



# Overview

1. Introduction to Web Service?
2. Restful Web Services with Spring Boot
3. Spring Boot Programming & Basics
4. Spring Data JPA

# What is Web Service?

A service which is delivered over the web/internet

# User Management Portal


Users



Users











Settings

Welcome, Rick 

Search users...

+New User






Photo	User ID	First Name	Last Name	Username	Email	Status	Actions
	1091961105	Rick	Richarson	rick	rick@email.com	Active	 
	2527123140	John	Doe	johnny	john@email.com	Active	 
	6394024156	Vaishnavi	Hiwase	vaisnavi	vaish@gmail.com	Active	 

Search users...

+New User



Photo	User ID	First Name	Last Name	Username	Email	Status	Actions
	1091961105	Rick	Richarson	rick	rick@email.com	Active	 

## Rick Richarson



Rick Richarson

Joined Apr 6, 2021, 9:28:18

rick

PM

Status: **Active**

Last Login:

1091961105



rick@email.com



SUPER\_ADMIN



Account Unlocked



Close

## Edit Rick Richarson



First Name

Rick

Last Name

Richarson

Username

rick

Email

rick@email.com

Role

SUPER ADMIN



Profile Picture

Choose File

Browse

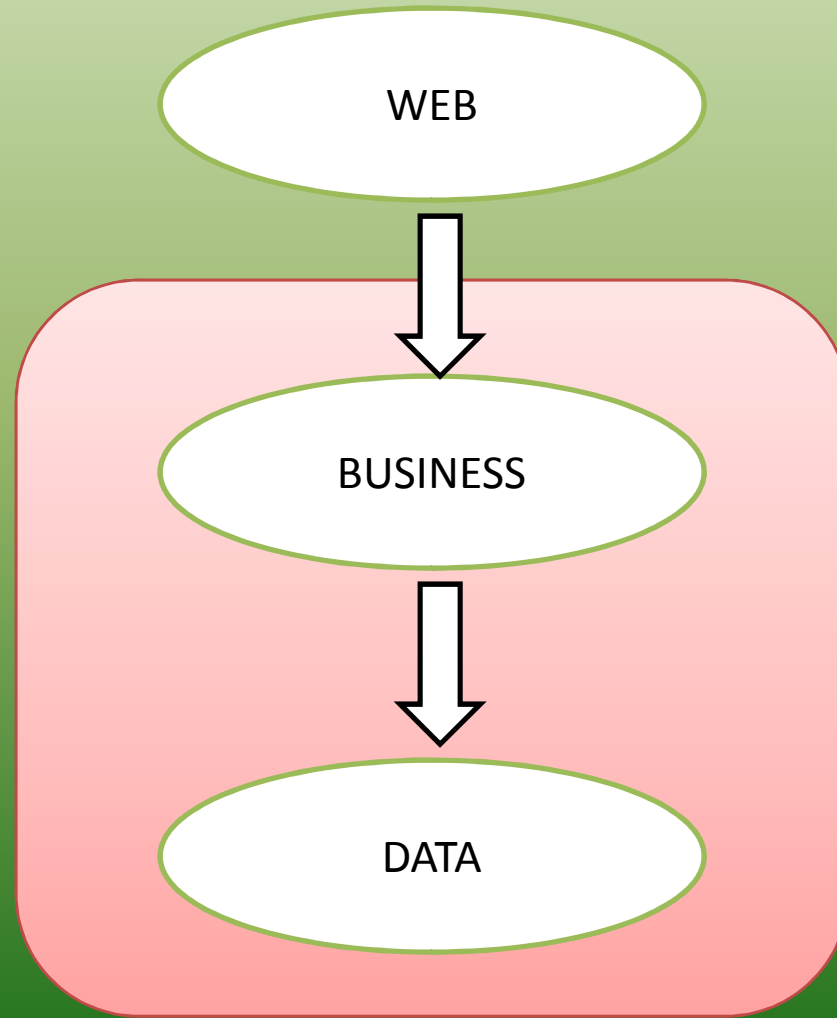
☒ Active

☒ Unlocked

Close

Save changes

# Application Layers

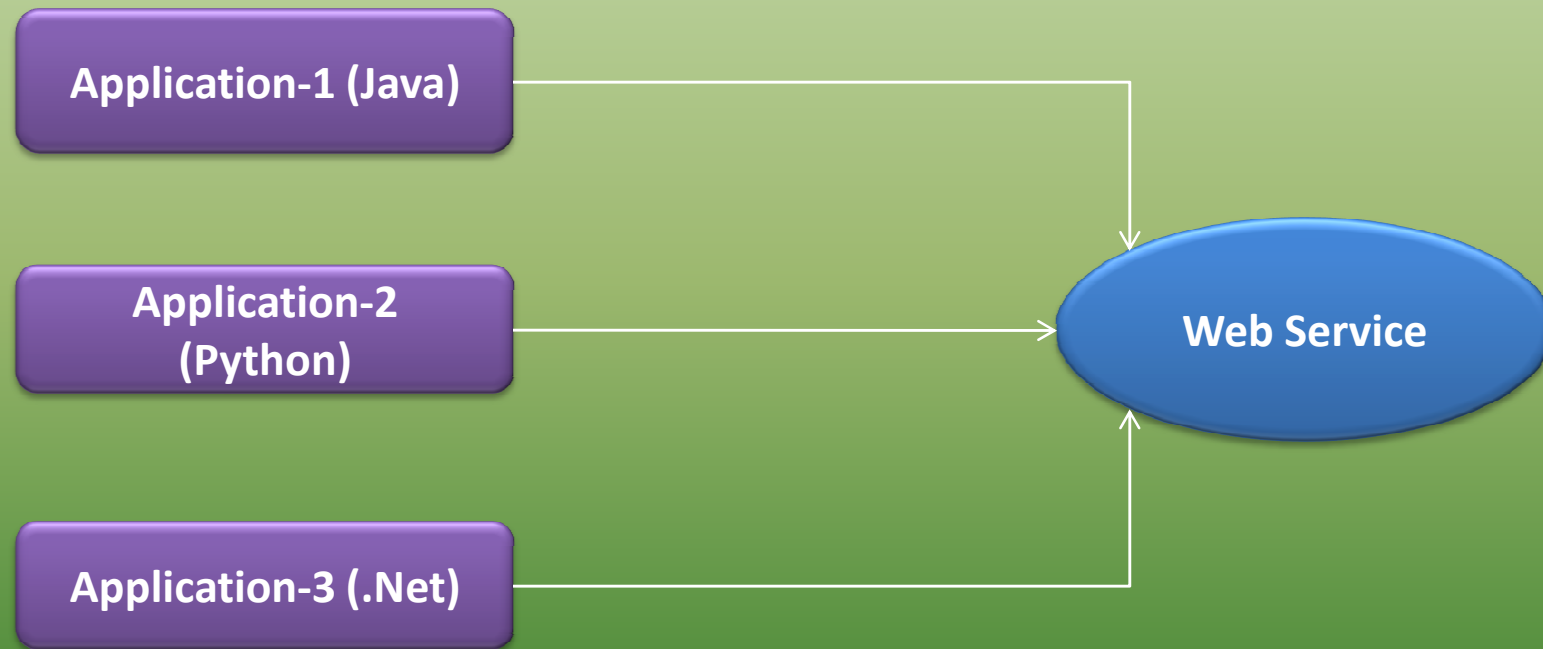


# Web Service – W3C Definition

A web service is a software system designed to support interoperable machine-to-machine interaction over a network

