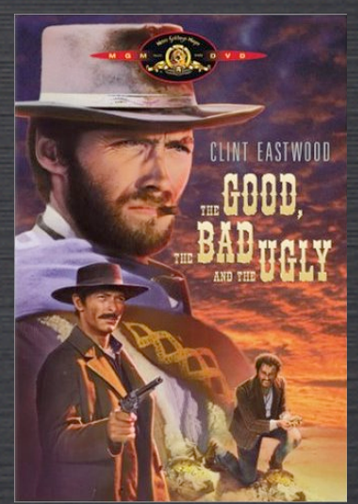
The Good, The Bad and The Ugly

WS-* Specifications For SOA

Peter Tröger



Web Services And SOA

- If a SOA is the solution, what was the original problem?
  - **Interoperate** and **integrate** legacy systems
  - Inappropriate **mapping** between business **processes** and IT processes
- Service-Oriented Computing (M.P. Papazoglou and D. Georgakopoulos)

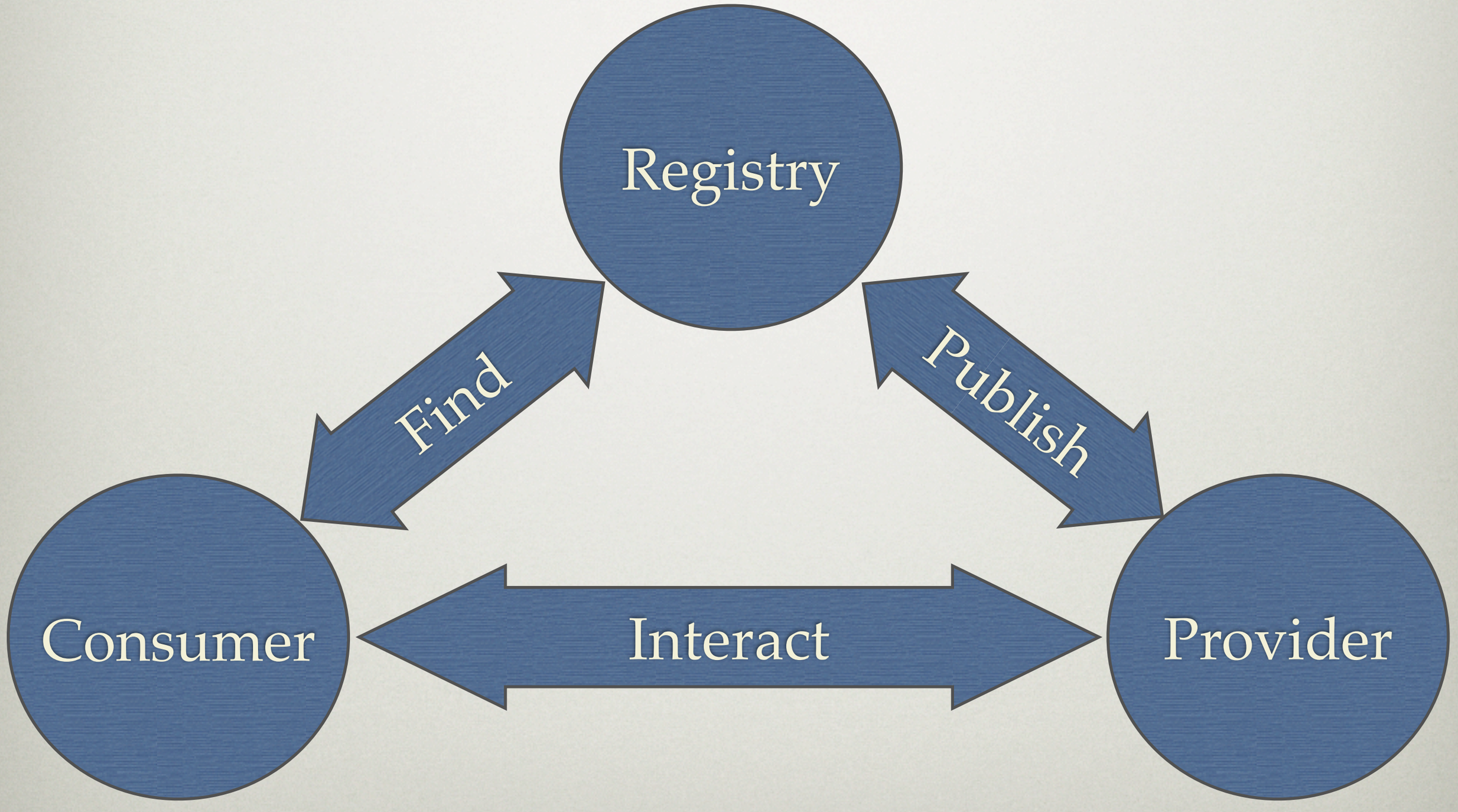
Terminology

- Web Service (W3C Web Service Architecture Group)
  - Machine-to-machine network interaction
  - Machine - processable **WSDL** interface description
  - Interaction using **SOAP** messages
  - Typically conveyed using **HTTP** (but not a must)
  - **XML** serialization
- Endpoint: Reference of target for Web service messages

