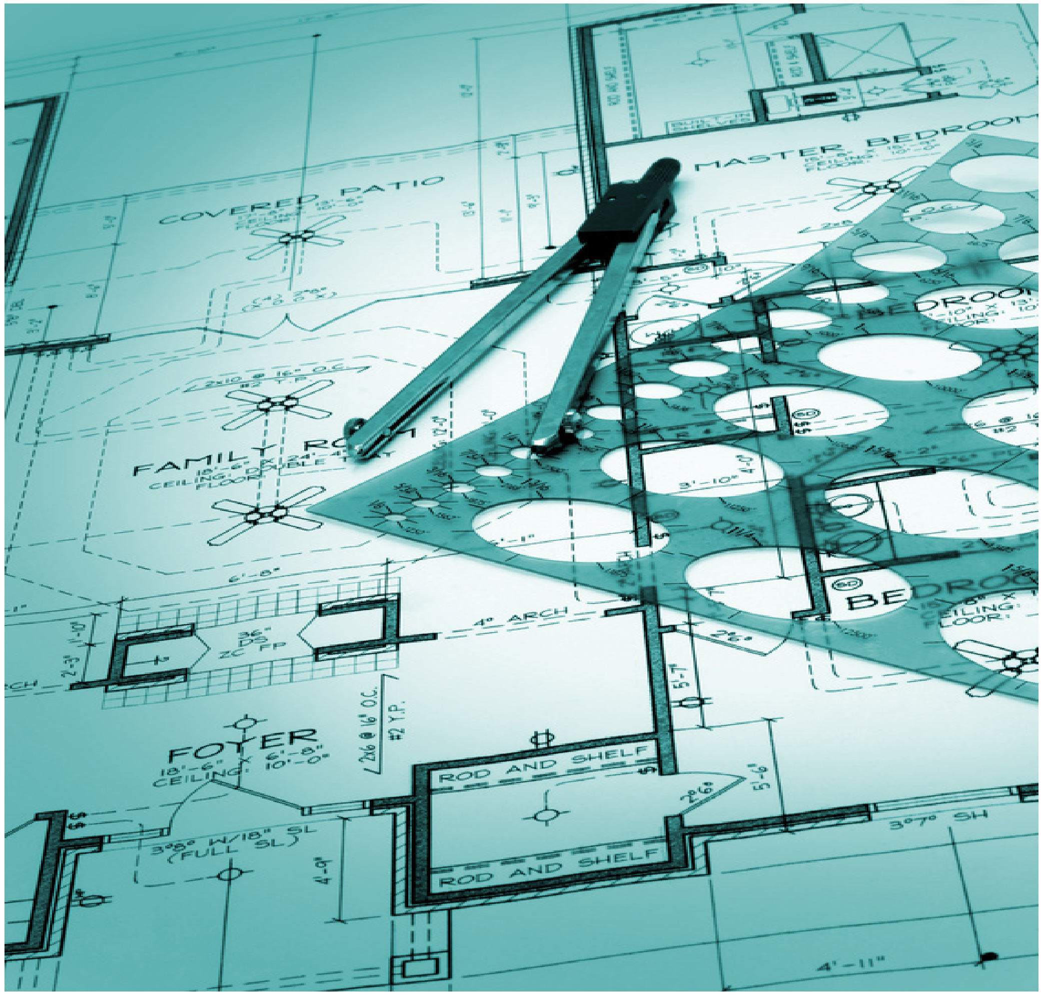
Understanding Apache CXF

Michael Hoffman
@mhi_inc

hardcore dev and IT training

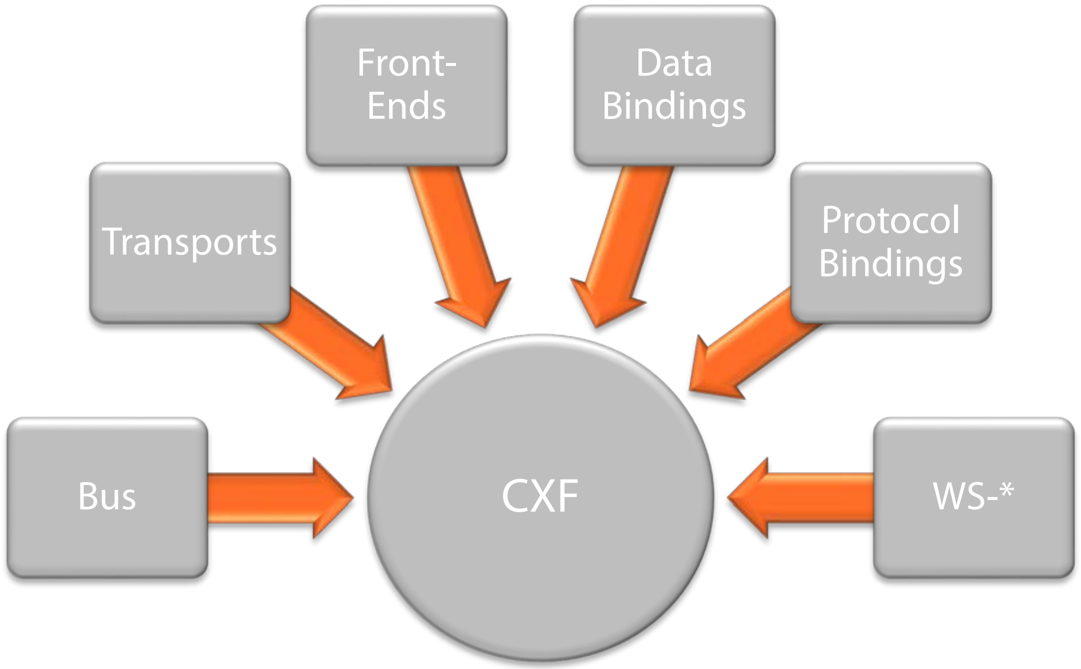
Module Objectives



The Apache Software Foundation

TM The Apache
