Developing a Contract-First Web Service
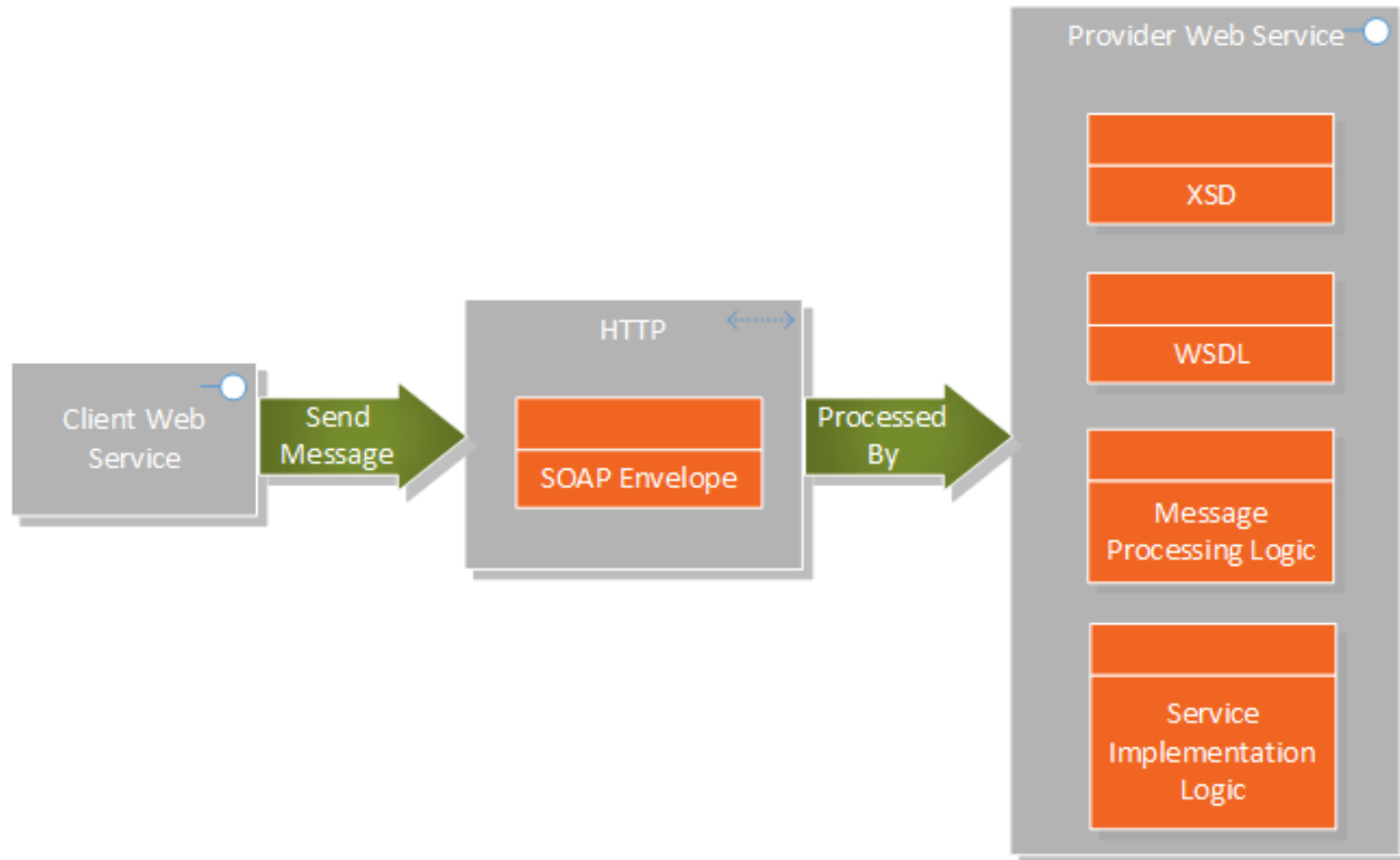
Michael Hoffman
@mhi_inc

Module Objectives

What is a Contract-First Web Service?

