

DAIMLERCHRYSLER

Extending SOAP to Adhere to Session-oriented Communication Principles

Mario C. Jeckle

DaimlerChrysler Forschungszentrum Ulm

mario.jeckle@daimlerchrysler.com

mario@jeckle.de

www.jeckle.de

Contents

- Definitions:
 - *Context vs. Transaction*
- Motivation:
 - Necessity and practical usage of context-aware communication
- Technical Background:
 - Existing Solutions and related work
 - W3C's SOAP v1.2
- The Challenge
- Solution:
 - SOAP's built-in extensibility mechanisms
 - Description Model
 - Execution model
- Deployment scenarios and practical examples

Basic Definitions

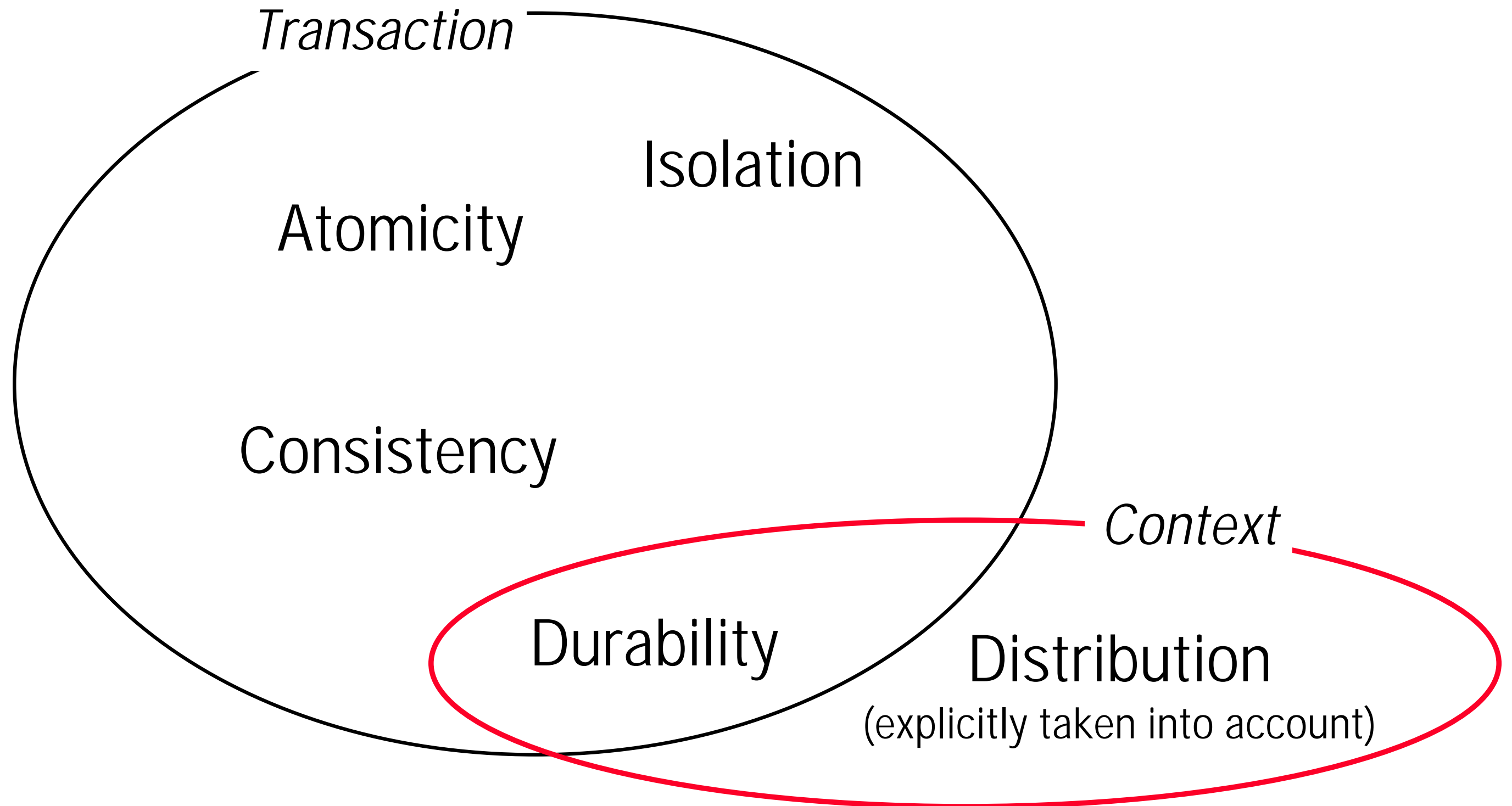
- Transaction

Combined treatment of singular processing steps (like database write operations) for integrity's sake. (-> ACID principles)

- Context (Session)

Stateful connection between two parties during which one or more communications take place.