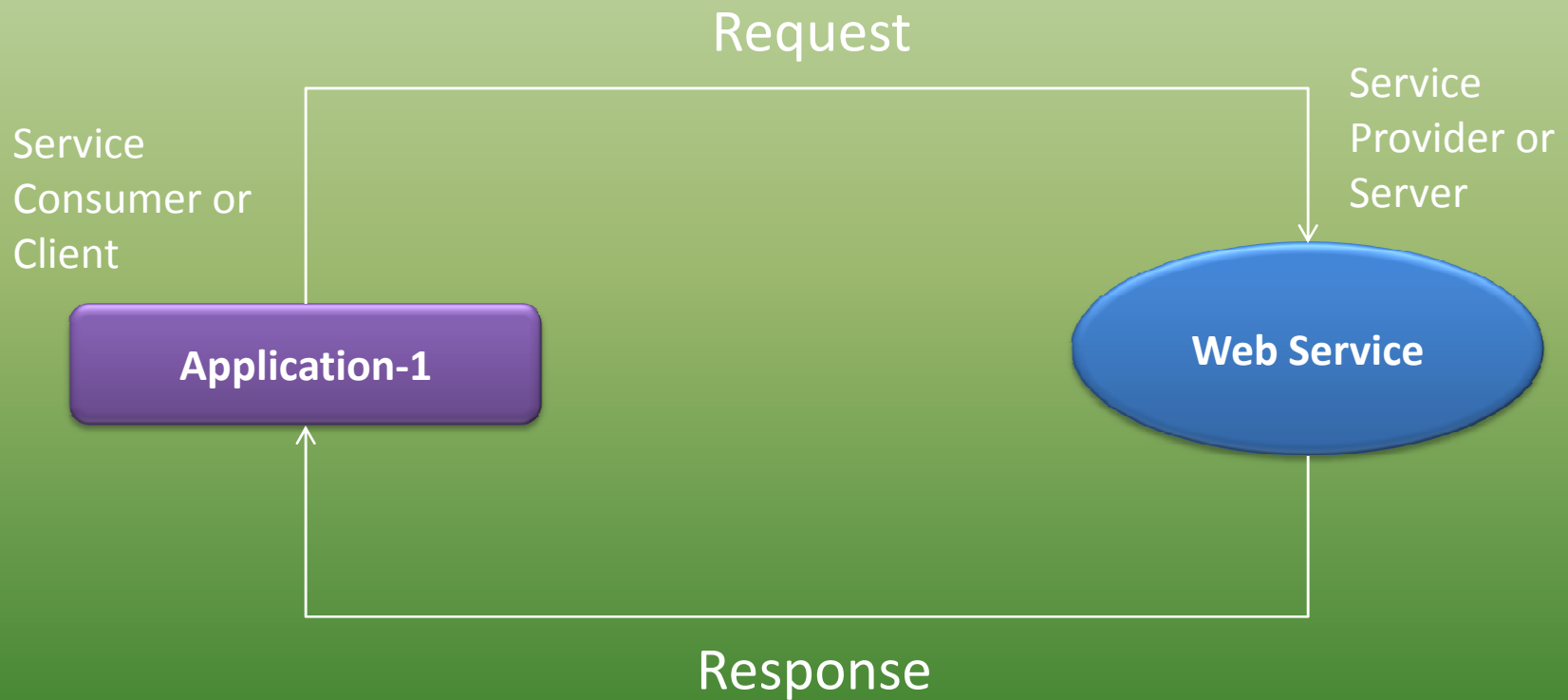
1. Design for Machine to machine (or application to application) interaction.
2. It should be Interoperable (Not platform dependent)
3. Should allow communication over network

