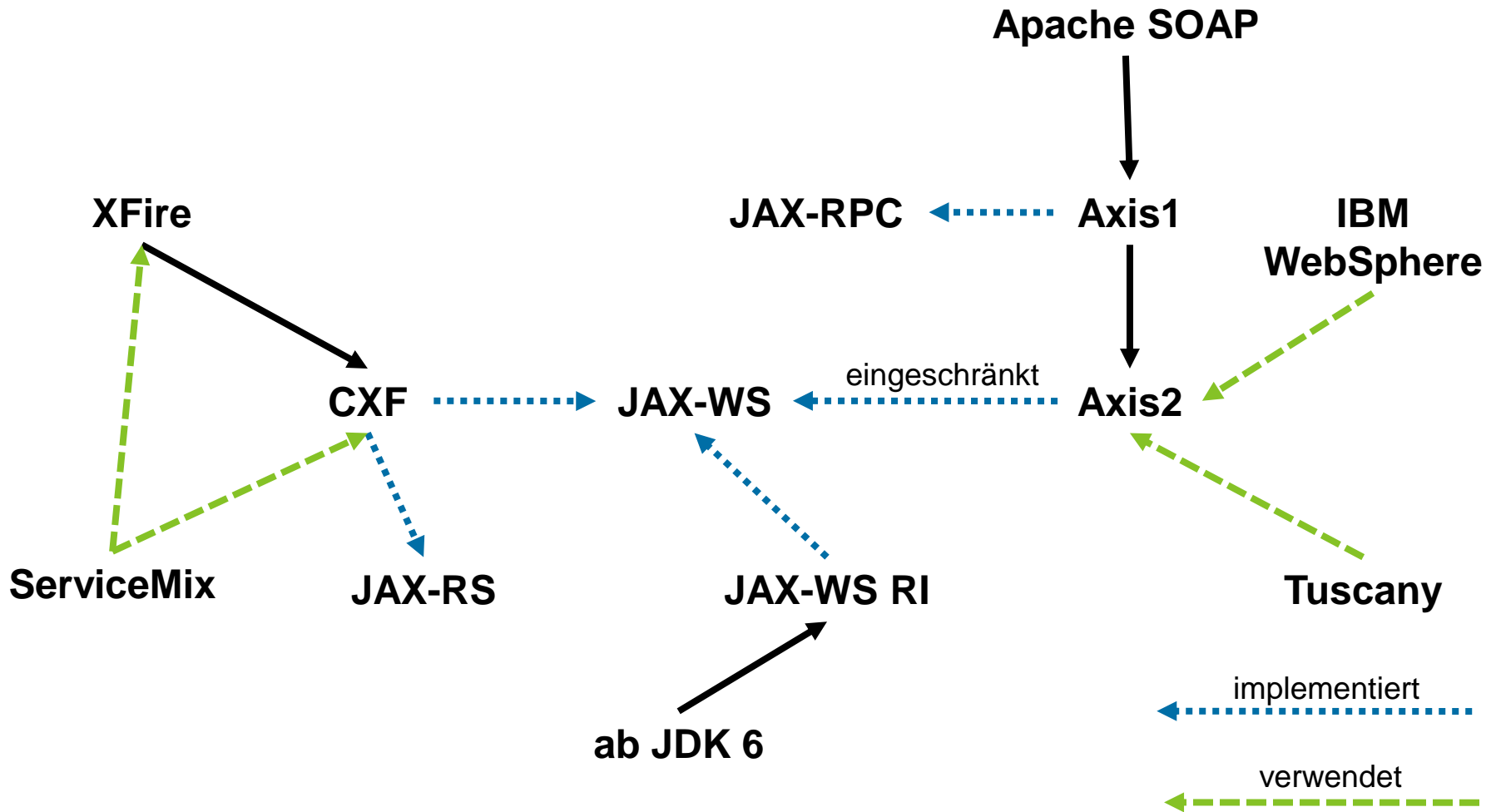


Axis2, CXF und JAX-WS RI im Vergleich

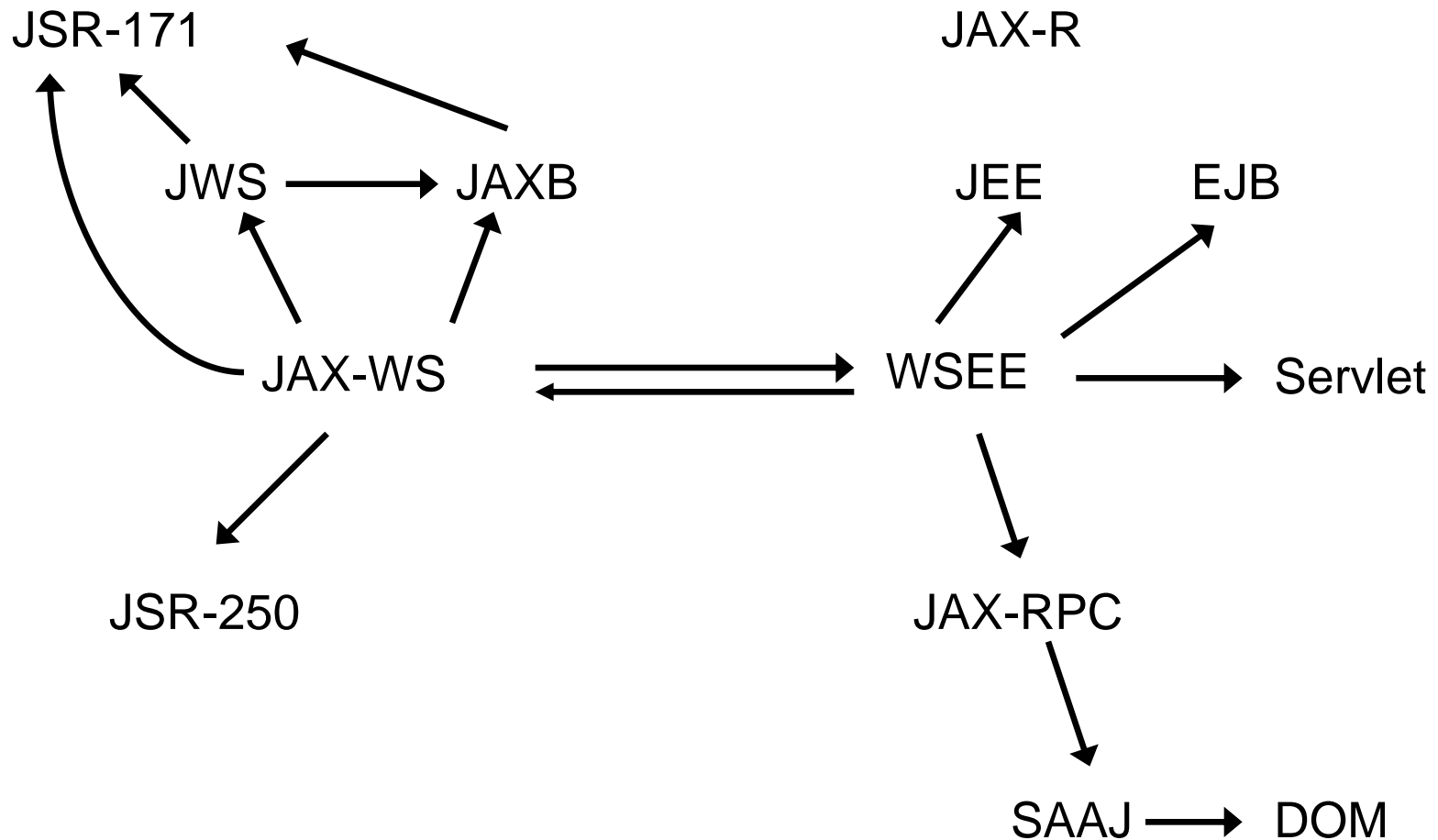
