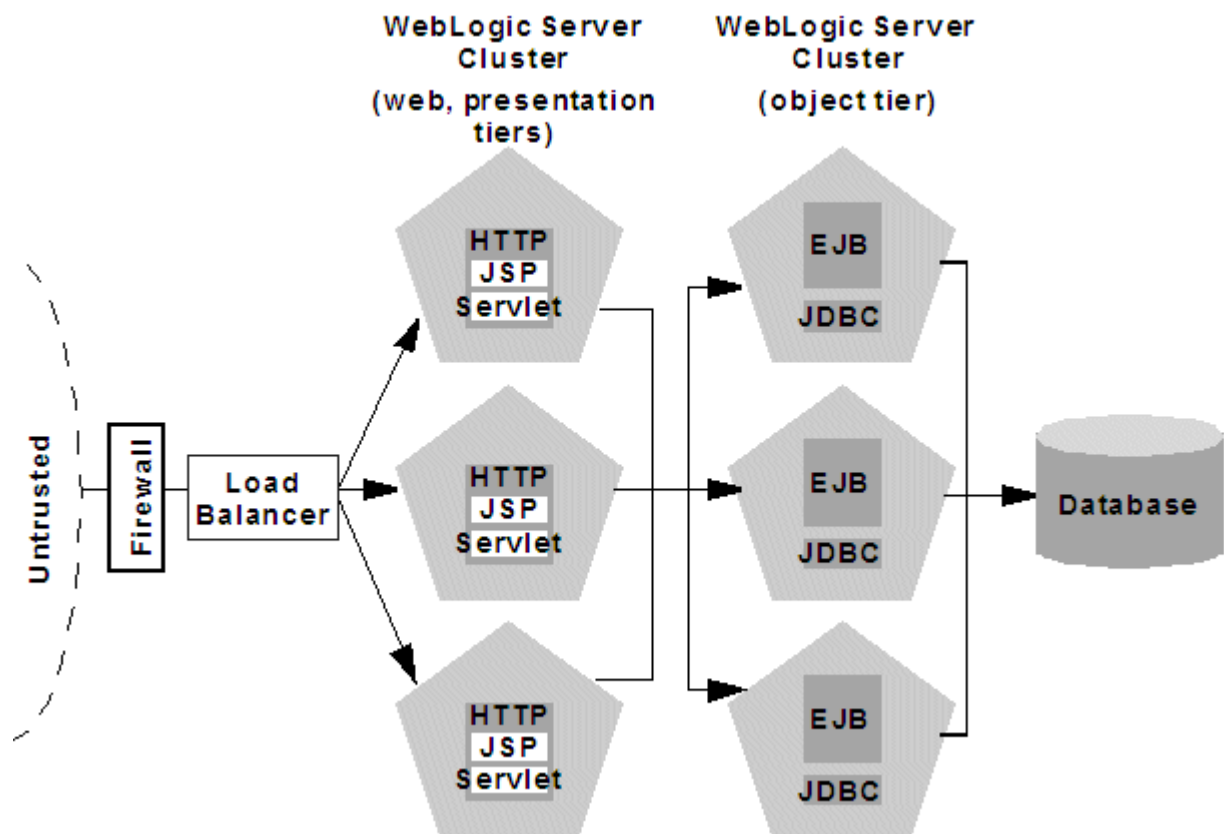


参考 edoc: [Recommended Basic Architecture](#)

多层结构的集群架构（Cluster）

这种架构使用两个 WLS 集群，一个放置静态内容和 JSP/Servlet，另一个放置集群 EJB。一般应用于下面这些情况：

- 在负载均衡机制需要调用集群 EJB 中的方法时；
- 在提供内容与提供对象的服务之间需要更大的机动性时；
- 在需要更高的系统稳定性时。



参考 edoc: [Recommended Multi-Tier Architecture](#)

配置集群应用的必要条件

- 集群中的所有 Server 必须位于同一网段，并且必须是 IP 广播(UDP)可达的
- 集群中的所有 Server 必须使用相同的版本,包括 Service Pack
- 集群中的 Server 必须使用永久的静态 IP 地址。动态 IP 地址分配不能用于集群环境。如果服务器位于防火墙后面，而客户机位于防火墙外面，那么服务器必须有公共的静态 IP 地址，只有这样，客户端才能访问服务器
- 要以 CLUSTER 方式运行，必须有包含 CLUSTER 许可的 LICENSE 才行（从 Oracle 网站上下载的最新版本已经包含了此许可，可进行 Cluster 配置）

配置前的准备工作

在配置集群应用前要对集群的配置信息有一个良好的设计，下面就是我们这次配置的集群信息：在同一台机器上配置集群（在一台机上只安装一套 WebLogic 11g 产品就可以了）

提示：建议您的测试服务器有 4G 以上的内存，否则服务器启动过程将会很漫长。

机器类型	操作系统	网络配置	角色	备注
HP PC	Windows XP SP3	IP: 192.168.0.4 PORT:7001	Administrator Server	管理服务器

HP PC	Windows XP SP3	IP: 192.168.0.4 PORT:8080	Proxy Server	代理服务器
HP PC	Windows XP SP3	IP: 192.168.0.4 PORT:7003	Managed Server	受管理服务器
HP PC	Windows XP SP3	IP: 192.168.0.4 PORT:7004	Managed Server	受管理服务器
HP PC	Windows XP SP3	IP: 192.168.0.4 PORT:7005	Managed Server	受管理服务器

使用 Configuration Wizard 进行配置

创建新的 Domain

选择 “「开始」菜单 → 程序 → Oracle WebLogic → WebLogic Server 11gR1 → Tools → Configuration Wizard”，单击 “下一步” 按钮



