

## What is SOAP?

- Simple Object Access Protocol
- XML-based protocol over HTTP
- SOAP is designed to communicate via Internet
- Platform-independent, simple, and extensible

## Why SOAP?

- HTTP is supported by all Internet browsers and servers
- Communicate between applications on different operating systems, with different technologies and programming languages.
- Connecting desktop applications to powerful Internet servers using the standards of the Internet: HTTP and XML.

## SOAP Syntax

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-
envelope"
  soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <soap:Header>
    ...
  </soap:Header>
  <soap:Body>
    ...
    <soap:Fault>
      ...
    </soap:Fault>
  </soap:Body>
</soap:Envelope>
```

## SOAP Syntax

- MUST use XML
- MUST use the SOAP Envelope namespace
- MUST use the SOAP Encoding namespace
- NOT contain a DTD reference
- NOT contain XML Processing Instructions
  - <?CDParser MessageBox("There are songs missing!")?>

## SOAP Envelope Element

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-
envelope"
</soap:Envelope>
```

Namespace: <http://www.w3.org/2001/12/soap-envelope>

## SOAP Header Element

- Optional
- contains application specific information like authentication, payment, etc.
- intended for endpoints on the message path
- mustUnderstand (0/1; true/false): the receiver processing the header must recognize the element
- <soap:Header>
  - <notary:token xmlns:notary="http://notaries-r-us.com/" soap:mustUnderstand="1">
  - XQ34Z-4G5
  - </m:Trans>
  - </soap:Header>

## Intermediary

- Securing message exchanges
  - Encryption, digital sign
- Notarization
- Message tracing

## Body Element

- intended for the ultimate endpoint of the message

```
<soap:Body>
  <m:GetPrice xmlns:m="http://www.w3schools.com/prices">
    <m:Item>
      Apples
    </m:Item>
  </m:GetPrice>
</soap:Body>
```

```
<soap:Body>
  <m:GetPriceResponse xmlns:m="http://www.w3schools.com/prices">
    <m:Price>
      1.90
    </m:Price>
  </m:GetPriceResponse>
</soap:Body>
```

## Fault Element

- used to hold error information

Sub Element	Description
<code>&lt;code&gt;</code>	A code for identifying the fault
<code>&lt;subcode&gt;</code>	additional Information
<code>&lt;reason&gt;</code>	A human readable explanation
<code>&lt;detail&gt;</code>	application specific info

## Fault Values

Value	Description
• VersionMismatch	Found an invalid namespace for the SOAP Envelope
• MustUnderstand	mustUnderstand attribute set to "1", not understood
• Sender	message was incorrectly formed
• Receiver	a problem with the server

```
<soap:Body>
  <soap:Fault>
    <soap:code>
      <soap:Value>soap:Sender</soap:Value>
      <soap:subCode>
        <soap:Value>st:InvalidPurchaseOrder</soap:Value>
      </soap:subCode>
    </soap:code>
    <soap:Reason>
      <soap:Text xml:lang="en">Version mismatch</soap:Text>
    </soap:Reason>
    <soap:Detail> ... </soap:Detail>
  </soap:Fault >
</soap:Body>
```

## A SOAP Request

```
POST /InStock HTTP/1.1
Host: www.example.org
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnn
```

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
  soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <soap:Body xmlns:m="http://www.example.org/stock">
    <m:GetStockPrice>
      <m:StockName>IBM</m:StockName>
    </m:GetStockPrice>
  </soap:Body>
</soap:Envelope>
```

## A SOAP Response

```
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnn
```

```
<?xml version="1.0"?>
<soap:Envelope xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
  soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <soap:Body xmlns:m="http://www.example.org/stock">
    <m:GetStockPriceResponse>
      <m:Price>34.5</m:Price>
    </m:GetStockPriceResponse>
  </soap:Body>
</soap:Envelope>
```