Basic Definitions



Motivation

Necessity and practical usage of context-aware communication...

- Typical communication flows in a technical and/or business environment span multiple singular communications
 - correlation of singular communications has to be explicitly encoded into the data transmitted
 - or has to be handled by the communicating application in a proprietary manner
 - contexts are used as a mechanism for semantically embracing a number of communications
- Removing context-related information from the payload reduces amount of data to transfer and cleans up cluttered semantics
- Often *real* transactional behavior (i.e. the ACID principles) is not needed for various reasons
- Application scenarios for deploying session oriented communication:
 - Shopping basket
 - Login-in, Log-out which are embracing a number of interactive data accesses
 - Partial delegation of incoming service requests

Technical Background

Existing solutions and related work:

- Transaction handling:
 - *OASIS Business Transaction Protocol (BTP)*
Oriented on classical (distributed) transactional processing (i.e. 2PC).
 - *Java Transaction Service (JTS)*,
Specifies a *Transaction Manager* which conforms to the *Java Transaction API*.
In essence: a Java Mapping of CORBA's Object Transaction Service
 - *XML Transaction Authority Markup Language (XAML)*
... no longer under development.

Technical Background

Existing solutions and related work:

- Context handling:
 - HTTP/1.1
 - keep alive just allows it to inform the requesting client about connection's termination.
 - Cookies (IETF RFC 2109, 2964)
 - Just small name value pairs which are transmitted solely over HTTP.
 - Optional part of HTTP.
 - URL enrichment (IETF RFC 1945, 2616; URL Rewrite)
 - Solely defined for HTTP.
 - Questionable from the standpoint of security.
 - Cannot be used within Web Services since the service endpoint (i.e. the URI) dynamically changes due to rewrite.

Technical Background

W3C's XML Protocol-Standard -- SOAP v1.2:

- Currently under standardization (which is in fact almost finished) by a W3C working group
- Based on the *Simple Object Access Protocol* proposed by Microsoft, IBM, Lotus, et al.
- Defines an extensible framework for processing XML encoded messages and remote procedure calls transported using protocols of the Internet stack.
- User extensible by the use of SOAP's header elements.
- SOAP nodes (*Intermediaries*) are full fledged SOAP processors which are allowed to process parts of the transported information along the path between the sender and the ultimate receiver.
- SOAP's most prominent ...
 - ... usage: XML encoded RPCs (SOAP part 2, chap. 4)
 - ... transport protocol: HTTP (SOAP, part 2, chap. 7)

Technical Background

W3C's XMLP/SOAP Standard

```
<?xml version="1.0" encoding="UTF-8"?>
<env:Envelope
  xmlns:env="http://www.w3.org/2001/09/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:m="http://example.org/2001/06/quotes">
  <env:Header>
    <m:doSomething>42</m:doSomething>
  </env:Header>
  <env:Body>
    <m:GetLastTradePrice
      env:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
      <m:Symbol>DCX</m:Symbol>
    </m:GetLastTradePrice>
  </env:Body>
</env:Envelope>
```