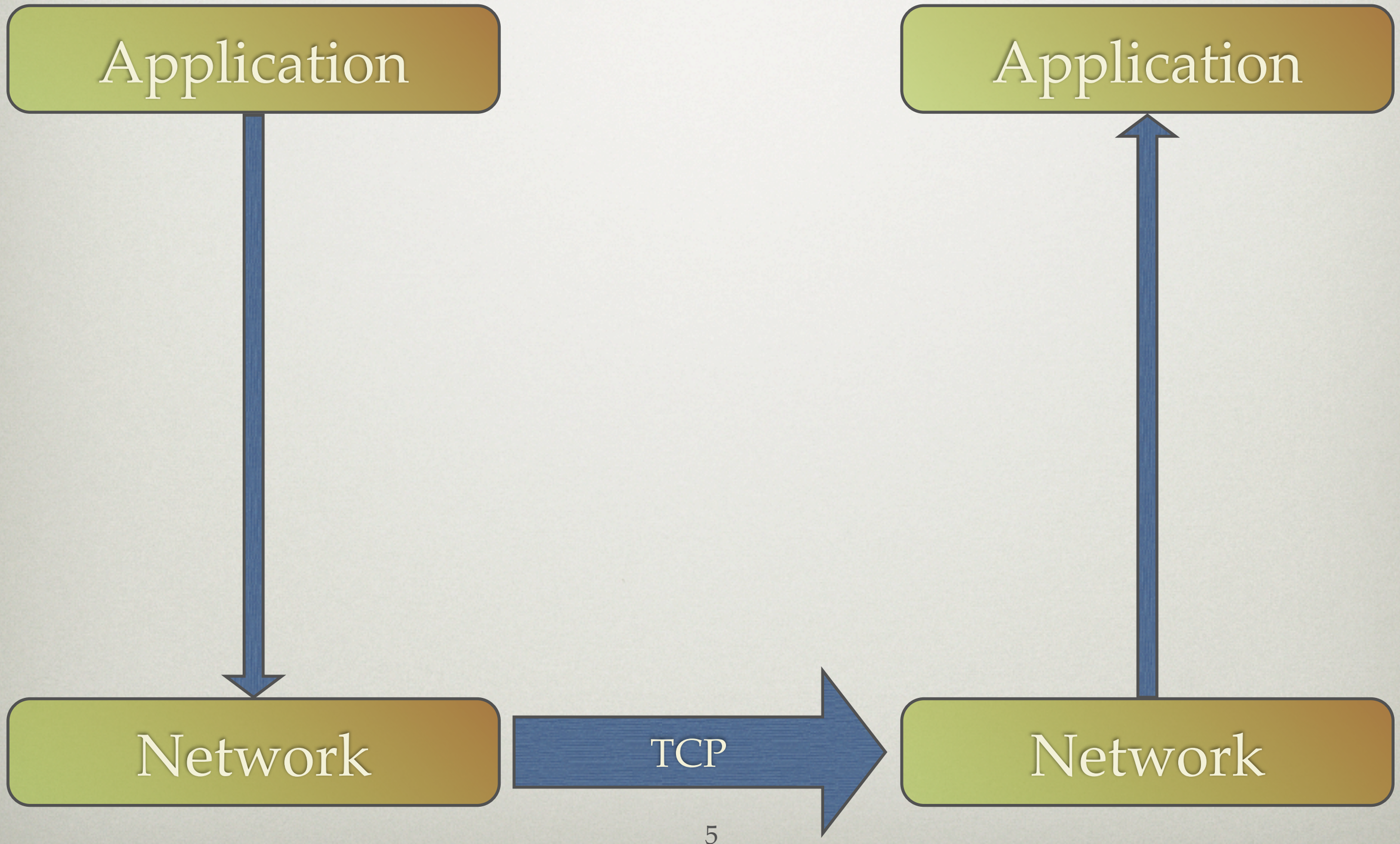
3

Service Interaction

Service Interaction

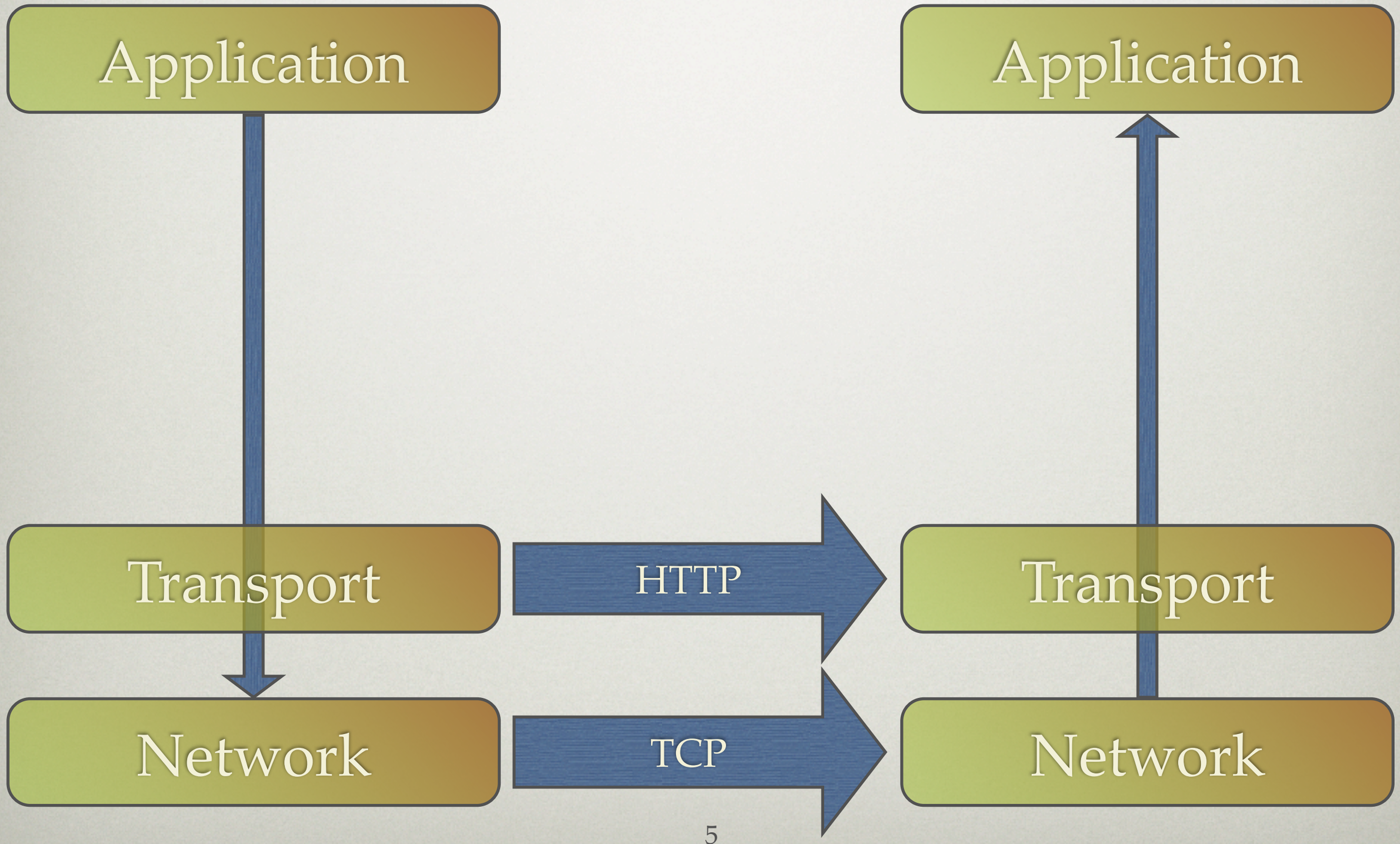


Interoperability On Different Layers

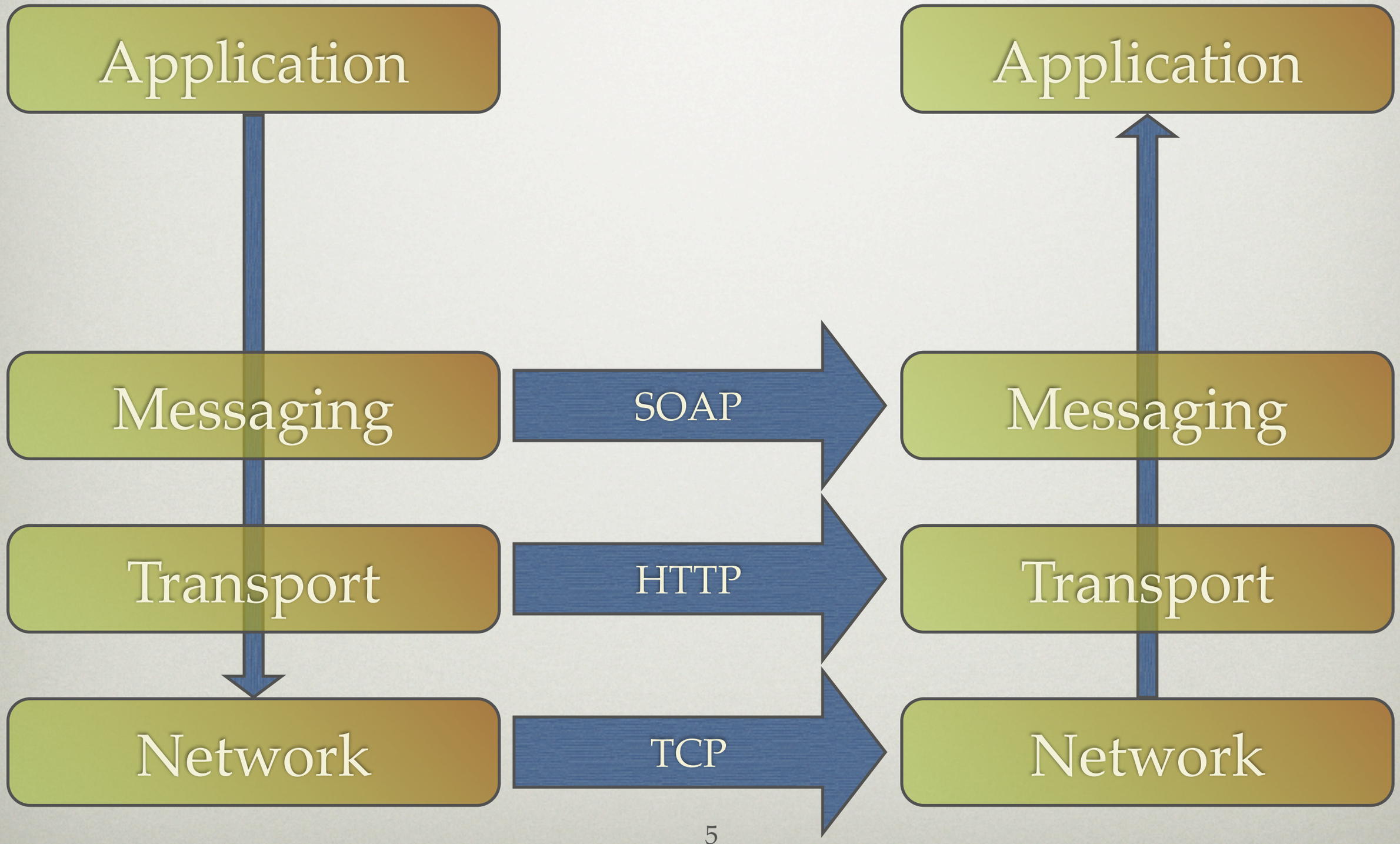


5

Interoperability On Different Layers

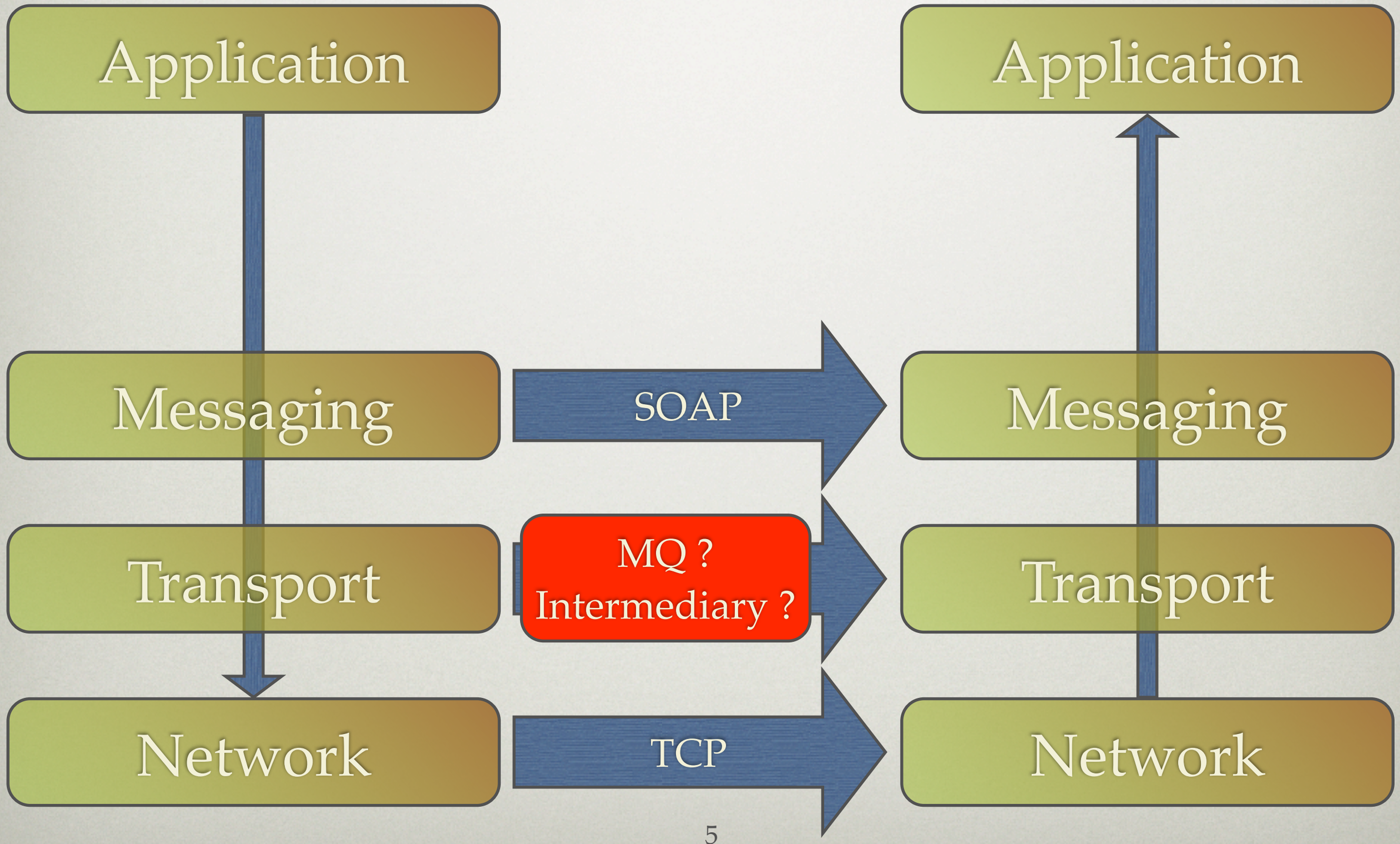


Interoperability On Different Layers

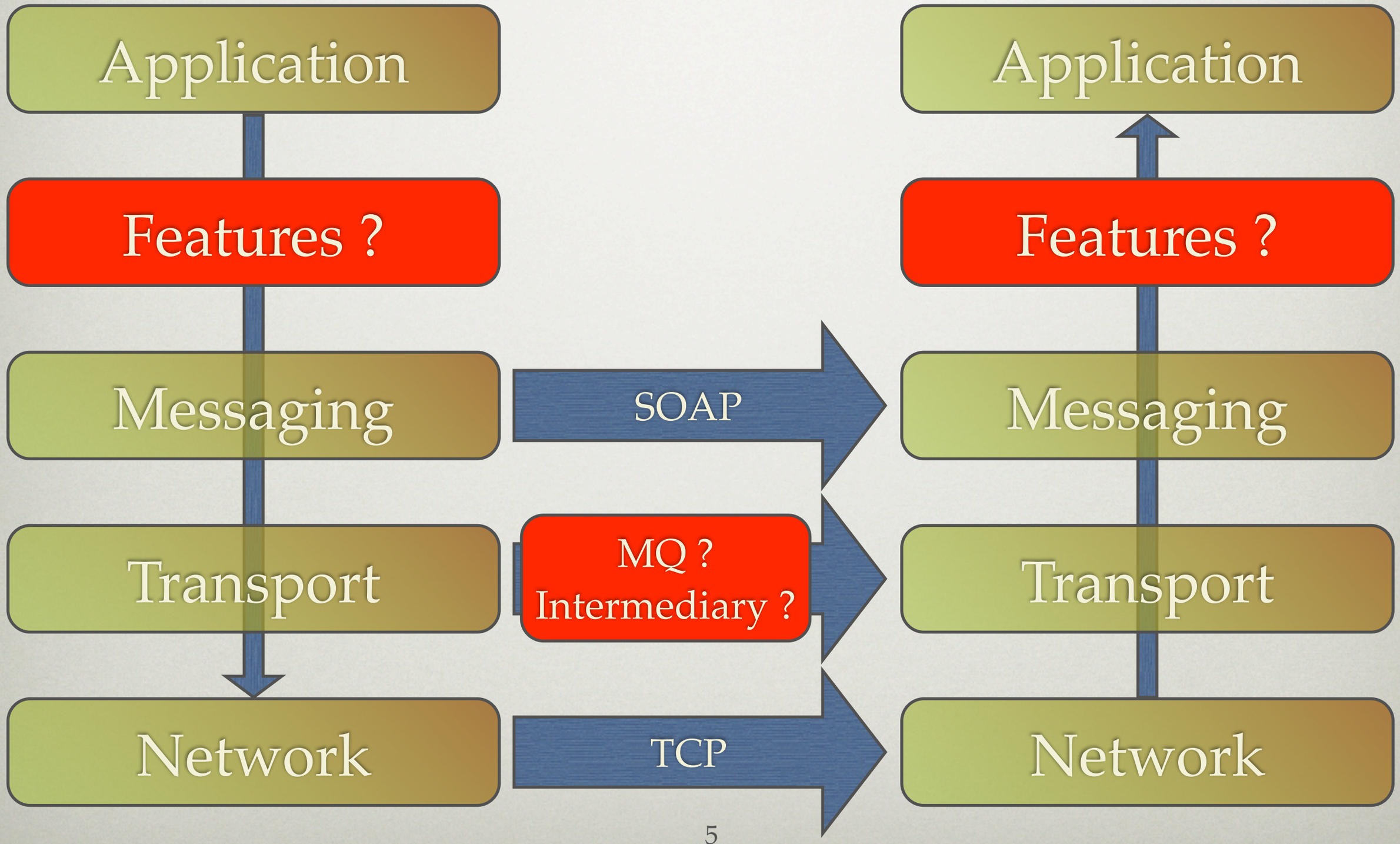


5

Interoperability On Different Layers

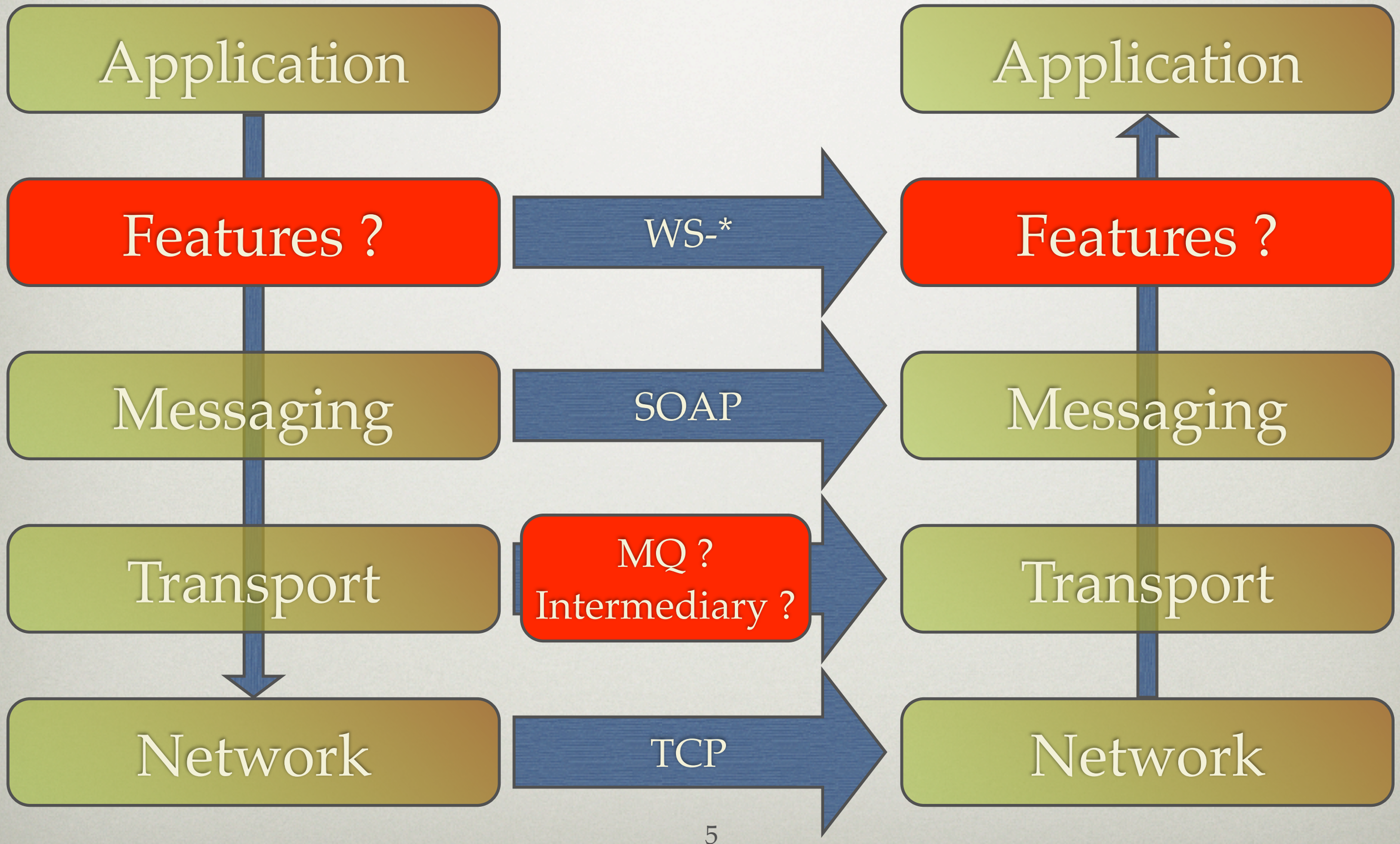


Interoperability On Different Layers



5

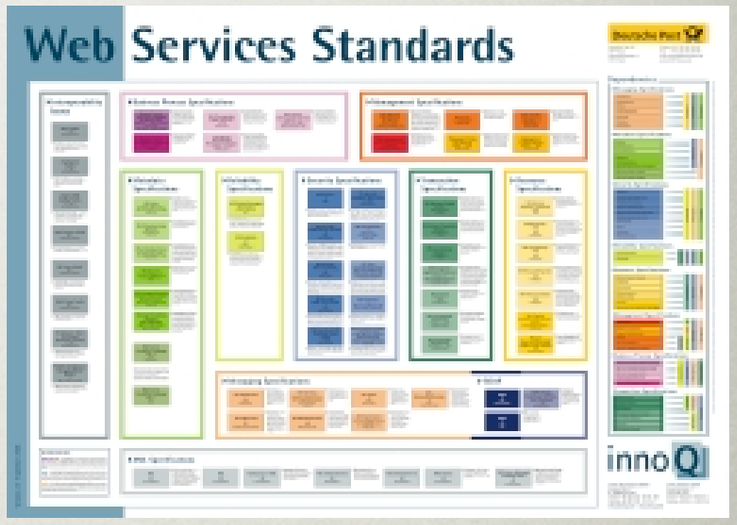
Interoperability On Different Layers



5

WS-* Standards

- Everything on-top-of SOAP, WSDL, and UDDI
- Huge number of documents, topics, and versions
- Different organizations with different influence and power
- Concurrent specifications, competing implementations
- Microsoft with / against IBM against the rest



innoq.com

6

WS-Why (Don Box)