Software Foundation
http://www.apache.org/

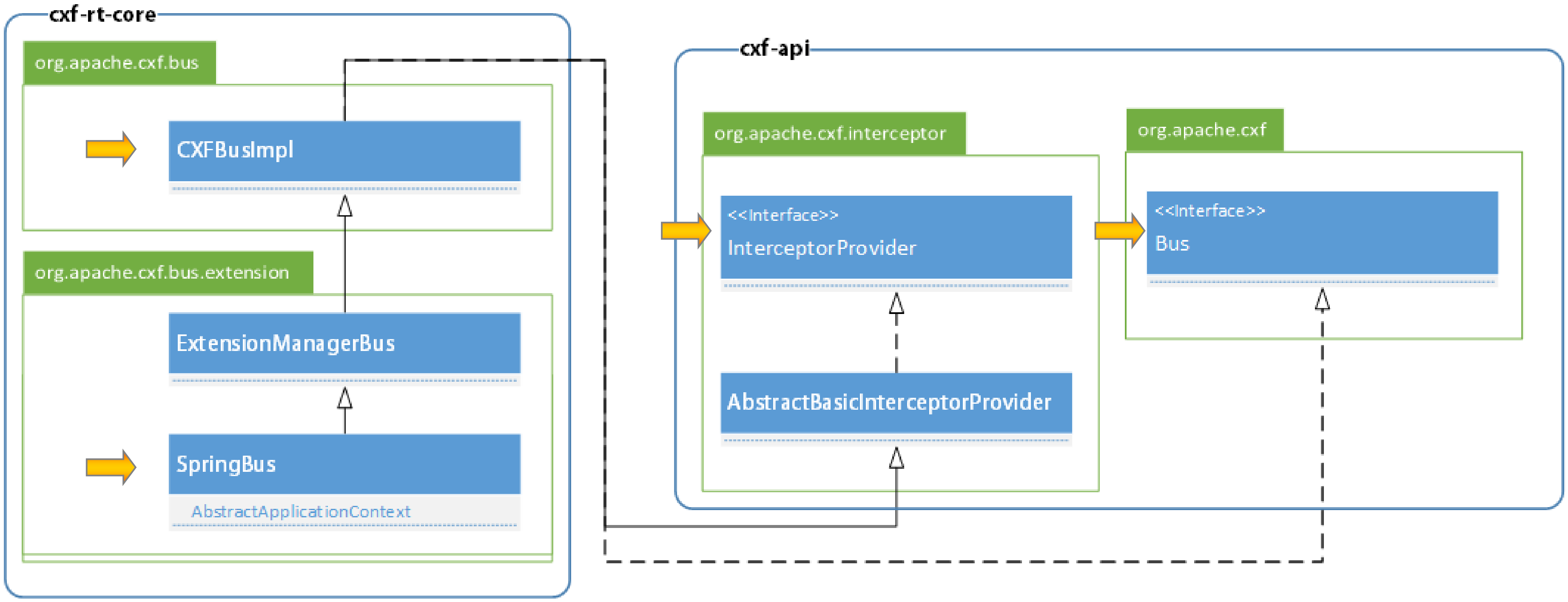
CXF Architecture Overview



Introduction to the CXF Bus

- Spring-based registry of components
- Highly customizable
- Minimal configuration