SOAP-Message

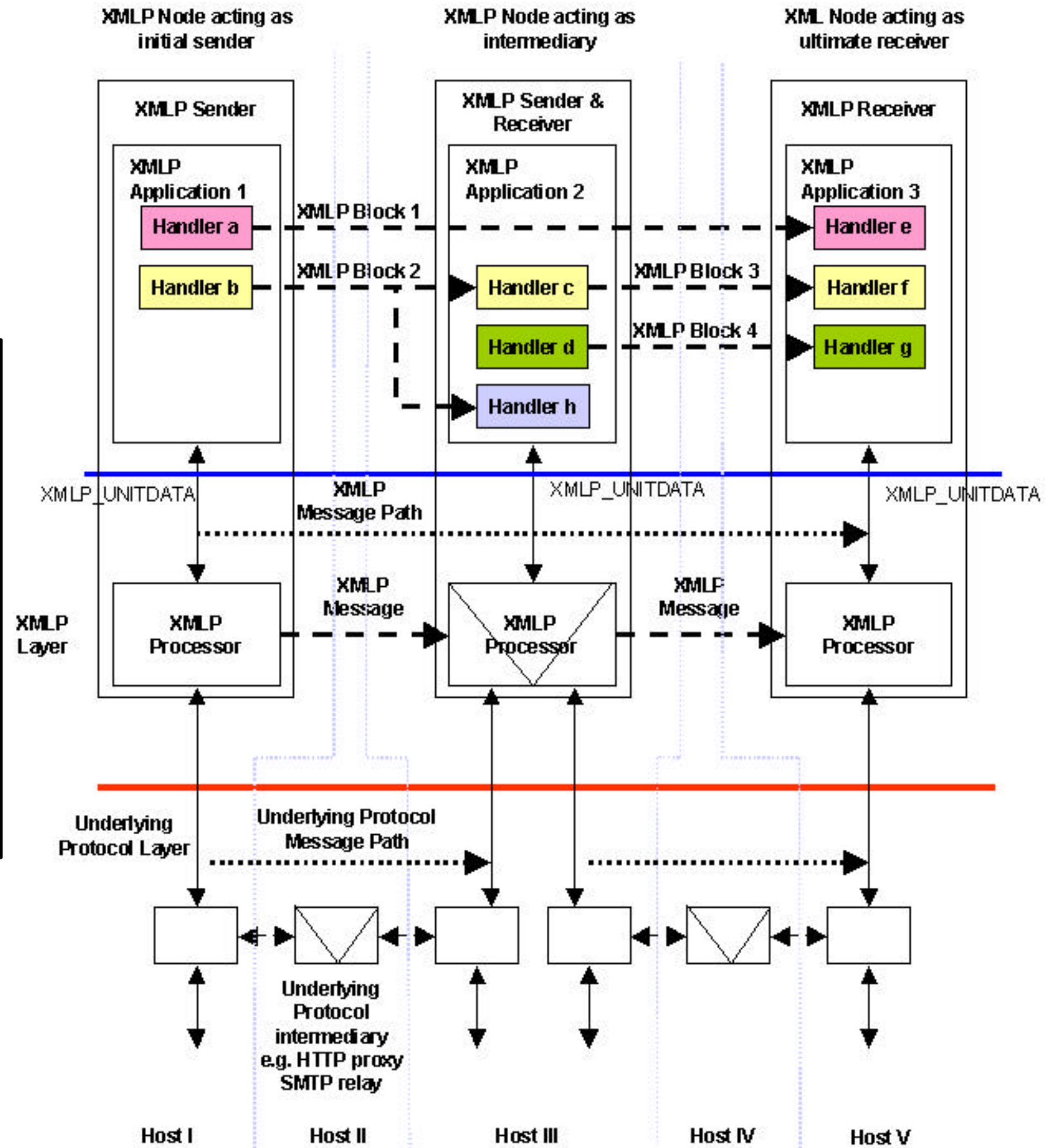
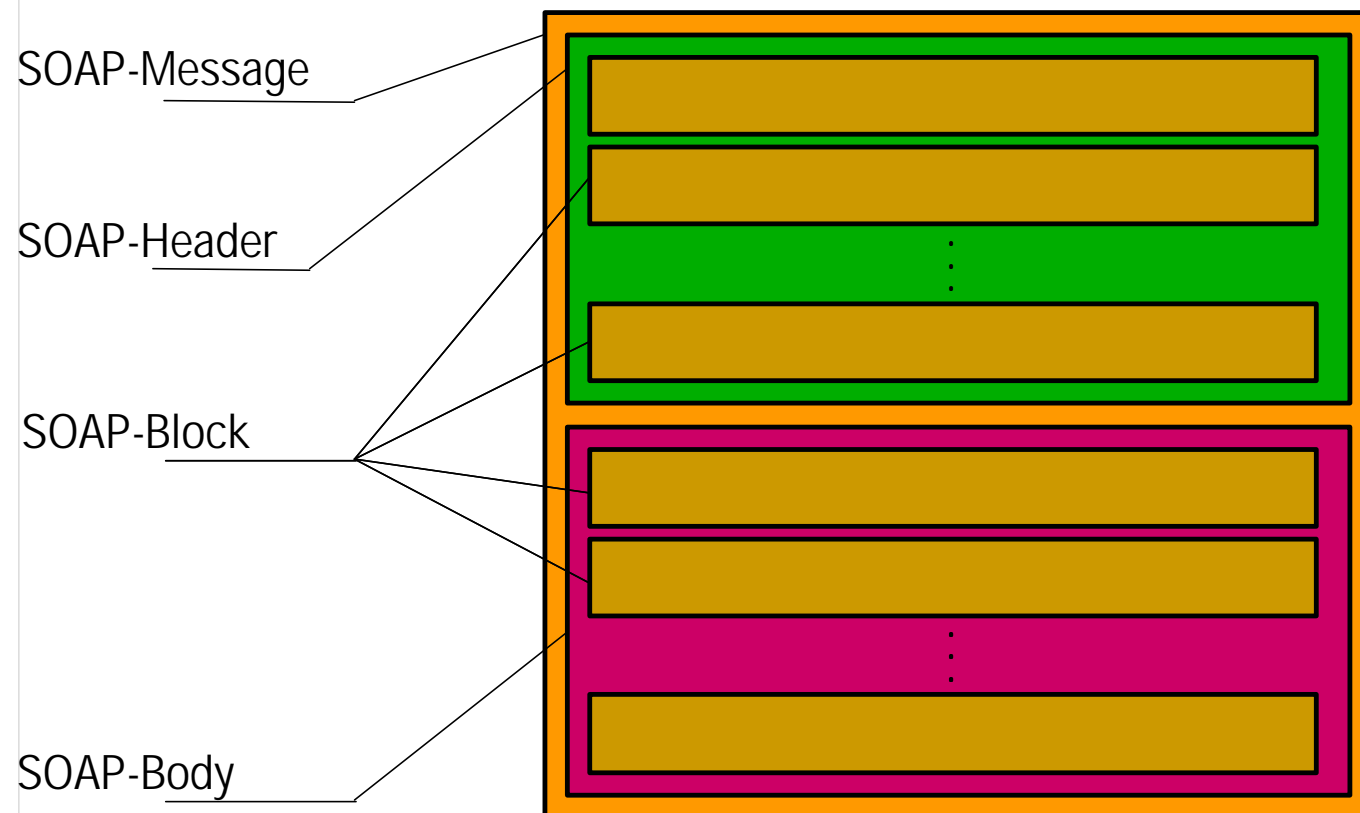
SOAP-Header

