

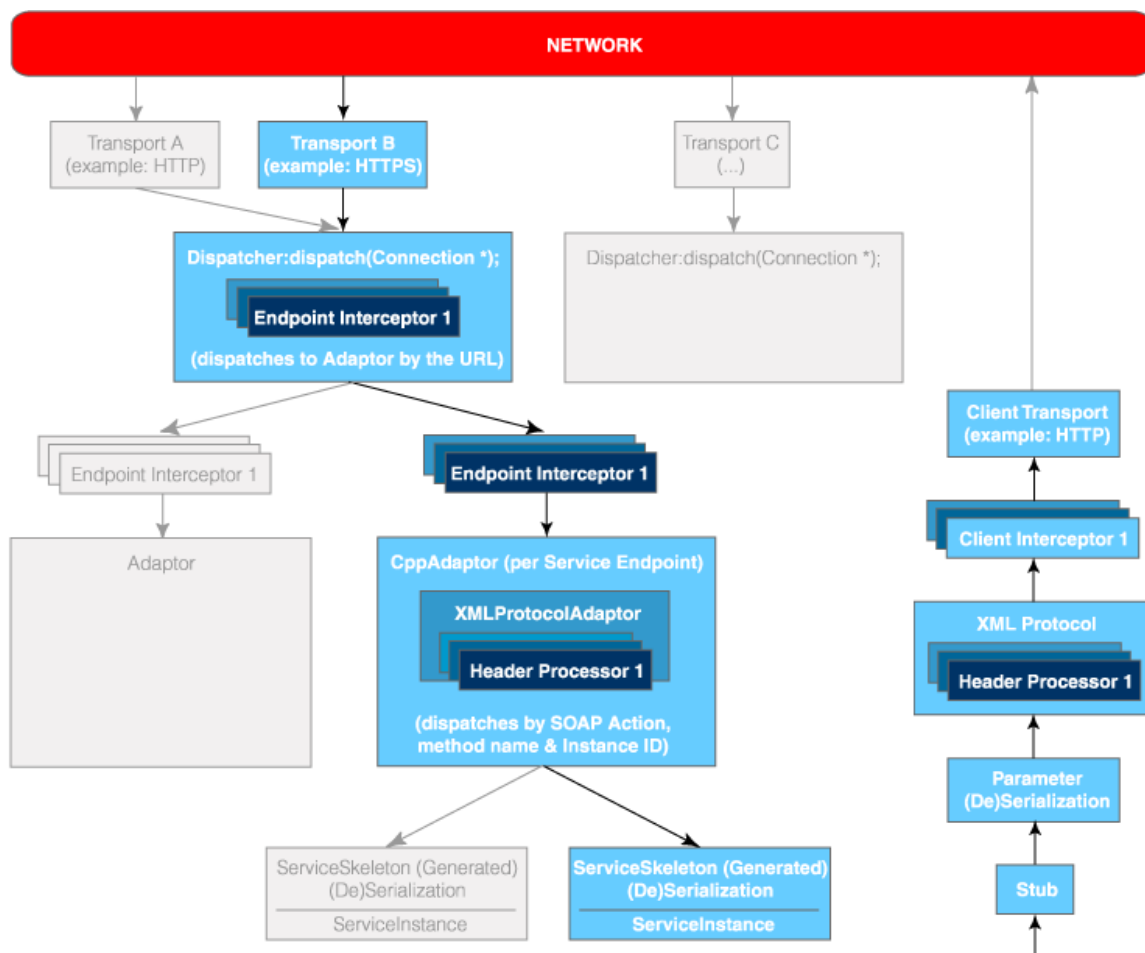
# WSO2 SOA Enablement Server for C++

## Introduction

WSO2 SOA Enablement Server for C++ is a high-performance, industrial-strength Web services API and development and runtime environment for building and deploying Web services in C/C++ that provides proven interoperability with Java-based Web services frameworks and Microsoft .NET.

WSO2 SOA Enablement Server for C++ is an ideal solution for exposing the functionality of existing C/C++ applications as Web services, and can be used to bridge Java, C++, and MS .NET applications running on different platforms easily. WSO2 SOA Enablement Server for C++ offers unique portability, modularity, asynchronous processing, UDDI client libraries and a comprehensive API that allows it to be seamlessly ported into existing applications.

## Product Architecture



WSO2 SOA Enablement Server for C++ runtime consists of three layers. Each of these layers deals with the message on some level of abstraction. Any of these layers may be replaced with the user's own implementation or extended to customize processing.

**Transport layer** - sends and receives binary data. There can be several Transport implementations each using a different protocol. There are several transports available: HTTP, HTTPS, ISAPI (for MS IIS), NSAPI (for Netscape Server) or FastCGI (for Apache).

**XML Protocol layer** - reads the message provided by the Transport and converts it to XML tokens. It can split the message into several parts defined by a particular XML Protocol (for example, SOAP). These parts, headers and bodies, are then passed to Header Processors. Their main purpose is to process the headers contained in the message but they can also affect the message body. These can be used for implementation of XML signatures, client identification etc.