- **WS-DesertIsland** –
  a must have for core XML Web services
- **WS-IslandResort** -
  the next layer of important specs
- **WS-NewZealand** -
  specs you'd probably need once in a lifetime
- **WS-IslandOfDoctorMoreau** -
  the ugly step children of the WS-* spec family
- **WS-FantasyIsland** - specs Don would love to see

7

OASIS
Standardization Body

- Organization for the Advancement of Structures Information
- Founded 1993 as SGML Open, till 1998
- Focus on high-level Web services
- Royalties and Patents
- Standardization track: TC draft, TC specification, OASIS standard

8

W3C
Standardization Body

- Founded 1994 by Tim Berners-Lee
- Internet standards (HTTP, HTML, XML)
- All ratified standards must be royalty-free
- Standardization track:
  - Working group note and working draft
  - Candidate recommendation
  - Proposed / W3C recommendation
- Member submission

9

WS-I
Standardization Body

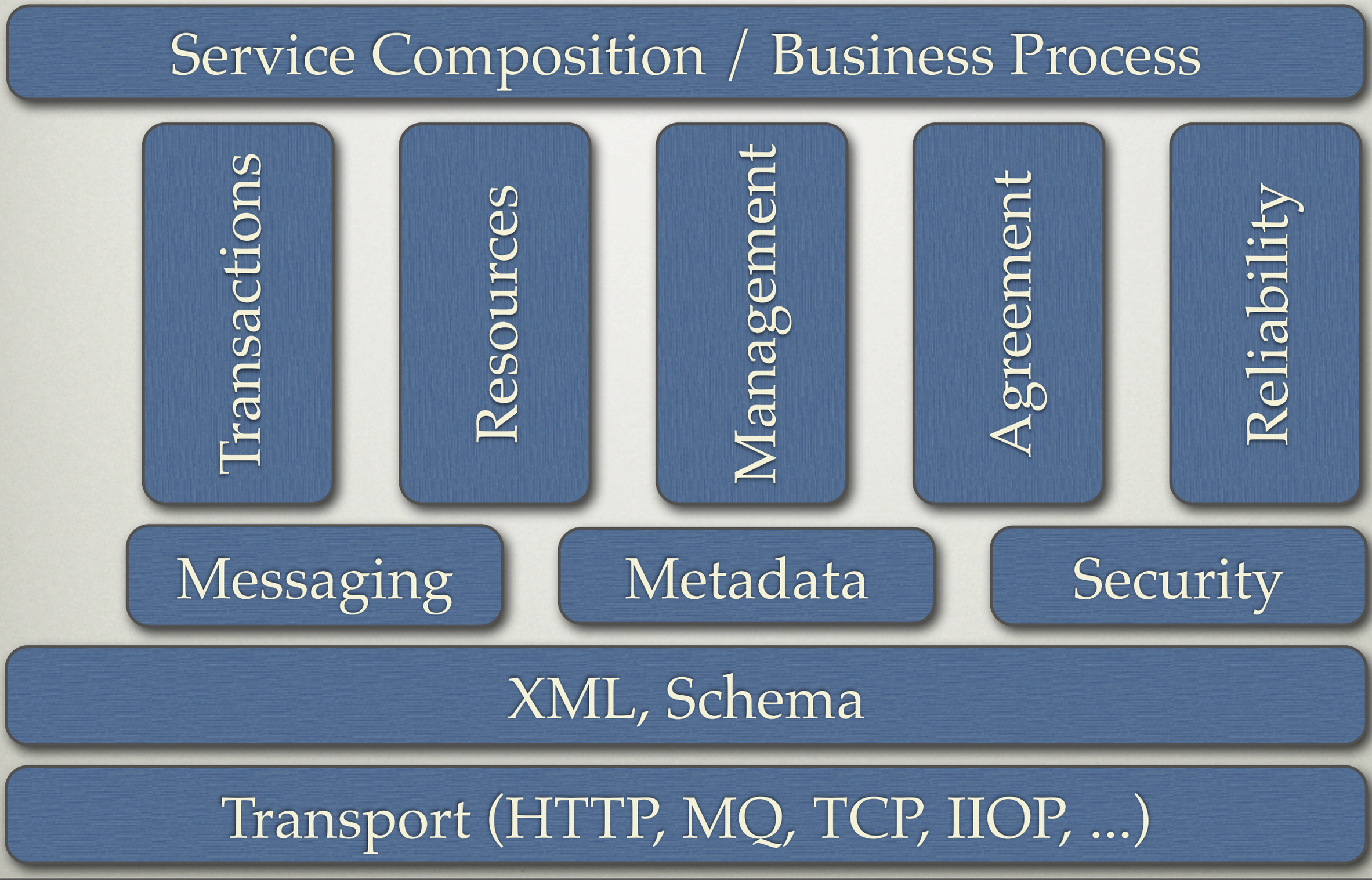
- Web Services Interoperability Organization
- Founded in 2002 by Microsoft, IBM and others
- Clarifies ambiguities and restricts WS specifications
- Profiles for basic specs (SOAP, WSDL, UDDI) and Security
- Conformance test tool chain (Java, C#)

10

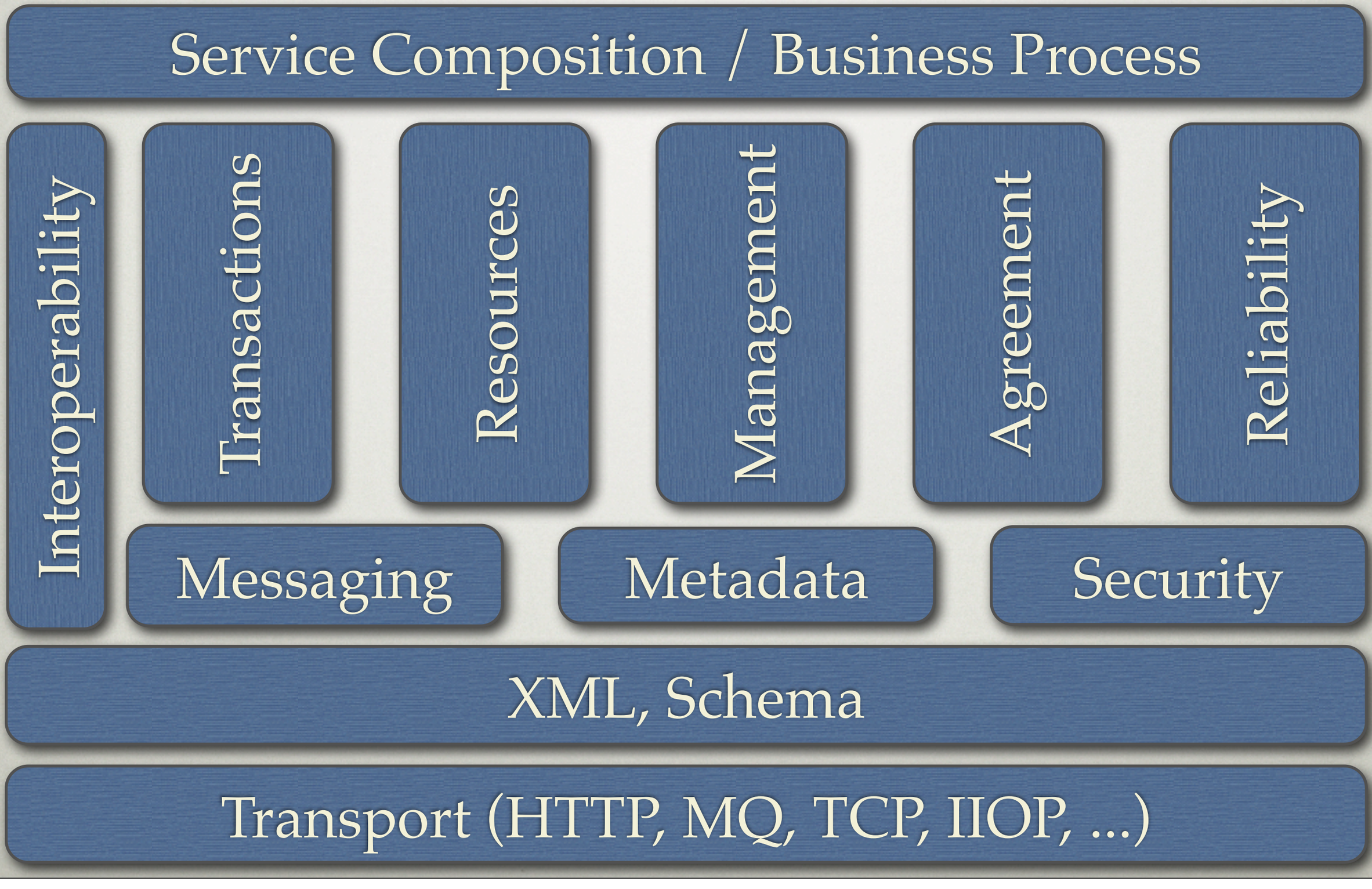
Web Service Specification Landscape

Web Service Specification Landscape

Web Service Specification Landscape



Web Service Specification Landscape



Implementations

- SOAP, WSDL, UDDI - Endless number of implementations
- WS-* specifications
  - Apache projects (Axis, WSFX, Muse, ...)
  - IBM Emerging Technologies Toolkit (ETTK)
  - Microsoft Web Services Enhancements toolkit (WSE) / Windows Communication Foundation (WCF)
  - Sun Java Java Web Services Developer Pack (WSDP)
  - Verisign Trust Service Integration Kit (TSIK)

12

WS-CheckList

- Participating Companies ?
  - Either agreed by IBM and MS,
    or we have two concurrent specifications
- Status in standardization body ?
  - Maturity of the document
- Implementations ?
  - More than one implementation is an indicator for real-world adoption
- Moving target !!!

13

WS-CheckList

- Participating Companies ?
  - Either agreed by IBM and MS,
    or we have two concurrent specifications

- Status in standardization body ?
  - Maturity of the document

- Implementations ?
  - More than one implementation is an indicator for real-world adoption

- Moving target !!!

13

Web Service Specification Landscape

Service Composition / Business Process

Interoperability

Transactions

Resources

Management

Agreement

Reliability

Messaging

Metadata

Security

XML, Schema

Transport (HTTP, MQ, TCP, IIOP, ...)

SOAP

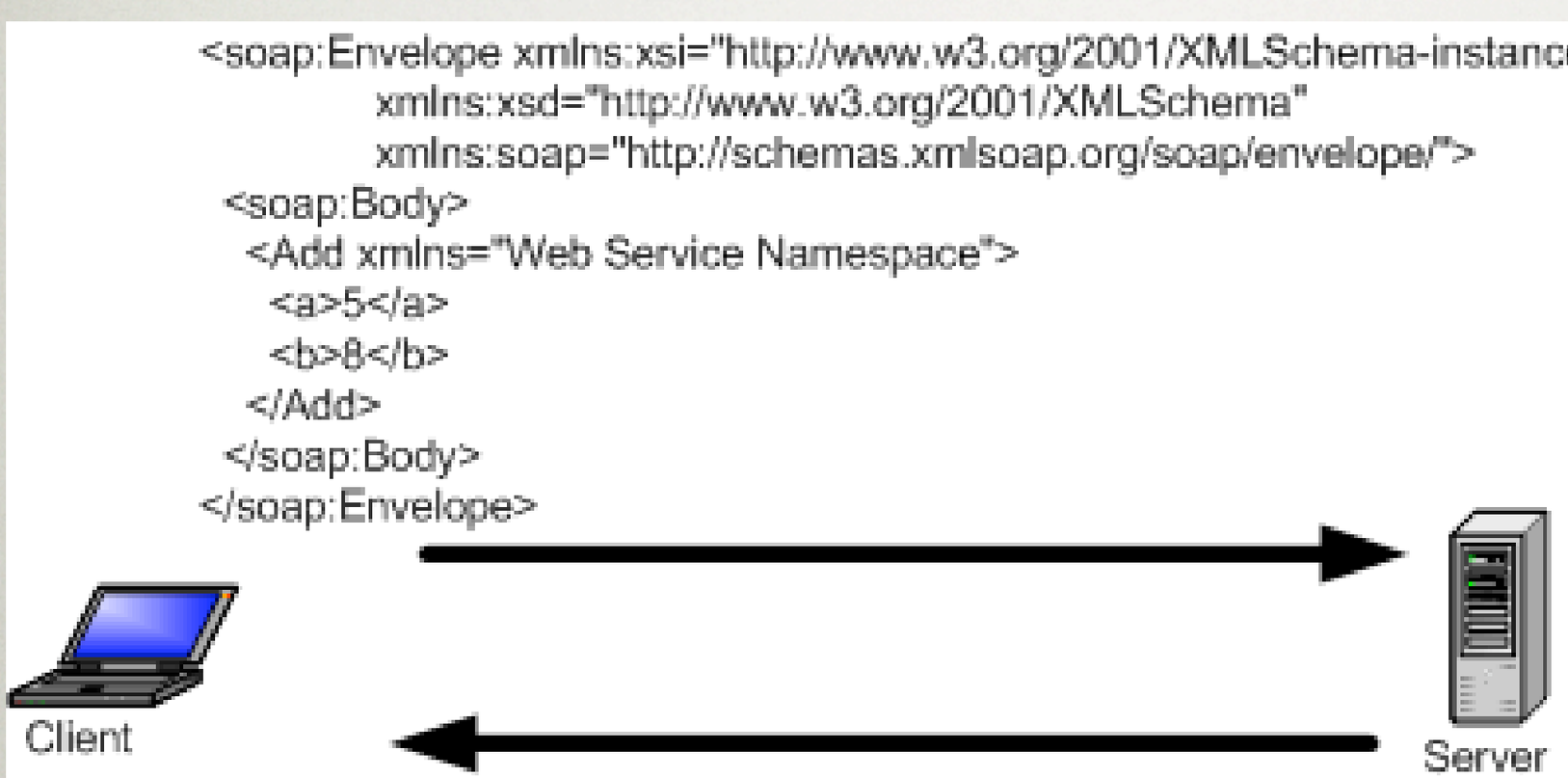
- Current version 1.2 (W3C), version 1.1 widely used (Don Box)
- Envelope structure for defining messages
- Processing model (roles, relays)
- Set of encoding rules for application specific data types
- Fault information
- Convention for doing RPC