# Web Service





# Web Service



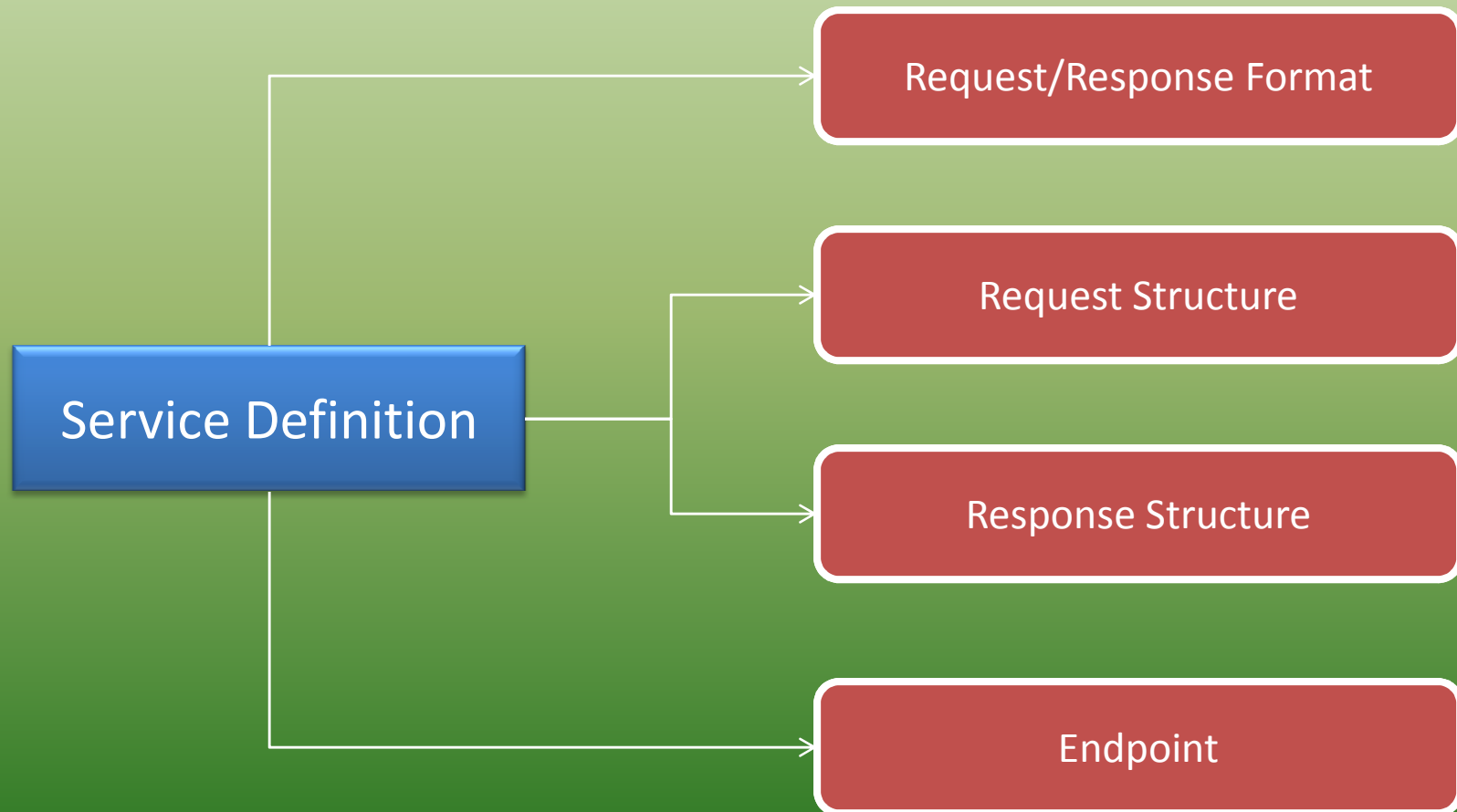
# Request / Response Formats

1. XML (Extensible Markup Language)
2. JSON (Java Script Object Notation)

```
- <Parts>
- <Part>
  <Id>4478</Id>
  <Part_Name>1000 Ohm Resistor</Part_Name>
  <Total_Available>25000</Total_Available>
  <Price>0.01</Price>
</Part>
- <Part>
  <Id>3328</Id>
  <Part_Name>15000 Ohm Resistor</Part_Name>
  <Total_Available>75000</Total_Available>
  <Price>0.02</Price>
</Part>
- <Part>
  <Id>4725</Id>
  <Part_Name>555 Timer IC</Part_Name>
  <Total_Available>1500</Total_Available>
  <Price>0.25</Price>
</Part>
</Parts>
```

```
{
  "orders": [
    {
      "orderno": "748745375",
      "date": "June 30, 2088 1:54:23 AM",
      "trackingno": "TN0039291",
      "custid": "11045",
      "customer": [
        {
          "custid": "11045",
          "fname": "Sue",
          "lname": "Hatfield",
          "address": "1409 Silver Street",
          "city": "Ashland",
          "state": "NE",
          "zip": "68003"
        }
      ]
    }
  ]
}
```