SOAP-Block

SOAP-Body

Technical Background

W3C's XMLP/SOAP Standard



Technical Background

W3C's XMLP/SOAP Standard

- SOAP Header (SOAP v1.2, part 1, sect. 5.2)
 - ... are basically meta information to the SOAP message
 - ... are user definable
 - ... are located within a XML namespace which is separated from SOAP one's
 - ... could be processed by a SOAP node
- SOAP Node (SOAP v1.2, part 1, sect. 2.1)
 - ... are able to process SOAP messages along their message path,
 - ... may consume *SOAP headers*
 - ... could be explicitly addressed within a header using a URI

Challenge

Propagating context information needed within Web Services in a way, form or shape which is neutral w.r.t. programming languages as well as technical infrastructures and which additionally relies solely on the SOAP protocol

- Solution's characteristics to achieve :
 - Propagation of context information
 - => no transactionality
 - => Business Transaction Protocol, XML Transaction Authority Markup Language are discharged
 - Programming language independence
 - => Language centric approaches (e.g. Java Transactions) are discharged
 - Infrastructure independence:
 - => HTTP-based approaches (HTTP/1.1, Cookies, URL Enrichment) are discharged
 - (exclusively) based on SOAP
 - => Usage of SOAP's built-in extensibility mechanisms