predic8 GmbH
Moltkestr. 40
53173 Bonn

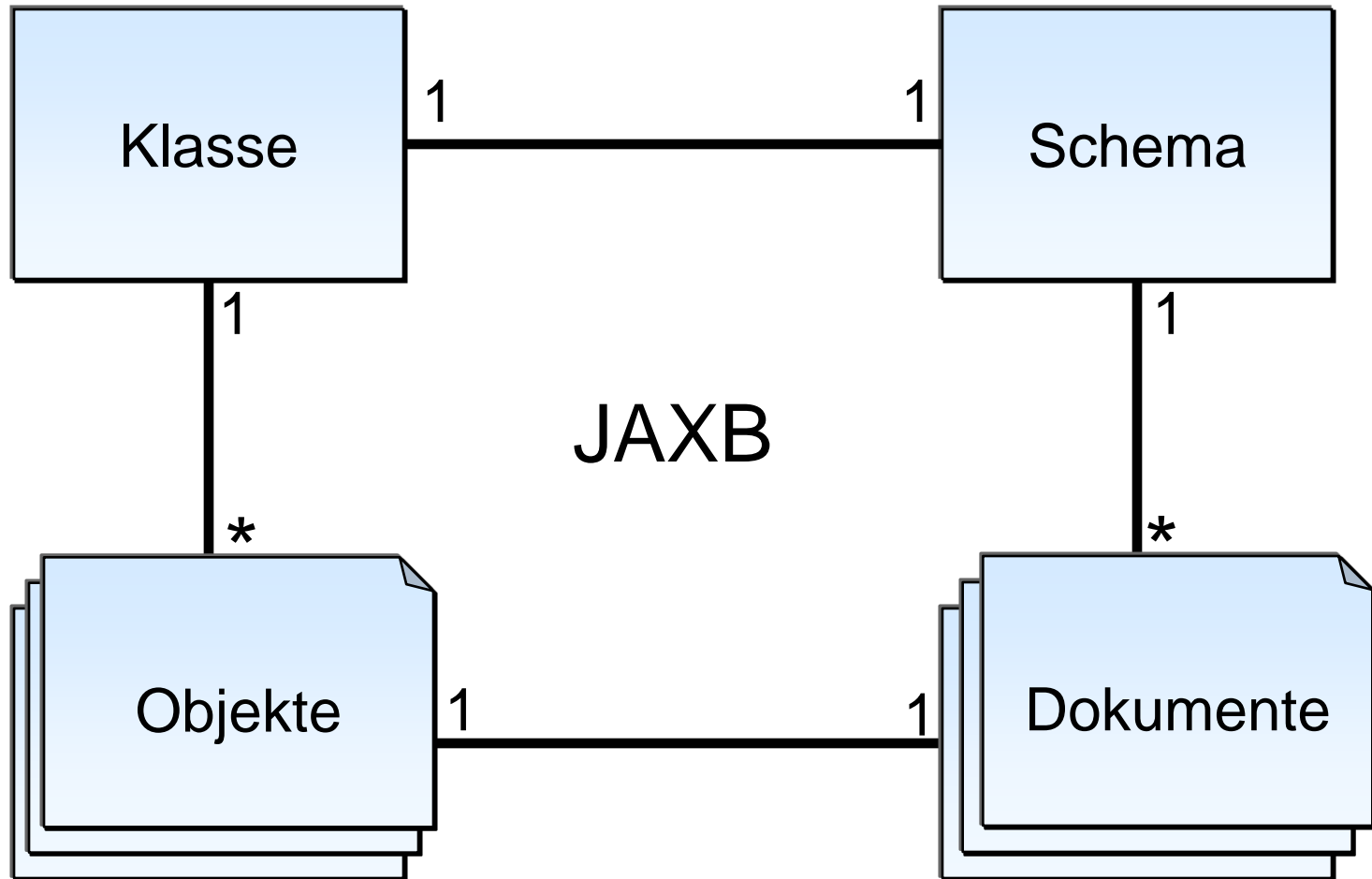
Tel: (0228)5552576-0
www.predic8.de
info@predic8.de