选择安装域源

选择安装“Basic WebLogic Server Domain”，单击“下一步”按钮



指定域的名称和位置

输入要创建的域名和位置，此处域名可输入为 `cluster_domain`，输入完成后单击“下一步”按钮



配置管理员用户名和密码

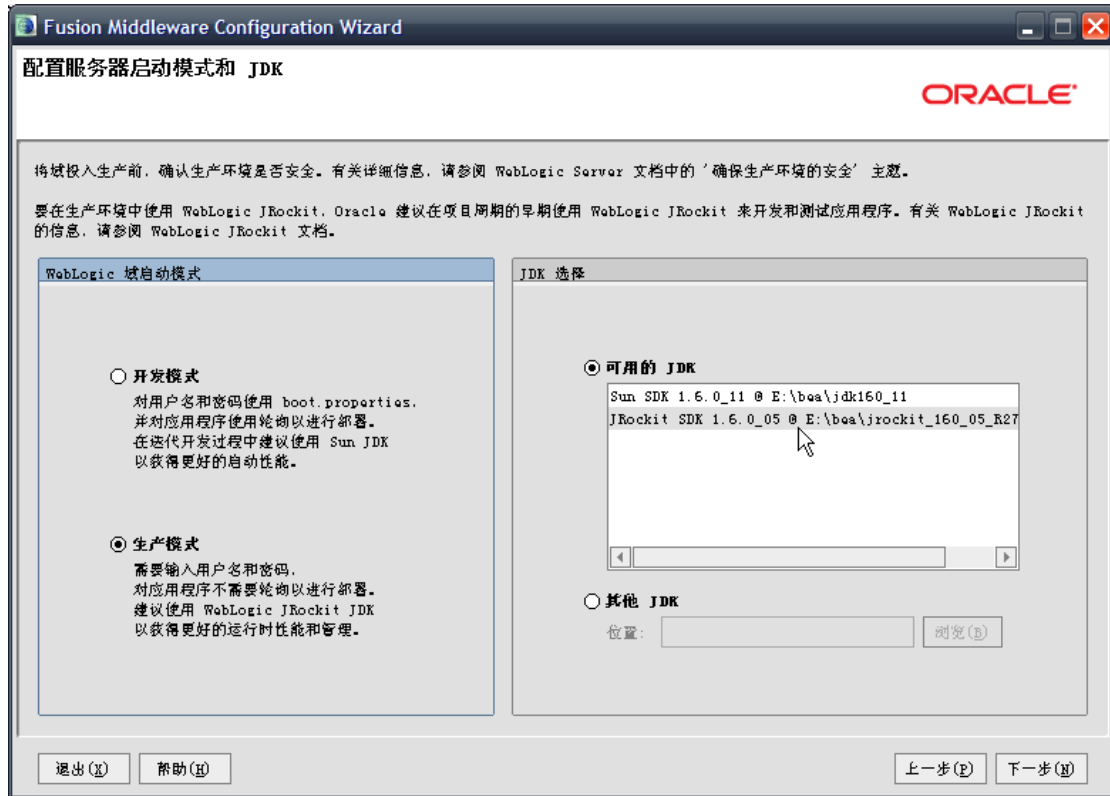
输入 WebLogic 管理员管理服务器 (AdminServer) 的用户名和密码，在此以 “weblogic” 作用登录管理服务器的用户名，密码为 “weblogic1”，单击 “下一步” 按钮

The image shows a screenshot of the 'Fusion Middleware Configuration Wizard' window. The title bar reads 'Fusion Middleware Configuration Wizard'. The main title is '配置管理员用户名和密码' (Configure Administrator Username and Password). The Oracle logo is in the top right corner. Below the title, there is a '放弃更改 (C)' (Discard Changes) button. The main area contains four input fields: '*User name:' with 'woblogic', '*User password:' with '*****', '*Confirm user password:' with '*****', and 'Description:' with 'This user is the default administrator.'. At the bottom, there are four buttons: '退出 (X)' (Exit), '帮助 (H)' (Help), '上一步 (P)' (Previous Step), and '下一步 (N)' (Next Step). A mouse cursor is pointing at the '下一步 (N)' button.

*User name:	woblogic
*User password:	*****
*Confirm user password:	*****
Description:	This user is the default administrator.

