

**Main page:** [Cisco Unity Express -- Programmatic Interface Service Programming Guide](#)

**Previous page:** [Cisco Unity Express -- Quick Start for RESTful Web Services](#)

**Next page:** [Cisco Unity Express -- REST PI Service Conventions](#)

**Search the Cisco Unity Express documentation on the DocWiki:**

Loading

---

Most of the interaction between client and the CUE PI service takes place by exchanging XML documents using HTTP calls. Thus, CUE PI service clients need an HTTP library to make HTTP calls and an XML parser to process the XML documents. Though a client does not necessarily need to use a Java-based environment, this document discusses some of the software choices available for the Java-based clients only.

## Contents

- [1 HTTP Library](#)
  - ◆ [1.1 CURL](#)
- [2 XML Processing](#)
  - ◆ [2.1 XML Schema and Validation](#)
  - ◆ [2.2 Java Architecture for XML Binding](#)

## HTTP Library

For making HTTP requests to the CUE PI service, it is highly recommended to use a library compliant with HTTP/1.1 specification. For Java-based implementation, some of the common available choices are:

- HTTP client `java.net.HttpURLConnection`, part of the Java standard library
- Apache Jakarta Commons HttpClient
- The Jersey Client API - Part of the Jersey framework, JAX-RS reference implementation by Sun

## CURL

In Linux-based environments, "curl" command line utility provides a simple and quick way for interacting with the CUE PI service. This document uses "curl" CLI snippets to provide examples while explaining various configuration operation requests and responses.

## XML Processing

An XML parser is required to extract information from the XML document returned by the CUE PI service. CUE PI service does not require use of any specific XML parser or approach (SAX or DOM) by its clients.

## XML Schema and Validation

The CUE PI service provides XML Schema for its resources defined with XML representation.

- Clients can download the set of XML schema from the CUE PI service as explained in the [Cisco Unity Express -- XML Schema Download REST PI Service Operation](#) section.
- The XML documents returned by the CUE PI service are guaranteed to be compliant with the XML schema published by the CUE PI service so clients do not need to validate the XML response against the schema.
- The CUE PI service does not perform validation of XML payload received from the clients against the schema. Thus, clients must make sure that any XML document sent to the CUE PI service is compliant with the published schema.
- Because schema validation is not performed by the service, any unrecognized XML elements present in the payload are ignored. An error is returned only if the XML document is not well formed.

To make sure that the XML document passed to the CUE PI service are always schema compliant, clients can choose to perform schema validation before passing the payload to HTTP request. Another approach to ensure schema compliance is to generate the XML payloads using the Java Architecture for XML binding (JAXB) library with Java classes generated from the CUE PI XML schema.

## Java Architecture for XML Binding

The JAXB provides a fast and convenient way to bind between XML schema and Java representations. JAXB provides methods for unmarshalling XML instance documents into Java content trees and also marshalling Java content trees into XML instance documents. JAXB also provides a way to generate Java classes from XML schema documents. The following sequence provides a high-level view of JAXB usage with the CUE PI service.

- Acquire XML schema published by the CUE PI ([Cisco Unity Express -- XML Schema Download REST PI Service Operation](#)) service and run JAXB XJC (Java to XML) compiler on the schema set. The compiler produces a JAXB-compliant Java representation of the #: XML schema.

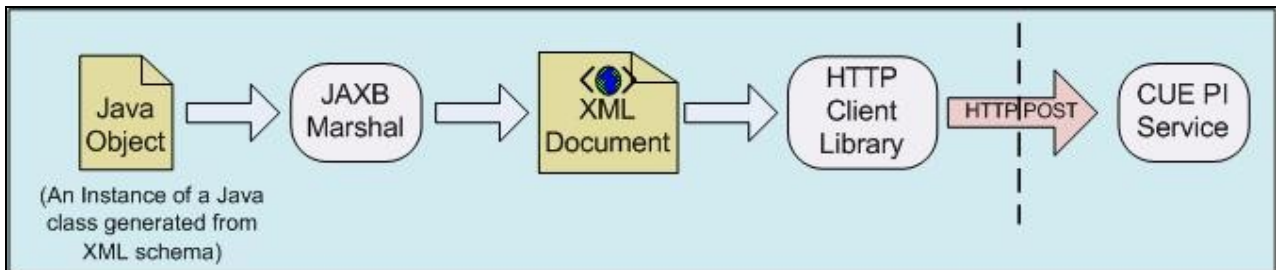
**Figure 1: Running the XJC (JAXB Schema compiler) on CUE PI XML schema**



- While making POST or PUT (for create or update of a resource) calls, the client populates the resource representation as Java objects. These java objects are then marshalled to XML documents

using JAXB library. This ensures that the XML documents produced are compliant to the XML schema.

**Figure 2: Using JAXB to marshall Java objects to XML documents**



- Similarly while processing the response for GET request, clients can use JAXB library to unmarshall the XML payload to Java object tree.

**Figure 3: Using JAXB to unmarshall XML documents**