- Einführung
- Die Standards: JAXB, JWS, JAX-WS
- Axis2
- CXF
- JAX-WS RI / Metro
- Deployment
- Fazit

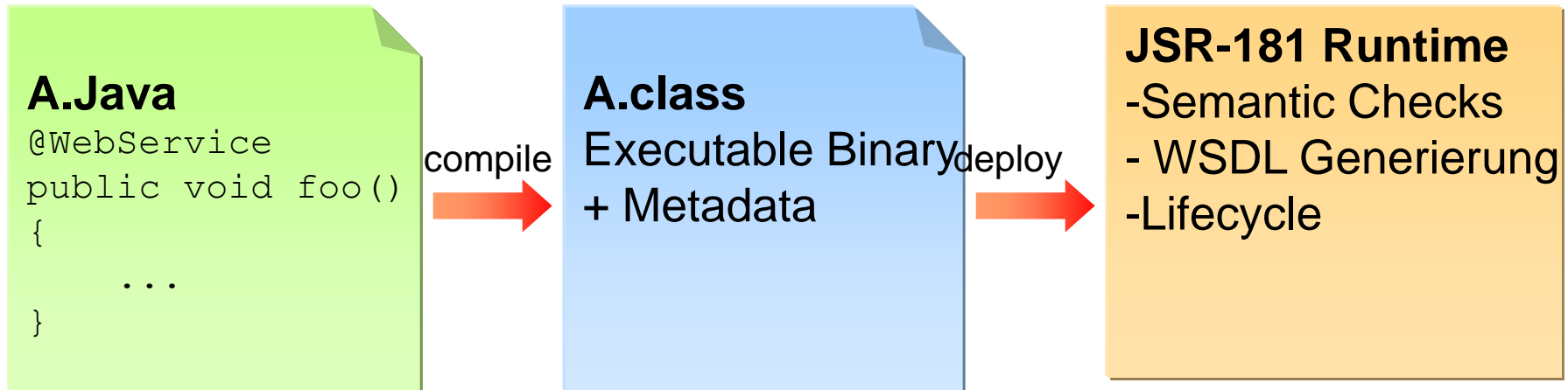


Die Standards: JWS, JAXB und JAX-WS





```
import javax.jws.WebService;  
import javax.jws.WebMethod;  
  
@WebService  
public class HelloWorldService {  
  
    @WebMethod  
    public String helloWorld() {  
        return "Hello World!";  
    }  
}
```



- SOAP 1.1, 1.2
- WSDL 1.1 nicht WSDL 2.0
- WS-I Basic Profile 1.0
- Annotations JSR-175
- WS Metadata JSR-181
- Enterprise WS JSR-109
- WS Security JSR-183

- Rückwärtskompatibilität mit JAX-RPC 1.X
- Plugable Data Binding
- Unterstützung für SOAP Encoding
- Unterstützung für J2SE < 5.0