配置域启动模式和 JDK

选择“生产模式”和“JRockit SDK”后，单击“下一步”按钮



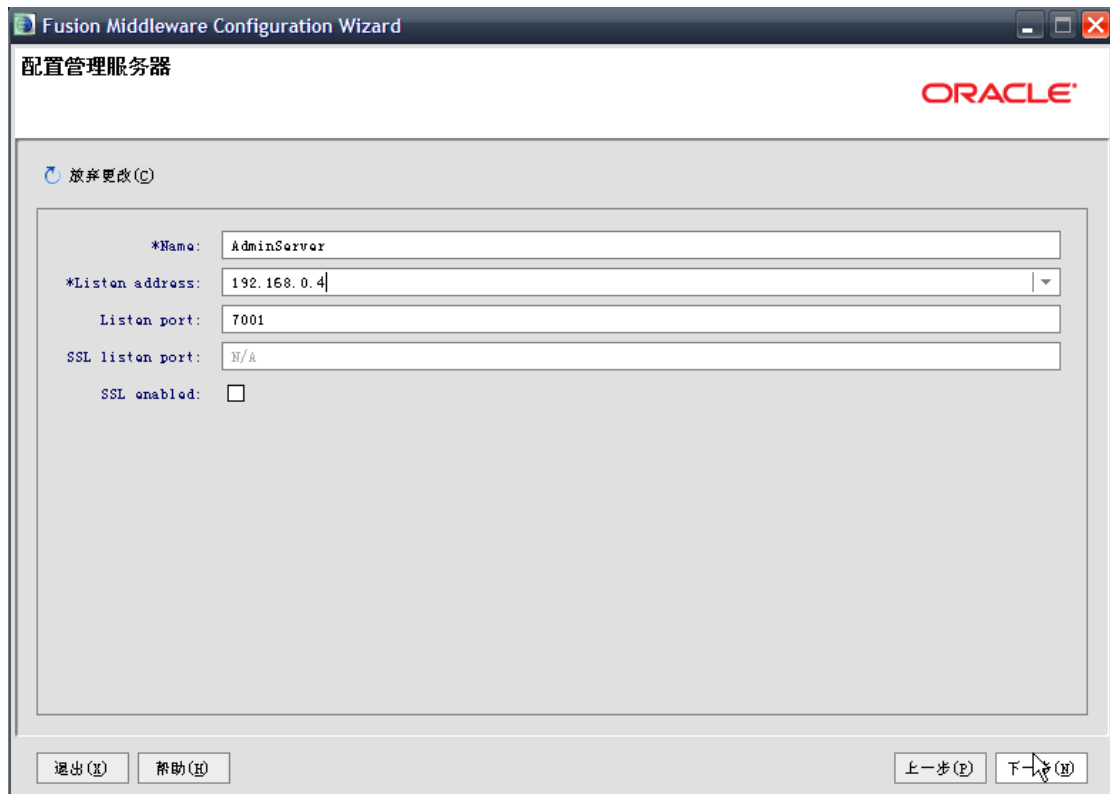
选择是否高级配置

选择“管理服务器”，“受管服务器、群集和计算机”后，单击“下一步”按钮



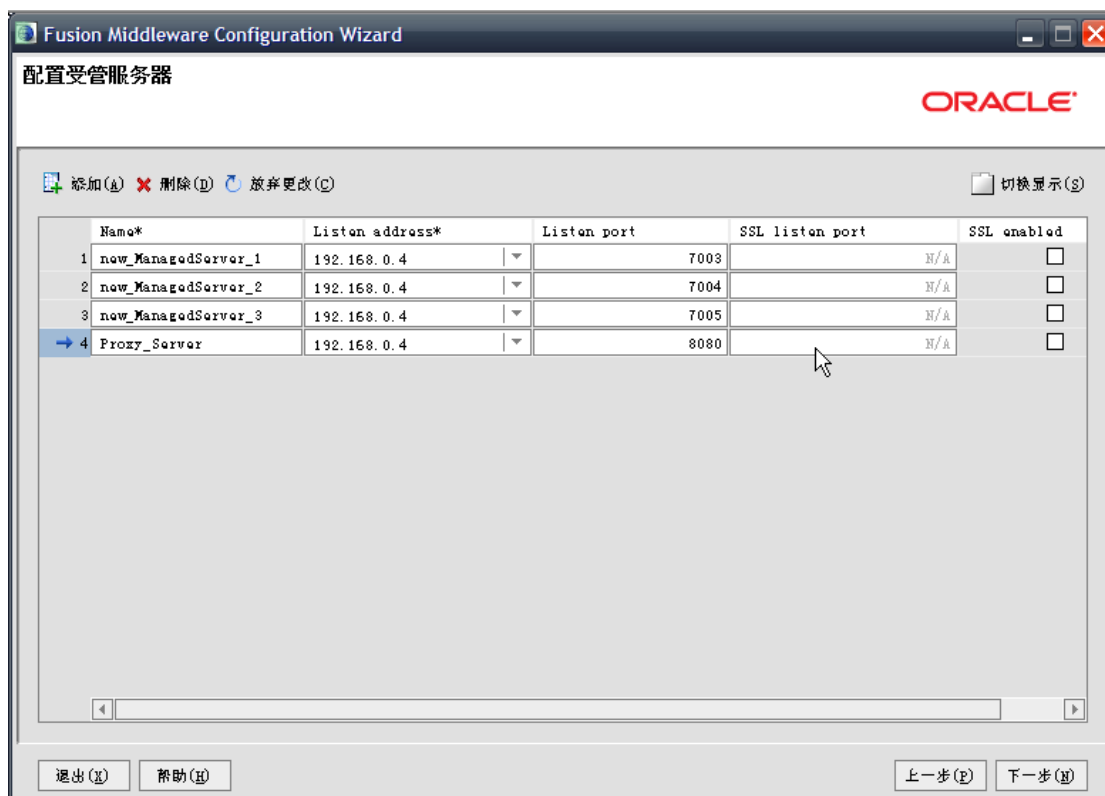
配置管理服务器

输入 Administrator Server 的名称，监听地址（可选所有 IP 或者指定 IP），监听端口，如果需要 SSL 支持的话可以在“SSL enabled”后面的复选取框上打勾，配置 SSL 监听端口，单击“下一步”按钮



配置受管理服务器（代理服务器实际上也是一个受管理服务器）

输入 Managed Server 的名称，监听地址，监听端口，如果需要 SSL 支持的话，可以在“SSL enabled”复选框上打勾，配置 SSL 监听端口。点击添加后可以配置多个 Managed Server，完成后单击“下一步”按钮。