CXF Bus Class Diagram



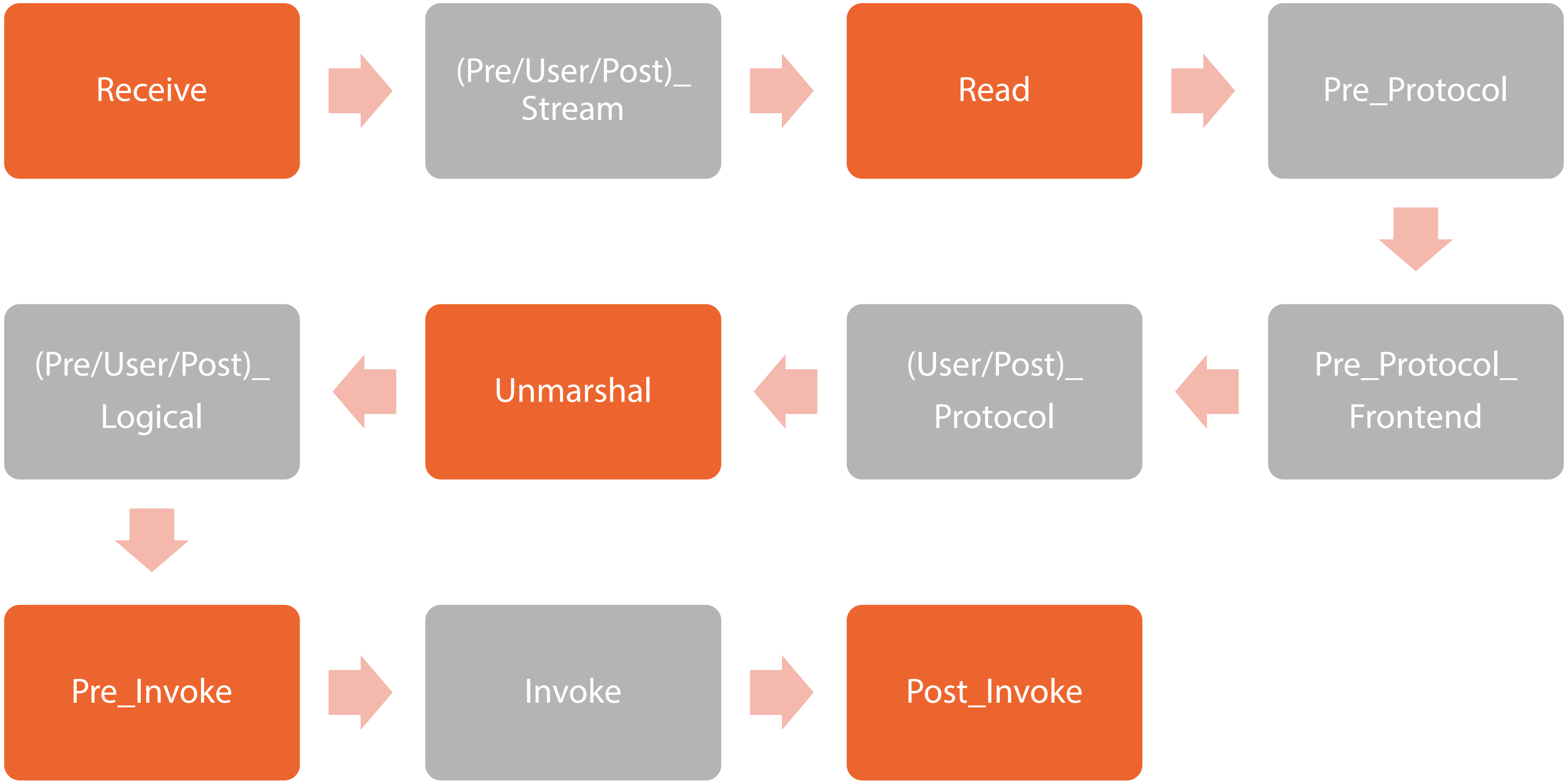
Interceptors and Messages

- **Interceptors**
  - Follows Interceptor design pattern
  - Provides message and fault handlers
  - Can be chained together
- **Messages**
  - Container for data to be passed through interceptor chains

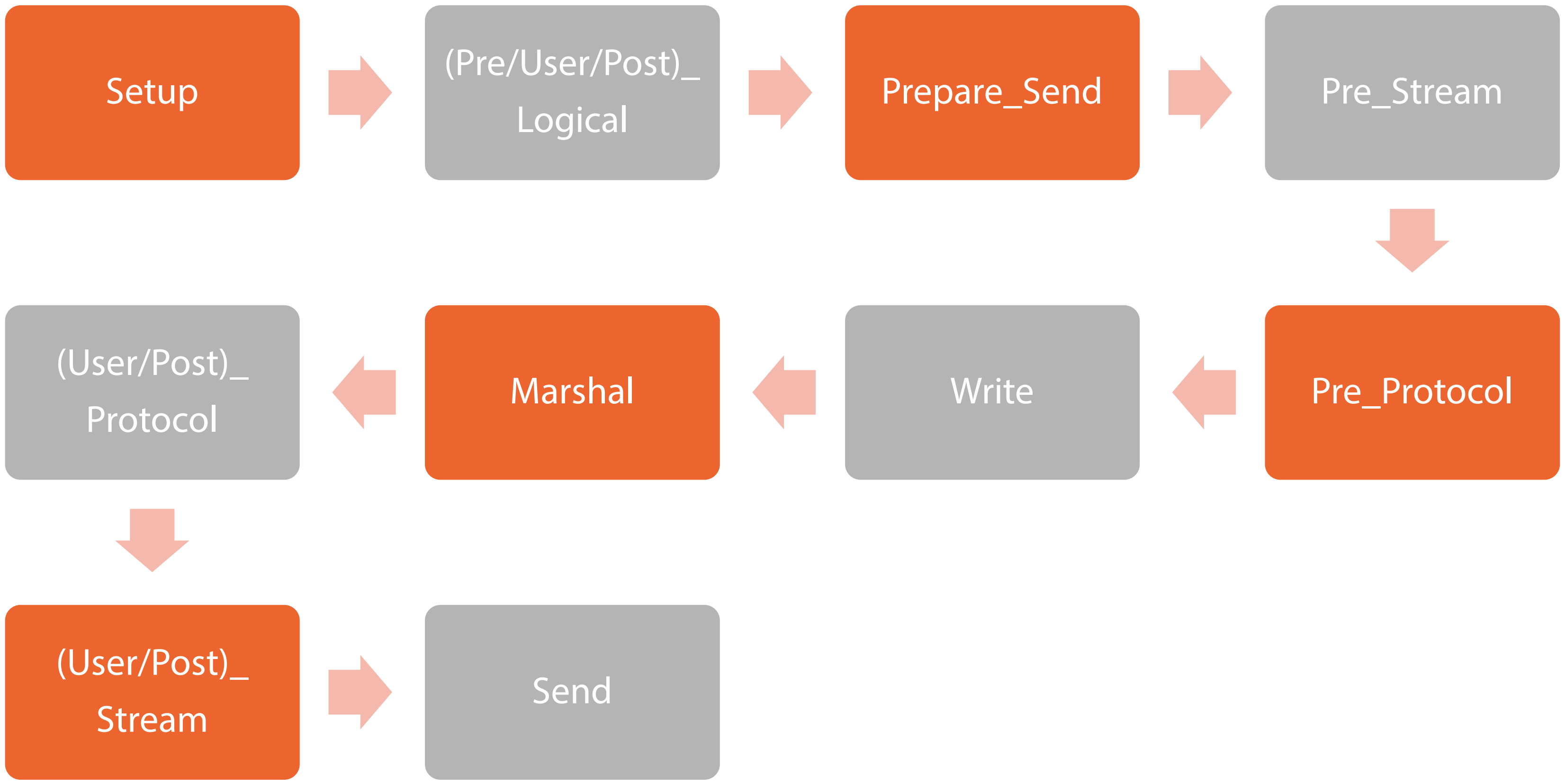
Interceptor Phases

- **Three logical groupings of phases**
  - Incoming chains
  - Outgoing chains
  - Post-processing chains