Axis2

Architektur 1

Basis für Apache SOAP

IBM SOAP4J

Architektur 2

Basiert auf DOM

Apache SOAP

Architektur 3

Neuimplementierung mit SAX

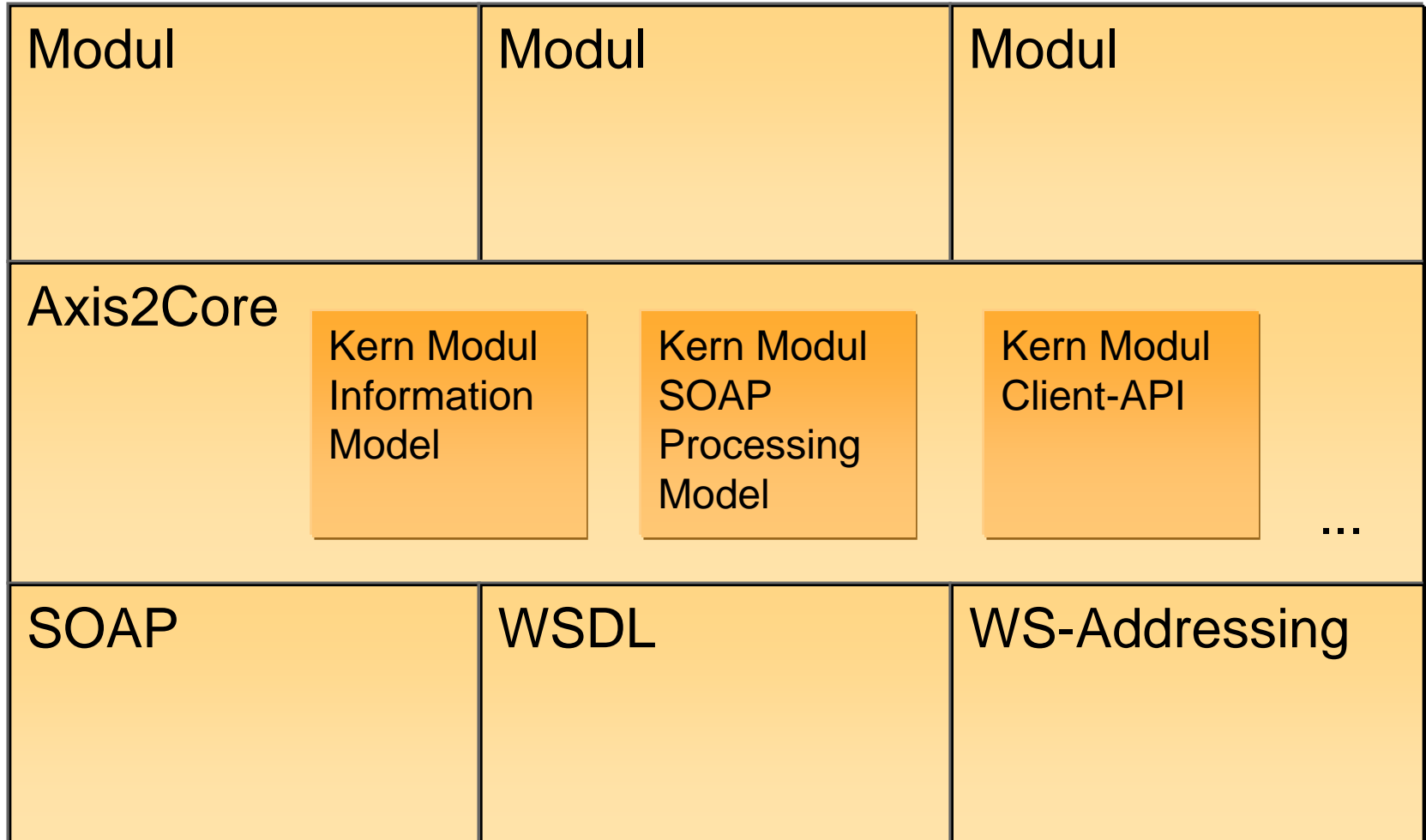
Axis1

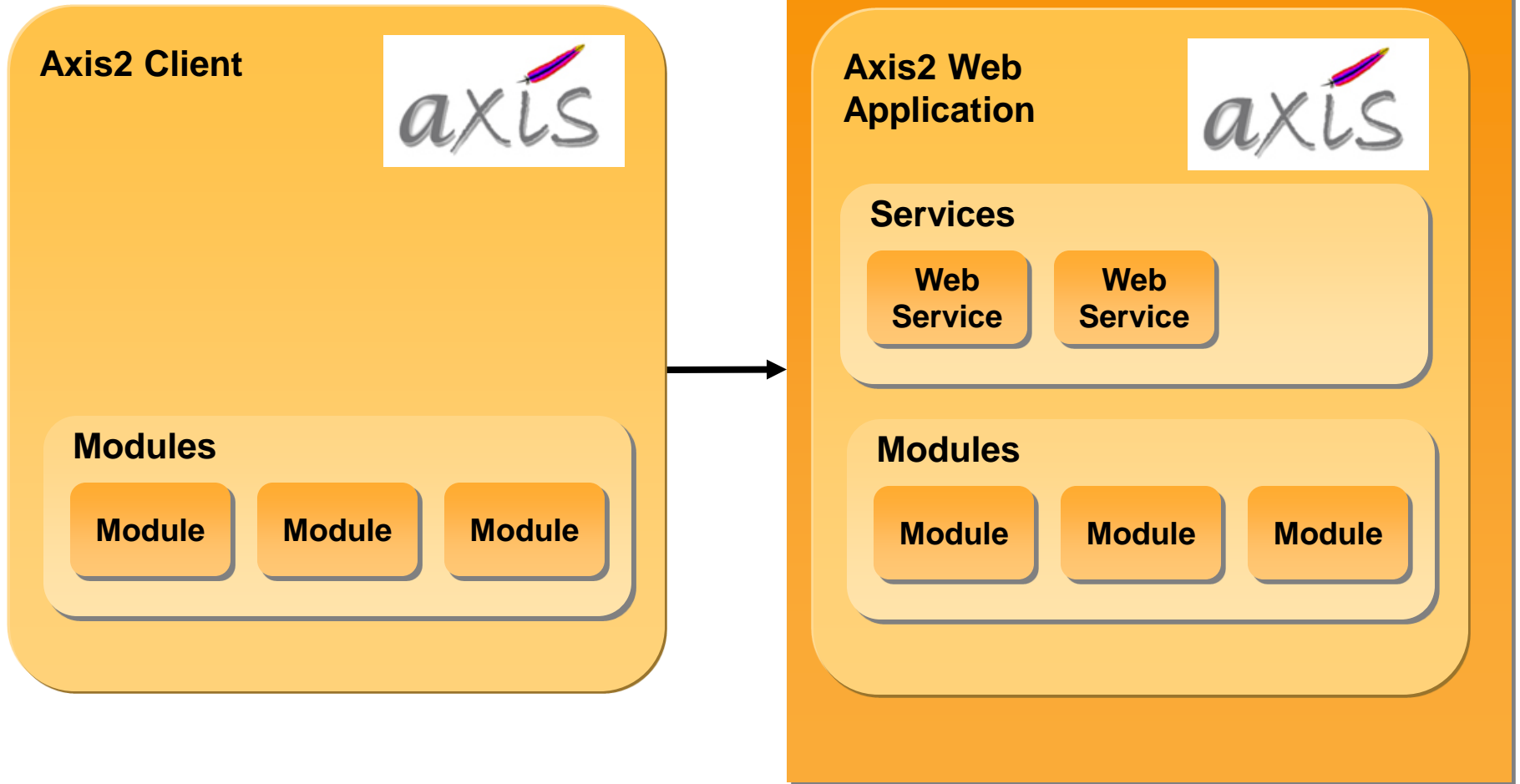
Architektur 4

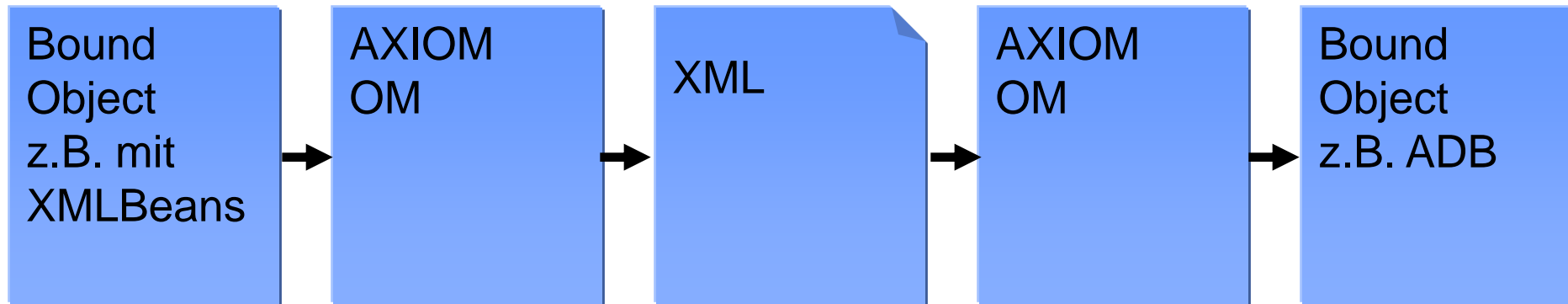
Neuimplementierung mit StAX

Axis2

- Pojo Web Services
- Spring Integration
- WSDL2Java und Java2WSDL Generator
- SOAP with Attachments
- REST Style Web Services
- Unterstützung für einige WS-* Standards
- Modulare Struktur
- Deployment: Service und Modul Archive







- Ab Axis2 Version 1.2
- Axis2 1.4.1 unterstützt JAX-WS 2.0 und JAXB 2.0
- JAXB Unterstützung ist noch „experimental“ (Axis2 1.4.1)
- JAXB 2.1 u. JAX-WS 2.1 wahrscheinlich ab Version 1.5
- Axis2 Subprojekt: axis2-jaxws
- wsgen und wsimport aus Java SDK 6 verwenden
- Auch ?wsdl verwendet Java 6 Werkzeuge



Apache CXF

predic8 GmbH

Molkestr. 40
53173 Bonn

Tel: (0228)5552576-0
www.predic8.de
info@predic8.de

- Nachfolger von **Celtix** und **Xfire**
- Services Framework für die „Protokolle“
 - ▶ SOAP, REST, JMS, HTTP, ...
- Open Source unter ASF Lizenz
- Lightweight