Usage of SOAP's header mechanism
to add session orientation to Web Services

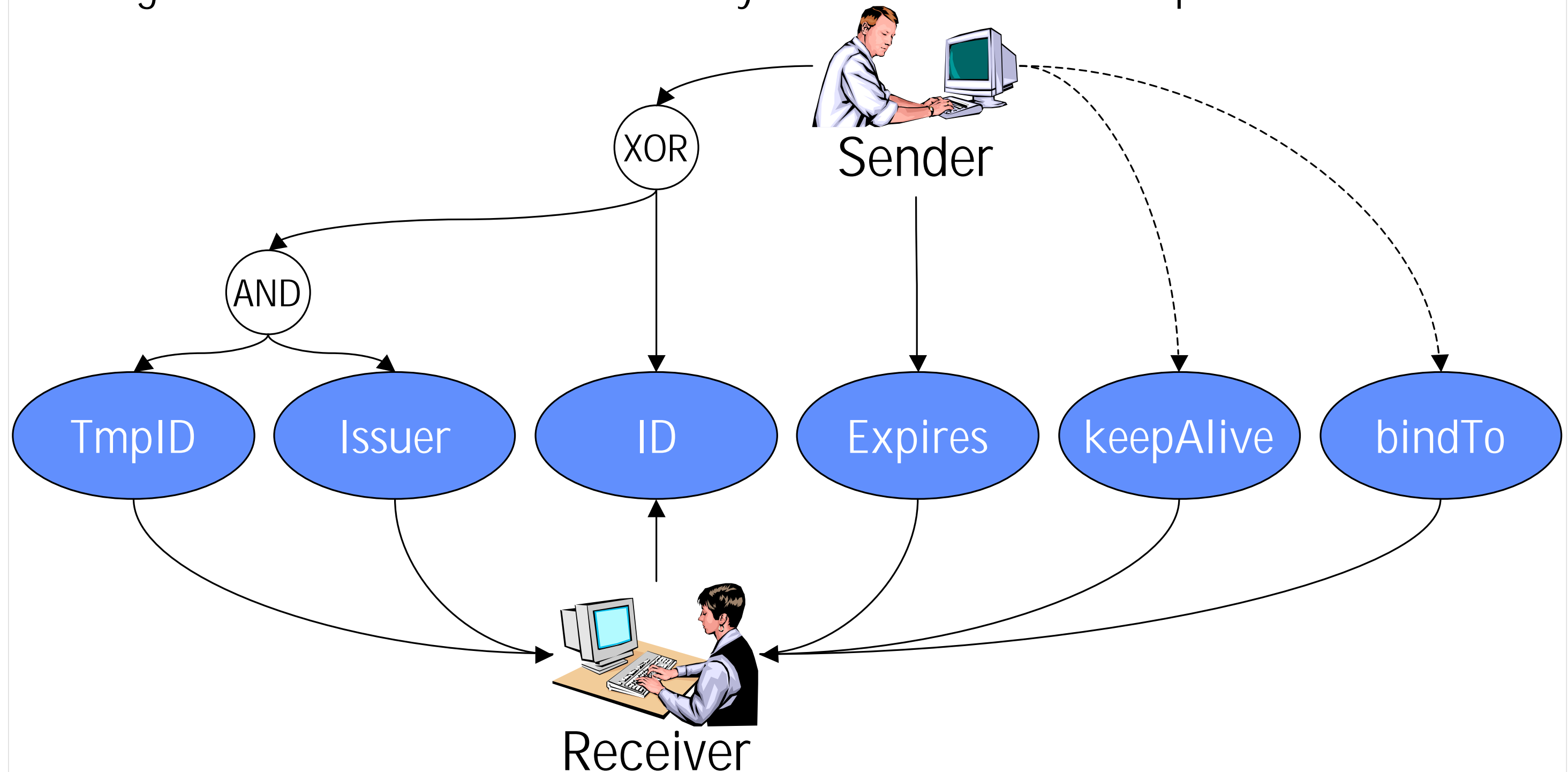
Solution

Usage of SOAP's built-in Extensibility Mechanisms

- Description model
 - XML Information Set based model abstracted from concrete syntax
 - XML vocabulary organized in an own namespace
 - Session identification
 - Session expiration
 - Context issuer (optional)
- Execution model
 - To be executed by every SOAP node
(i.e. intermediaries along the message path as well as the ultimate receiver)
 - Session states and transitions
 - Negotiation mechanism (optional)

Solution

Usage of SOAP's built-in Extensibility Mechanisms -- Description Model



Solution

Usage of SOAP's built-in Extensibility Mechanisms -- Description Model

```
<?xml version="1.0" encoding="UTF-8"?>
<Envelope xmlns="http://www.w3.org/2001/09/soap-envelope" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.w3.org/2001/09/soap-envelope">
  <Header>
    <ctx:Context xmlns:ctx="http://www.example.com/soap-envelope/Context"
      soap:mustUnderstand="true"
      soap:role="http://www.w3.org/2001/12/soap-envelope/role/next"
      keepAlive="true">
      <ctx:ID>9db60798-2c74-46eb-968a-370bfc333c60</ctx:ID>
      <ctx:Expires>2002-08-01T16:00:00:00+01:00</ctx:Expires>
      <ctx:Issuer>http://www.ssgrr.it</ctx:Issuer>
    </ctx:Context>
  </Header>
  <Body>
    <Login xmlns:baz="http://www.ssgrr.it">
      <Userna
      <EncPas
    </Login>
  </Body>
</Envelope>
```