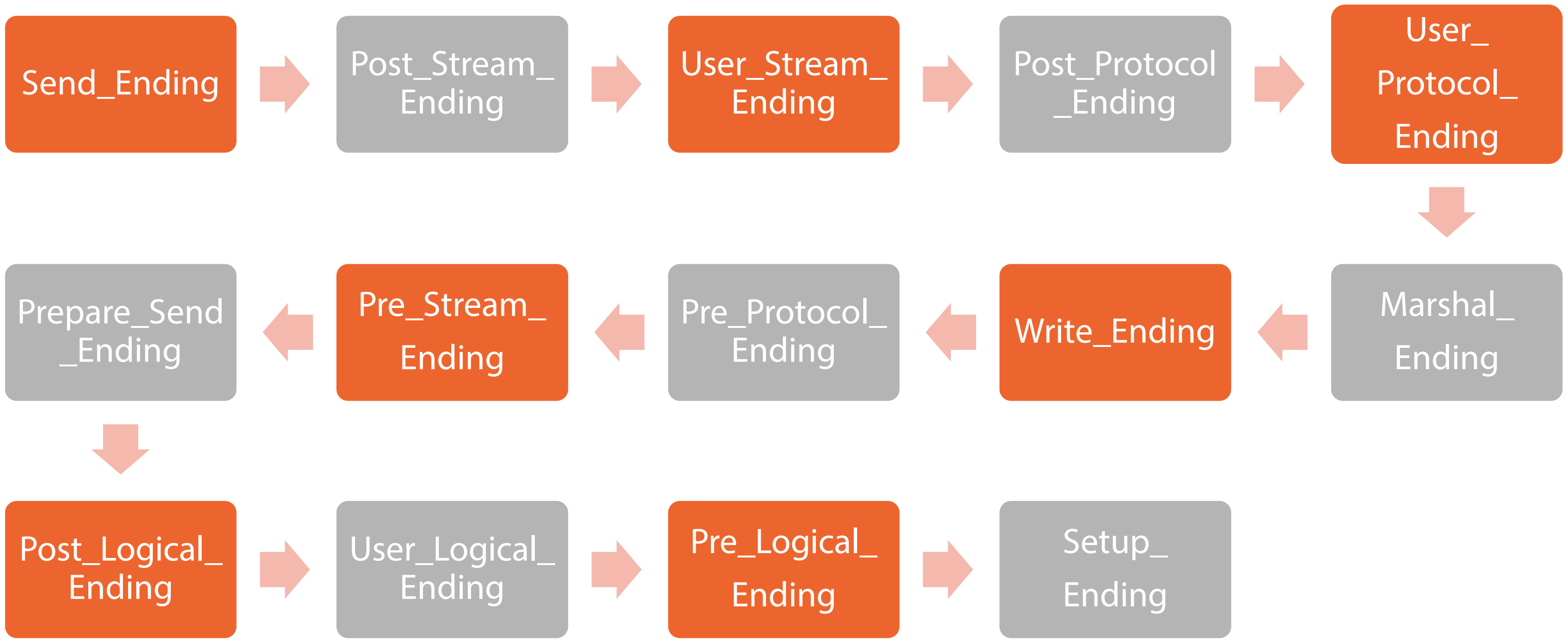
Phases for Incoming Interceptor Chains



Phases for Outgoing Interceptor Chains



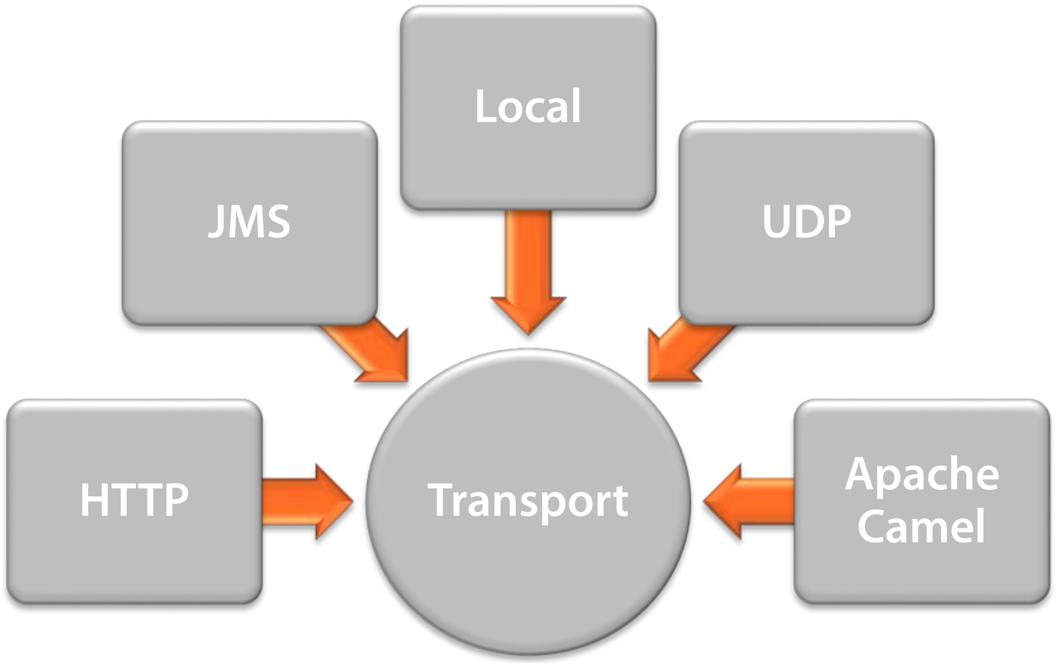
Phases for Post-Processing Interceptor Chains