# Web Service



# Web Services Group

- SOAP Base Web Services (XML)
- Restful Web Services (Architectural approach)

# SOAP Based Web Services

- Simple Object Access Protocol
- No longer used this abbreviation, now its simple SOAP
- SOAP defines specifics way of building web services
- SOAP used XML based request formats (XML request and response)

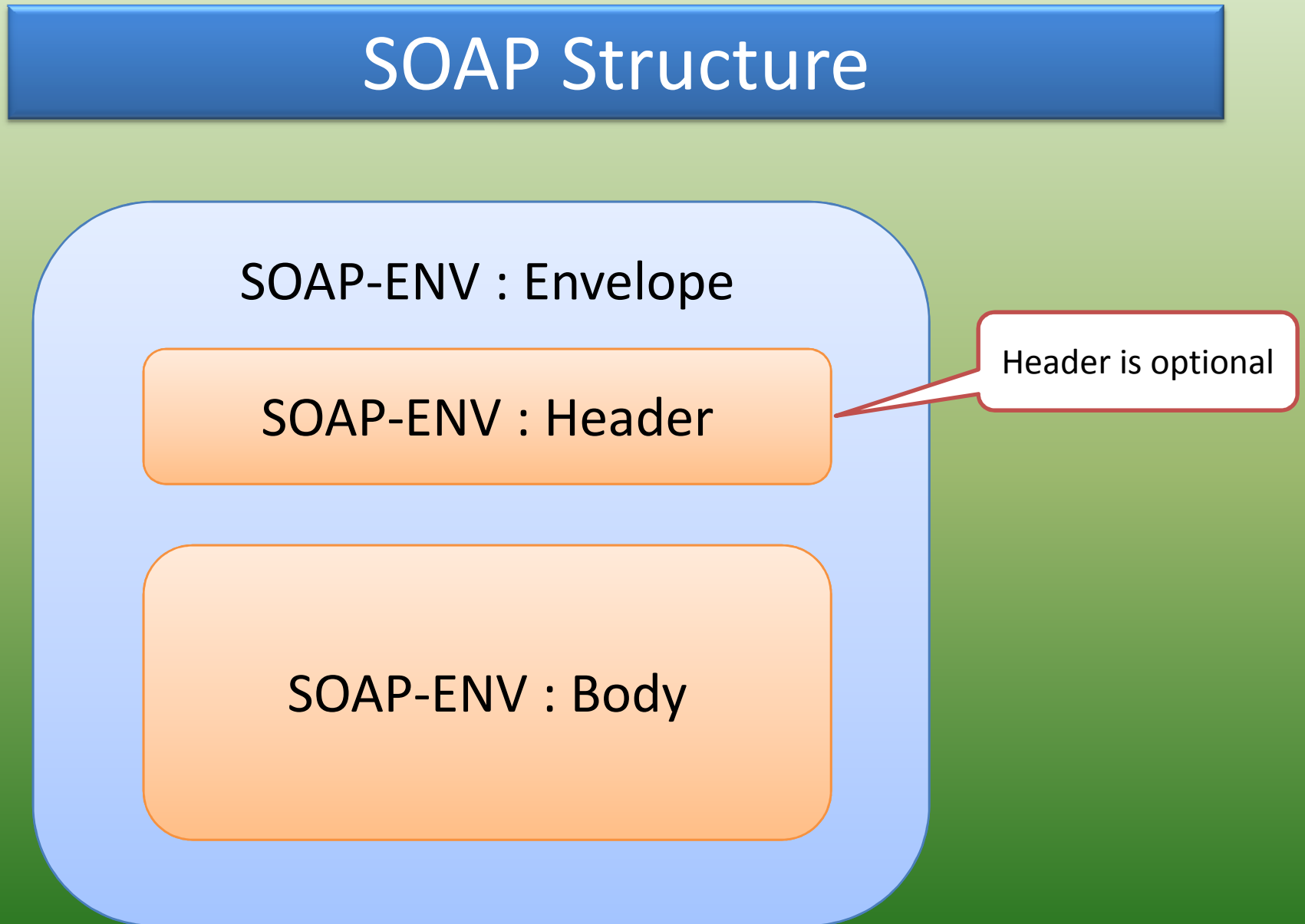
# SOAP Structure

SOAP-ENV : Envelope

SOAP-ENV : Header

Header is optional

SOAP-ENV : Body



# SOAP Structure

POST http://www.webservicex.com/globalweather.asmx HTTP/1.1

Host: www.webservicex.com

Content-Length: 341

Content-Type: application/soap+xml

```
<?xml version="1.0"?>
```

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope">
```

```
  <soap:Header></soap:Header>
```