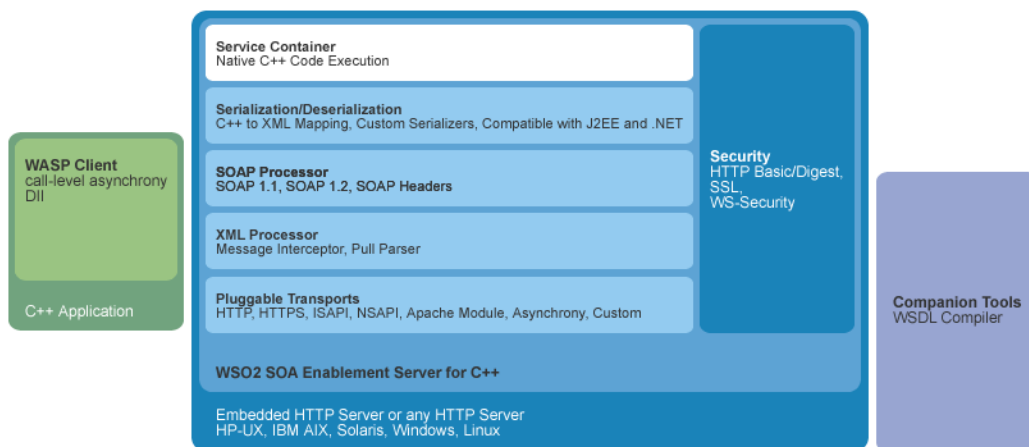
**Serialization layer** - converts the incoming XML constructs to data structures of the given implementation language, C++ in this case. As a result, the original binary data stream is converted into C++ data structures which can be easily read and modified.

## Features

### Portability

WSO2 SOA Enablement Server for C++ is shipped on all platforms including Windows, Solaris, RedHat, Debian, HP-UX, and IBM AIX. Plus WSO2 provides source code to select customers. Also WSO2 SOA Enablement Server for C++ is available on both 32 and 64-bit platforms.

WSO2 SOA Enablement Server for C++ offers support for both native and portable exceptions. Because some platforms do not support exceptions, the implementation of exceptions is not always so simple. There are also platforms which are unable to use exceptions in shared libraries. The use of portable exceptions ensures cross platform compatibility whether native C++ exceptions are supported or not.



### Modularity

The architecture of WSO2 SOA Enablement Server for C++ is modular and flexible. Each component in it is configurable and loadable on demand according to your setup. WSO2 SOA Enablement Server for C++ supports shared objects on UNIX and shared libraries on Windows, resulting in better modularity of applications, improved performance and application load time. Client and server side functionality are equally powerful.

### Embedding

WSO2 SOA Enablement Server for C++ can run in a variety of modes - standalone, ported to Apache as native module or as a FastCGI script, ported to Netscape using NSAPI or IIS using ISAPI. Because of its transport modularity, WSO2 SOA Enablement Server for C++ can be embedded into many other environments with minimal effort. For standalone or embedded usage, WSO2 SOA Enablement Server for C++ includes a high

performance HTTP server which runs in process. WSO2 SOA Enablement Server for C++ also provides excellent performance when running behind a Web server such as Apache. Plus, you can run multiple instances of WSO2 SOA Enablement Server for C++ on the same machine.

## **Transports**

WSO2 SOA Enablement Server for C++ includes a flexible, pluggable transport framework, which makes it far easier to customize it to support reliable transport mechanisms. The framework provides a layer of abstraction between the WSO2 SOA Enablement Server for C++ Server and its hosting environment and between the message processing subsystem and the transport protocol. The pluggable transport framework uses a plug-in architecture that allows you to support almost any host and transport.

## **Asynchronous Invocation**

WSO2 SOA Enablement Server for C++ provides an API for both invocation and service asynchrony. WSO2 SOA Enablement Server for C++ Web service clients can initiate a call to a Web service and continue on without having to wait for an immediate response from the server. When the resulting message arrives from the service, WSO2 SOA Enablement Server for C++ Server notifies the client, returning the results back to the client. This architecture scales well and results in higher server throughput. Plus, it also ensures more efficient use of networking resources and memory on the server.

## **Stateful Services**

Session management in WSO2 SOA Enablement Server for C++ is done via SOAP headers, which makes it easier to implement stateful services which are not dependent on the underlying transport.

## **SOAP Attachments**

WSO2 SOA Enablement Server for C++ has a fully interoperable implementation of MIME, DIME and MTOM for both client and server side.

## **XML Pull Parser**

- Custom mapping between XML and code at both compile-time and runtime.
- Namespaces mapping between XML and C++ namespaces
- Doc/lit and RPC/encoded SOAP messages

## **Security**

HTTP authentication (Basic/Digest)

SiteMinder

WS-Security - WS-Security implementation in the WSO2 SOA Enablement Server for C++ is based on the following specifications.

*OASIS Web Services Security: SOAP Message Security 1.0, Errata 1.0*

*OASIS Web Services Security: UsernameToken Profile 1.0, Errata 1.0*

*OASIS Web Services Security: X.509 Token Profile 1.0, Errata 1.0*

## **WS-Addressing**

Supports wsa:Action and wsa:To only.

## **Dynamic Invocation**

WSO2 SOA Enablement Server for C++ supports both Dynamic Interface Invocation (DII) and Dynamic Service Invocation (DSI).

## **Unicode Support**

WSO2 SOA Enablement Server for C++ has full support for Unicode characters.

## **Memory Debugger**

The memory debugging facility in the WSO2 SOA Enablement Server for C++ tracks memory leaks by remembering all new/delete uses.

## **Admin Console**

WSO2 SOA Enablement Server for C++ has a user-friendly, easy-to-use graphical user interface for runtime management.

## **Field Testing**

WSO2 SOA Enablement Server for C++ is a robust field-tested solution with a number of live deployments.