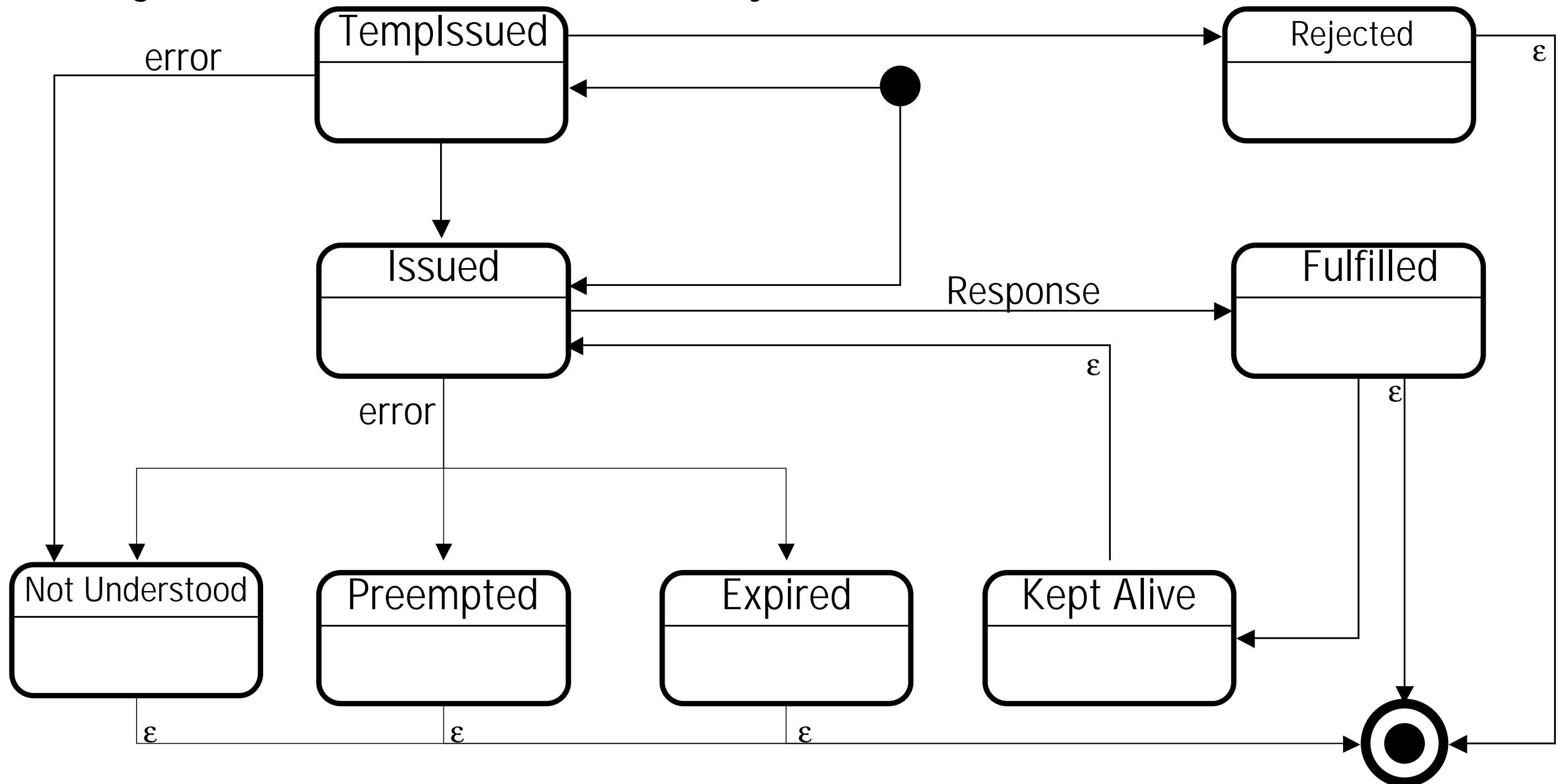
The diagram shows a callout box with a white background and a black border. It contains the following text:

```
soap:mustUnderstand="true"
soap:role="http://www.w3.org/2001/12/soap-envelope/role/next"
```

Two lines from the XML code above point to this callout box: one from the `soap:mustUnderstand="true"` attribute in the `<ctx:Context>` element, and another from the `soap:role="http://www.w3.org/2001/12/soap-envelope/role/next"` attribute in the same element.