C	IBM, MS, DevelopMentor
S	W3C Recommendation
I	1.2: All, but MS .NET 1.x

15

SOAP in 30 Seconds

```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xmlns:xsd="http://www.w3.org/2001/XMLSchema"
               xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Add xmlns="Web Service Namespace">
      <a>5</a>
      <b>8</b>
    </Add>
  </soap:Body>
</soap:Envelope>
```

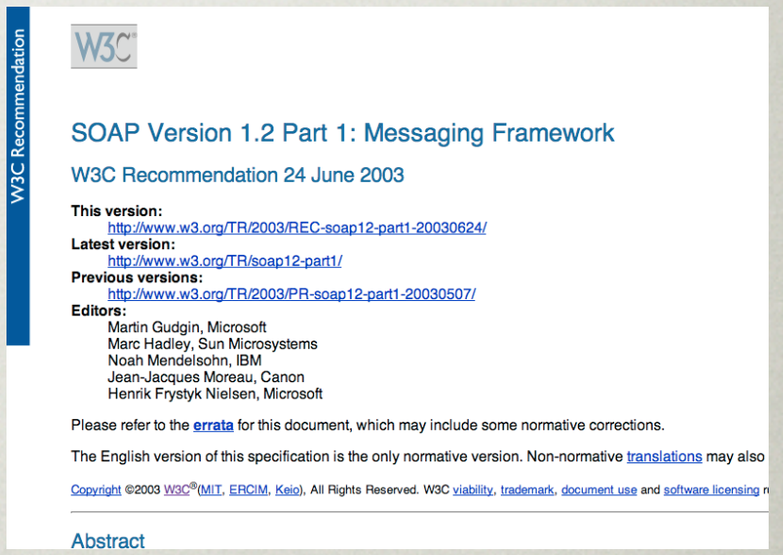



```
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
               xmlns:xsd="http://www.w3.org/2001/XMLSchema"
               xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddResponse xmlns="Web Service Namespace">
      <AddResult>13</AddResult>
    </AddResponse>
  </soap:Body>
</soap:Envelope>
```

16

Academic SOAP Usage Scenarios

- Fire-and-forget to single/multiple receiver (notifications)
- RPC, Request/response asynchronous communication
- (Multiple) Third party intermediary
- Request with acknowledgment
- Request with encrypted payload
- Multiple asynchronous responses
- Caching
- Routing


W3C Recommendation

W3C

SOAP Version 1.2 Part 1: Messaging Framework

W3C Recommendation 24 June 2003

This version:
http://www.w3.org/TR/2003/REC-soap12-part1-20030624/
Latest version:
http://www.w3.org/TR/soap12-part1/
Previous versions:
http://www.w3.org/TR/2003/PR-soap12-part1-20030507/
Editors:
Martin Gudgin, Microsoft
Marc Hadley, Sun Microsystems
Noah Mendelsohn, IBM
Jean-Jacques Moreau, Canon
Henrik Frystyk Nielsen, Microsoft

Please refer to the **errata** for this document, which may include some normative corrections.

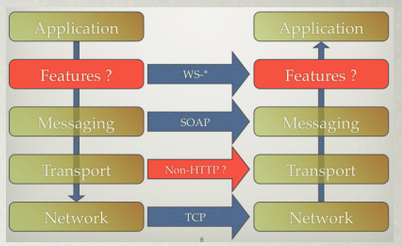
The English version of this specification is the only normative version. Non-normative translations may also

Copyright ©2003 W3C®(MIT, ERCIM, Keio), All Rights Reserved. W3C viability, trademark, document use and software licensing r

Abstract

17

WS-Addressing



- Scenario: We need a transport-neutral session identifier in our SOAP requests
  - Altering (HTTP) URL does not help
  - Standardized extension of SOAP header needed
- Scenario: Transport-neutral transmission through processing nodes
  - Asynchronous /event-based processing
- Endpoint reference (EPR)
- Message Information (MI) header

C	IBM, Bea, MS, Sun, SAP, ...
S	W3C Recommendation
I	Axis, ETTK, WSE, ...

18

WS-Addressing Endpoint Reference

- Convey the information for endpoint identification and referencing in an XML document
  - WSDL is for the service, EPR is for the endpoint
- Inclusion of parameters in endpoint description
  - Provided by reference issuer, opaque to client
- Inclusion of metadata
  - WSDL information can be part of the EPR
- Mapping to SOAP 1.1 / 1.2 and WSDL 1.1 / 2.0

19

Endpoint Reference Example

```
<wsa:EndpointReference
  xmlns:wsa="http://www.w3.org/2005/08/addressing"
  xmlns:shop="mailto:admin@example.org">
  <wsa:Address>
     http://exampleshopping.net/hello/index.asmx
  </wsa:Address>
  <wsa:ReferenceParameters>
     <shop:UserProfile>123456</shop:UserProfile>
     <shop:SessionID>987654</shop:SessionID>
  </wsa:ReferenceParameters>
  <wsa:Metadata>
     <wsdl:definitions
        targetNamespace="http://example.org/shop"
        xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
        ...
     </wsdl:definitions>
  </wsa:Metadata>
</wsa:EndpointReference>
```


20

WS-A Message Addressing Headers

- Transport-neutral end-to-end message characteristics as additional SOAP header entries
- Communication of information that is distributed over SOAP and transport layer
  - Source and destination endpoint
  - Reply and fault endpoint (reliable messaging)
  - Message identifier
  - Policy
- Basis for many on-top WS specifications

21

Message Addressing Properties

```
<wsa:To>
  xs:anyURI</wsa:To>
<wsa:From>
  wsa:EndpointReferenceType</wsa:From>
<wsa:ReplyTo>
  wsa:EndpointReferenceType</wsa:ReplyTo>
<wsa:FaultTo>
  wsa:EndpointReferenceType</wsa:FaultTo>
<wsa:Action>
  xs:anyURI</wsa:Action>
<wsa:MessageID>
  xs:anyURI</wsa:MessageID>
<wsa:RelatesTo
RelationshipType="xs:anyURI">
  xs:anyURI</wsa:RelatesTo>
<wsa:ReferenceParameters>
  xs:any*</wsa:ReferenceParameters>
```

Message Addressing Properties

- Destination (mandatory)
  - IRI of the intended receiver, independent from transport destination information
  - Equal to HTTP request URL in usual SOAP case
- Source endpoint (optional)
  - Endpoint where the message was originated
- Predefined address values
  - http://www.w3.org/2005/08/addressing/anonymous
  - http://www.w3.org/2005/08/addressing/none

23

Message Addressing Properties

- Reply endpoint (optional)
  - Intended receiver for replies to this message
  - Source endpoint MAY be used when absent
  - Requires also message ID
- Fault endpoint (optional)
  - Intended receiver for faults related to this message
  - Reply or source endpoint MAY be used when absent
  - Requires also message ID

24

Message Addressing Properties

- Relationship value (optional)
  - Relationship type as attribute
  - Only reply type is predefined by the spec
- Action specifier (optional)
  - Similar to SOAP action specifier
- Message identifier (optional)
  - Identifies the message uniquely in time and space
  - Must be set when a reply is expected

25

Example SOAP Request

```
<S:Envelope
  xmlns:S="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <S:Header>
    <wsa:MessageID>http://example.com/4711</wsa:MessageID>
    <wsa:ReplyTo>
      <wsa:Address>http://example.com/client1</wsa:Address>
    </wsa:ReplyTo>
    <wsa:To>mailto:fabrikam@example.com</wsa:To>
    <wsa:Action>http://example.com/mail/Delete</wsa:Action>
  </S:Header>
  <S:Body>
    <f:Delete xmlns:f="http://example.com/">
       <maxCount>42</maxCount>
    </f:Delete>
  </S:Body>
</S:Envelope>
```

Example SOAP Response

```
<S:Envelope
  xmlns:S="http://www.w3.org/2003/05/soap-envelope"
  xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <S:Header>
    <wsa:MessageID>http://example.com/1147</wsa:MessageID>
    <wsa:RelatesTo>http://example.com/4711</wsa:RelatesTo>
    <wsa:To>http://example.com/client1</wsa:To>
    <wsa:Action>http://example.com/mail/DeleteAck</wsa:Action>
  </S:Header>
  <S:Body>
    <f:DeleteAck xmlns:f="http://example.com/"/>
  </S:Body>
</S:Envelope>
```

WS-Addressing Conclusion

- All relevant information for processing and routing inside of the SOAP envelope
- Less problems with intermediaries or non-HTTP transport protocols
- Client is unaware of additional information needed by the provider
  - Supports factory / registry patterns
- Support for asynchronous messaging
- Standardized way to pass around WS references

C	IBM, Bea, MS, Sun, SAP, ...
S	W3C Recommendation
I	Axis, ETTK, WSE, ...

28

Message Transmission Optimization Mechanism

- Optimization of the transmission and/or wire format of SOAP messages
- Backward compatible to *SOAP with Attachments*
- Based on XOP (XML-binary optimized packaging)
  - Repackaging & transmitting *base64Binary* data in native format, keep link in original Infoset
  - Base64 data comes in separate MIME parts
  - MIME / Multipart serialization of SOAP messages
- Hop-by-hop contract

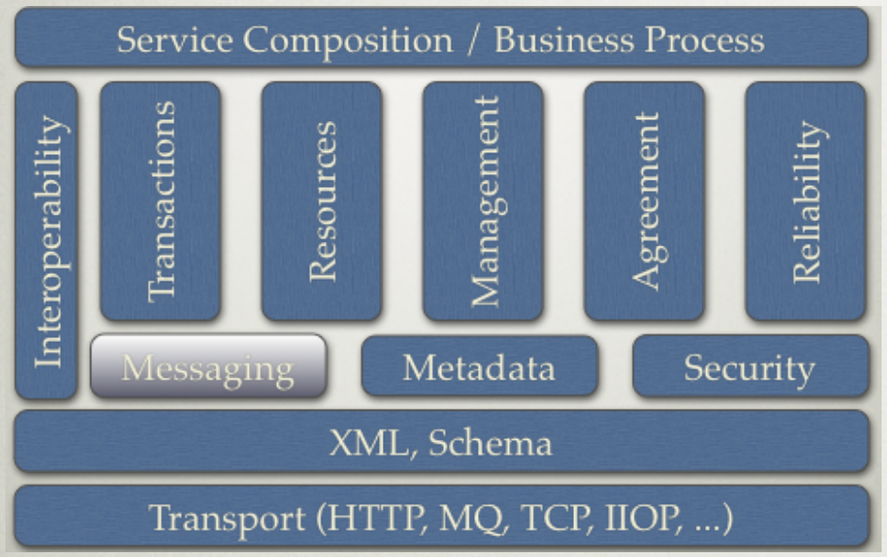
C	MS, IBM, Bea, Canon
S	W3C Recommendation
I	Axis2, WCF

29

```
Content-Type: multipart/related;
boundary=MIMEBoundary4A7AE55984E7438034;
        type="application/xop+xml";
...
--MIMEBoundary4A7AE55984E7438034
content-type: application/xop+xml; charset=utf-8;
type="application/soap+xml;"
content-transfer-encoding: binary
...
<?xml version='1.0' encoding='utf-8'?>
<soapenv:Envelope>
  ...
        <xop:Include href="cid:1.A91D6D2E3D7AC4D580@apache.org"
             xmlns:xop="http://www.w3.org/2004/08/xop/include">
        </xop:Include>
  ...
</soapenv:Envelope>
--MIMEBoundary4A7AE55984E7438034
content-type: application/octet-stream
content-transfer-encoding: binary
content-id: <1.A91D6D2E3D7AC4D580@apache.org>

Binary Data.....
--MIMEBoundary4A7AE55984E7438034--
```

