

# Deploying on Solaris (WO 5.3.3) (DELETABLE)

Setup WebObjects 5.3.3 on Solaris 10 x86

Rev 1 1.01.2008

Johann Werner (werner at isd dot uni-stuttgart dot de)

based on description of Francois Bientz for setup on FreeBSD

Rev 2 28.04.2008

John Bruce (john at kinrossgroup dot com)

added initial SMF scripts for wotaskd & javamonitor

That's the very first time I am using Solaris so I don't guarantee that my instructions are complete nor comply to standard Solaris naming/best-practice /whatsoever conventions. Feel free to comment/correct it!

What I used to test this setup:

Parallels Desktop

Solaris 10 8/07 x86 (Entire Solaris Software Group installation)

Access to an OS X Server 10.4.x machine that has the WO 5.3.3 files

## WebObjects 5.3.3 installation

This section discusses how you "install" WO 5.3.3 on Solaris to use it as pure application server. You will be able to connect to WO apps by direct connect or use that machine from another machine that runs Apache with WOADaptor.

Under Solaris create the following directory where all WO related files will be contained:

```
/opt/Apple
```

Inside of this directory create:

```
/opt/Apple/Logs  
/opt/Apple/Library/Frameworks  
/opt/Apple/Library/PrivateFrameworks  
/opt/Apple/Library/WebObjects  
/opt/Apple/Local/Library/WebObjects/Applications  
/opt/Apple/Local/Library/WebObjects/Extensions
```

Then copy from the OS X Server machine these files from /System/Library/Frameworks

```
JavaDirectToWeb.framework
JavaDTWGeneration.framework
JavaEOAccess.framework
JavaEOApplication.framework
JavaEOCocoa.framework
JavaEOControl.framework
JavaEODistribution.framework
JavaEOGeneration.framework
JavaEOInterface.framework
JavaEOInterfaceCocoa.framework
JavaEOInterfaceSwing.framework
JavaEOProject.framework
JavaEORuleSystem.framework
JavaEOTool.framework
JavaFoundation.framework
JavaJDBCAdaptor.framework
JavaJNDIAdaptor.framework
JavaWebObjects.framework
JavaWebServicesClient.framework
JavaWebServicesGeneration.framework
JavaWebServicesSupport.framework
JavaWOExtensions.framework
JavaWOJSPServlet.framework
JavaXML.framework
```

to /opt/Apple/Library/Frameworks

Next copy from the OS X Server machine these files from /System/Library/PrivateFrameworks

```
EOPlaceholders.framework
JavaMonitor.framework
```

to /opt/Apple/Library/PrivateFrameworks

Copy from the OS X Server machine these files from /System/Library/WebObjects

```
JavaApplications
WODocumentRoot
```

to /opt/Apple/Library/WebObjects

Finally you have to copy these files from /Library/WebObjects/Extensions

```
axis-ant.jar
axis.jar
commons-discovery.jar
commons-logging.jar
jaxrpc.jar
log4j-1.2.8.jar
saa.jar
wsdl4j.jar
```

to /opt/Apple/Local/Library/WebObjects/Extensions

## Create startup scripts for wotaskd and javamonitor

To be able to use the machine as an application server wotaskd has to run. To automatically start wotaskd on startup create the file /etc/rc3.d/S110.wotaskd

```
#!/bin/sh

NEXT_ROOT=/opt/Apple

case "$1" in
    start)
        export NEXT_ROOT
        ${NEXT_ROOT}/Library/WebObjects/JavaApplications/wotaskd.woa/wotaskd &
        echo -n ' wotaskd'

        ;;
    stop)
        lsof -i tcp:1085 |grep LISTEN | sort -u | awk '{print"kill -9 " $2}' | sh
        ;;
    *)
        echo ""
        echo "Usage: `basename $0` { start | stop }"
        echo ""
        exit 64
        ;;
esac
```

Then

```
chmod +x /etc/rc3.d/S110.wotaskd
```

If you need to start JavaMonitor too on that machine (remember: only **one** JavaMonitor per subnet) create /etc/rc3.d/S150.javamonitor

```
#!/bin/sh

NEXT_ROOT=/opt/Apple

case "$1" in
    start)
        export NEXT_ROOT
        ${NEXT_ROOT}/Library/WebObjects/JavaApplications/JavaMonitor.woa/JavaMonitor -WOPort 56789 &
        echo -n 'JavaMonitor'

        ;;
    stop)
        lsof -i tcp:56789 |grep LISTEN | sort -u |awk '{print"kill -9 "$2}' | sh
        ;;
    *)
        echo ""
        echo "Usage: `basename $0` { start | stop }"
        echo ""
        exit 64
        ;;
esac
```

Then

```
chmod +x /etc/rc3.d/S150.javamonitor
```

To be done: Starting wotaskd and JavaMonitor under a specific user/group. As it is shown here the two apps will run with root privileges. Another approach would be to put them into a Solaris Zone.