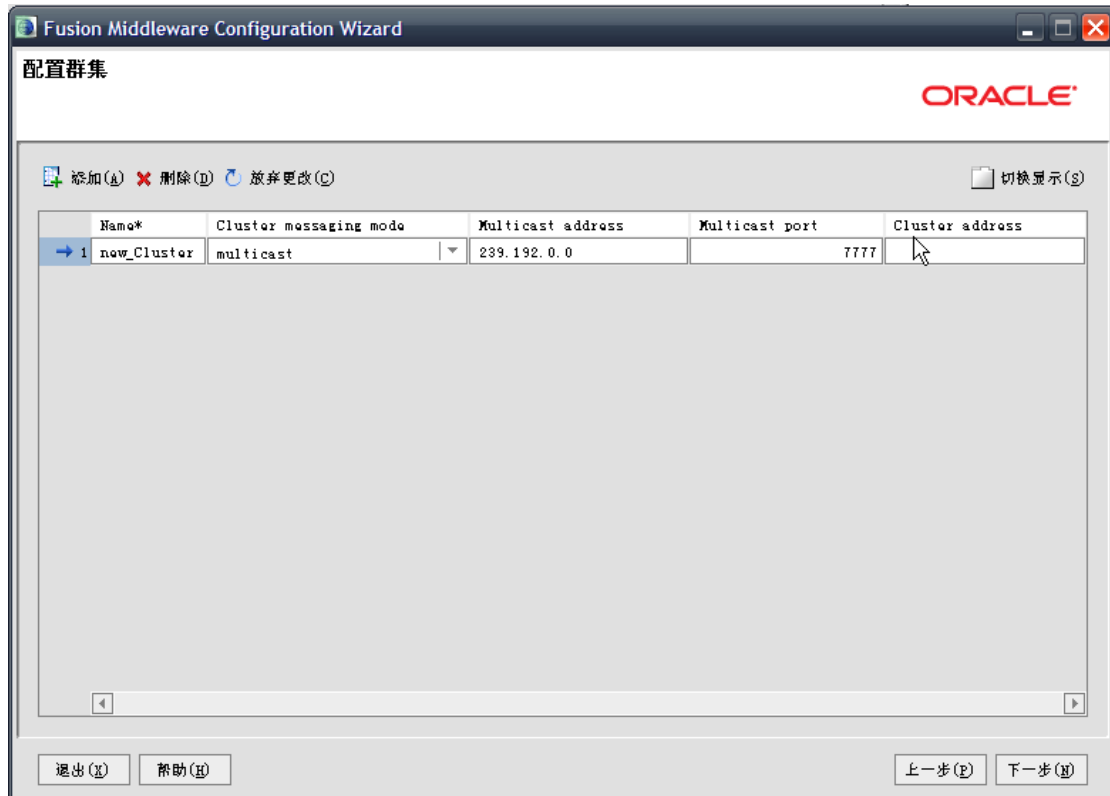


配置群集信息

单击“添加”，输入 Cluster 的名称 new_Cluster，Multicast address 组播地址（用来进行集群间的状态通信）和端口 7777，Cluster 的各个服务器地址和端口（Cluster 地址可输可不输，输入地址的例子：“192.168.0.4:7003,192.168.0.4:7004,192.168.0.4:7005”）。完成后单击“下一步”按钮。

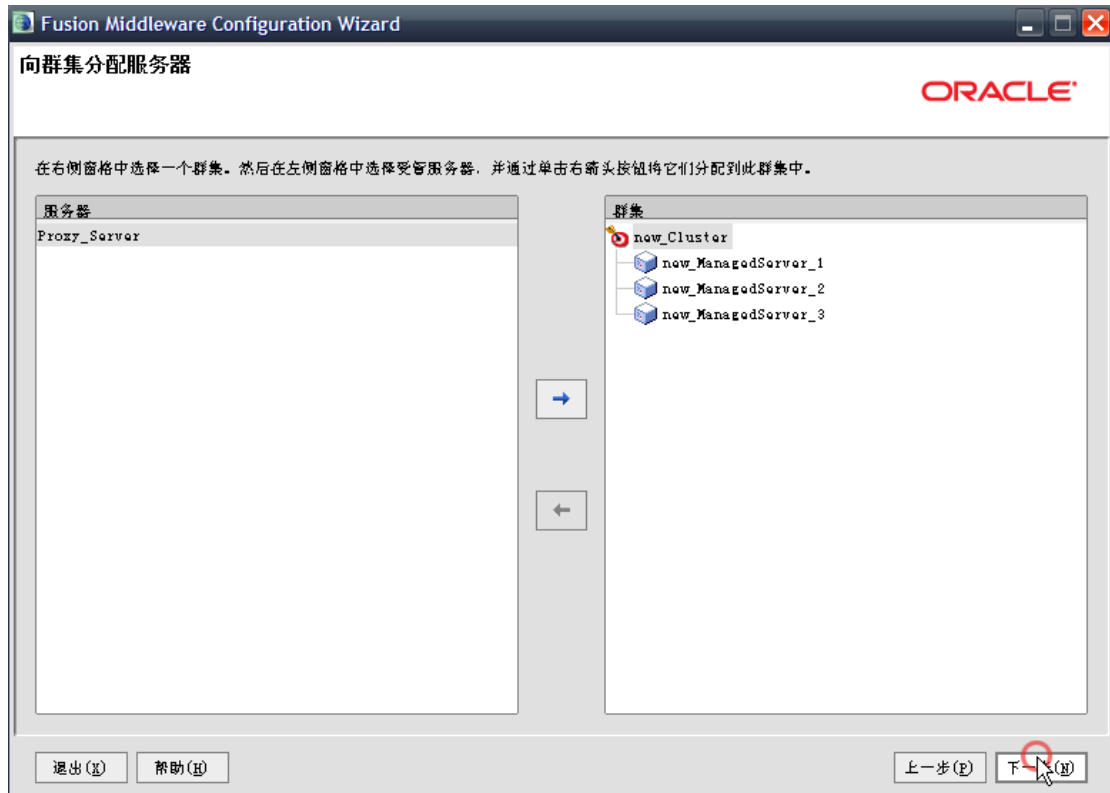
提示：当用 Configuration Wizard 创建集群时，您只能选择 multicast，虽然官方文档推荐新建集群时使用 unicast 消息机制；要选择 unicast，您只能在 Administration Console 中修改集群的配置，参考 Oracle Fusion Middleware Oracle WebLogic Server Administration Console 帮助文档中的：“Create and configure clusters”。

提示：multicast address 的 IP 范围必须位于 224.0.0.0 到 239.255.255.255 之间。WebLogic Server 使用的 multicast 默认值为 239.192.0.0。您不能使用类似于 x.0.0.1 的多播地址。