Solution

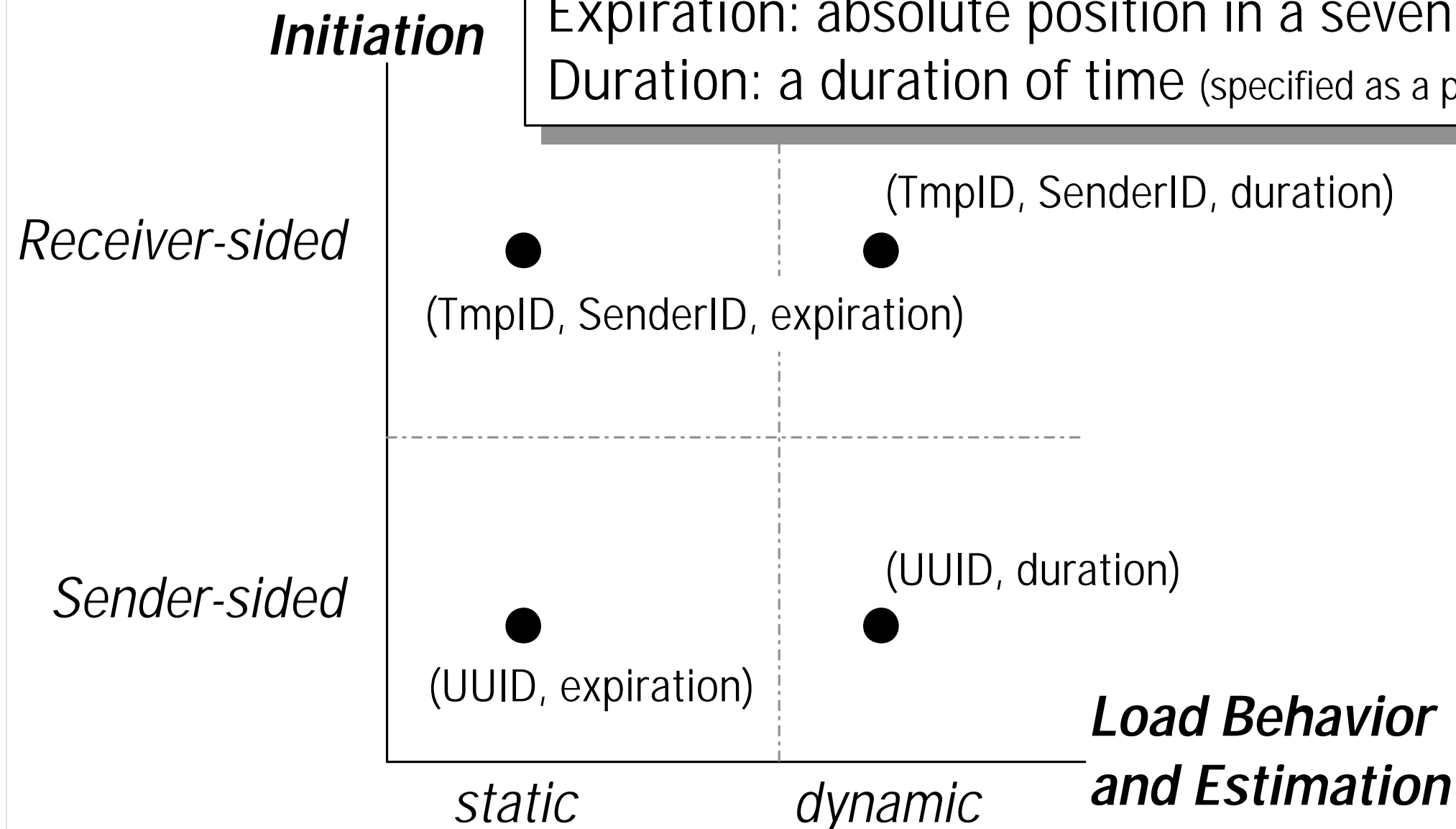
Usage of SOAP's built-in Extensibility Mechanisms -- Execution Model



Solution

Usage of SOAP's built-in Extensibility Mechanisms -- Execution Model

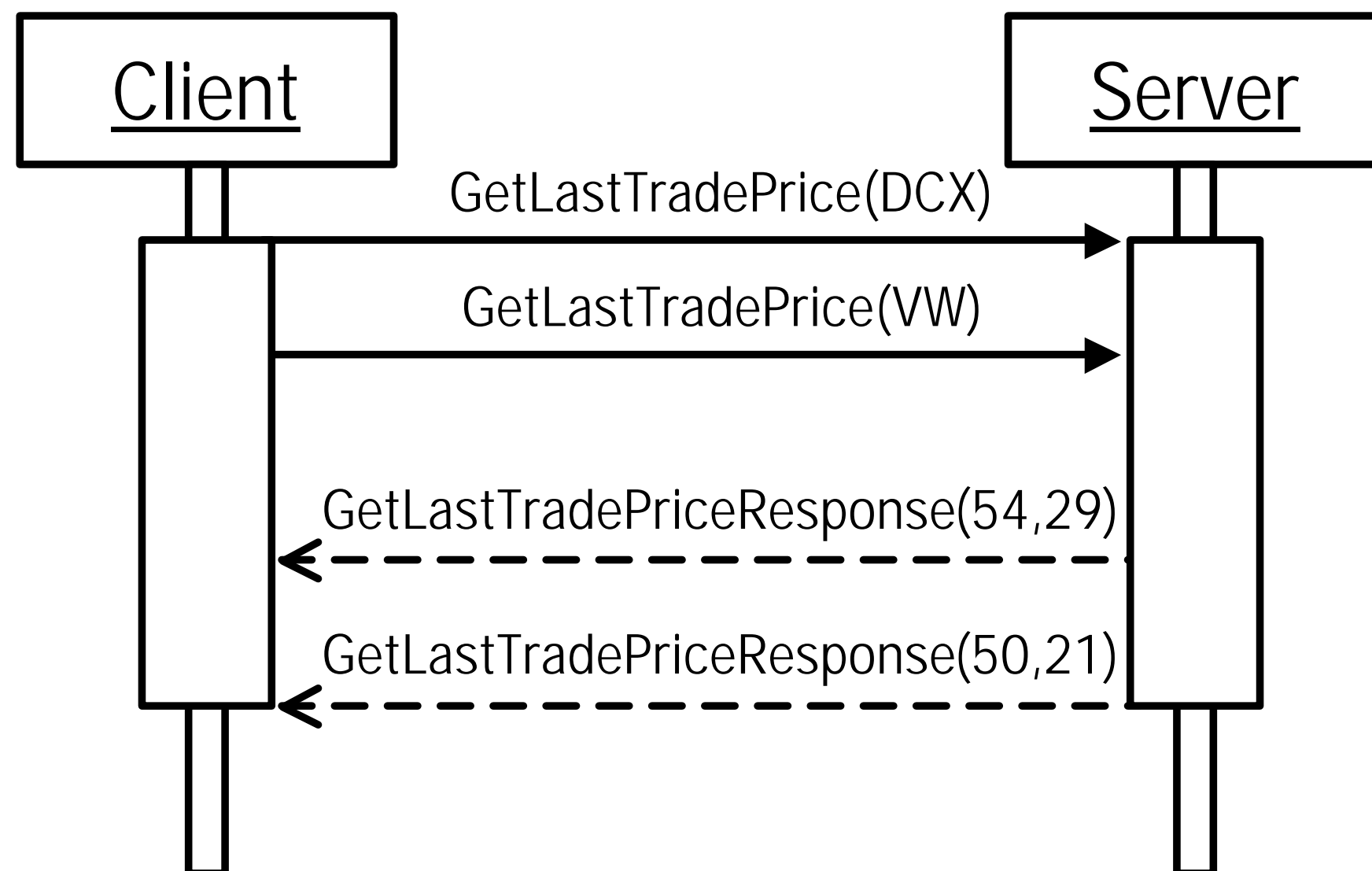
TmpID: Temporary identification (sender unique)
SenderID: an unique identification of a sender
UUID: Universally Unique Identifier (globally unique)
Expiration: absolute position in a seven-dimensional space
Duration: a duration of time (specified as a point within a six-dimensional space)