Also to be done: Instead of using init.d starting these services by SMF.

## Using SMF To Manage wotaskd & JavaMonitor

If you are running WO in a non global Zone the **lsof** command will not work as the `/devices` directory is missing. An alternative to the above scripts are the [wotaskd.smf](#) and [javamonitor.smf](#) smf scripts attached to this page. I've tried them on a test server and they work for me but I'm a total SMF novice so **use these files at your own risk**.

Import the files with:

```
sudo svccfg import wotaskd.smf
sudo svccfg import javamonitor.smf
```

They should start the wotaskd and javamonitor services but if not they can be enabled with:

```
sudo svcadm enable svc:/webobjects/wotaskd:default
sudo svcadm enable svc:/webobjects/javamonitor:default
```

To confirm they are working kill the running java processes and they should restart. Also you can check the logs usually in `/var/svc/logs`.

## Installing JDBC drivers

If you want to access a database from your WO apps you have to install appropriate JDBC drivers. In this guide I will use FrontBase drivers as an example.

Download the database specific drivers from [www.frontbase.com](http://www.frontbase.com) (called "WebObjects 5 Plugin" on the download page) and extract the archive. Then you should have two files:

```
FrontBasePlugIn.framework
frontbasejdbc.jar
```

The first file is the one you have to add to your WO project. If you have WOnder you can exchange it with the one that comes with WOnder as it is an improved version (though I did not check for differences 😊). Put that file into

```
/opt/Apple/Local/Library/Frameworks
```

The second file has to be put into the extensions directory of java

```
/usr/java/jre/lib/ext
```

Reboot and you should be able to connect to a FB database from WO.

## Compiling WOAdaptor and Setting up Apache

On Solaris 10 8/07 there is preinstalled Apache in versions 1.3 and 2.0. In this how-to we will compile an adaptor for Apache 2.0. It took me quite long to get the compile process right on Solaris so I changed some things and won't refer to the adaptor files in WOnder CVS. Instead get this [SolarisAdaptor archive](#).

Put it e.g. into

```
/tmp
```

Then unzip it with

```
unzip SolarisAdaptor.zip
```

Next type in these commands

```
cd SolarisAdaptor
./makeAdaptor.sh
```

The file makeAdaptor.sh contains:

```
#!/bin/sh
/usr/apache2/bin/apxs -c -i -a -DSINGLE_THREADED_ADAPTOR mod_WebObjects.c appcfg.c cfgparse.c config.c
hostlookup.c
list.c listing.c loadaverage.c loadbalancing.c log.c MoreURLCUtilities.c nbsocket.c PB.project Platform.c
random.c
request.c response.c roundrobin.c shmем.c strdict.c strtbl.c transaction.c transport.c wastring.c womalloc.c
WOURLCUtilities_3.c WOURLCUtilities.c xmlparser.c xmltokenizer.c xmlparse.c
```

It uses the apache extension tool to compile all source files, links them together and installs them in the right location for apache 2. It even inserts the LoadModule directive into your httpd.conf file! What a luxury 😊

You notice the "-DSINGLE\_THREADED\_ADAPTOR". You can change it to -DMULTIPLE\_THREADED\_ADAPTOR though I don't know if it works and exactly what advantages you will get from (besides having multiple threads).

The next step is to put the WO config file into the right location:

```
cp webobjects.conf /etc/apache2/
```

That config file has been altered in three ways:

- the LoadModule directive is commented out as it was automatically added to the main config file by apxs
- the WebObjectsDocumentRoot points to the standard apache2 location /var/apache2/htdocs  
Change it if you have moved your htdocs to another place
- the WebObjectsAlias has been changed to /Apps/WebObjects  
Feel free to modify the alias to any string you want. It has to be different from "cgi-bin" as it does not work. I didn't investigate on that. Note that you have to set the appropriate setting in JavaMonitor too. Go to JavaMonitor and click on the "Site" tab. On top there is a text field "URL to Adaptor". Put in the URL of your webserver with the changed WO alias.

Then lastly add as the last line of your current apache configuration file:

```
Include /etc/apache2/webobjects.conf
```

**Please note that I don't guarantee that this adaptor works as intended. Use only at your own risk.**

## Running Apache

The very last step is to start apache. On Solaris use the command:

```
svcadm enable apache2
```

And check if it is really running

```
svcs -x apache2
```