# Axis C++ Windows Installation Guide

<!-- --> <!-- -->

## 1. Axis C++ Windows Installation Guide

This document contains how to use the binary distribution of Axis C++. These binaries have been tested on following Microsoft platforms,
Windows 2000 Professional
Windows XP 2002 Professional

## 2. Contents

## 3. What binary distribution contains

1. The binary distribution contains the following folders
-> bin
-> deploy
-> docs
-> include
-> lib
-> samples
2. The folder "bin" contains sample clients and other dlls needed to run the Axis c++ server
3. Inside the "deploy" folder following sub folders exist
-> conf (This includes the server.wsdd, which is the deployment descriptor used for deploying services.)
-> handlers (This includes sample handler dlls)
-> libs (Contains dlls which are needed by axis.)
-> logs (Directory for logs)
-> webservices (Contains sample web services that can be deployed with Axis C++)

---

-> wsdls (Contains sample wsdls)

4. The "include" folder contains all the include files which are necessary for building Axis c++ clients.

5. "lib" folder contains jars needed to use the wsdl2ws tool.

6. "samples" folder contains sources of all the server and client samples including interop tests.

## 4. Installation with Apache 1.3.X/2.X web server

Steps:

1. Install apache web server and locate the installation folder(For the default installation (apache 1.3.X) the path is "C:\Program Files\Apache Group\Apache" and for apache 2.X the path is "C:\Program Files\Apache Group\Apache2" ). Lets say this folder is APACHE_FOLDER.

2. Unzip the binary distribution to a folder of your choice. Say this folder is AXIS_EXTRACT.

3. Inside the extracted folder AXIS_EXTRACT there is a sub folder called deploy. Copy the whole deploy folder to APACHE_FOLDER.

4. Rename copied "deploy" folder to "Axis". Lets say this folder is AXIS_FOLDER.

5. Set environment variable AXISCPP_DEPLOY to point to above folder.

6. Add AXIS_FOLDER/libs to the PATH environment variable.

7. Edit Apache configuration file httpd.conf located in APACHE_FOLDER/conf and add following lines at the end.

LoadModule axis_module modules/AXISMODULE

<Location>

SetHandler axis

<\Location>

Where AXISMODULE is either mod_axis.dll for apache 1.3.X or mod_axis2.dll for apache 2.X.

8. Copy AXISMODULE from the AXIS_EXTRACT/bin folder to APACHE_FOLDER/modules folder.

9. Copy AxisServer.dll and AxisTransport.dll from the AXIS_EXTRACT/bin folder to AXIS_FOLDER/libs

10.If you want to use the Expat parser with Axis C++,rename AxisXMLParser_Expat.dll from AXIS_EXTRACT/bin folder to AxisXMLParser.dll and copy it to AXIS_FOLDER/libs

**Installing the Expat XML parser**

If you decide to use the Expat parser you need to install the Expat parser. Download the Expat-1.95.7 binary version from http://sourceforge.net/projects/expat/

Extract the Expat binaries to a folder Called Expat ,we will refer to this folder as

[EXPAT_EXTRACT].

Copy libexpat.dll file from [EXPAT_EXTRACT]\Libs to [AXISCPP_DEPLOY]/libs/

If you want to use the Xerces parser with Axis C++, rename AxisXMLParser_Xerces.dll from AXIS_EXTRACT/bin folder to AxisXMLParser.dll and copy it to AXIS_FOLDER/libs

**Installing the Xerces XML parser**

If you decide to use the Xerces parser you need to install the Xerces parser. Download the Xerces-C 2.2.0( 32 bit binary release)from: http://xml.apache.org/xerces-c/download.cgi

Extract the Xerces-C 2.2.0 binary version. We will refer to this folder as [XERCES_EXTRACT]. Copy xerces-c_2_2_0.dll from [XERCES_EXTRACT]/bin/ to [AXISCPP_DEPLOY]/libs/

11. Open the AXIS_FOLDER/axiscpp.conf_win for editing. The contents of the file will look as follows.

LogPath:XXXX

WSDDFilePath:YYYY

XXXX is the path to a file named AxisLog (The log file)and YYYY is the path to the server.wsdd file. Give the appropriate values for XXXX and YYYY as follows.

i.e.

