Axis C++ Linux Installation Guide

<!-- --> <!-- --> .command {color: #B20000; text-indent: 60pt; font-family: helvetica; font-weight: bold} <!-- -->

1. Installing and deploying web applications using xml-AxisC++

2. Contents

3. Introduction

This document describes how to install Apache Axis C++. It assumes you already know how to write and run C++/C code and not scared away by XML. You should also have familiarity with Web servers. This version of Axis C++ is tested on Apache web server.

4. What You need

It needs a few helper libraries, for logging, WSDL processing and introspection. You need to have following in order to run Axis C++ engine.

- Apache 1.3 or 2
- expat (if use expat parser)
- Xerces-c (if use xerces parser)

5. Installation On Linux

We tested with the following

- RedhatLinux9.0(2.4.20-8), Dabian(2.6.3-1-686)(unstable), Redhat Linux8.0(2.4.18-14)

- Apache 1.3 [apache_1.3.27.tar .gz, apache_1.3.28.tar.gz)] (Source), Apache2.0 (source)
- expat [expat-1.95.7.tar.gz](Source)

Note

- autoconf 2.57, automake 1.6.3, libtool 1.4.3

Downloading the source or binary from the mirror site and setting the environment variables

You can download the Axis C++ source or binary from one of the apache mirror siteshttp://ws.apache.org/axis/cpp/download.html

Your downloaded source distribution is **axis-c-src-1-1-linux.tar.gz**.
Binary distribution is **axis-c-1-1-linux.tar.gz.**
After you extracting it as, **/home/axisuser/projects/ axis-c-src-1-1-linux** or **/home/axisuser/projects/ axis-c-1-1-linux** you can rename it as **axis_c**

You have to set the environment variable $AXISCPP_HOME to the directory where you extracted the tar ball.. We further assume that the user dose the installation and has the linux user account axisuser. For example I have my <AXISCPP_HOME> as following.

/home/axisuser/projects/axis_c

6. Installing expat

You can get expat from the uri http://sourceforge.net/projects/expat/

You have to follow the expat installation guide to install it

7. Installing Xerces-c

You can get xercesc from the uri http://www.xml.apache.org/xerces-c/download.cgi
You have to follow the the xercesc installation giude to install it.

8. Installing Apache

Get the apache downloadable software. (We used the source apache_1.3.27.tar.gz). Build it with sharable module support.

```
$ ./configure --enable-module=so
```

Note:- Here "so" is simple letters

```
$ make
$ make install
```

Starting the Apache server

```
$ usr/local/apache/bin/apachectl start
```

Stopping the Apache Server

```
$ usr/local/apache/bin/apachectl stop
```

9. Installing Axis C++

If you downloaded the source distribution

Create an environment variable called AXISCPP_HOME.

```
$ cd /home/axisuser
```

```
[axisuser@localhost axisuser]$ vi ./.bash_profile
```

```
AXISCPP_HOME="/home/axisuser/projects/axis_c"
```

```
export AXISCPP_HOME
```

Save it and back in the terminal window.

```
$ source ~/.bash_profile
```

Verify the above change had been correctly effected by typing

```
$ echo $AXISCPP_HOME
```

Copy apache include files

```
$ cp -f <apacheinstdirectory>/include/* $AXISCPP_HOME/include/apache1_3/
```

If you are using expat parser copy the expat include files

```
$ cp -f <expatinstdirectory>/include/expat.h $AXISCPP_HOME/include/expat/
```

If you are using xercesc parser copy the xercesc include files

```
$ cp -rf <xercescinstdirectory>/include/xercesc/* $AXISCPP_HOME/include/xercesc/
```

```
$ cd $AXISCPP_HOME
```

The Folder called deploy in the $AXISCPP_HOME/ should be copied to your place of choice. Rename the deploy folder as "Axis" . Give all permissions to this folder.

```
$ cp -rf $AXISCPP_HOME/deploy  /usr/local
$ cd /usr/local
$ mv deploy Axis
$ chmod -R 777 Axis
```

Now set the environment variable **AXIS_HOME** pointing to this directory.
AXIS_HOME="/usr/local/Axis"

You also need to rename the following files

```
mv $AXIS_HOME/axiscpp.conf_linux $AXIS_HOME/axiscpp.conf
mv $AXIS_HOME/conf/server.wsdd_linux $AXIS_HOME/conf/server.wsdd
mv $AXIS_HOME/conf/client.wsdd_linux $AXIS_HOME/conf/client.wsdd
```

If you are using expat parser do the following

set **EXPAT_HOME**="<Your expat installation root folder>"

set
LD_LIBRARY_PATH="$LD_LIBRARY_PATH:$EXPAT_HOME/lib:$AXISCPP_HOME/bin:$AXI
(do this in your .bash_profile)

```
cd $AXISCPP_HOME/src/soap
cp -f Makefile.am_expat Makefile.am
cd $AXISCPP_HOME/src/wsdd
cp -f Makefile.am_expat Makefile.am
```

If you are using Xerces-c parser do the following

set **XERCESC_HOME**="<Your xercesc installation root folder>"

set
LD_LIBRARY_PATH="$XERCESC_HOME/lib:$AXISCPP_HOME/bin:$AXIS_HOME"
(do this in your .bash_profile)

```
cd $AXISCPP_HOME/src/soap
cp -f Makefile.am_xercesc Makefile.am
cd $AXISCPP_HOME/src/wsdd
cp -f Makefile.am_xercesc Makefile.am
```

in $AXISCPP_HOME/configure.ac comment expat related things and uncomment xercesc related things as described in it.
for samples

in $AXISCPP_HOME/samples/client/configure.ac comment expat related things and uncomment xercesc related
things as described in it.

