

# Axis C++

<!-- -->

## 1. Welcome to Axis CPP

Axis C/C++ (Axis CPP) is a non-Java implementation of Axis. At its core Axis CPP has a C++ runtime engine. The provided tooling allows you to create C++ client-side stubs and server-side skeletons. The server skeletons can be deployed to either a full Apache web server using the supplied apache module or a "simple\_axis\_server" - which is a simple HTTP listener (designed to help you test your services).

There is also limited support for C wrappers around the generated Server and client objects. However, at the current time it is not recommended that these be used in a production environment.

The implementation has concentrated on being ws-i profile compliant. The implementation does not have support for ws-\* specifications and it is envisaged that this will be provided by Axis2. However, at the current time, Axis CPP is still being actively developed and improved until such times as Axis2 C implementation has progressed far enough to take over its role.

## 1.1. Contents

- [Features](#)
- [Quality Assurance](#)
- [Schedules](#)
- [Axis C++ 1.6](#)
  - [Key Features](#)
  - [Known issues](#)

## 1.2. Features

- Soap engine with both client and server support
- WSDO based deployment with dynamic deployment tools.
- Support for all basic types, Complex types and Arrays
- WSDL2WS tool for building C/C++ components
- Server side - Skeletons and Wrappers
- Client side - Stubs
- Standalone server (with HTTP support)

- Web server modules for Apache 1.3 & Apache2 (Linux/Windows)
- Web interface to the deployed services and their WSDL s.
- Sample web services and client applications.

### 1.3. Quality Assurance

In the past two years, the comprehensive test-suite has been created and maintained to a high-level. The suite is run over all the nightly builds. If bugs are found then tests are added to ensure that the code does not regress - this is your assurance of quality.

### 1.4. Schedules

At any time you can get a copy of the latest [overnight build](#) and we try to produce a full release at least every 6 months.

The current level of Axis CPP that we recommend is 1.6. This has many bug fixes in it for the more complex scenarios that we have seen Axis CPP being used in over recent months.

## 2. Axis C++ 1.6

[Download the latest release of Axis C++](#)

### 2.1. Key features of Axis C++ 1.6

- WSDL tool fixes to support optional types.
- Supports all XSD built-in simple types.
- Supports Broader XSD Any types
- Some support for xsd:choice and xsd:all

### 2.2. Known Issues

- C support is not complete.
- There are no vc projects for samples
- Optional and nillable elements within xsd:choice or xsd:all
- Binary distribution requires Apache Xerces-C version 1.5, however, Axis C++ can be built from source, without modification, to use Xerces-C version 1.6.

We hope you will enjoy using Axis C++.

Numerous efforts are currently underway to improve Axis C++ as a whole.

## Axis C++

We value your feedback very much.

Please report any bugs in [Jira](#) and feel free to let us know your thoughts and/or problems in [axis-c-user@ws.apache.org](mailto:axis-c-user@ws.apache.org)

We welcome contributions to Axis C++ so please join the discussions in [axis-c-dev@ws.apache.org](mailto:axis-c-dev@ws.apache.org)