➡ ▪ **Inventory of services and schemas defined early in a design phase.**

➡ ▪ **Considered best practice**

  ▫ Schema types are portable

  ▫ Supports logic-to-contract coupling

➡ ▪ **For more on Contract-First vs. Contract-Last: tinyurl.com/spring-contract-first**

Contract-First Design Considerations

→ ■ **Contract granularity**

    □   Breadth of business context

    □   Amount of work

    □   Amount of data

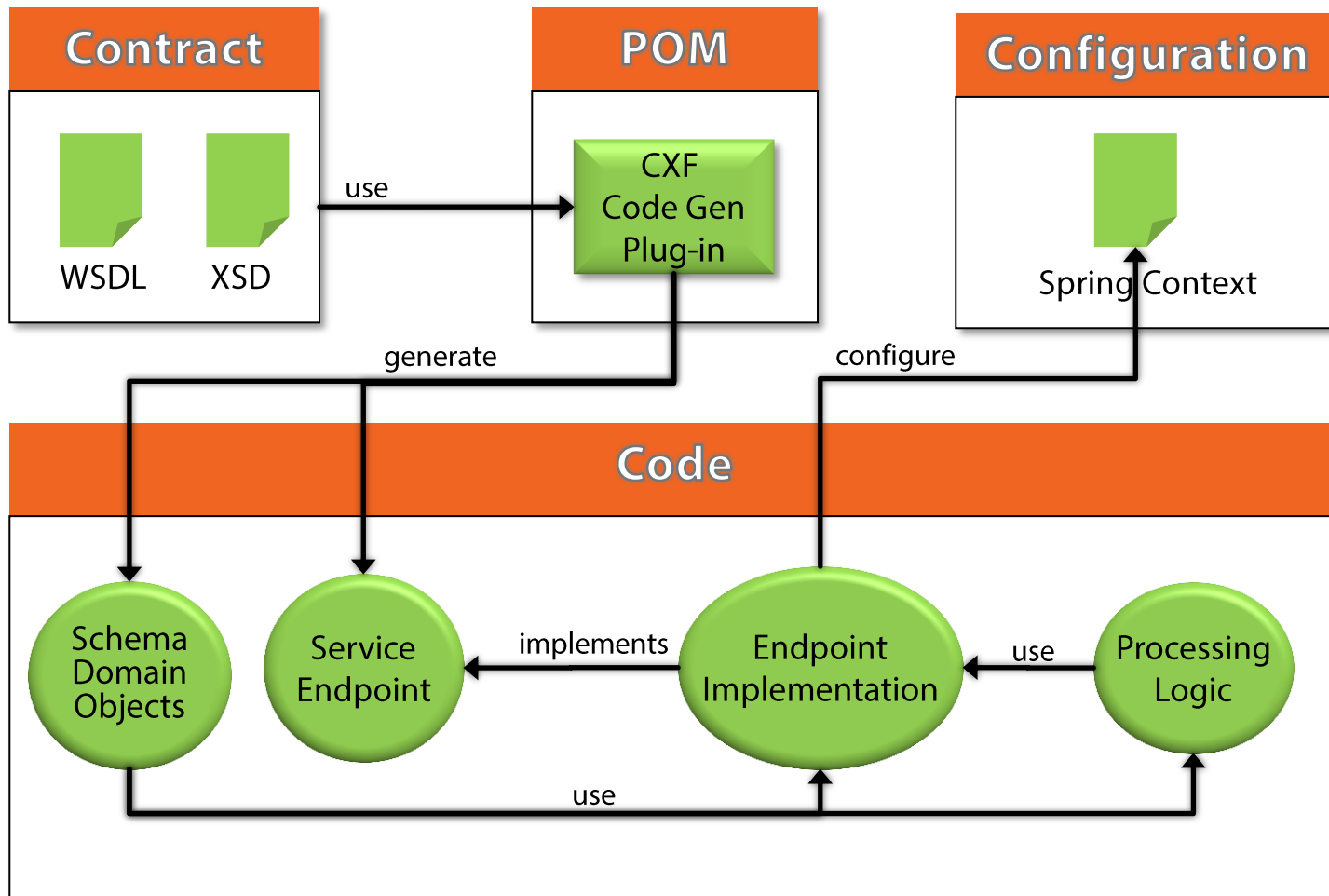   □   Amount of constraints

→ ■ **Re-usability**

→ ■ **Transparency**

Case Study – Book Order Placement

Order Web Service

- **Central place where their clients can place orders**

- **Each request requires three inputs**
  - Account
  - Book
  - Order Inquiry

- **Each response provides two outputs**
  - Account information
  - Order information

Web Service Implementation Overview

WSDL Contract

- What does the contract do?

- How should the contract services be accessed?

- Where can the contract services be accessed from?