- Programmiermodell für Web Services
- Verfügbare Frontends:
 - ▶ Simple
 - ▶ JAX-WS

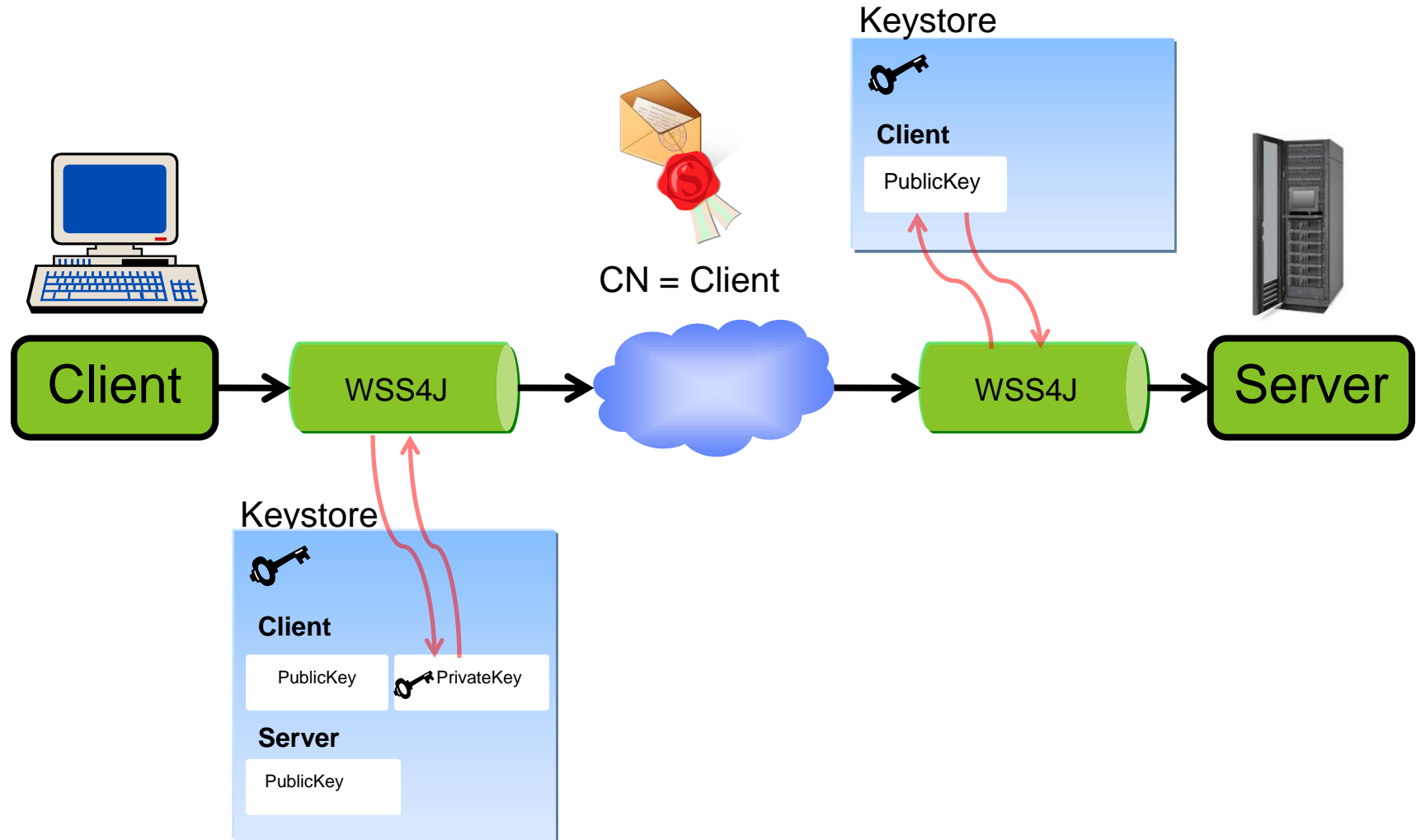
- Bildet Java Objekte auf XML ab
- Bietet sinnvolles Mapping ohne grossen Aufwand
- Detaillierte Customization ist möglich
- War Databinding in XFire
- Kann ohne CXF und XFire eingesetzt werden
- Nur für Einsatz mit Simple Frontend empfohlen (dokumentiert und getestet)

- Manche finden das Aegis Binding eleganter als JAXB
- Für Client ungeeignet, da wsdl2java fehlt
- Keine Unterstützung für elementForm= "unqualified"
- Kann alles deserialisieren
- Kann nur deserialisieren wenn Mapping vorliegt

- Bietet gemeinsam genutzte Ressourcen an
- Ressourcen:
 - ▶ WSDL Manager
 - ▶ Binding Factory Manager
 - ▶ HTTP Destination Factory
- Basiert auf Spring
 - ▶ Wird per DI konfiguriert
- Verwaltet Erweiterungen
- Interceptor Provider

