

*Devtech*

## **SOA AS I SEE IT**

A pragmatic view of Service Oriented Architecture

Devindra J. Werasooriya

*Devtech*

### **Disclaimer**

The opinions expressed in this presentation are mine and mine alone and do not necessarily reflect those of the organisations or individuals that are acknowledged at the end of the presentation.

## **Organisation of the Presentation**

- Architectural overview of SOA – where should it and should it not be applied
- Basic SOA models
- SOA applied to Transactional Processing
- References and acknowledgements
- Questions

3

## **Scope**

- Focused on architectures, patterns and standards
- Occasional digressions to the current default framework – SOAP protocol, http transport and XML messages
- Limited to processes and procedures between a Service Invoker and a Service Provider – essentially B2B
- Do not intend to get chest-deep into standards material – only references and high-level characteristics

4

## **SOA -- Definition vs. Implementation**

Defined as an architecture for delivering business functionality through a suite of distinct programs where inter program communication occurs by invoking loosely coupled interfaces, based on a formal contract that is platform independent. The following are usual features of the architecture

- The service access must not require knowledge of the underlying implementations of either the requester or responder in order to interoperate
- Services may be discovered at run-time.

Implemented using web-services -- self-contained, modular business process applications that are based on the technologies of WSDL to describe the contract, UDDI to advertise, SOAP/XML to communicate, HTTP to transport and the INTERNET to network.

5

## **When to do SOA**

- The figure displays the shape of the basic SOA model – the model is Organic
- When the invoker – or client – is managed by the principle program driver
- When the business requirement can be addressed through synchronous – request/response – without significant performance handicap
- Do SOA if there is a strong push for centralised governance at the invoker and the possibility of disseminating the common governance model through to all providers – centralised governance but distributed computing.

6

### **When not to SOA**

- The figure displays the shape of the basic non SOA model – this model is scalable
- When the downstream systems have complex user-interfaces that add value and are hard to replicate
- When system interoperability and dynamic discovery are not essential characteristics
- When the Invoker and the Provider are on a level playing field.
- If guaranteed message delivery and performance are important
- When the integration driver has a strong Data-Integration Focus as opposed to a Application-Integration Focus
- The diagram shows the shape of an NSOA architecture

7

### **A Basic SOA Model**

The most basic SOA model is of a look-up service. Following is the

- Protocol – Simple Object Access Protocol
- Message – principle elements
- Transport – Hypertext Transmission Protocol
- Fault-resolution
- Interoperability – is a definitional requirement of SOA.
- Possible client architectural models
  - Static Proxy
  - Dynamic Invocation
  - DII

8

### **Basic SOA Model -- Continued**

- Security – security must be a part of the basic architecture – not a monolithic inconvenience that has to be dealt with after the fact. Make a scenario based assessment of security adequacy. Every level of security comes at a price – performance and often cost.
- Standards relevant to a basic web-service
  - WS-I: does not guarantee interoperability – but is the best way to ensure interoperability
  - WS-Security
- Note that both of these standards are specific to a SOAP based SOA implementation.

9

### **WS-Interoperability**

The profile is based on matching of specific targets to specific requirements. Following are the important targets

- The Envelope – refers to its structure and content
- The Web-Service Contract or the interface – the WSDL description
- Registry Data – Elements related to the registration and discovery of web-services

The protocol elements that transport the message must be SOAP/http  
The scope is judiciously defined with in-scope elements and points of extension. The extensions can effect interoperability, which is a reason for not guaranteeing interoperability.

Following are some [examples of WS-I compliance](#)

10

*Devtech*

## **WS-Security**

- Enhances SOAP messaging through
  - Message Integrity
  - Message Confidentiality
  - Message Authentication
- Provides a General Purpose Mechanism for Associating Security Tokens with messages
- Specifies how to encode binary security tokens – gives specific techniques for
  - X.509 certificates
  - Kerberos tickets
- A specific description of WS-Security is provided in the [references](#)
- Consider the following generic view of [web-services security](#)

11

*Devtech*

## **Transactional SOA**

Support for Transactional Activity is essential in an Integration Framework – SOA is no exception. Have to overcome

- The Distributed computing environment
- SOA is founded on synchronous interactions

Critical to transactional activity is the establishment of co-ordination – defined as the process of the coordinator disseminating information to a number of participants as a part of a transactional process

- A coordination context must be associated with distributed transaction processing

12

## *Devtech*

### **Access Queensland – Choreography Requirements**

- Business Imperatives
  - Necessary to have shopping baskets that spanned delivery channels
  - A product in the basket must be delivered – product reservation is a must
  - Some products are of restricted availability – do not want people to simply reserve for long periods and not buy them
  - Some provider channels may have other user-interfaces
- Design Decisions from the above
  - A product in the basket will be reserved for a limited time.
  - Explicit un-reserve message issued from the client if reservation expires.
  - Transaction rollbacks on Payment failures and Session\_failure.
  - Rollbacks apply to the whole basket