将“受管理服务器”添加到“集群”中

选择左面列表中的 Managed Server，将其添加到右面的列表中，注意这里代理服务器 Proxy_Server 不要添加到右边，之后单击“下一步”按钮。



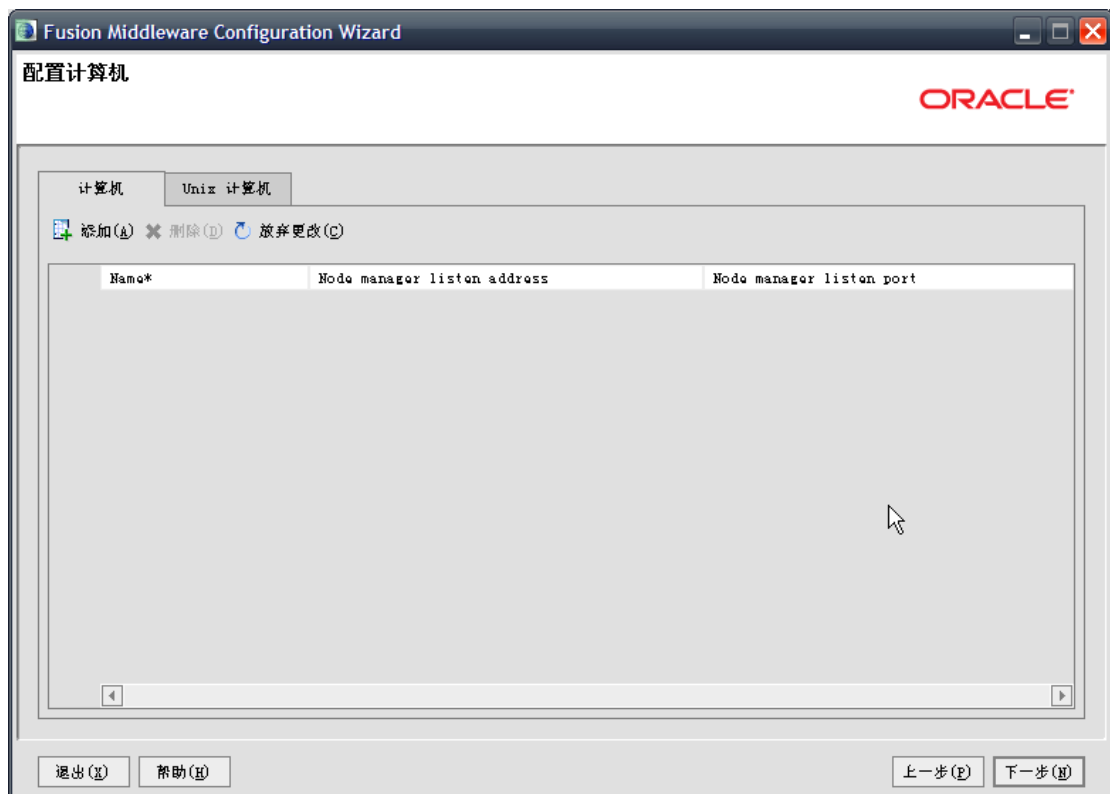
创建 HTTP 代理应用程序

选中“创建 HTTP 代理”复选框，new_Cluster 代理服务器选择前面没有加入 new_Cluster 的 Proxy_Server，单击“下一步”按钮。