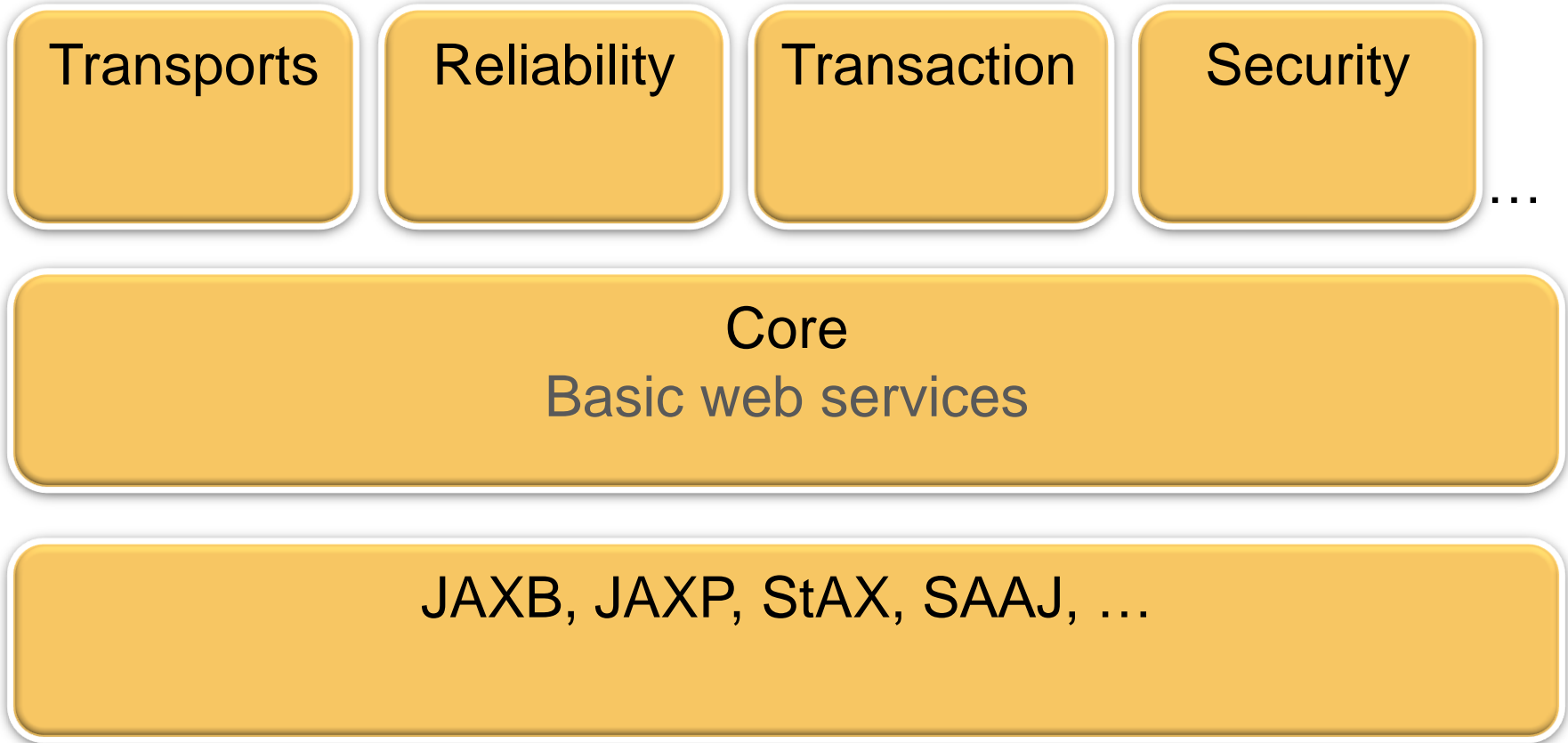
```
<cxf:bus>
  <cxf:features>
    <cxf:logging />
    <wsa:addressing />
    <wsrm-mgr:reliableMessaging>
      <wsrm-policy:RMAssertion>
        <wsrm-policy:BaseRetransmissionInterval Milliseconds="4000" />
        <wsrm-policy:AcknowledgementInterval Milliseconds="2000" />
      </wsrm-policy:RMAssertion>
      <wsrm-mgr:destinationPolicy>
        <wsrm-mgr:acksPolicy intraMessageThreshold="0" />
      </wsrm-mgr:destinationPolicy>
    </wsrm-mgr:reliableMessaging>
  </cxf:features>
</cxf:bus>
```