13

## *Devtech*

### **AQ – choreography continued**

- Products are reserved on confirmation of intent to purchase after selection – the clock starts ticking from the first reservation
- A product goes through the following cycle during purchase
  - Reserved
  - Confirmed
  - Committed
  - Fulfilled – not mandatory
- The reservation time applies to all contents of the shopping basket.
- There are coordination-id's for the shopping basket and each (channel, product) tuple – the latter set on a confirmed reservation
- Some items – such as licenses -- do not require a reservation, however in keeping with the general trend of messaging they were also nominally reserved prior to payment.

14

### **AQ – Choreography -- continued**

- Events that lead to revocation of reservations are
  - Payment failure
  - Product timeouts.
  - Session Timeouts
  - Session failure
- A successful payment preceded by a successful reservation is necessary and sufficient for a successful delivery of products.
- The standards that operate in this area are
  - WS-Coordination
  - WS-Transaction

15

### **WS-Coordination**

- WS-Coordination – involves
  - Activation : Creation of a coordination context
  - Registration : Registration of the specific delivery services
  - Propagation : Disseminating the coordination context
- Activation – Occurs during the creation of the shopping basket
- Registration – Unnecessary given that the delivery services are known and need not be re-discovered
- Propagation – Occurs with each tentative reservation
- Completion – Occurs at transaction commitment; followed by persistence and clean-up of the transaction context.

16

## WS-Transaction

- WS-Transaction
  - Remember the classical transaction properties
    - Atomic
    - Consistent
    - Isolated
    - Durable
  - Classified into two categories
    - Atomic Transaction – AT model
    - BusinessActivity – BA model
      - AT model
        - » Consists of Transaction aware [business services and Transaction Protocol aware participants](#)

17

## WS-Transaction

- » Participants are protocol aware, there is some flexibility here.
- » It is fair to say that the model is essentially based on 2 Phase Commit [diagram](#)
- » Both Prepare and Commit function on an all or nothing model
- » None of the phases have a concept of time-binding
- All succeed or nothing succeeds – the transaction may be rolledback based on the failure of one participant.
- Resources are locked for the duration of the transaction
- Also no provision to capture more states during the prepare phase
- Business Activity BA Model
  - » Independent tasks – and allows the transaction to progress while some subtasks have failed
  - » Can operate on a nested task model
  - » The child tasks may progress to their goals by pre-emptively.
  - » Possibility is open for subsequent compensation if necessary.
- [AQ Transactional Flow](#) is almost a prelude to the Business Activity Model.

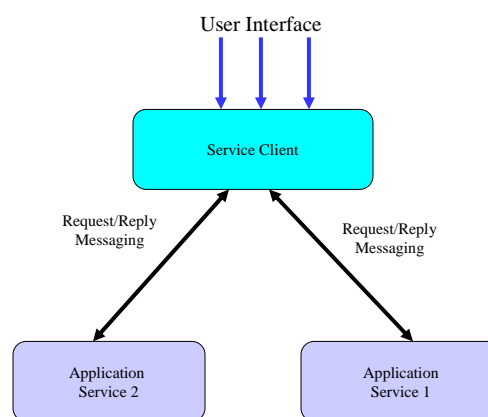
18

## Conclusion

- Standards in any area of Science or Engineering is a sign of maturity – SOA is no different.
- However SOA standards are global – applicable across diverse computing technologies.
- Standards are also a means of re-use of best practice and therefore of Risk Minimisation.
- However, the functionality required from the Software-project or framework could mean acquiring some clear notions regarding
  - Using a subset of the standard
  - The adequacy of the scope of the standard
- The first is easily solved if the standard is well understood, the second is harder given the need to discover the delta – between the standard and the expected.
- Whether to minimise the delta through functionality re-scoping or deal with the delta as a development proposition, may depend on the perceived long-term vision for the Software Framework of interest – SOA is no different.