```
  <soap:Body>
```

```
    <ns1:GetWeather xmlns:ns1='http://www.webserviceX.NET'>
```

```
      <ns1:CityName>Montreal</ns1:CityName>
```

```
      <ns1:CountryName>Canada</ns1:CountryName>
```

```
    </ns1:GetWeather>
```

```
  </soap:Body>
```

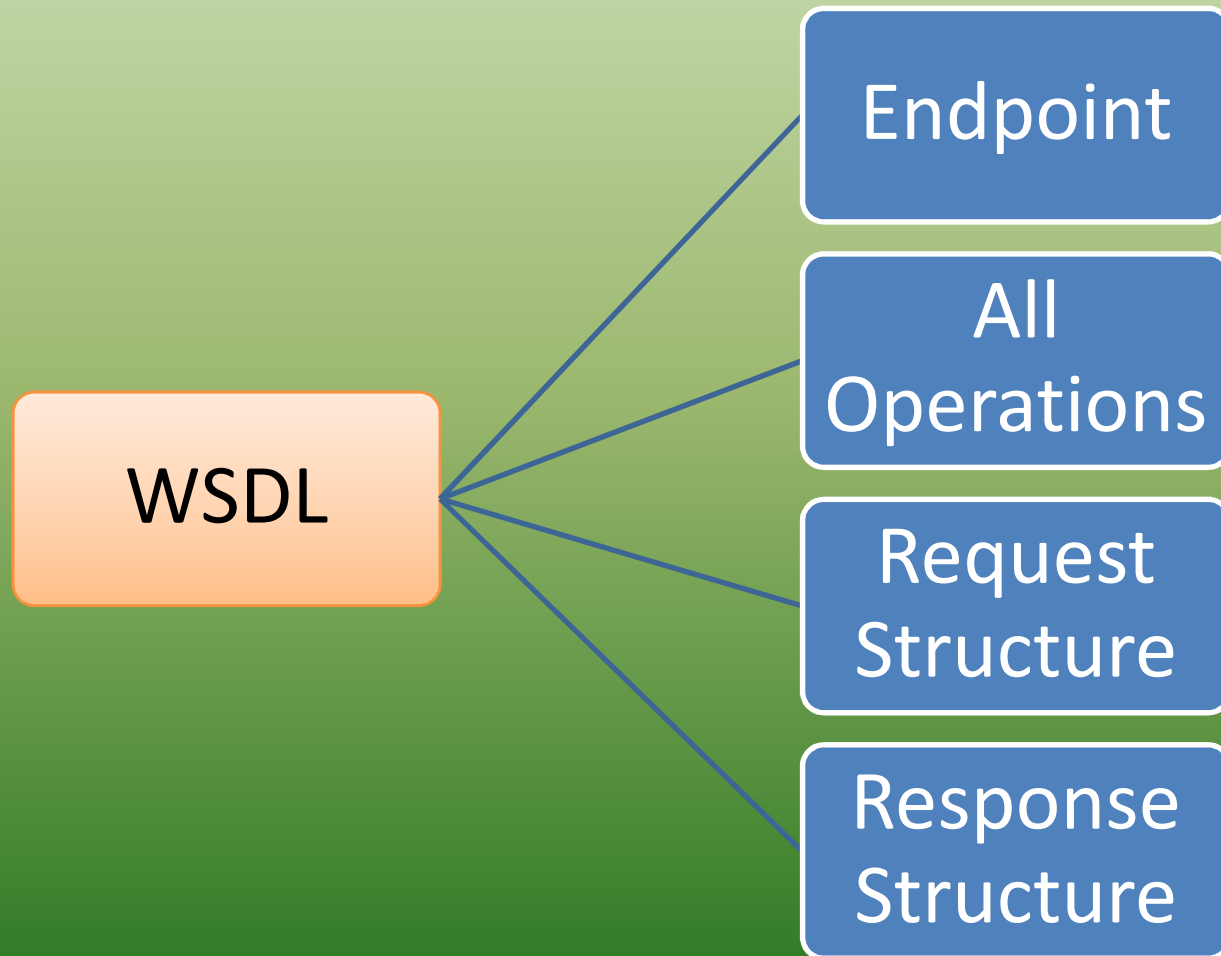
```
</soap:Envelope>
```

# SOAP Structure

- Format
  - SOAP XML Request
  - SOAP XML Response
- Transport
  - SOAP over MQ
  - SOAP over HTTP
- Service Definition
  - WSDL (Web Service Description Language)



# SOAP - WSDL



CalculatorSoap

Overview Service Endpoints WSDL Content WS-I Compliance

CalculatorSoap

- Anonymous Complex Types
- Global Elements
- Schemas
- Messages
- PortTypes
- Bindings
- Services

calculator.asmx?wsdl

http://www.dneonline.com/calculator.asmx?wsdl

