

SolrWebSphere

Solr with IBM WebSphere

Solr doesn't run out of the box on IBM [WebSphere](#) Application Server. The minimum version for Running Solar is WAS 6.1.x, cause WAS 6.0 doesn't support Java 5. This page is just about [WebSphere](#) specific instructions, see the instructions in the generic [Solr installation](#) page for general info before consulting this page.

- [Solr with IBM WebSphere](#)
 - [Required Modifications for WAS 6.1](#)
 - [Enabling Java 5 for JSP pages](#)
 - [Adding index.jsp to Welcome File List](#)
 - [Working with the Admin Console and Multicore](#)
 - [Enable remote JMX monitoring on WebSphere and register Solr MBeans](#)

Required Modifications for WAS 6.1

The following instructions are the required modifications for the apache-solr-1.3.0.war on [WebSphere](#) 6.1. This modifications should work with all versions of the the WAS 6.1 series.

Enabling Java 5 for JSP pages

Cause [WebSphere](#) 6.x is a J2EE 1.4 container, it doesn't support Java 5 out of the box. You have to explicit enable Java 5 for the JSP compiler. To do this, you've got to create (or modify) 2 files:

- WEB-INF/ibm-web-bnd.xmi
- WEB-INF/ibm-web-ext.xmi

ibm-web-bnd.xmi

```
<?xml version="1.0" encoding="UTF-8"?>
<webappbnd:WebAppBinding xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:webappbnd="webappbnd.xmi"
xmi:id="WebAppBinding_1213147492785" virtualHostName="default_host">
  <webapp href="WEB-INF/web.xml#WebApp_ID" />
</webappbnd:WebAppBinding>
```

ibm-web-ext.xmi

```
<?xml version="1.0" encoding="UTF-8"?>
<webappext:WebAppExtension xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI" xmlns:webappext="webappext.xmi"
xmi:id="WebAppExtension_1213147492785" reloadInterval="3" reloadingEnabled="true" additionalClassPath=""
fileServingEnabled="true" directoryBrowsingEnabled="false" serveServletsByClassnameEnabled="true">
  <webApp href="WEB-INF/web.xml#WebApp_ID" />
  <jspAttributes xmi:id="JSPAttribute_1" name="jdkSourceLevel" value="15" />
</webappext:WebAppExtension>
```

Adding index.jsp to Welcome File List

In the apache-solr-1.3.0.war there is now welcome file section. [WebSphere](#) doesn't handle index.jsp as *auto* welcome file, so you've got to modify the web.xml to get a page displayed. Add the 'welcome-file' section add the bottom of the web.xml file.

web.xml

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

Working with the Admin Console and Multicore

If you're having trouble using the various pages on the admin console when specifying the core name in your URL path you may see errors around [WebSp here](#) not finding the JSPs. This could be related to a known issue with Websphere not running filters when a static file is not found on the expected path.

PK33090 <http://www-01.ibm.com/support/docview.wss?uid=swg24014758>

The [SolrDispatchFilter](#) removes the core name from the path when forwarding to the correct JSPs. To force Websphere to run the filter before returning a 404 add the custom property from the linked issue above to the app server running your Solr instance.

Enable remote JMX monitoring on [WebSphere](#) and register Solr MBeans

When you try to access Solr on [WebSphere](#) (with `<jmx/>` config enabled in `solrconfig.xml`) it fails to register beans (query, facet, etc) and throws "javax.management.InstanceAlreadyExistsException". To resolve this you need to add following System property to your server (Application servers > server > Process definition > Java Virtual Machine > Custom properties)

```
javax.management.builder.initial=javax.management.MBeanServerBuilder
```

Restart the server and now [WebSphere](#) should not throw exception. To confirm that beans have been registered successfully make sure that you have following line in your logs

```
org.apache.solr.core.JmxMonitoredMap <init> JMX monitoring is enabled. Adding Solr mbeans to JMX Server:
```

Once you have the beans registered you can connect and access Solr configurations and statistics via JMX remotely by adding following System properties to server.

```
com.sun.management.jmxremote.port="any unused port number"  
com.sun.management.jmxremote.authenticate=false  
com.sun.management.jmxremote.ssl=false
```

Restart the server and now you can use tool like JConsole to connect and access Solr configurations and statistics remotely via JMX by specifying the hostname and port.