19

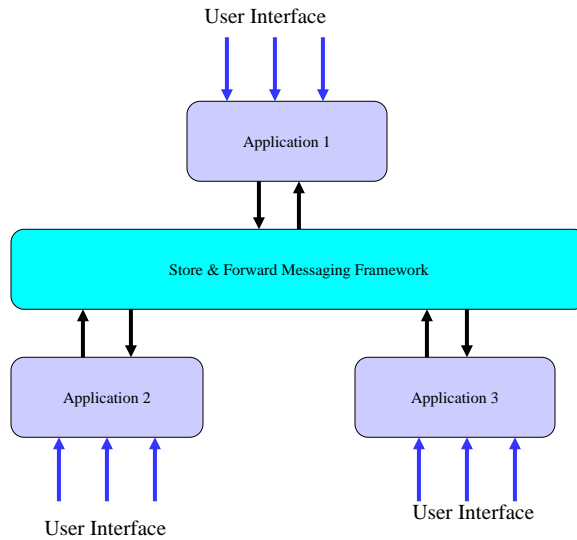
## The shape of an SOA architecture



[Back](#)

20

### The shape of a non SOA architecture



[Back](#)

### WS-Interoperability Examples

```
<soap:Envelope xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/' >  
  <soap:Body>  
    <p:Process xmlns:p='http://example.org/Operations' />  
  </soap:Body>  
  <m:Data xmlns:m='http://example.org/information' >  
    Non-compliant due to Envelope structure  
  </m:Data>  
</soap:Envelope>
```

```
=====
```

```
<soap:Envelope xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/' >  
  <soap:Body>  
    <p:Process xmlns:p='http://example.org/Operations' >  
      <m:Data xmlns:m='http://example.org/information' >  
        Compliant Envelope  
      </m:Data>  
    </p:Process>  
  </soap:Body>  
</soap:Envelope>
```

## WS-Interoperability Examples continued

```
<soap:Fault xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/' >
  <faultcode>soap:Client</faultcode>
  <faultstring>Invalid message format</faultstring>
  <faultactor>http://example.org/someactor</faultactor>
  <detail>There are <b>illegal </b> of elements in the fault message</detail>
  <m:Exception xmlns:m='http://example.org/faults/exceptions' >
    <m:ExceptionType>Severe</m:ExceptionType>
  </m:Exception>
</soap:Fault>
```

```
=====
<soap:Fault xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/' >
  <faultcode>soap:Client</faultcode>
  <faultstring>Invalid message format</faultstring>
  <faultactor>http://example.org/someactor</faultactor>
  <detail>
    <m:msg xmlns:m='http://example.org/faults/exceptions' > This is a
      <b>legal</b>message
    </m:msg>
    <m:Exception xmlns:m='http://example.org/faults/exceptions' >
      <m:ExceptionType>Severe</m:ExceptionType>
    </m:Exception>
  </detail>
</soap:Fault>
```

23

## WS-Interoperability Examples continued

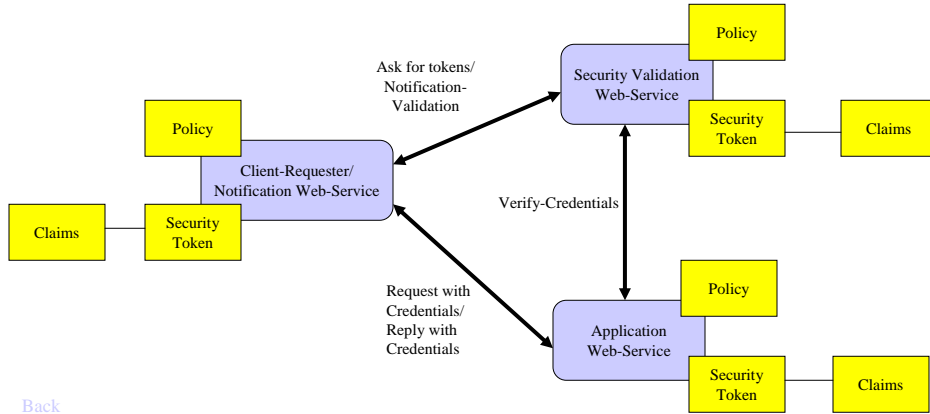
```
<soap:Fault xmlns:soap='http://schemas.xmlsoap.org/soap/envelope/' >
  <soap:faultcode>soap:Client</soap:faultcode>
  <soap:faultstring>Invalid message format</soap:faultstring>
  <soap:faultactor>http://example.org/someactor</soap:faultactor>
  <soap:detail>
    <m:most xmlns:m='http://example.org/faults/exceptions' >
      <b>Illegal</b> element qualification
    </m:most>
  </soap:detail>
</soap:Fault>
```

Remove soap qualification in child elements of soap:Faults

[Back](#)