LogPath: [APACHE_HOME]\Axis\logs\AxisLog

WSDDFilePath: [APACHE_HOME]\Axis\conf\WSDDFILE

WSDDFILE is [APACHE_HOME]\Axis\conf\server.wsdd for Apache 1.3 and Apache 2. (There will be three wsdd files in the [APACHE_HOME]\Axis\conf\ rename the relevant wsdd file of your platform to server.wsdd).

Also in order to run the provided samples do the following.

Open server.wsdd for editing. For each service element, locate the element "parameter", having the value "classname" for the "name" attribute. Change the value of the "value" attribute, to have the absolute path to the indicated webservice.dll in that value.

Rename AXIS_FOLDER/axiscpp.conf_win to AXIS_FOLDER/axiscpp.conf

12. Restart the apache web server. If you have done correctly apache server should start without giving any error.

**apache 1.3.X :**

It is recommended to start apache 1.3.X as "apache -k start" on the command line

**apache 2.X :**

While testing the apache 2 module a problem was noted with the apache 2 web server, in that the webserver takes about 98% of CPU time and the server does not even serve the startup page. The workaround for this is to start the apache web server as "apache -X" to make it run single threaded. While this is sufficient to test the apache 2 module this problem needs to be investigated further.

10. Open an internet browser and check http://localhost/axis. If the module has loaded properly then you should get the axis welcome page with a listing of deployed web services.

**5. Installation with Simple Axis Server**

STEPS:

1. Create a folder of your choice. We will call this folder as SIMPLE_AXIS_SERVER_FOLDER.
2. Unzip axis-c-1.1-win32.zip to a folder of your choice. Say this folder is AXIS_EXTRACT.
3. Inside the extracted folder AXIS_EXTRACT there is a sub folder called deploy. Copy whole deploy folder to SIMPLE_AXIS_SERVER_FOLDER.
4. Rename copied "deploy" folder to "Axis". Lets say this folder to be AXIS_FOLDER.
5. Set environment variable AXISCPP_DEPLOY to point to above folder.
6. Copy the exe SimpleAxisServer.exe from AXIS_EXTRACT/bin to the SIMPLE_AXIS_SERVER_FOLDER.
7. Copy the content of AXIS_FOLDER/libs to the SIMPLE_AXIS_SERVER_FOLDER.
8. Start the SimpleAxisServer by providing the port to which operate. (eg: c:\SimpleAxisServer>SimpleAxisServer 80)


**6. Running Provided Samples**

If Axis installation is a complete success, following sample web services and handlers have already been installed successfully.
You will find the built console applications at AXIS/c/bin directory. Following are the built .exe files.
1.base.exe
2.cbase.exe
3.doclitbase.exe
4.doclitgroupB.exe
5.groupB.exe
6.cgroupB
In order to run these samples you should have AxisClient.dll in the PATH environment variable or in the same directory where the client .exe is located.
You can run the provided client programs to confirm whether the Axis cpp server and clients are working fine. To run this sample client applications you need to have the path to libexapt.dll in the "PATH" environment variable or libexapt.dll should be placed where the apache.exe is.
**Note: These sample clients are compiled with the assumption that server is running at localhost port 80.**

**7. Using Visual C++ to play with provided samples.**

In your AXIS_EXTRACT/samples folder you have the samples. If you are using Visual C++ you can open the workspace file for client applications AXIS_EXTRACT\c\samples\client\interoptests\interoptests.dsw and For Deployable webservices, AXIS_EXTRACT\c\samples\server\interoptests\interoptests.dsw and play with them. There are seven projects in the workspace.

**8. Writing your web services and client applications using Axis C++.**

For writing your own web services and client applications see the **Windows User Guide**