Asynchronous Messaging



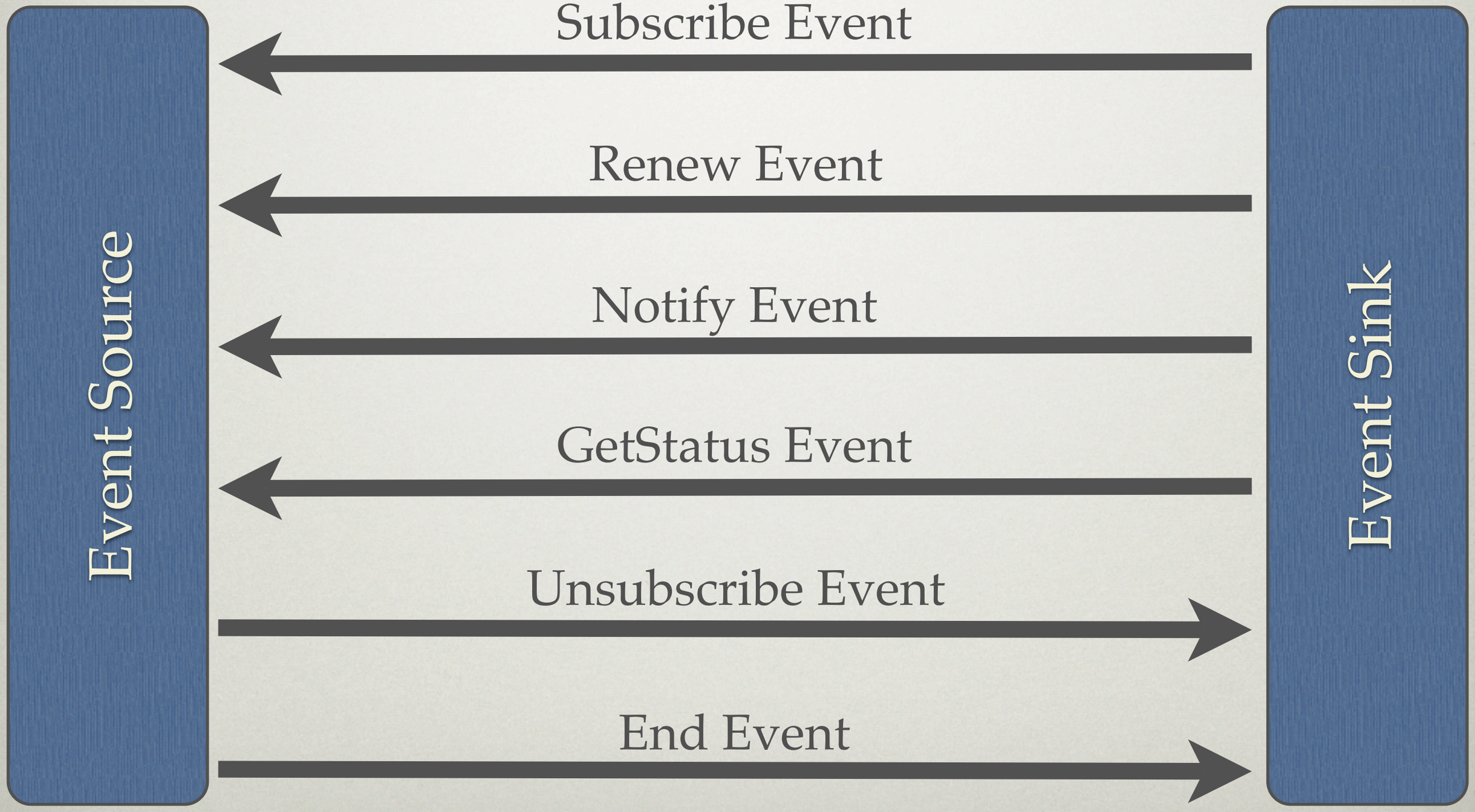
- WS-Notification 1.2 (IBM)
  - WS-BaseNotification
  - WS-BrokeredNotification
- WS-Eventing (IBM, MS)

C	IBM, Tibco, HP
S	OASIS Standard
I	ETTK, Axis, ...

C	IBM, Tibco, MS
S	W3C Member Sub.
I	ETTK, Axis, ...

31

Push Notification With Web Services



32

WS-Notification

- **Web Service Base Notification 1.3**
  - Interfaces for consumers and producers
- **Web Services Topics 1.3 (WS-Topics)**
  - Mechanisms to organize and categorize items of interest for subscription
- **Web Services Brokered Notification 1.3**
  - Interface for Notification Broker

33

WS-baseNotification

- Different roles with their endpoints -
  producer, consumer, subscription manager, subscriber
- Two ways of notification (no response expected)
  - Producer sends raw application-specific content
  - Producer send special Notify message
    - Subscription reference
    - Topic and topic dialect
    - Producer reference

34

Notify Message Infoset

```
...
<wsnt:Notify>
  <wsnt:NotificationMessage>
    <wsnt:SubscriptionReference>
      wsa:EndpointReferenceType
    </wsnt:SubscriptionReference> ?
    <wsnt:Topic Dialect="xsd:anyURI">
      {any} ?
    </wsnt:Topic>?
    <wsnt:ProducerReference>
      wsa:EndpointReferenceType
    </wsnt:ProducerReference> ?
    <wsnt:Message>
      {any}
    </wsnt:Message>
  </wsnt:NotificationMessage> +
  {any} *
</wsnt:Notify>
```


35

SOAP Notify Example

```
<s:Envelope><s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-2/NotificationConsumer/Notify
    </wsa:Action>
    ...
</s:Header><s:Body>
    <wsnt:Notify><wsnt:NotificationMessage>
      <wsnt:SubscriptionReference>
        <wsa:Address>http://www.example.org/SubManager</wsa:Address>
      </wsnt:SubscriptionReference>
      <wsnt:Topic Dialect=
        "http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
        npex:SomeTopic
      </wsnt:Topic>
      <wsnt:ProducerReference>
        <wsa:Address>http://www.example.org/NotiProd</wsa:Address>
      </wsnt:ProducerReference>
      <wsnt:Message>
        <npex:NotifyContent>exampleNotifyContent</npex:NotifyContent>
      </wsnt:Message>
    <wsnt:NotificationMessage></wsnt:Notify>
</s:Body></s:Envelope>
```

WS-Notification Subscription

- Several options for subscription
  - Notification consumer EPR
  - Boolean filter expressions
    - Order and timing for tests defined by producer
    - Topic, or XPath expression to check message content
  - Initial termination time
  - Subscription policy (e.g. # of messages)
  - Indicator for raw subscription
- Subscription response with EPR and termination time

Subscription Example

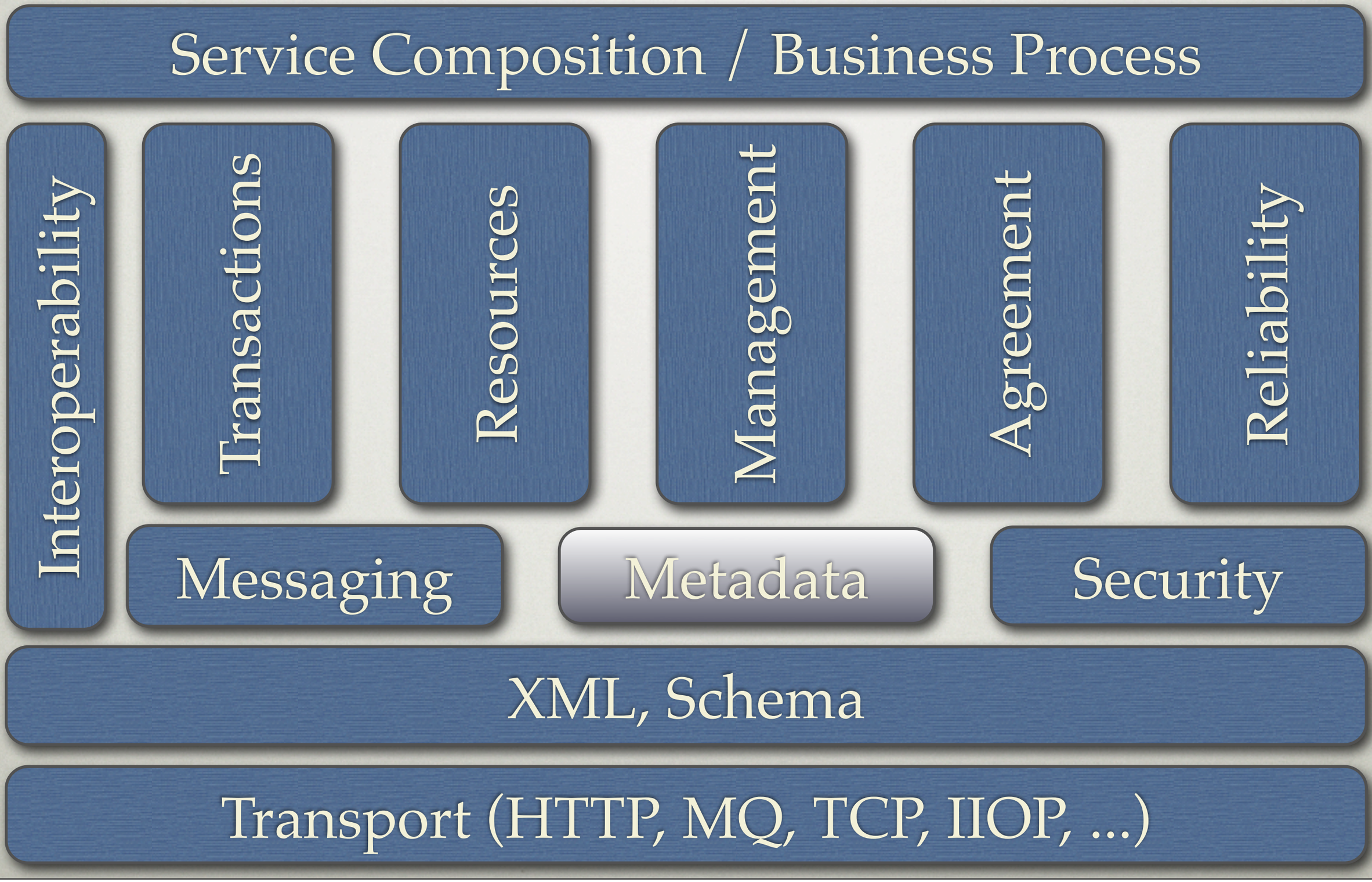
```
<s:Envelope><s:Header>
  <wsa:Action>
    http://docs.oasis-open.org/wsn/bw- 2/NotificationProducer/SubscribeRequest
  </wsa:Action>
  ...
</s:Header><s:Body>
  <wsnt:Subscribe>
    <wsnt:ConsumerReference>
      <wsa:Address>http://www.example.org/NotificationConsumer</wsa:Address>
    </wsnt:ConsumerReference>
    <wsnt:Filter>
      <wsnt:TopicExpression
          Dialect="http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple">
        npex:SomeTopic
      </wsnt:TopicExpression>
      <wsnt:MessageContent
          Dialect="http://www.w3.org/TR/1999/REC-xpath-19991116">
        boolean(ncex:Producer="15")
      </wsnt:MessageContent>
    </wsnt:Filter>
    <wsnt:InitialTerminationTime>
      2005-12-25T00:00:00.00000Z
    </wsnt:InitialTerminationTime>
  </wsnt:Subscribe>
</s:Body></s:Envelope>
```

WS-Notification vs. WS-Eventing

- WS-Notification features
  - Support for small devices
    (restricted set of mandatory features)
  - Support for direct and brokered notification
  - Transformation and aggregation of Topics
  - Runtime metadata (e.g. available subscription types)
  - Broker federations
  - Based on *WS-ResourceProperties* and
    WS-ResourceLifetime (from WSRF)

39

Web Service Specification Landscape



Metadata

- Web Service Description Language (WSDL) 1.1 / 2.0
- Universal Description, Discovery and Integration (UDDI)
- WS-MetadataExchange
- WS-Policy (WS-PolicyAssertions, WS-PolicyAttachment)
- WS-Discovery (Bea, Canon, Intel, Microsoft)
- WS-Agreement
- WS-Inspection

41

UDDI

- Universal Description Discovery and Integration
- Repository broker service for discovery of services, offered as Web service
- Data model for service info and metadata
- UDDI Business Registry
- Part of WS-I Basic Profile
- Unbeloved child of the WS community

C	Ariba, IBM, Microsoft
S	v3 OASIS standard
I	Microsoft, IBM, SAG, ...

42

WS-MetadataExchange

- Web Service consumers need metadata to interact
  - Messages, protocols, endpoint addresses
  - Policies

- Request for specific metadata type (dialect)
- Simple interaction to get service description (communication bootstrapping)
- Intended for retrieval of WSDL and WS-Policy

C	SAP, IBM, Microsoft, Bea, CA, ...
S	Industry Proposal
I	Microsoft, IBM, ...

43

Metadata Request Example

```
<s12:Envelope
   xmlns:s12='http://www.w3.org/2003/05/soap-envelope'
   xmlns:wsa='http://schemas.xmlsoap.org/ws/2004/08/addressing'
   xmlns:wsx='http://schemas.xmlsoap.org/ws/2004/09/mex'>
<s12:Header>
   <wsa:Action>
       http://schemas.xmlsoap.org/ws/2004/09/
                mex/GetMetadata/Request
   </wsa:Action>
   <wsa:MessageID>4f6d5c6027f4</wsa:MessageID>
   <wsa:ReplyTo>
      <wsa:Address>http://example.com/MyEndpoint</wsa:Address>
   </wsa:ReplyTo>
   <wsa:To>http://server.example.org/YourEndpoint</wsa:To>
</s12:Header>
<s12:Body>
   <wsx:GetMetadata>
   <wsx:Dialect>http://schemas.xmlsoap.org/wsdl/</wsx:Dialect>
</s12:Body>
</s12:Envelope>
```

Metadata Response Example

```
<s12:Envelope
   xmlns:s12='http://www.w3.org/2003/05/soap-envelope‘
   xmlns:wsa='http://schemas.xmlsoap.org/ws/2004/08/addressing'
   xmlns:wsx='http://schemas.xmlsoap.org/ws/2004/09/mex' >
<s12:Header>
   <wsa:Action>
       http://schemas.xmlsoap.org/ws/2004/09/
              mex/GetMetadata/Response
   </wsa:Action>
   <wsa:RelatesTo>4f6d5c6027f4</wsa:RelatesTo>
   <wsa:To>http://client.example.com/MyEndpoint</wsa:To>
</s12:Header>
<s12:Body>
   <wsx:Metadata>
     <wsx:MetadataSection
        Dialect='http://schemas.xmlsoap.org/wsdl/'>
          <wsdl:definitions … </wsdl:definitions>
     </wsx:MetadataSection>
   </wsx:Metadata>
</s12:Body>
</s12:Envelope>
```