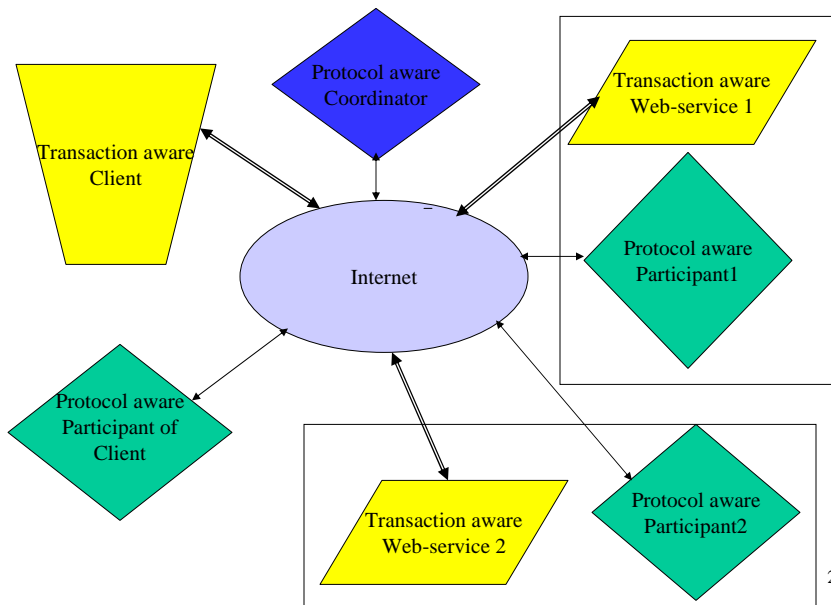
24

### WS-Security

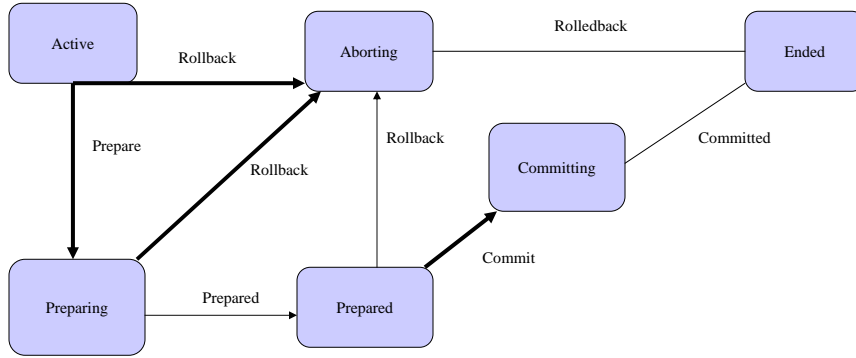


[Back](#)

### WS-Transaction – AT – Services & Participants

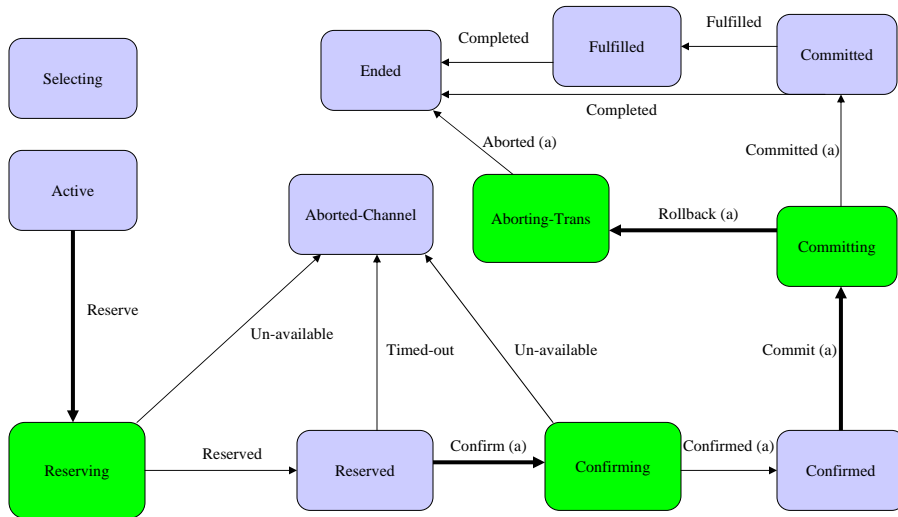


WS-Transaction – AT – 2PC Protocol



27

AQ Transactional Flow



[Back](#)

28

## References

- Reference on WS-Security : <http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-message-security-1.0.pdf>
- Jim Webbers Blog : <http://jim.webber.name/>
- <http://www.ws-i.org/>
- The Serverside Portal
- IBM Developer Works and BEA Dev2Dev

[Back](#)

29

## Acknowledgements

- Access Queensland Project and its Management
- DataAgility and its Management
- Jim Webbers Socsig presentation and Blog
- Previous SOCSIG presentations – especially those of Peter Chillcot, David Glower, Brad Kassell.
- The SOC Research Interest Group of the Swinburne University of Technology.
- **You all for your presence and attention**

30