Signieren von Nachrichten mit WSS4J



JAX-WS RI und Metro

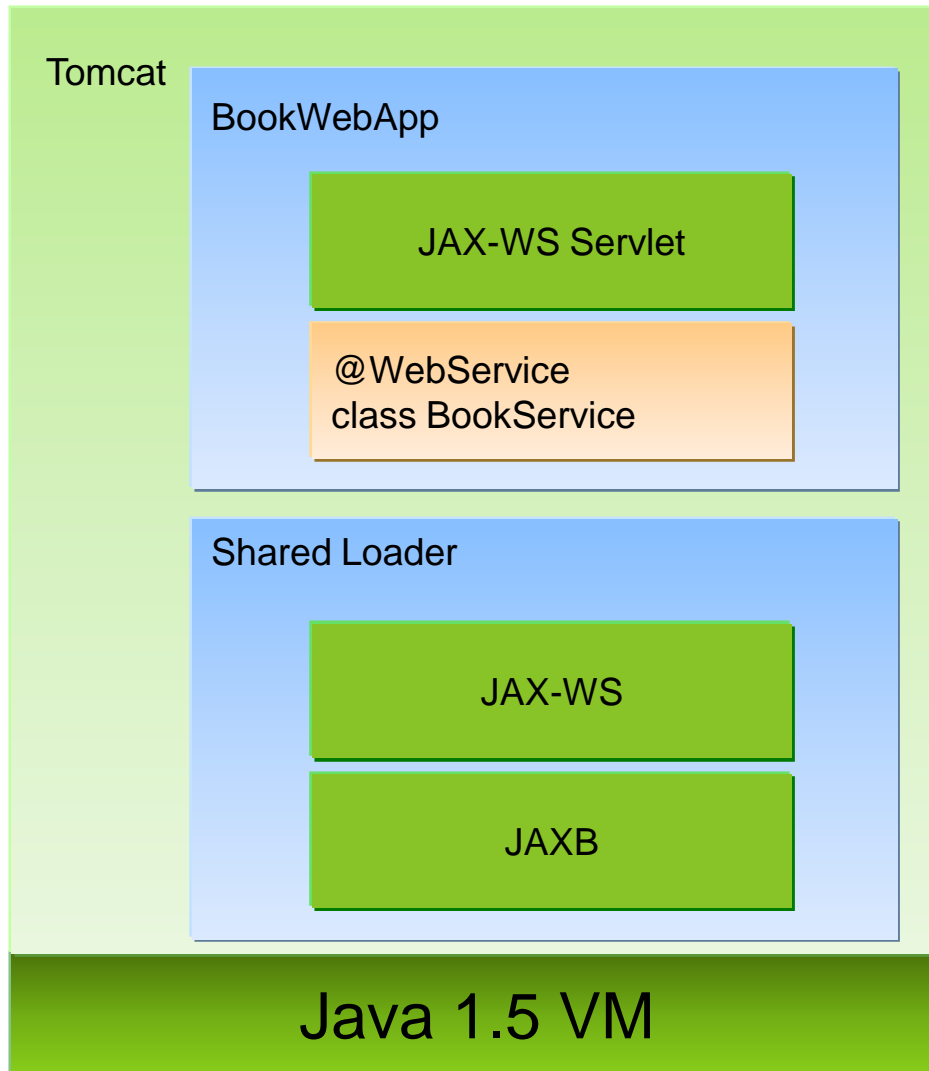
- Früherer Projektname: Tango
- Bestandteil von Glassfish V2
- Open Source Implementierung von WS-* Standards
- Ziel: Interoperabilität mit .NET und WCF



Quelle: <https://metro.dev.java.net/discover>

Deployment mit JAX-WS

JAX-WS in JEE Container (< Version 5)



Deployment in JEE 5 Server



Glassfish Application Server

Web Container

Book Web App

@WebService
Class BookService

VM >= 1.5

```
@WebService  
Class BookService
```

```
Endpoint.publish("http://localhost:8080/book",new BookServiceImpl())
```


Fazit

Feature	Axis1 1.4	Axis2 1.4.1	CXF 2.1.3	JAX-WS RI 2.1.5
Basic Profile 1.1 Compilant	x	x	x	x
JAX-RPC	x			
JAX-WS		(x)	x	x
POJO Services		x	x	x
MTOM		x	x	x
REST Support		x	JAX-RS	-(JAX-RS RI)

Unterstützung für WS-* Standards (Stand Juli 2008)



Standard	Axis2	CXF	JAX-WS/Metro
WS-Addressing	X	X	X
WS-Coordination	X(2)		X
WS-MetadataExchange			
WS-Policy	X	X	X
WS-ReliableMessaging	X(3)	X	X
Web Services Security	X(1)	X(4)	X
WS-SecureConversation	X(1)		X
WS-SecurityPolicy			X
WS-Transaction	X(2)		X
WS-Federation			

- (1) Unterstützt durch das zusätzliche Modul Apache Rampart
- (2) Unterstützt durch das zusätzliche Modul Apache Kandula2
- (3) Unterstützt durch das zusätzliche Modul Apache Sandesha2
- (4) über Apache WSS4J Interceptor

- JAX-WS und JAXB sind Standard
 - ▶ Mit JAX-WS sind Axis2, CXF und RI austauschbar
- Axis2
 - ▶ Bietet beste Schema Unterstützung
- CXF
 - ▶ Ist leichtgewichtig
 - ▶ Am besten (mit Spring) integrierbar
- JAX-WS RI
 - ▶ Einfach einzusetzen
 - ▶ Ausgereift
 - ▶ Bietet mit Metro beste WS-* Unterstützung



Fragen ?