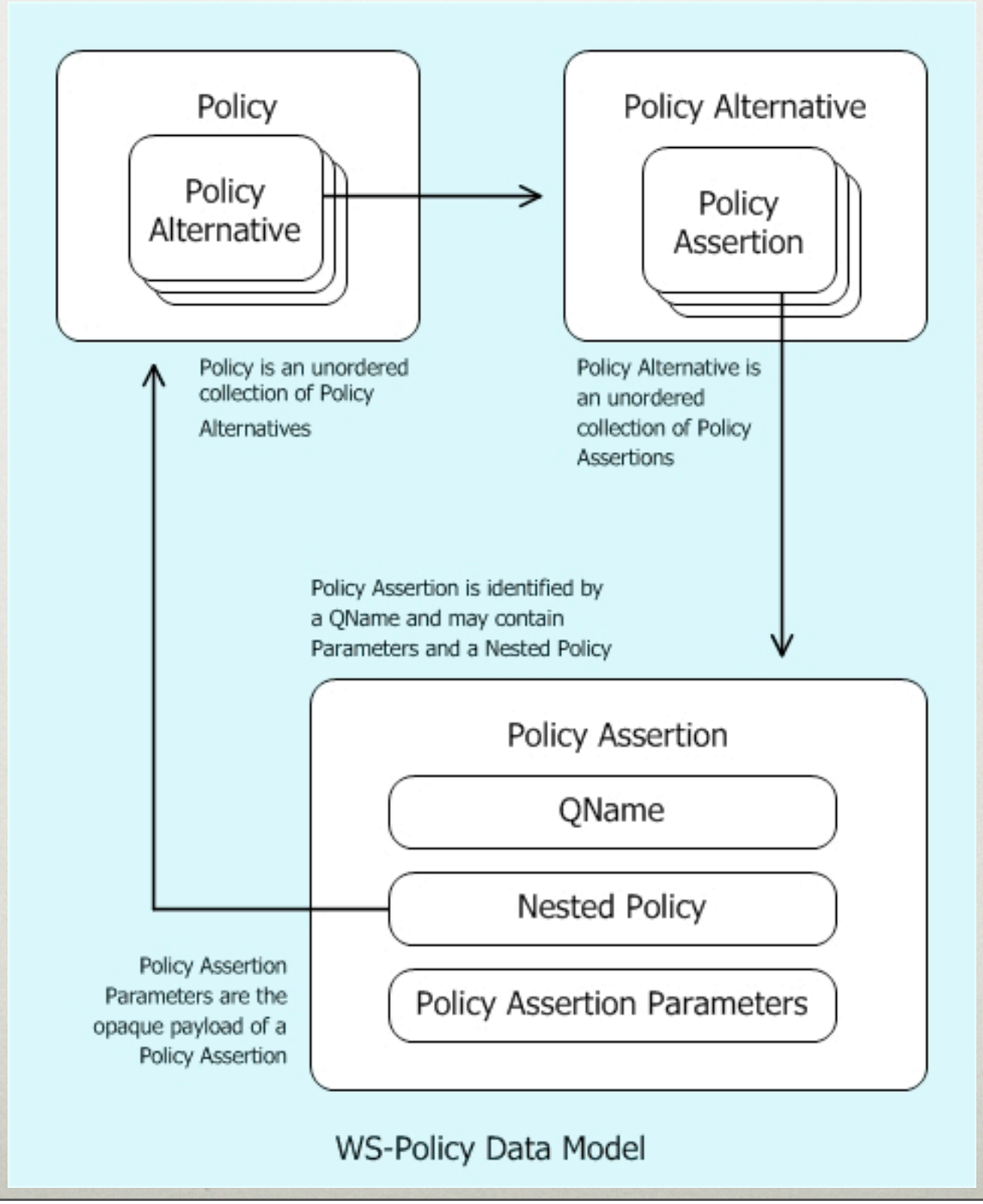
WS-Policy

- Describe and communicate the policies of a web service
  - Policy: possibly empty collection of policy alternatives
  - Policy alternative: possibly empty collection of policy assertions
  - Policy assertion: Domain-specific requirement or capability, or other property of behaviour
    - ‘Wire’ properties (authentication, transport)
    - Extended properties (e.g. QoS characteristics)

C	SAP, IBM, Microsoft, Bea, ...
S	W3C Working Draft
I	Microsoft, Axis2, IBM, ...

46

WS-Policy Data Model



WS-Policy Data Model

WS-Policy Example

Policy Alternative

Policy Assertion

```
<wsp:Policy>
  <wsp:ExactlyOne>
    <wsp:All>
      <wsse:SecurityToken>
        <wsse:TokenType>wsse:Kerberosv5TGT</wsse:TokenType>
      </wsse:SecurityToken>
    </wsp:All>
    <wsp:All>
      <wsse:SecurityToken>
        <wsse:TokenType>wsse:X509v3</wsse:TokenType>
      </wsse:SecurityToken>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
```


48

WS-Policy Extensions

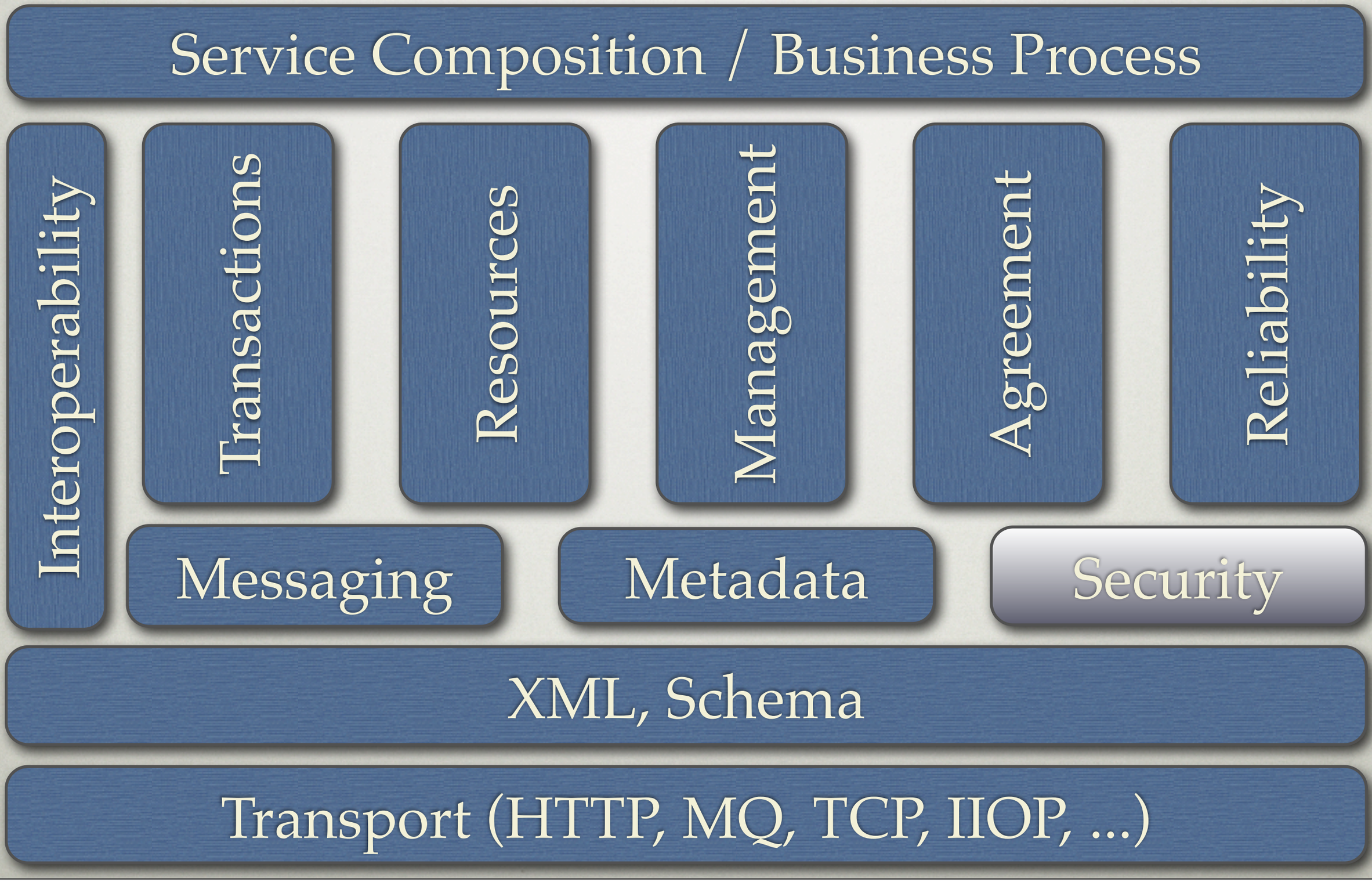
- WS-PolicyAttachment
  - Associating policies with their subjects
  - Reference policies from WSDL definitions
  - Associate policies with deployed WS endpoints
  - Associate policies with UDDI entities
- WS-SecurityPolicy
  - WS-Policy assertions that apply to WS-Security
- WS-PolicyAssertions
  - Common message policy assertions

49

Attaching Policy Expressions To WSDL

```
<Policy wsu:Id="common2">
  <mtom:OptimizedMimeSerialization wsp:Optional="true"/>
  <wsap:UsingAddressing />
</Policy>
<Policy wsu:Id="secure2">
  <ExactlyOne>
   <sp:TransportBinding>…</sp:TransportBinding>
   <sp:AsymmetricBinding>…</sp:AsymmetricBinding >
  </ExactlyOne>
</Policy>
<wsdl:binding name="SecureBinding" type="tns:RealTimeDataInterface" >
  <PolicyReference URI="#secure2" />
  <wsdl:operation name="GetRealQuote">…</wsdl:operation>
  …
</wsdl:binding>
<wsdl:service name="RealTimeDataService">
  <wsdl:port name="RealTimeDataPort" binding="tns:SecureBinding">
    <PolicyReference URI="#common2"/>
    …
  </wsdl:port>
</wsdl:service>
```

Web Service Specification Landscape



Security

- WS-Security
  - SOAP message security
- Several on-top-of specifications
  - WS-Trust
    - Dissemination of credentials between different trust domains
    - Security token service
  - WS-SecureConversation
    - Establish and share security context for connections

C	IBM, Microsoft, Sun, Verisign
S	v1.x: OASIS Standard
I	WSE, ETTK, ...

52

Security Triad - CIA

- Confidentiality
  - Keeping secrets a secret
  - Encryption and access control
  - Ensure that parties are really who they claim to be
- Integrity
  - No unintended modification
  - Signing and hashing
  - Might include non-repudiation
- Availability (DDOS)

WS-Security

- Generic mechanism to associate security tokens with messages
- How to encode binary security tokens (X.509, Kerberos)
- Based on collection of different specifications
  - *XML Signature* and *XML Encryption* to secure XML elements in a standardized manner
- Different profile specifications for authentication mechanisms -> unsigned / signed security tokens
  - Username / X.509 / SAML / Kerberos Token Profile

54

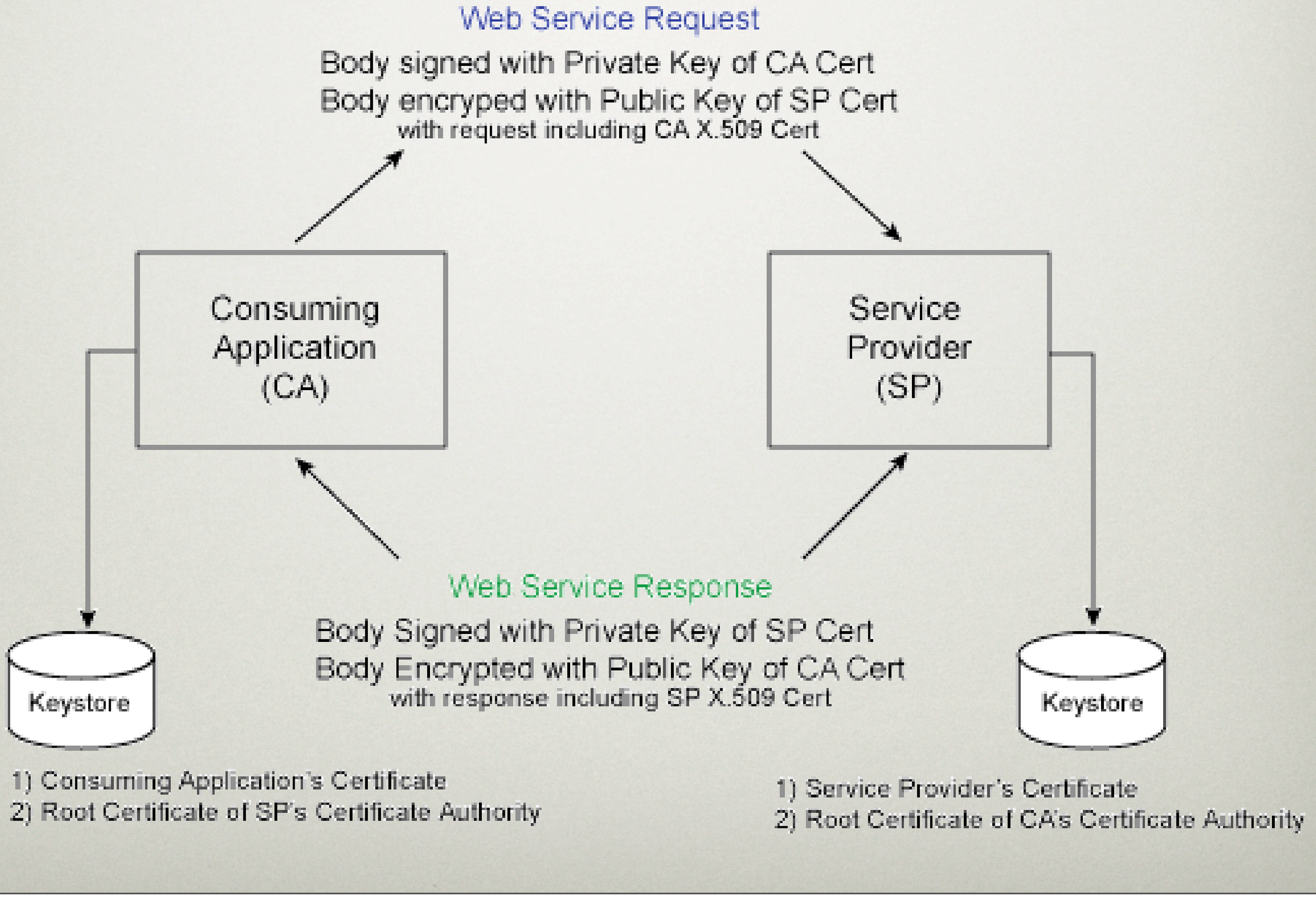
Token Examples

```
<wsse:UsernameToken>
    <wsse:Username>Sven</wsse:Username>
    <wsse:Password Type="wsse:PasswordText">Kennwort</wsse:Password>
</wsse:UsernameToken>
```

```
 <wsse:UsernameToken>
    <wsse:Username>Sven</wsse:Username>
    <wsse:Password Type="wsse:PasswordDigest">
        KE6QugOpkPyT3Eo0SEgT30W4Keg=</wsse:Password>
    <wsse:Nonce>5uW4ABku/m6/S5rnE+L7vg==</wsse:Nonce>
    <wsu:Created>2002-08-19T00:44:02Z</wsu:Created>
</wsse:UsernameToken>
```

```
  <wsse:BinarySecurityToken
      ValueType="wsse:X509v3"
      EncodingType="wsse:Base64Binary">
        MIIHdjCCB...</wsse:BinarySecurityToken>
```

