

# Web services hints and tips: JAX-RPC versus JAX-WS, Part 1

## Introduction

Russell Butek  
Nicholas Gallardo

October 06, 2006

JAX-WS 2.0 is the successor to JAX-RPC 1.1. This article introduces a series that compares these two Java™ web services programming models.

[View more content in this series](#)

## Introduction

Web services have been around a while now. First there was SOAP. But SOAP only described what the messages looked like. Then there was WSDL. But WSDL didn't tell you how to write web services in Java™. Then along came JAX-RPC 1.0. After a few months of use, the Java Community Process (JCP) folks who wrote that specification realized that it needed a few tweaks, so out came JAX-RPC 1.1. After a year or so of using that specification, the JCP folks wanted to build a better version: JAX-RPC 2.0. A primary goal was to align with industry direction, but the industry was not merely doing RPC web services, they were also doing message-oriented web services. So "RPC" was removed from the name and replaced with "WS" (which stands for web Services, of course). Thus the successor to JAX-RPC 1.1 is JAX-WS 2.0 - the Java API for XML-based web services.

## What remains the same?

Before we itemize the differences between JAX-RPC 1.1 and JAX-WS 2.0, we should first discuss what is the same.

- JAX-WS still supports SOAP 1.1 over HTTP 1.1, so interoperability will not be affected. The same messages can still flow across the wire.
- JAX-WS still supports WSDL 1.1, so what you've learned about that specification is still useful. A WSDL 2.0 specification is nearing completion, but it was still in the works at the time that JAX-WS 2.0 was finalized.

## What is different?

- SOAP 1.2  
JAX-RPC and JAX-WS support SOAP 1.1. JAX-WS also supports SOAP 1.2.
- XML/HTTP  
The WSDL 1.1 specification defined an HTTP binding, which is a means by which you can send XML messages over HTTP without SOAP. JAX-RPC ignored the HTTP binding. JAX-WS adds support for it.
- WS-I's Basic Profiles  
JAX-RPC supports WS-I's Basic Profile (BP) version 1.0. JAX-WS supports BP 1.1. (WS-I is the web services interoperability organization.)
- New Java features
  - JAX-RPC maps to Java 1.4. JAX-WS maps to Java 5.0. JAX-WS relies on many of the features new in Java 5.0.
  - Java EE 5, the successor to J2EE 1.4, adds support for JAX-WS, but it also retains support for JAX-RPC, which could be confusing to today's web services novices.
- The data mapping model
  - JAX-RPC has its own data mapping model, which covers about 90 percent of all schema types. Those that it does not cover are mapped to `javax.xml.soap.SOAPElement`.
  - JAX-WS's data mapping model is JAXB. JAXB promises mappings for all XML schemas.
- The interface mapping model  
JAX-WS's basic interface mapping model is not extensively different from JAX-RPC's; however:
  - JAX-WS's model makes use of new Java 5.0 features.
  - JAX-WS's model introduces asynchronous functionality.
- The dynamic programming model
  - JAX-WS's dynamic client model is quite different from JAX-RPC's. Many of the changes acknowledge industry needs:
    - It introduces message-oriented functionality.
    - It introduces dynamic asynchronous functionality.
  - JAX-WS also adds a dynamic server model, which JAX-RPC does not have.
- MTOM (Message Transmission Optimization Mechanism)  
JAX-WS, via JAXB, adds support for MTOM, the new attachment specification. Microsoft never bought into the SOAP with Attachments specification; but it appears that everyone supports MTOM, so attachment interoperability should become a reality.
- The handler model
  - The handler model has changed quite a bit from JAX-RPC to JAX-WS.
  - JAX-RPC handlers rely on SAAJ 1.2. JAX-WS handlers rely on the new SAAJ 1.3 specification.

In the remainder of this tip, we will discuss SOAP 1.2, XML/HTTP, the WS-I Basic Profiles, and Java 5. Each of the remaining five bullets above will be a separate tip in this series.

### SOAP encoding

SOAP encoding has fallen out of favor in the web services community. It is not supported by the WS-I basic profile. So JAX-WS, as the latest incarnation of Java web services, has

disallowed SOAP encoding. JAX-RPC supports SOAP encoding, so if you really must use SOAP encoded messages, stick with JAX-RPC.