JAX-WS Interceptor Chain

Attachment In	• RECEIVE
StAX In	• POST_STREAM
SOAP Action In	• READ
Must Understand	• PRE_PROTOCOL
SOAP Handler	• PRE_PROTOCOL_FRONTEND
Check Fault	• POST_PROTOCOL
SOAP Header	• UNMARSHAL
Wrapper Class In	• POST_LOGICAL
SwA In	• PRE_INVOKE
Service Invoker	• INVOKE

CXF Transport Options



HTTP Transport Options

Deploy web services to a web container?

OR

Deploy web services with an embedded web container?

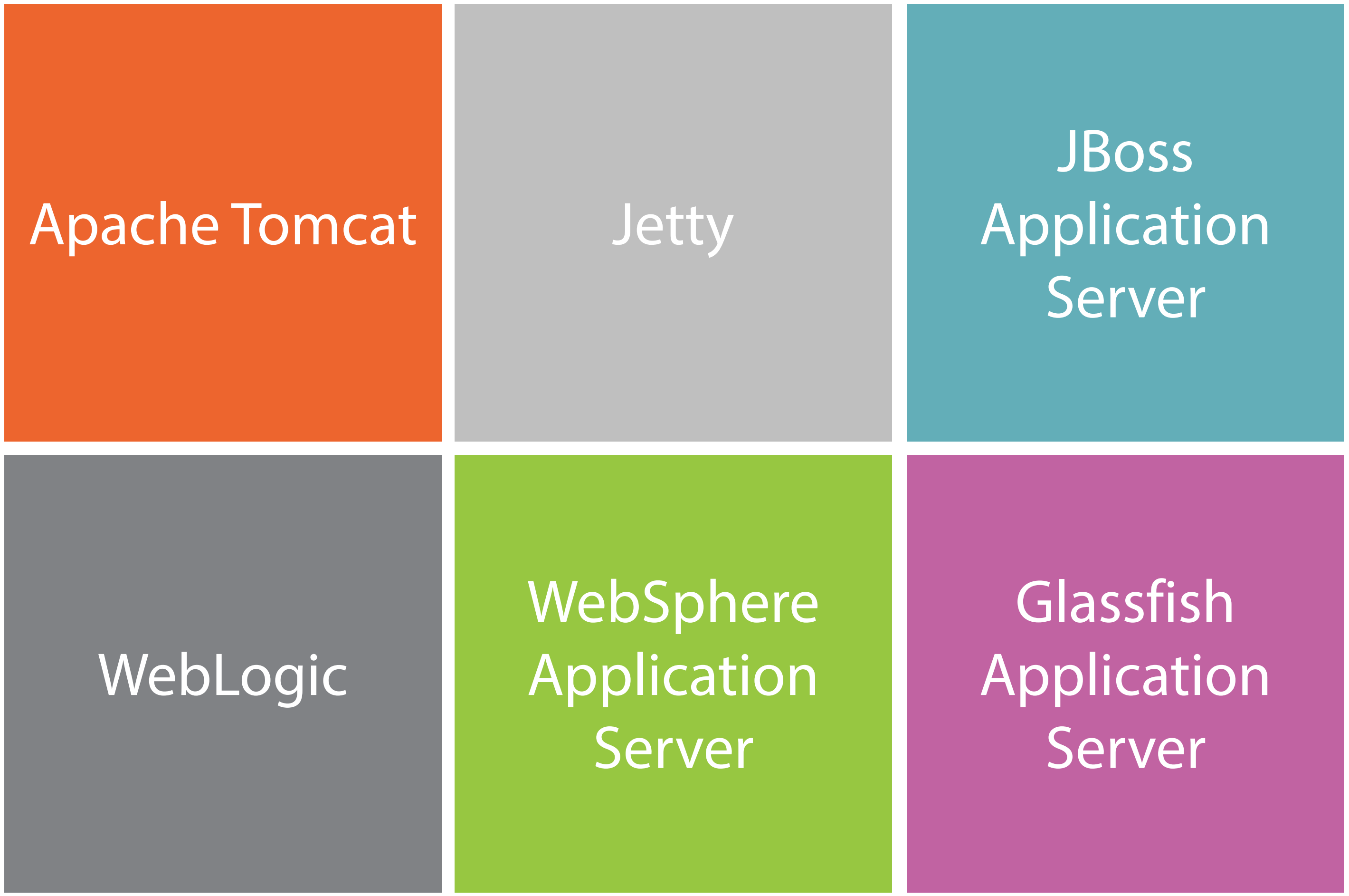
CXF Servlet

- Request processing for web service endpoints
- Available in the cxf-rt-transports-http library
- Supports the creation of Spring’s application context