To Build from the Axis C++ source execute these.

```
$cd $AXISCPP_HOME/

$ sh autogen.sh

$ sh runconfig

$ make

$ make install
```

OR Run the build.sh as follows

```
sh build.sh
```

libaxiscpp_mod.so (If you built for apache2 this is libaxiscpp_mod2.so), libaxiscpp_client.so, libserver_engine.so, libAdminService.so, adminclient, simple-axis_server should have been created in $AXISCPP_HOME/bin directory. Note:- you can give the install path inside the runconfigure script.

Now there is no need to seperately build client side and server side. When execute build.sh it will lookafter everything.

NOTE: If you use apache2.0 following two small changes has to be done in $AXISCPP_HOME/configure.ac and $AXISCPP_HOME/src/server/Makefile.am)

in $AXISCPP_HOME/configure.ac

comment

```
AC_OUTPUT(Makefile src/Makefile src/common/Makefile src/engine/Makefile src/soap/Makefile src/wsdd/Makefile src/xml/Makefile src/server/Makefile src/server/apache/Makefile)
```

and uncomment

```
#AC_OUTPUT(Makefile src/Makefile src/common/Makefile src/engine/Makefile src/soap/Makefile src/wsdd/Makefile src/xml/Makefile src/server/Makefile src/server/apache2/Makefile)
```

in $AXISCPP_HOME/src/server/Makefile.am

change SUBDIRS = apache to SUBDIRS = apache2

Then to deploy it on apache

First you need to edit <apache install directory>/conf/httpd.conf

```
$ vi <apache install directory>/conf/httpd.conf
```

At the bottom of the file you have to include following lines and save it.

```
LoadModule axis_module libexec/libaxiscpp_mod.so (in apache2 replace libexec with modules and libaxiscpp_mod.so with libaxiscpp_mod2.so)

<Location /axis>

SetHandler axis

</Location>
```

```
cd /usr/local/Axis

cp -f $AXIS_HOME/deploy.sh_apache ./deploy.sh
(if apache2) cp -f $AXIS_HOME/deploy.sh_apache2 ./deploy.sh

sh deploy.sh
```

deploy.sh is a script which copies files to neccessary places and start apache.

--

If you downloaded the binary distribution

Create an environment variable called AXISCPP_HOME.

```
$ cd /home/axisuser

[axisuser@localhost axisuser]$ vi ./.bash_profile

AXISCPP_HOME="/home/axisuser/projects/axis_c"

export AXISCPP_HOME
```

Save it and back in the terminal window.

```
$ source ~/.bash_profile
```

Verify the above change had been correctly effected by typing

```
$ echo $AXISCPP_HOME

$ cd $AXISCPP_HOME
```

The Folder called deploy in the $AXISCPP_HOME/ should be copied to a folder of your choice .Rename the deploy folder as "Axis" . Give all permissions to this folder.

```
$ cp -rf $AXISCPP_HOME/deploy  /usr/local
$ cd /usr/local
$ mv deploy Axis
$ chmod -R 777 Axis
```

Now set the environment variable **AXIS_HOME** pointing to this directory.

```
AXIS_HOME="/usr/local/Axis"
```

You also need to rename the following files

```
mv $AXIS_HOME/axiscpp.conf_linux $AXIS_HOME/axiscpp.conf
mv $AXIS_HOME/conf/server.wsdd_linux $AXIS_HOME/conf/server.wsdd
mv $AXIS_HOME/conf/client.wsdd_linux $AXIS_HOME/conf/client.wsdd
```

NOTE: Binary distribution is built to work with expat parser. If you need xerces parser you need to build
from the source.

Then to deploy it on apache

First you need to edit <apache install directory>/conf/httpd.conf

```
$ vi <apache install directory>/conf/httpd.conf
```

At the bottom of the file you have to include following lines and save it.

```
LoadModule axis_module libexec/libaxiscpp_mod.so (in apache2 replace libexec with modules and libaxiscpp_mod.so with libaxiscpp_mod2.so)

<Location /axis>

SetHandler axis

</Location>

cd /usr/local/Axis

cp -f $AXIS_HOME/deploy.sh_apache  ./deploy.sh
(if apache2) cp -f $AXIS_HOME/deploy.sh_apache2 ./deploy.sh

sh deploy.sh
```

deploy.sh is a script which copies files to neccessary places and start apache.

10. Validating The Installation

If you have done installation successfully it will display the Axis C++ welcome page when you point to URI http://localhost/axis .
Note: In the axis welcome page all the services in $AXIS_HOME/conf/server.wsdd are listed. This does not mean
that the libraries corresponding to these services are deployed yet. It merely list whatever in the server.wsdd.

Deploying a Web Service

You can deploy your own web service or the sample webservices with the guidance in the userguide