```
1 <wsdl:definitions targetNamespace="http://tempuri.org/" xmlns:soap="http://schemas.xmlsoap.org/soap/">  
2 <wsdl:types>  
3 <s:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/">  
4 <s:element name="Add">  
5 <s:complexType>  
6 <s:sequence>  
7 <s:element minOccurs="1" maxOccurs="1" name="intA" type="s:int"/>  
8 <s:element minOccurs="1" maxOccurs="1" name="intB" type="s:int"/>  
9 </s:sequence>  
10 </s:complexType>  
11 </s:element>  
12 <s:element name="AddResponse">  
13 <s:complexType>  
14 <s:sequence>  
15 <s:element minOccurs="1" maxOccurs="1" name="AddResult" type="s:int"/>  
16 </s:sequence>  
17 </s:complexType>  
18 </s:element>  
19 <s:element name="Subtract">  
20 <s:complexType>  
21 <s:sequence>  
22 <s:element minOccurs="1" maxOccurs="1" name="intA" type="s:int"/>  
23 <s:element minOccurs="1" maxOccurs="1" name="intB" type="s:int"/>  
24 </s:sequence>  
25 </s:complexType>
```

# Restful Web Services

- REpresentational State Transfer
- Use only HTTP (Hyper Text Transfer Protocol)
- Browser sends HTTP Request & get HTTP Response back
- HTTP Methods (GET, PUT, POST, DELETE)
- HTTP Response Code (200, 404 ...)

# Restful Web Services

- **Resource** is anything that you want to exposed through your application
- For example User or Task is resource in User Management Application
- We assign URI's to each resource
  - /user/John
  - /user/John/task
  - /user/John/task/101

# Restful Web Services

- **Resource** can use any representation
  - XML
  - HTML
  - JSON
- Example
  - Create a User : POST /users
  - Delete a User : DELETE /users/101
  - Get all Users : GET /users
  - Get one User : GET /users/101

# Restful Web Services

- Data Exchange Format
  - No restrictions on data format
  - JSON is popular
- Transport
  - HTTP only
- Service Definition
  - No standards as SOAP
  - WADL (Web Application Definition Language)
  - Swagger

# SOAP vs REST

SOAP	REST
Format based on XML	Architectural approach
Data Exchange : XML only	Data Exchange : Can be anything (XML, JSON etc)
Service Definition : WSDL	Service Definition : No standards defined (WADL but not widely used)
Transport : HTTP, MQ	Transport : HTTP only
Ease of Implementation : Difficult to implement. Requires to follow WSDL	Ease of Implementation : Easy to implement. No need to follow any WSDL

**QUESTIONS ?**



# Summary

- What is Web Service?
- Service Definition
- The Types of Web Services
- SOAP Based & Restful Web Services
- Restful Web Service Application
- Spring Doc Open API & JPA

# Web Services vs Application