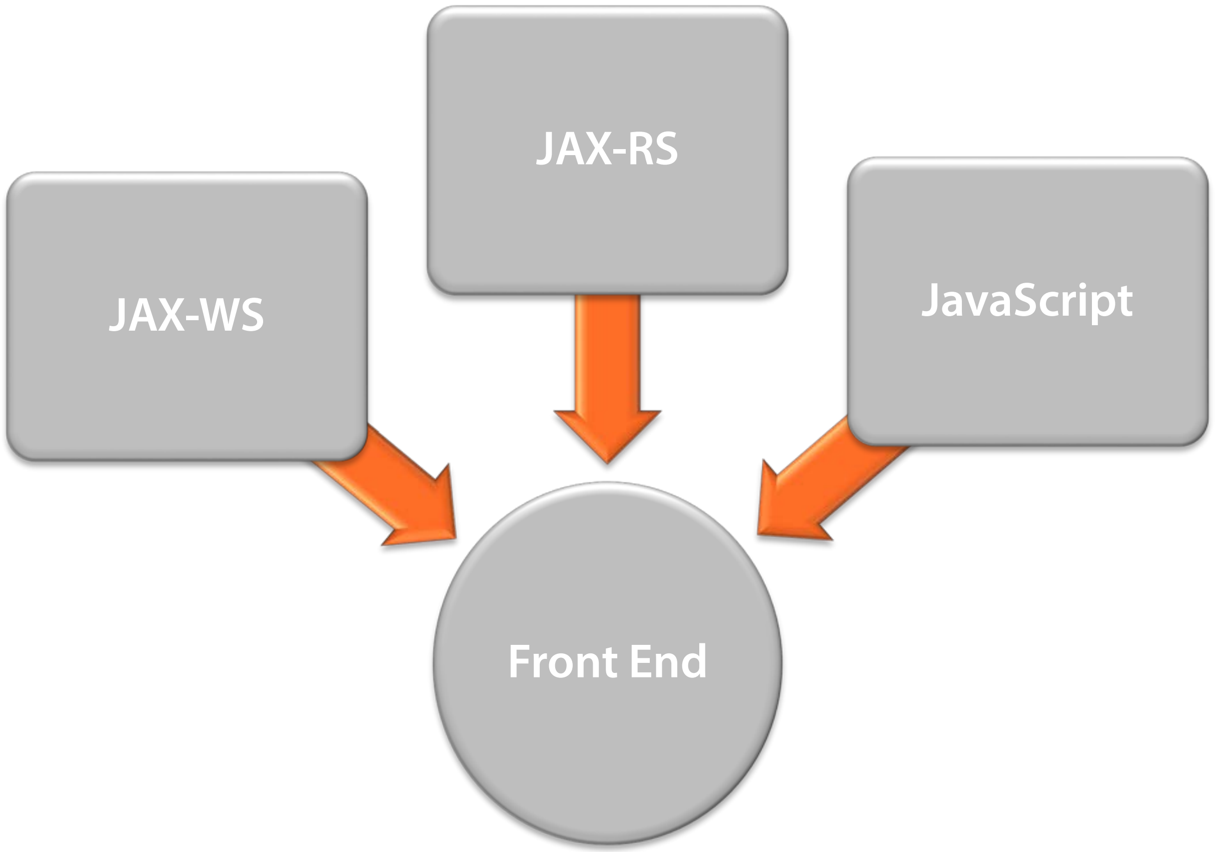
CXF Servlet Web Descriptor

```
<servlet>
    <servlet-name>CXFServlet</servlet-name>
    <display-name>CXF Servlet</display-name>
    <servlet-class>
        org.apache.cxf.transport.servlet.CXFServlet
    </servlet-class>
    <load-on-startup>1</load-on-startup>
</servlet>

<servlet-mapping>
    <servlet-name>CXFServlet</servlet-name>
    <url-pattern>/*</url-pattern>
</servlet-mapping>
```

CXF Web Container Support



CXF Front End Options



JAX-WS

- Specification for Java XML-based web services
- Typically implemented using a WSDL with SOAP over HTTP
- CXF provides full support for JAX-WS

JAX-RS

- **Specification for Java RESTful web services**
  - Supports plain old Java objects (POJO) through URIs.
- **CXF Support for JAX-RS versions**
  - JAX-RS 2.0 – CXF 2.x mostly supports, CXF 3.x fully supports
  - JAX-RS 1.1 – CXF 2.x+ fully supports.

Deciding Between JAX-WS and JAX-RS

JAX-WS	JAX-RS
☐ Distributed component integration	☐ Mobile and web view integration
☐ Complex operations	☐ Simple transactions
☐ Standards-based	☐ Limited constraints
☐ Multiple transports	☐ HTTP transport

Implementing JAX-WS

- Available in the cxf-rt-frontend-jaxws library
- Supported by a variety of transports
- Configured through Spring application context

JAX-WS Endpoint Bean Configuration

```
<jaxws:endpoint

  id="helloWorld"

  implementor="com.pluralsight.cxfdemo.HelloWorldImpl"

  address="/HelloWorld" />
```