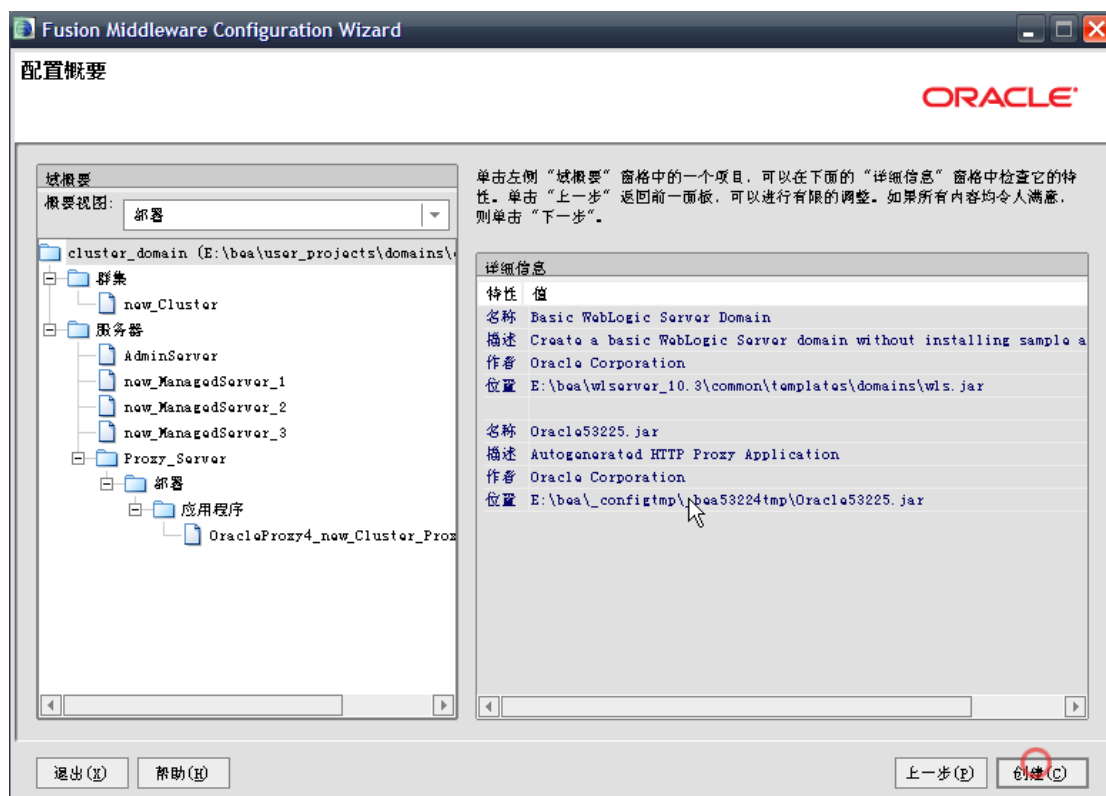
配置计算机

在此不对计算机进行配置，单击“下一步”按钮。



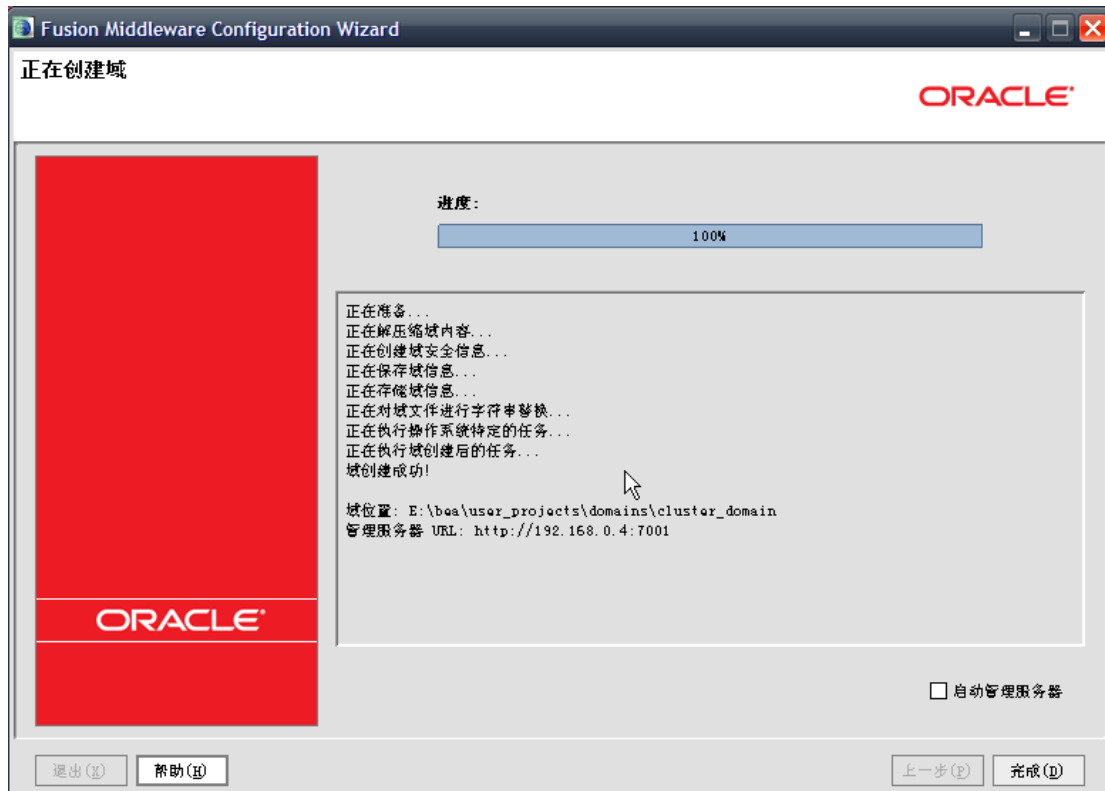
检查 WebLogic 域配置

在此检查前面所做的配置，如果没有问题单击“创建”按钮。



正在创建域

创建无误后，单击“完成”按钮



编辑 WebLogic 服务启动文件

➤ 编辑口令文件

在刚才创建的域目录下建立 **boot.properties** 文件，内容为：

```
username=weblogic  
password=weblogic1
```

在笔者的机器上，该文件的存储位置是：

E:\bea\user_projects\domains\cluster_domain\boot.properties。

➤ 编辑启动管理服务器的文件(可选，可直接运行 startWebLogic.cmd 即可)

在刚才创建的域目录下的 bin 目录中创建 startAdminServer.cmd，内容为：

```
@ECHO OFF  
@REM 启动管理服务器。  
start .\startWebLogic
```

在笔者的机器上，该文件的存储位置是：

E:\bea\user_projects\domains\cluster_domain\bin\startAdminServer.cmd。

➤ 编辑启动代理服务器的文件

在刚才创建的域目录下的 bin 目录中创建 startProxyServer.cmd，内容为：

```
@ECHO OFF
```

```
@REM 启动代理服务器。
```

```
@REM 使用 startManagedWebLogic 命令启动 Managed Server，它的命令格式是：
```

```
@REM startManagedWeblogic.cmd Managed Server 的名称 Administrator Server 地址
@REM startManagedWeblogic 受管理服务器名称 管理服务器 URL
```

```
start .\startManagedWebLogic Proxy_Server http://192.168.0.4:7001
```

在笔者的机器上，该文件的存储位置是：

E:\bea\user_projects\domains\cluster_domain\bin\startProxyServer.cmd。

- 编辑启动受管理服务器(ManagedServer_1)的文件

在刚才创建的域目录下的 bin 目录中创建 startManagedServer_1.cmd，内容为：

```
@ECHO OFF
```

```
@REM 启动受管理服务 new_ManagedServer_1.
```

```
@REM 使用 startManagedWeblogic 命令启动 Managed Server，它的命令格式是：
```

```
@REM startManagedWeblogic.cmd Managed Server 的名称 Administrator Server 地址
```

```
@REM startManagedWeblogic 受管理服务器名称 管理服务器 URL
```

```
start .\startManagedWebLogic new_ManagedServer_1 http://192.168.0.4:7001
```

在笔者的机器上，该文件的存储位置是：

E:\bea\user_projects\domains\cluster_domain\bin\startManagedServer_1.cmd。

- 编辑启动代理服务器(ManagedServer_2)的文件

在刚才创建的域目录下的 bin 目录中创建 startManagedServer_2.cmd，内容为：

```
@ECHO OFF
```

```
@REM 启动受管理服务 new_ManagedServer_2.
```

```
@REM 使用 startManagedWeblogic 命令启动 Managed Server，它的命令格式是：
```

```
@REM startManagedWeblogic.cmd Managed Server 的名称 Administrator Server 地址
```

```
@REM startManagedWeblogic 受管理服务器名称 管理服务器 URL
```

```
start .\startManagedWebLogic new_ManagedServer_2 http://192.168.0.4:7001
```

在笔者的机器上，该文件的存储位置是：

E:\bea\user_projects\domains\cluster_domain\bin\startManagedServer_2.cmd。

- 编辑启动代理服务器(ManagedServer_3)的文件

在刚才创建的域目录下的 bin 目录中创建 startManagedServer_3.cmd，内容为：

```
@ECHO OFF
```

```
@REM 启动受管理服务 new_ManagedServer_3.
```

```
@REM 使用 startManagedWeblogic 命令启动 Managed Server，它的命令格式是：
```

```
@REM startManagedWeblogic.cmd Managed Server 的名称 Administrator Server 地址
```

```
@REM startManagedWeblogic 受管理服务器名称 管理服务器 URL
```

```
start .\startManagedWebLogic new_ManagedServer_3 http://192.168.0.4:7001
```

在笔者的机器上，该文件的存储位置是：

E:\bea\user_projects\domains\cluster_domain\bin\startManagedServer_3.cmd。

(注：使用 startManagedWebLogic 命令启动 Managed Server，它的命令格式是：
startManagedWebLogic.cmd Managed Server 的名称 Administrator Server 地址)

启动 WebLogic 服务

建议您按照顺序启动服务器，只有当管理服务器首先启动完毕后，受管服务器才能启动完成。当所有服务器的控制台上出现 Server started in RUNNING mode 字样时，整个集群启动完毕。