Building the WSDL

- **Create a new project**
  - Maven archetype:

- **Contract should tell how services can be accessed**

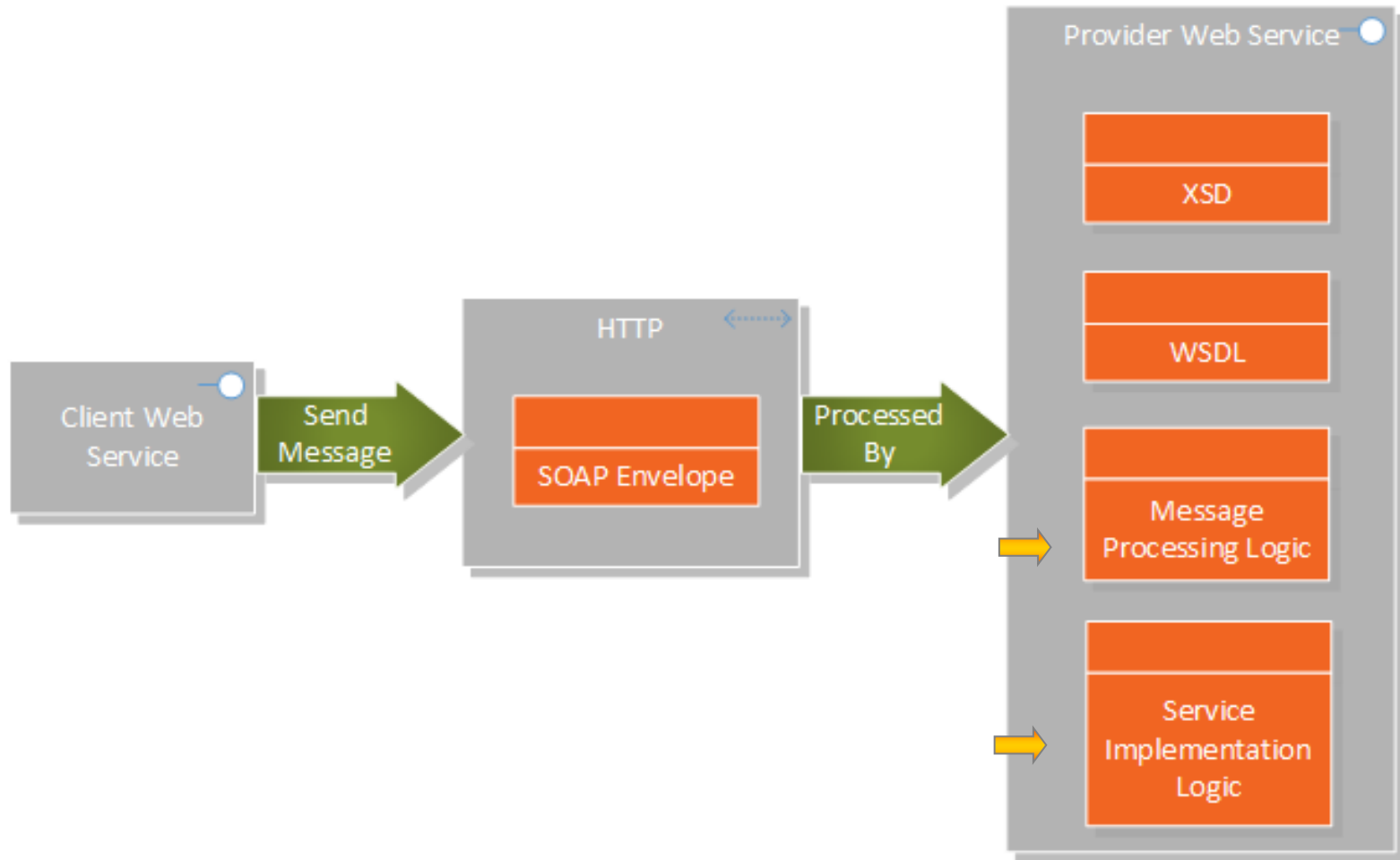- **Contract should tell where services can be accessed from**

Building the XML Schema

- Organizes messages into re-usable structures

- Defines the business domain language

- Separated from the WSDL

CXF CodeGen Maven Plugin

- Generates Java code based on a WSDL

- Executes wsdl2java through a command

- Reduces custom coding for WSDL binding

Implementing the Service Endpoint

JAX-WS Endpoint Bean Configuration

```xml
<jaxws:endpoint

  id="orders"

  implementor=
   "com.pluralsight.cxfdemo.orders.DefaultOrdersEndpoint"

  address="/services/orders" />
```

JAX-WS Endpoint Bean Configuration

```
<jaxws:endpoint

    address=""
    id=""
    implementor=""
    bindingUri=""
    bus=""
    endpointName=""
    implementorClass=""
    publish=""
    publishedEndpointUrl=""
    serviceName=""
    transportId=""
    wsdlLocation=""

    />
```

JAX-WS Endpoint Bean Configuration

```
<jaxws:endpoint>

    <jaxws:dataBinding />
    <jaxws:features />
    <jaxws:handlers />
    <jaxws:inInterceptors />
    <jaxws:inFaultInterceptors />
    <jaxws:outInterceptors />
    <jaxws:outFaultInterceptors />
    <jaxws:properties />

</jaxws:endpoint>
```

Module Summary