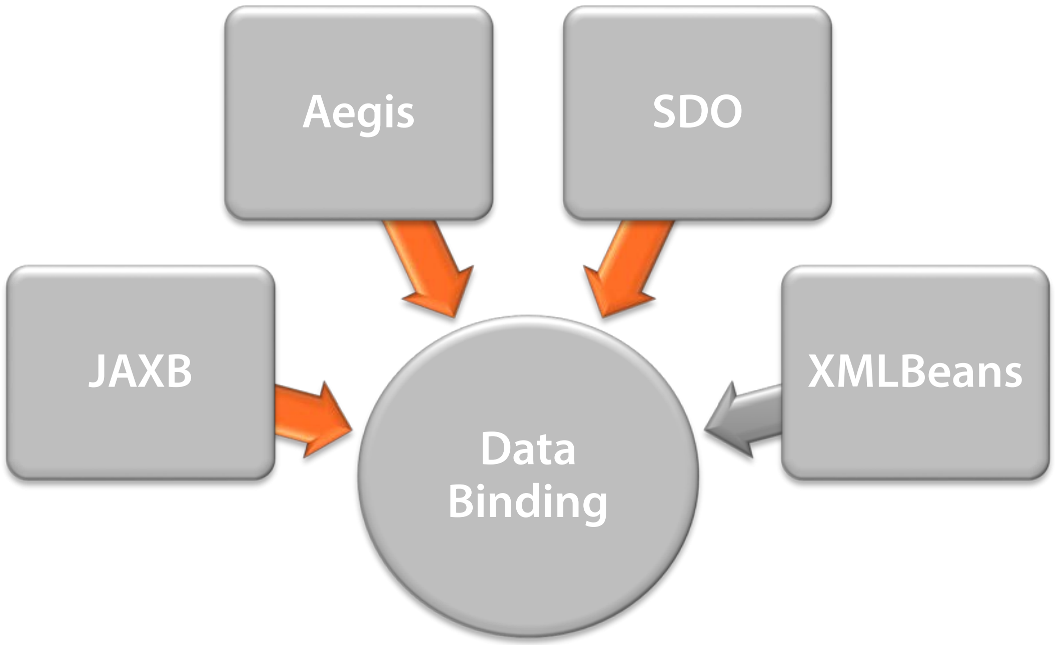
Implementing JAX-RS

- Available in the cxf-rt-frontend-jaxrs library
- Supported by the HTTP transport and CXF Servlet
- Configured through Spring application context

JAX-RS Endpoint Bean Configuration

```
<jaxrs:server id="services" address="/">

  <jaxrs:serviceBeans>

    <bean class="com.pluralsight.demo.jaxrs.HelloWorld" />

  </jaxrs:serviceBeans>

  <jaxrs:providers>

    <bean
   class="org.codehaus.jackson.jaxrs.JacksonJsonProvider"/>

  </jaxrs:providers>

</jaxrs:server>
```

CXF Data Binding Options



JAXB

- Java architecture for XML binding
- Binding based on XML schema defintion
- Unmarshal and marshal

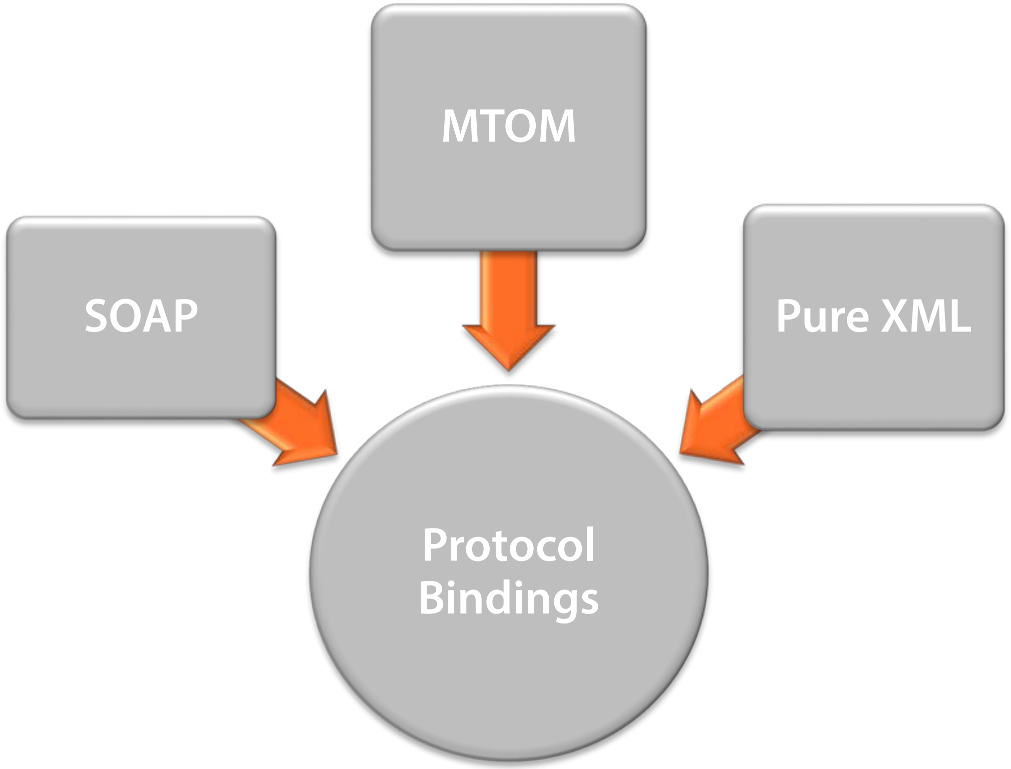
JAXB Configuration

```
<jaxws:endpoint

  id="helloWorld"

  implementor="com.pluralsight.cxfdemo.HelloWorldImpl"

  address="/HelloWorld">

  <jaxws:dataBinding>

    <bean class="com.apache.cxf.jaxb.JAXBDataBinding" />

  </jaxws:dataBinding>

</jaxws:endpoint>
```

CXF XJC Maven Plugin

- XJC is a binding compiler executed through a command prompt
- Generates Java code based on an XSD
- CXF supports XJC through a Maven plug-in