- 双击 “startAdminServer.cmd” 或者 “startWebLogic.cmd” 启动管理服务器
- 双击 “startProxyServer.cmd” 启动代理服务器
- 双击 “startManagedServer_1.cmd” 启动受管理服务器 1
- 双击 “startManagedServer_2.cmd” 启动受管理服务器 2
- 双击 “startManagedServer_3.cmd” 启动受管理服务器 3

开发 Web 应用 DefaultWebApp

提示：在此 PDF 的附件栏，已经附带了 **DefaultWebApp.war** 包，您可使用 WinZIP、WinRAR 等软件对其进行解压缩和修改。

DefaultWebApp/test.jsp

```
<%
    out.println("OK");
    System.out.println("OK");
%>
```

DefaultWebApp/WEB-INF/web.xml

```
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">
<web-app>
    <servlet>
        <servlet-name>servlet1</servlet-name>
        <servlet-class>servlet.Servlet1</servlet-class>
    </servlet>

    <servlet-mapping>
        <servlet-name>servlet1</servlet-name>
        <url-pattern>/servlet1.do</url-pattern>
    </servlet-mapping>

    <welcome-file-list>
        <welcome-file>index.html</welcome-file>
    </welcome-file-list>
```

```
</web-app>
```

DefaultWebApp/WEB-INF/weblogic.xml

```
<!DOCTYPE weblogic-web-app PUBLIC "-//BEA  
Systems, Inc.//DTD Web Application 8.1//EN"  
"http://www.bea.com/servers/wls810/dtd/weblogic810-web-jar.dtd">  
  
<weblogic-web-app>  
    <context-root>/</context-root>  
</weblogic-web-app>
```

DefaultWebApp/WEB-INF/classes/servlet/ Servlet1.java

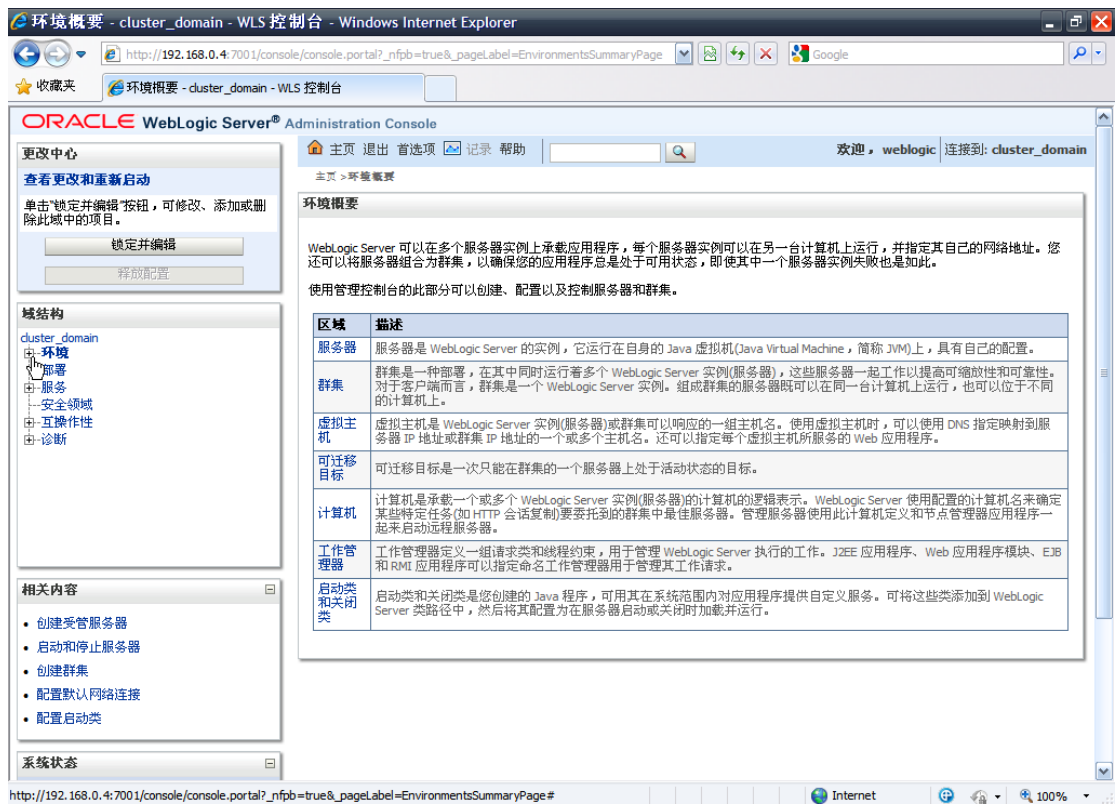
```
package servlet;  
  
import javax.servlet.*;  
import javax.servlet.http.*;  
import java.io.*;  
import java.util.*;  
  
public class Servlet1 extends HttpServlet {  
    private static final String CONTENT_TYPE = "text/html; charset=GBK";  
    //Initialize global variables  
    public void init() throws ServletException {  
    }  
    //Process the HTTP Get request  
    public void doGet(HttpServletRequest request, HttpServletResponse response) throws  
ServletException, IOException {  
        response.setContentType(CONTENT_TYPE);  
        PrintWriter out = response.getWriter();  
        out.println("<html>");  
        out.println("<head><title>Servlet1</title></head>");  
        out.println("<body bgcolor=\"#ffffff\">");  
        out.println("<p>The servlet has received a GET. This is the reply.</p>");  
        out.println("</body></html>");  
    }  
    //Process the HTTP Post request  
    public void doPost(HttpServletRequest request, HttpServletResponse response) throws  
ServletException, IOException {  
        response.setContentType(CONTENT_TYPE);  
        PrintWriter out = response.getWriter();  
        String username = (String)request.getParameter("username");  
        out.println("<html>");  
        out.println("<head><title>Servlet1</title></head>");  
        out.println("<body bgcolor=\"#ffffff\">");  
        out.println("<p>username : " + username + "</p>");  
    }  
}
```

```
out.println("<p>The servlet has received a POST. This is the reply.</p>");
out.println("</body></html>");
}
//Clean up resources
public void destroy() {
}
}
```