X.509 Scenario



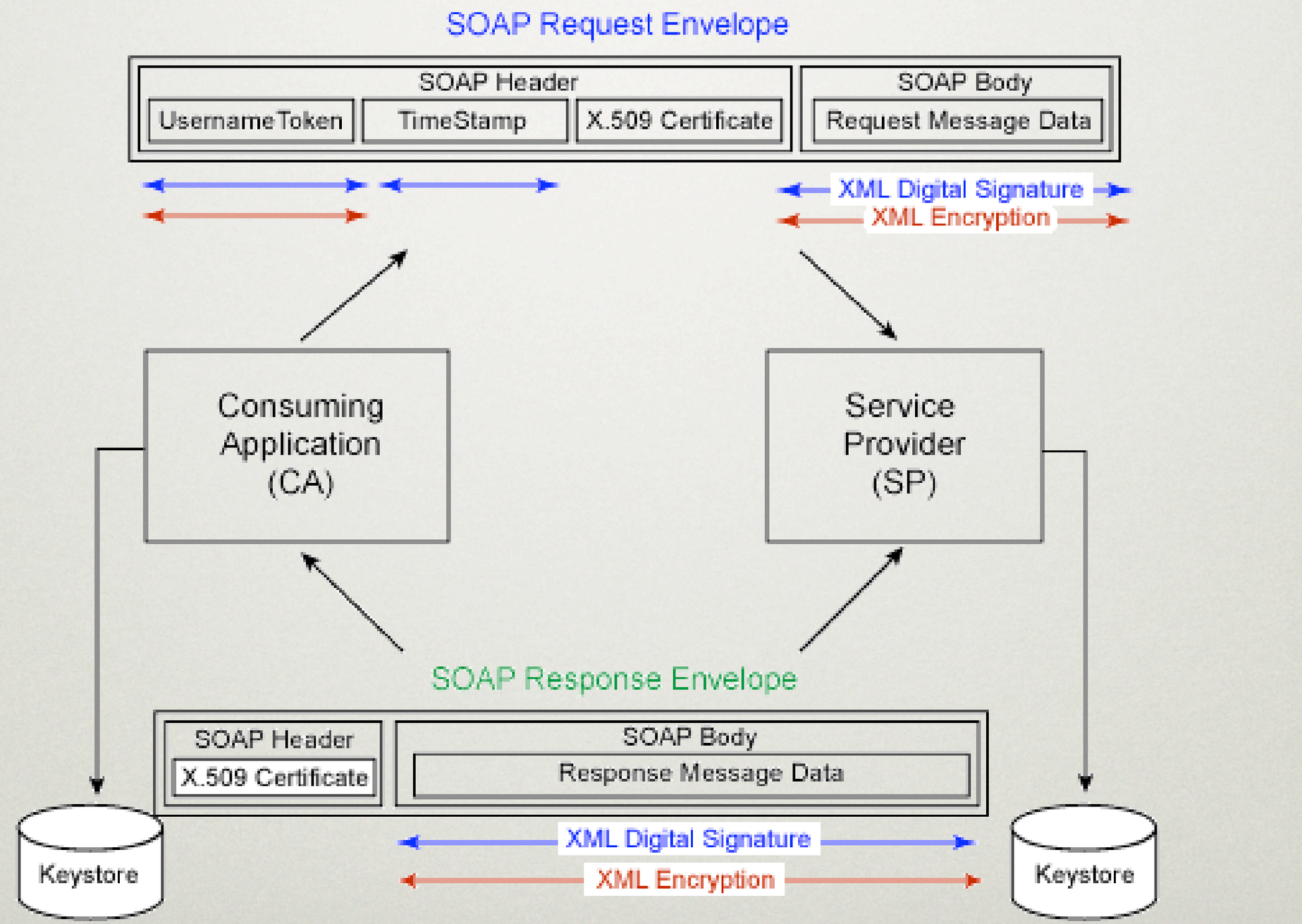
XML Signature + WS-Security Example

```
<?xml version="1.0" encoding="utf-8"?>
<S:Envelope xmlns:S="http://www.w3.org/2001/12/soap-envelope"
            xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
            xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/04/secext"
            xmlns:xenc="http://www.w3.org/2001/04/xmlenc#">
<S:Header>
   <wsse:Security>
      <wsse:BinarySecurityToken ValueType="wsse:X509v3" EncodingType="wsse:Base64Binary"
                                Id="X509Token">
            MIIEZzCCA9CgAwIBAgIQEmtJZc0rqrKh5i...
      </wsse:BinarySecurityToken>
      <ds:Signature>
          <ds:SignedInfo>...</ds:SignedInfo>
          <ds:SignatureValue>BL8jdfToEb1l/vXcMZNNjPOV...</ds:SignatureValue>
          <ds:KeyInfo>
              <wsse:SecurityTokenReference>
                 <wsse:Reference URI="#X509Token"/>
              </wsse:SecurityTokenReference>
          </ds:KeyInfo>
      </ds:Signature>
   </wsse:Security>
</S:Header>
<S:Body>
   <tru:StockSymbol xmlns:tru="http://fabrikam123.com/payloads">
      QQQ
   </tru:StockSymbol>
</S:Body>
</S:Envelope>
```

Encryption Example

```
<S11:Envelope xmlns:S11="..." xmlns:ds="..." xmlns:wsse="..." xmlns:xenc="...">
<S11:Header>
  <wsse:Security>
     <xenc:EncryptedKey>
        <xenc:EncryptionMethod Algorithm="..."/>
        <ds:KeyInfo>
           <wsse:SecurityTokenReference>
             <ds:X509Data>
               <ds:X509IssuerSerial>
                 <ds:X509IssuerName>DC=ACMECorp, DC=com</ds:X509IssuerName>
                 <ds:X509SerialNumber>12345678</ds:X509SerialNumber>
               </ds:X509IssuerSerial>
             </ds:X509Data>
           </wsse:SecurityTokenReference>
        </ds:KeyInfo>
        <xenc:CipherData><xenc:CipherValue>...</xenc:CipherValue></xenc:CipherData>
        <xenc:ReferenceList>
           <xenc:DataReference URI="#encrypted"/>
        </xenc:ReferenceList>
     </xenc:EncryptedKey>
  </wsse:Security>
</S11:Header><S11:Body>
  <xenc:EncryptedData Id="encrypted" Type="...">
     <xenc:CipherData><xenc:CipherValue>...</xenc:CipherValue></xenc:CipherData>
  </xenc:EncryptedData>
</S11:Body></S11:Envelope>
```

WS-Security Usage



Web Service Specification Landscape

Service Composition / Business Process

Interoperability

Transactions

Resources

Management

Agreement

Reliability

Messaging

Metadata

Security

XML, Schema

Transport (HTTP, MQ, TCP, IIOP, ...)

Reliability

- Reliable point-to-point transmission of messages (message acknowledgment, retransmission)

- WS-ReliableMessaging

C	IBM, MS, Bea, TIBCO, ...
S	Industry proposal
I	Apache, ETTK, WSE, ...

- WS-Reliability

C	Sun, Oracle, Fujitsu, ...
S	v1.1: OASIS standard
I	Apache, WSE, ...

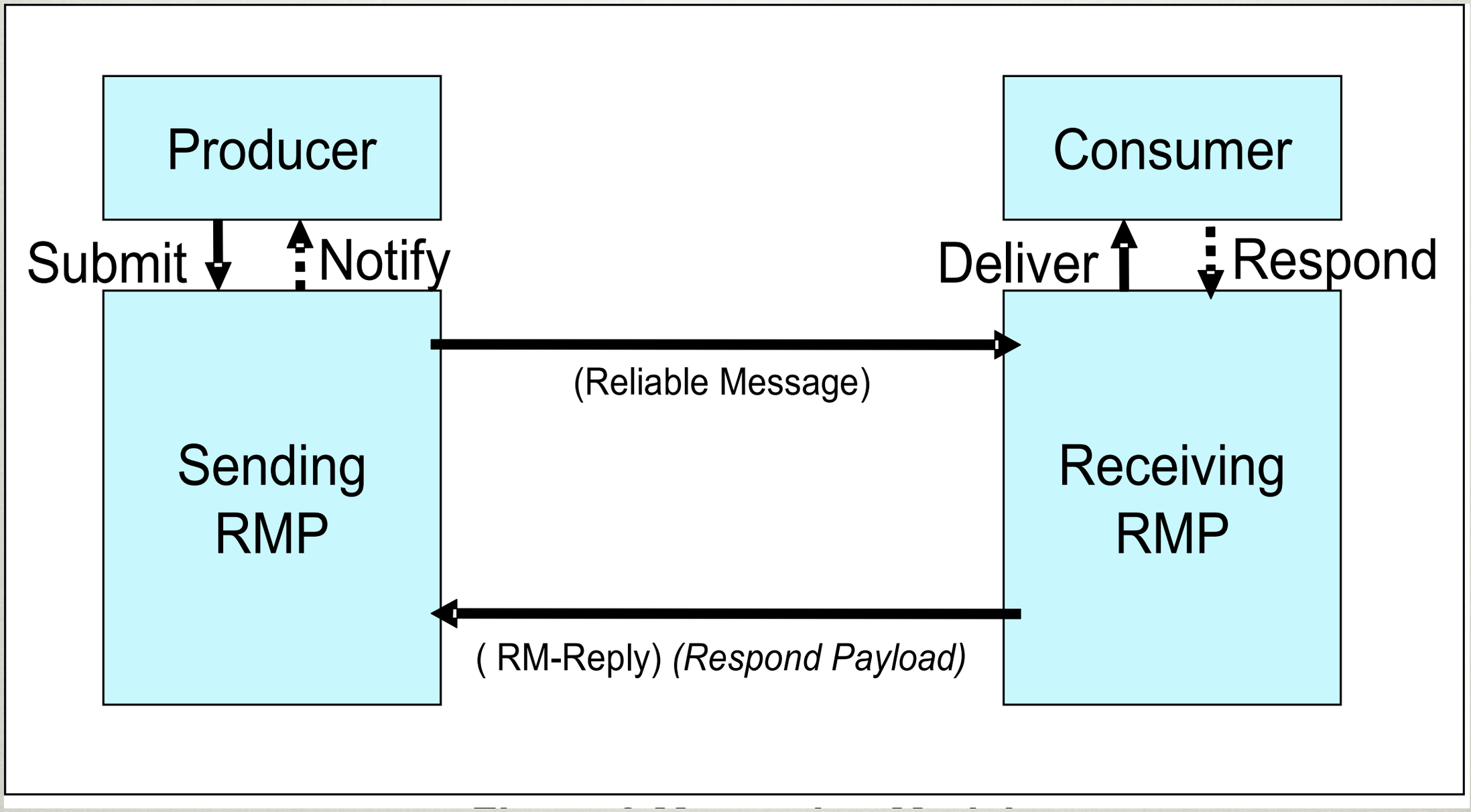
61

Delivery Assurance

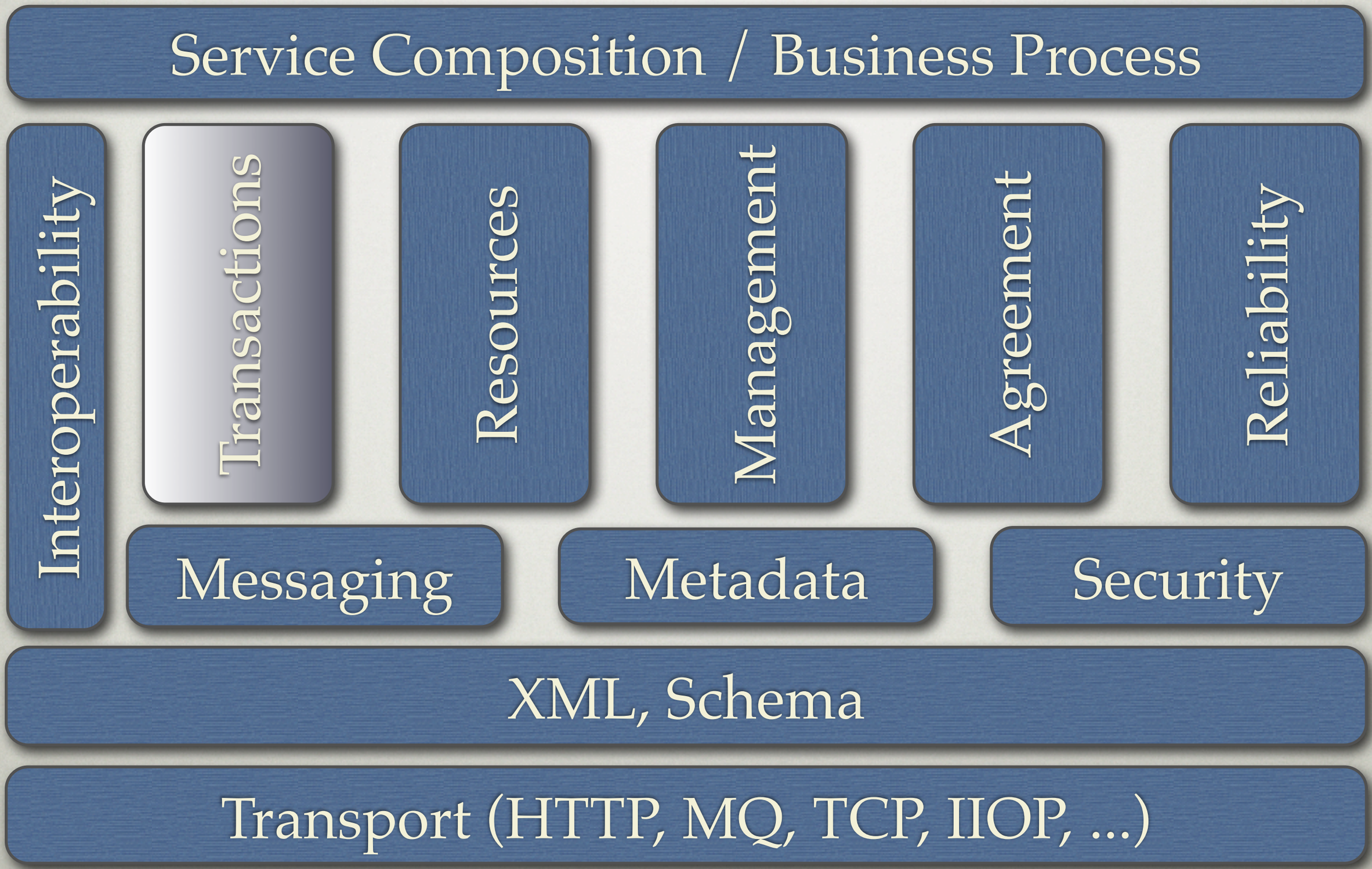
- Messages might be lost, duplicated, reordered, or in an unknown state
- Well-known concepts, realized with SOAP-based message protocol
  - AtMostOnce delivery
  - AtLeastOnce delivery
  - ExactlyOnce delivery
  - InOrder delivery

62

WS-Reliab* Concept



Web Service Specification Landscape



Transactions

- Define mechanisms for transactional interoperability in WS applications
- Huge set of specifications
  - WS-BusinessActivity, WS-Coordination, WS-AtomicTransaction, WS-Transactions, WS-Context, WS-CompositeApplicationFramework, WS-CoordinationFramework, WS-TransactionManagement, ...