Deployment Scenarios and Practical Examples

Message Correlation

- Basic idea:
Correlating asynchronously sent messages at the sender as well as the receiver side
- Example:
Stock quote inquiry



Deployment Scenarios and Practical Examples

Message Correlation

- Example:
Stock quote inquiry
- Problem:
When asynchronous communication mechanisms are used, messages have to be correlated on the basis of contextual information which is not part of the message's informational payload

```
<env:Envelope
  xmlns:env="http://www.w3.org/2001/09/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <env:Body>
    <m:GetLastTradePrice
      env:encodingStyle="http://www.w3.org/2001/12/soap-encoding"
      xmlns:m="http://example.org/2001/06/quotes">
      <m:Symbol>DCX</m:Symbol>
    </m:GetLastTradePrice>
  </env:Body>
</env:Envelope>
```

```
<env:Envelope
  xmlns:env="http://www.w3.org/2001/09/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <env:Body>
    <m:GetLastTradePriceResponse
      env:encodingStyle="http://www.w3.org/2001/12/soap-encoding"
      xmlns:m="http://example.org/2001/06/quotes">
      <m:Price>50,21</m:Price>
    </m:GetLastTradePriceResponse>
  </env:Body>
</env:Envelope>
```

Deployment Scenarios and Practical Examples

Message Correlation

- Solution:
 - Addition of session-related information to the sent message

```
<env:Envelope
  xmlns:env="http://www.w3.org/2001/09/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <ctx:Context xmlns:ctx="http://www.example.com/SOAPContext">
    <ctx:ID>f2f7c5ec-5123-40dc-be64-2b9953e60201</ctx:ID>
    <ctx:Expires>2002-06-28T08:25:00:00Z</ctx:Expires>
    <ctx:Issuer>URN:pin:bs4321234</ctx:Issuer>
  </ctx:Context>
  <env:Body>
    <m:GetLastTradePrice
      env:encodingStyle="http://www.w3.org/2001/12/soap-encoding"
      xmlns:m="http://example.org/2001/06/quotes">
      <m:Symbol>DCX</m:Symbol>
    </m:GetLastTradePrice>
  </env:Body>
</env:Envelope>
```

Deployment Scenarios and Practical Examples

Message Correlation

- Solution:
 - Retain the context information within the response message

```
<env:Envelope
  xmlns:env="http://www.w3.org/2001/09/soap-envelope"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <ctx:Context xmlns:ctx="http://www.example.com/SOAPContext">
    <ctx:ID>f2f7c5ec-5123-40dc-be64-2b9953e60201</ctx:ID>
  </ctx:Context>
  <env:Body>
    <m:GetLastTradePriceResponse
      env:encodingStyle="http://www.w3.org/2001/12/soap-encoding"
      xmlns:m="http://example.org/2001/06/quotes">
      <m:Price>50,21</m:Price>
    </m:GetLastTradePriceResponse>
  </env:Body>
</env:Envelope>
```

Deployment Scenarios and Practical Examples

Message Correlation

- Putting it all together