部署 WEB 应用

进入 Console 控制台（<http://192.168.0.4:7001/console>），使用管理员账户 weblogic/weblogic1 登录。

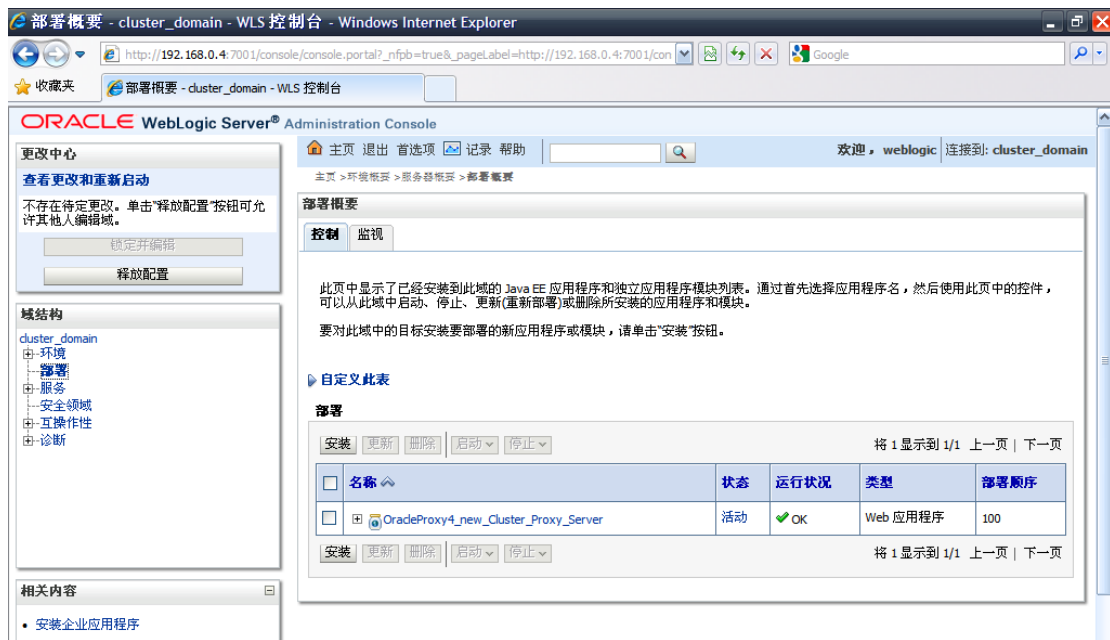
进入 console 控制台后，在左边域结构树中点击“环境”前的“+”，如下图所示：



在展开的菜单中点击“服务器”，在右边的列表中可以看到各服务器是否正常启动。



如果正常启动则单击左侧域结构下的“部署”，再单击“锁定并编辑”，



再单击右侧的“安装”按钮, 选择要部署的 WEB 应用程序, 笔者这里在“路径”输入框中输入了 Web 应用所在目录:



完成后，单击“下一步”，再单击“下一步”，



选择群集“new_Cluster”，再选择“群集中的所有服务器”，之后单击“下一步”，再单击“下一步”，再单击“完成”。最后再单击左上角的“激活更改”。

启动 WEB 应用

进入 Console 控制台（<http://192.168.0.4:7001/console>），使用管理员账户 weblogic/weblogic1 登录。

进入 console 控制台后，点击左侧的域结构下的“部署”，接着单击选中部署列表中的应用（这里是 DefaultWebApp），然后单击“启动”按钮下的菜单项“为所有请求提供服务”，



，接下来在出现的启动部署页再单击“是”按钮。



当看到 DefaultWebApp 右侧的状态从“准备就绪”转换为“活动”时就表示 WEB 应用启动完成。

测试集群的分发功能

编写测试 WEB 应用并部署

我们来编写一个简单的 WEB 应用，它会在控制台和浏览器上同时打印出“OK”字样，然后将这个 WEB 应用部署到集群中所有 Managed Server 上面。（代码见附件）

在这里我们将通过 Apache 中所带的 ab 包（只需要单个文件 ab.exe 即可）来进行并发访问的模拟测试，使用如下的命令就可以完成压力测试。

```
ab.exe -n 100 -c 10 http://192.168.0.4:8080/test.jsp
```

```
pause
```

ab 是测试程序的名称

参数 n 代表请求的总数量

参数 c 代表并发的请求数

url 为要测试压力的页面

注：使用这个命令时，一定要在系统路径中能够找到该程序，否则不能执行。

压力测试完成后，我们从 Managed Server 的控制台上可以看到，ManagedServer_1，ManagedServer_2，ManagedServer_3 都打印出了“OK”字样，这说明，在并发请求的情况下，集群能够将请求进行分发，以达到负载均衡的目的。

注：单独通过浏览器输入地址 <http://192.168.0.4:8080/test.jsp> 然后刷新，您可能发现服务器一直将请求转发到单独的一台受管服务器上，这是因为连接数过少所致，并非服务器集群配置失败；您可以停掉这台服务器，可以看到客户端访问仍能正常进行，而访问则转发到了另外一台服务器上。

Session 共享问题解决

编辑 **weblogic.xml** 配置文件放到 WEB 应用程序的 WEB-INF 目录下，内容如下：

```
<?xml version="1.0" encoding="UTF-8"?>
<wls:weblogic-web-app          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wls="http://www.bea.com/ns/weblogic/90"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd          http://www.bea.com/ns/weblogic/90
http://www.bea.com/ns/weblogic/90/weblogic-web-app.xsd">
<session-descriptor>
<persistent-store-type>replicated</persistent-store-type>
<sharing-enabled>true</sharing-enabled>
</session-descriptor>
<wls:context-root></wls:context-root>
</wls:weblogic-web-app>
```

集群配置中要注意的问题

- Admin Server 只用于集群的管理，而不能参与集群事务。
- Web 应用应该部署到集群上（不能部署到 AdminServer 和 Proxy_Server 上），文件的同步是由 WebLogic 来完成的。

附录

WebLogic Server 11gR1 文档 http://download.oracle.com/docs/cd/E12839_01/wls.htm

WebLogic 9.2 中文文档 <http://edocs.weblogicfans.net/wls/docs92/index.html>

WebLogic 中文爱好者 <http://weblogicfans.net/>

PDF 中所附带的附件：

- boot.properties
- 集群启动脚本.zip

- DefaultWebApp.war
- 并发测试.zip