CXF Protocol Binding Options



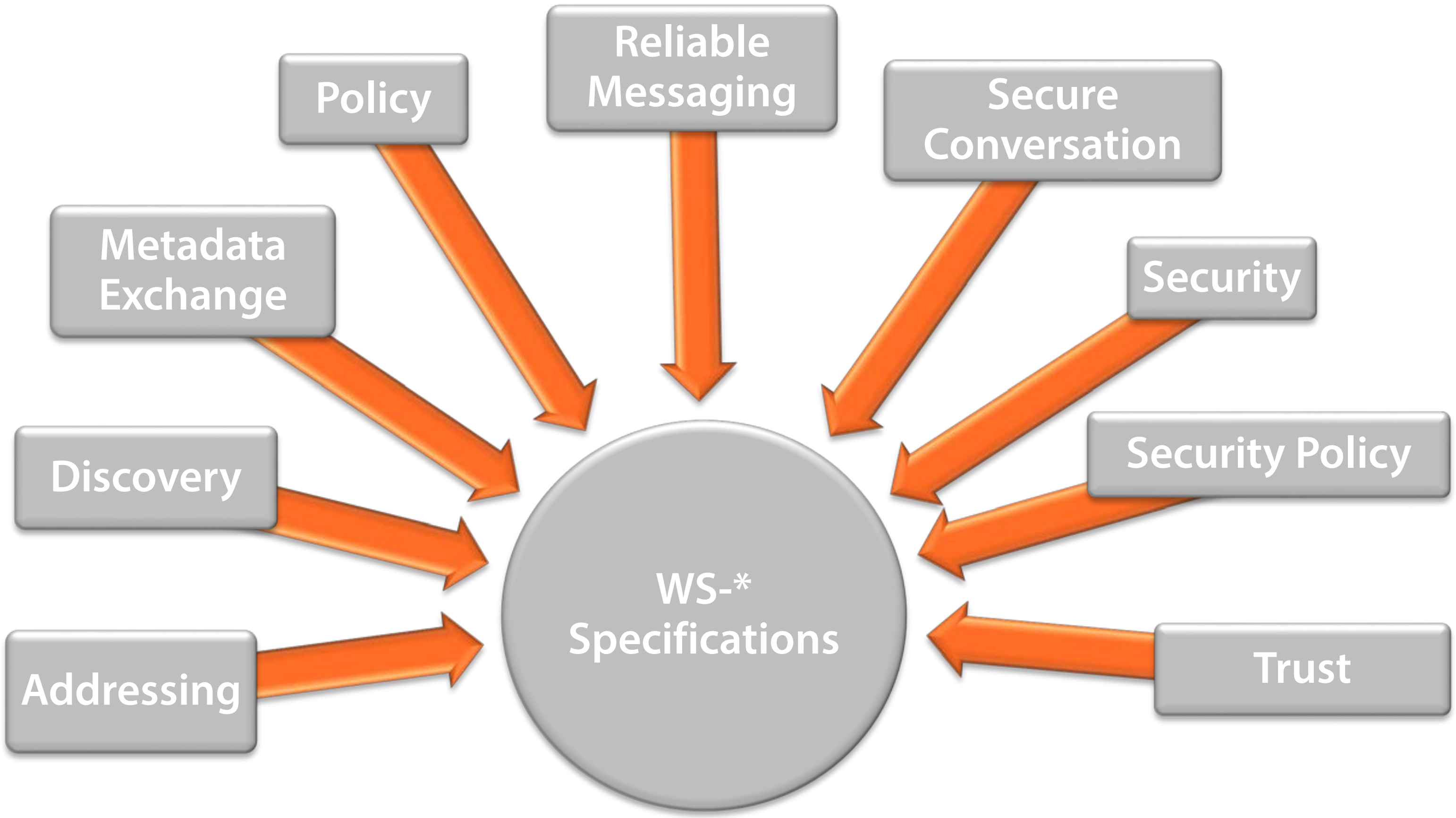
SOAP Protocol Binding

- Language that defines service message format
- Data is passed in an envelope that contains a header and body
- Configured as part of the WSDL binding section

SOAP Protocol Configuration

```
<wsdl:binding name="HelloWorldImplServiceSoapBinding"
   type="tns:HelloWorld">

   <soap:binding transport="http://schemas.xmlsoap.org/soap/http"
     style="document"/>

   <wsdl:operation name="sayHi">
     <soap:operation style="document" soapAction=""/>
     <wsdl:input name="sayHi">
       <soap:body use="literal"/>
     </wsdl:input>
     <wsdl:output name="sayHiResponse">
       <soap:body use="literal"/>
     </wsdl:output>
   </wsdl:operation>

</wsdl:binding>
```

WS-* Specification Options



WS-Addressing

- Provides a standard way of adding address information to a SOAP header
- Intended to support more complex enterprise-level solutions
- Include an endpoint reference and endpoint reference properties
- CXF supports as part of the JAX-WS endpoint schema

WS-Discovery

- Provides support for a multicast protocol that auto-discovers services on a local network
- Leverages the UDP protocol for communication
- CXF supports discovery through an API
- Available across two dependencies, cxf-services-ws-discovery-service and cxf-services-ws-discovery-api

WS-MetadataExchange

- Defines how metadata is represented for a web service endpoint
- Configured in the SOAP header
- Available through the dependency cxf-rt-ws-mex

WS-Policy

- **A framework and model for web service policies**
  - Domain-specific capabilities
  - Requirements
  - General characteristics
- **Configured in CXF through WSDL, Spring or API**
- **Available through the dependency cxf-rt-ws-policy**

WS-ReliableMessaging

- Defines a protocol for reliable message delivery between distributed systems
- Available through the dependency cxf-rt-ws-mex

WS-Security

- Provides security features beyond the transport level protocol
- Available through the dependency cxf-rt-ws-security

WS-SecurityPolicy

- Provides an easier, standards-based way to configure security
- Uses policies based on the web service policy framework
- Available through the dependency cxf-rt-ws-security

WS-SecureConversation

- Part of the WS-SecurityPolicy specification
- Provides a better performing approach for encrypted communication
- Available through the dependency cxf-rt-ws-security

WS-Trust

- Part of the WS-SecurityPolicy specification
- Supports the issuing, renewing and validation of security tokens
- Available through the dependency cxf-rt-ws-security

Module Summary