## SOAP 1.2

There is really not a lot of difference, from a programming model point of view, between SOAP 1.1 and SOAP 1.2. As a Java programmer, the only place you will encounter these differences is when using the handlers, which we will cover in a future tip. SAAJ 1.3 has been updated to support SOAP 1.2.

## XML/HTTP

Like the changes for SOAP 1.2, there is really not a lot of difference, from a programming model point of view, between SOAP/HTTP and XML/HTTP messages. As a Java programmer, the only place you will encounter these differences is when using the handlers, which we will cover in a future tip. The HTTP binding has its own handler chain and its own set of message context properties.

## WS-I's basic profiles

JAX-RPC 1.1 supports WS-I's Basic Profile (BP) 1.0. Since that time, the WS-I folks have developed BP 1.1 (and the associated AP 1.0 and SSBP 1.0). These new profiles clarify some minor points, and more clearly define attachments. JAX-WS 2.0 supports these newer profiles. For the most part, the differences between them do not affect the Java programming model. The exception is attachments. WS-I not only cleared up some questions about attachments, but they also defined their own XML attachment type: wsi:swaRef.

Many people are confused by all these profiles. You will need a little history to clear up the confusion.

WS-I's first basic profile (BP 1.0) did a good job of clarifying the various specs. But it wasn't perfect. And support for SOAP with Attachments (Sw/A) in particular was still rather fuzzy. In their second iteration, the WS-I folks pulled attachments out of the basic profile - BP 1.1 - and fixed some of the things they missed the first time around. At that point they also added two mutually exclusive supplements to the basic profile: AP 1.0 and SSBP 1.0. AP 1.0 is the Attachment Profile which describes how to use Sw/A. SSBP 1.0 is the Simple SOAP Binding Profile, which describes a web services engine that does not support Sw/A (such as Microsoft's .NET). The remaining profiles that WS-I has and is working on build on top of those basic profiles.

## Java 5

There are a number of changes to the Java language. JAX-WS relies on: annotations, generics, and executors. We will detail exactly how JAX-WS relies on this new functionality in follow-on tips. For information on these new features of Java, see the Java 5 link in [Related topics](#).

## Summary

JAX-WS 2.0 is the successor to JAX-RPC 1.1. There are some things that haven't changed, but most of the programming model is different to a greater or lesser degree. The topics introduced in

this tip will be expanded upon in a series of tips which we will publish over the coming months that will compare, in detail, JAX-WS and JAX-RPC. At a high level though, here are a few reasons why you would or would not want to move to JAX-WS from JAX-RPC.

#### Reasons you may want to stay with JAX-RPC 1.1:

- If you want to stay with something that's been around a while, JAX-RPC will continue to be supported for some time to come.
- If you don't want to step up to Java 5.
- If you want to send SOAP encoded messages or create RPC/encoded style WSDL.

#### Reasons to step up to JAX-WS 2.0:

- If you want to use the new message-oriented APIs.
- If you want to use MTOM to send attachment data.
- If you want better support for XML schema through JAXB.
- If you want to use an asynchronous programming model in your web service clients.
- If you need to have clients or services that can handle SOAP 1.2 messages.
- If you want to eliminate the need for SOAP in your web services and just use the XML/HTTP binding.
- If you like playing with leading edge technology.

## Related topics

- Build your next development project with [IBM trial software](#), available for download directly from developerWorks.
- Get the specification, API classes, and Javadocs for [JAX-RPC 1.1](#).
- Get the specification, API classes, and Javadocs for [JAX-WS 2.0](#).
- Get the specification, API classes, and Javadocs for [JAXB 2.0](#).
- Learn about all the features of [Java 5](#).
- Read the [WSDL 1.1 specification](#).
- Read the [SOAP version 1.2 primer](#).
- Read the [MTOM specification](#).
- Read the [SOAP with Attachments specification](#).
- Visit [WS-I's web site](#).
- You can begin to evaluate JAX-WS features with the alpha release of the [IBM WebSphere Application Server Version 6.1 Feature Pack for Web Services](#).

© Copyright IBM Corporation 2006

([www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml))

**Trademarks**

([www.ibm.com/developerworks/ibm/trademarks/](http://www.ibm.com/developerworks/ibm/trademarks/))