




Framework	Key Positives	Key Concerns
Apache AXIS2	<p>❶ Most Commonly Used, Matured &amp; Stable Web Services Development Framework</p> <p>❷ Supports Multiple Languages (C++, Java)</p> <p>❸ Supports both Contract-first &amp; Contract-last Approach</p> <p>❹ In context of Orchestration &amp; Web Services Transaction (long-running transactions) it supports wide variety of related WS-* specifications:  <i>WS-Atomic Transaction, WS-Business Activity, WS-Coordination, WS-Eventing, WS-Transfer</i>  <i>Compatible with Spring Framework</i></p>	<p>❶ Comparatively More Code Required/Generated w.r.t. Spring WS/CXF</p> <p>❷ Is being phased out gradually (mostly by Apache CXF)</p> <p>❸ It is not fully compliant for JAX-WS JAX-RS</p>

<b>Apache CXF</b>	<ul style="list-style-type: none"> <li>❶ Most widely used Web Services Standard Now; Improvement over AXIS2, which is now gradually being replaced by Apache CXF</li> <li>❷ Intuitive &amp; Easy to Use (less coding required as compared to AXIS2)</li> <li>❸ Clean separation of front-ends, like JAX-WS, from the core code</li> <li>❹ Fully compliant with JAX-WS, JAX-RS &amp; others</li> <li>❺ Best Performance across all available framework with minimum computation overhead</li> <li>❻ Supports wide variety of front-end models</li> <li>❼ Supports both JAX-WS &amp; JAX-RS (for Restful Services)</li> <li>❽ Supports JBI &amp; SDO (not supported in AXIS2)</li> <li>❾ Compatible with Spring Framework</li> </ul>	<ul style="list-style-type: none"> <li>❶ Does not support Orchestration &amp; WS Transactions yet</li> <li>❷ Does not support WSDL 2.0 yet</li> </ul>
<b>Spring WS</b>	<ul style="list-style-type: none"> <li>❶ Best in terms of supporting Contract-first Web Services Development Approach</li> <li>❷ Enforces Standards &amp; Best Practices by Framework Constraints (no way out of it &amp; hence limitation as well)</li> <li>❸ Supports Spring Annotations as well as JAX-WS</li> <li>❹ Least code from developer's perspective</li> </ul>	<ul style="list-style-type: none"> <li>❶ Least number of WS-* Specifications supported (does not fully compliant with JAX-WS)</li> <li>❷ Spring offers itself as standard &amp; hence other Java-compliant frameworks support better standards support</li> <li>❸ Only support Contract-first Web Services Development Model</li> </ul>

5 Best Aligned with Spring Technology Stack (also similar architectural stack as Spring MVC) including Spring Security

Pattern	Overall Score
Apache AXIS2	 2.59
Apache CXF	 2.624
Spring WS	 2.384

Criteria	Weight	AXIS2	CXF	Spring WS
Development Features	10.0%	2.6	2.8	2.7
Security	15.0%	2.3	2.7	2.7
Performance	20.0%	2	3	2
Reusability	5.0%	3	3	3
Data Bindings Support	5.0%	3	3	3
TCO (Total Cost of Ownership)	10.0%	3	2	2
Web Services Standards Support	10.0%	2.85	2.39	1.59
Transport Mechanism Support	5.0%	3	3	3
Vendor Alignment	5.0%	3	3	3
JSR Standards Alignment	5.0%	3	2	2
Required Skill sets	5.0%	2	2	3
Governance	5.0%	3	2	2

### Conclusion:

- Apache AXIS2 is relatively most used framework but Apache CXF scores over other Web Services Framework comparatively considering ease of development, current industry trend, performance, overall scorecard and other features (unless there is Web Services Orchestration support is explicitly needed, which is not required here)
- Though Spring Core Framework is an established technology, Spring Web Services is still evolving in comparison with Apache CXF and CXF has wider support of standards and proven framework from performance perspective.
- *Hence, Apache CXF is the recommended framework and clearly the most preferred in my opinion.*