65

WS-Coordination

- Framework for the coordination of a distributed application
- When application needs to reach consistent agreement between parts
- Integrates existing legacy transaction processing or workflow systems
- Definition of a *context*
  - Context propagation between services

C	IBM, MS, Bea, Iona, ...
S	Industry proposal
I	ETTK, WCF, ...

66

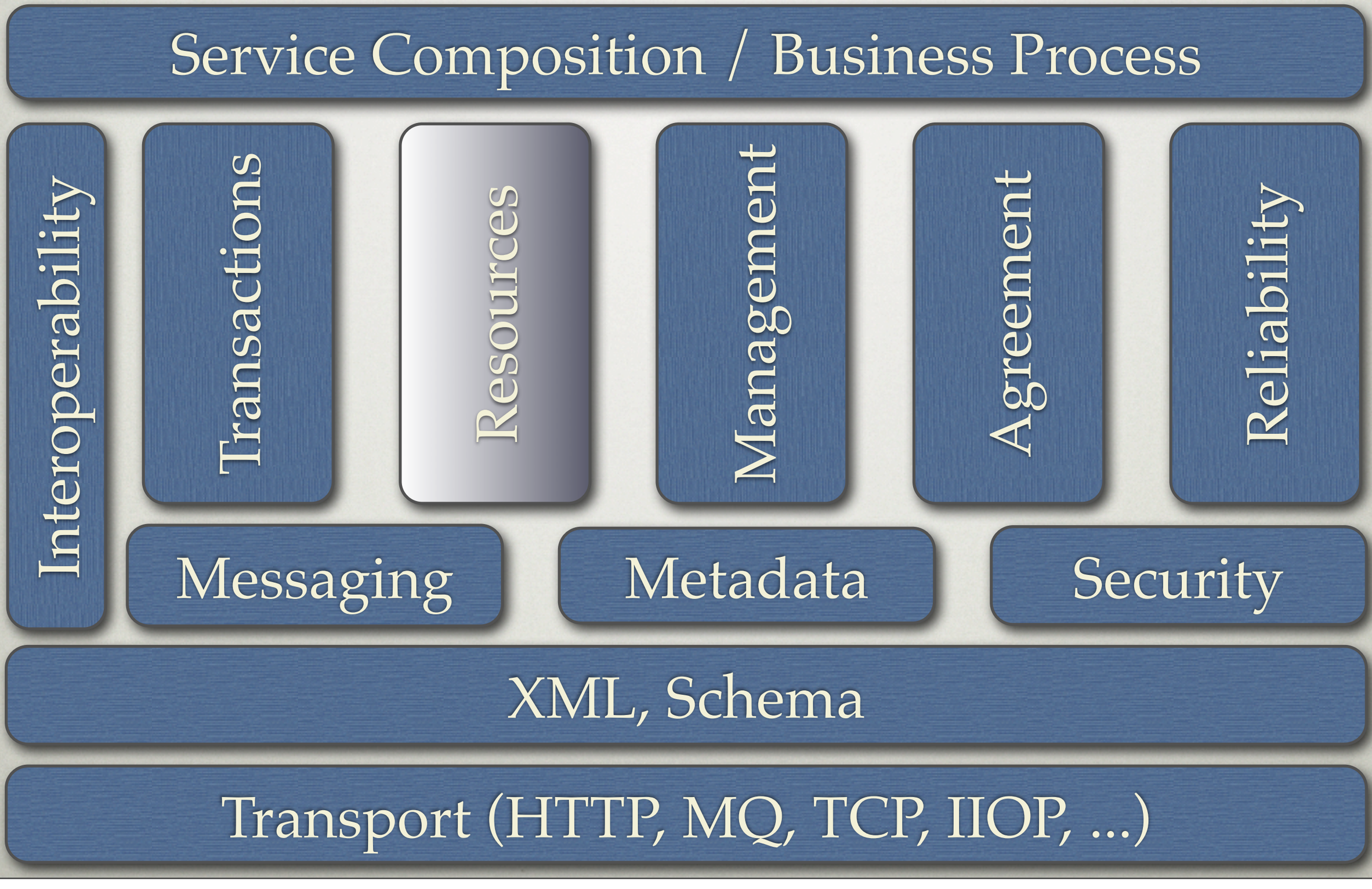
WS-AtomicTransaction

- Based on WS-Coordination
- For short-lived, all-or-nothing activities
- Defines three specific agreement protocols
  - Volatile two-phase commit
  - Durable two-phase commit
  - Completion

C	IBM, MS, Bea, Iona, ...
S	Industry proposal
I	ETTK, WCF, ...

67

Web Service Specification Landscape



Resources

- Web Services Resource Framework (WSRF)
  - WS-BaseFaults
  - WS-ServiceGroup
  - WS-ResourceProperties
  - WS-ResourceLifetime
- WS-Transfer

C	IBM, Oracle, HP, CA, ...
S	OASIS standard
I	Apache, ETTK, Globus, ...

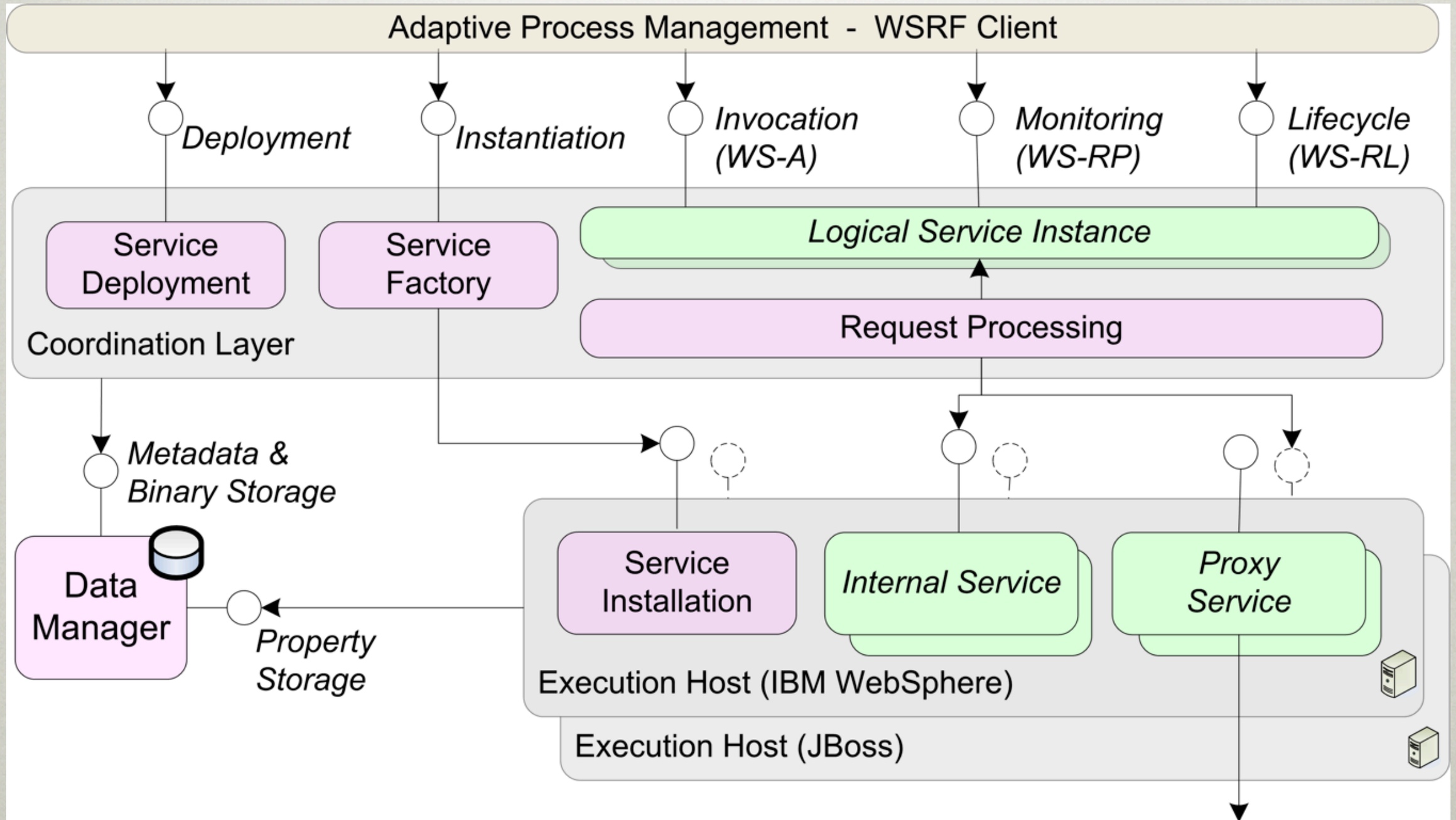
69

Web Services Resource Framework (WSRF)

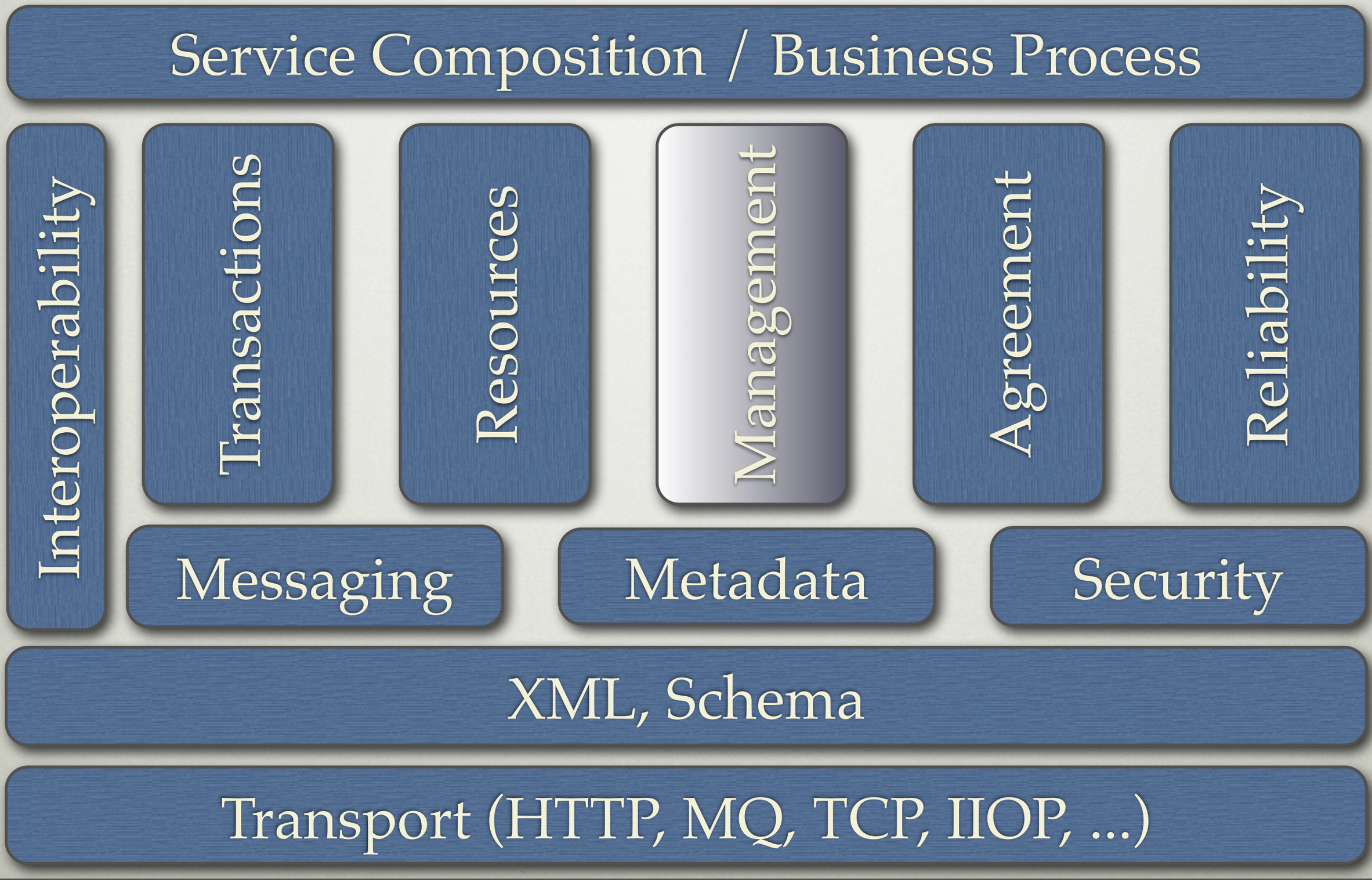
- Covers representation of stateful resources as Web service ('WS-Resource')
- Updated version of GGF OGSI approach
  - WS-ResourceProperties: Querying and changing state information
  - WS-ResourceLifetime: Free WS-Resources
  - WS-BaseFaults, WS-ServiceGroup, WS-Notification

70

ASG Services Infrastructure



Web Service Specification Landscape



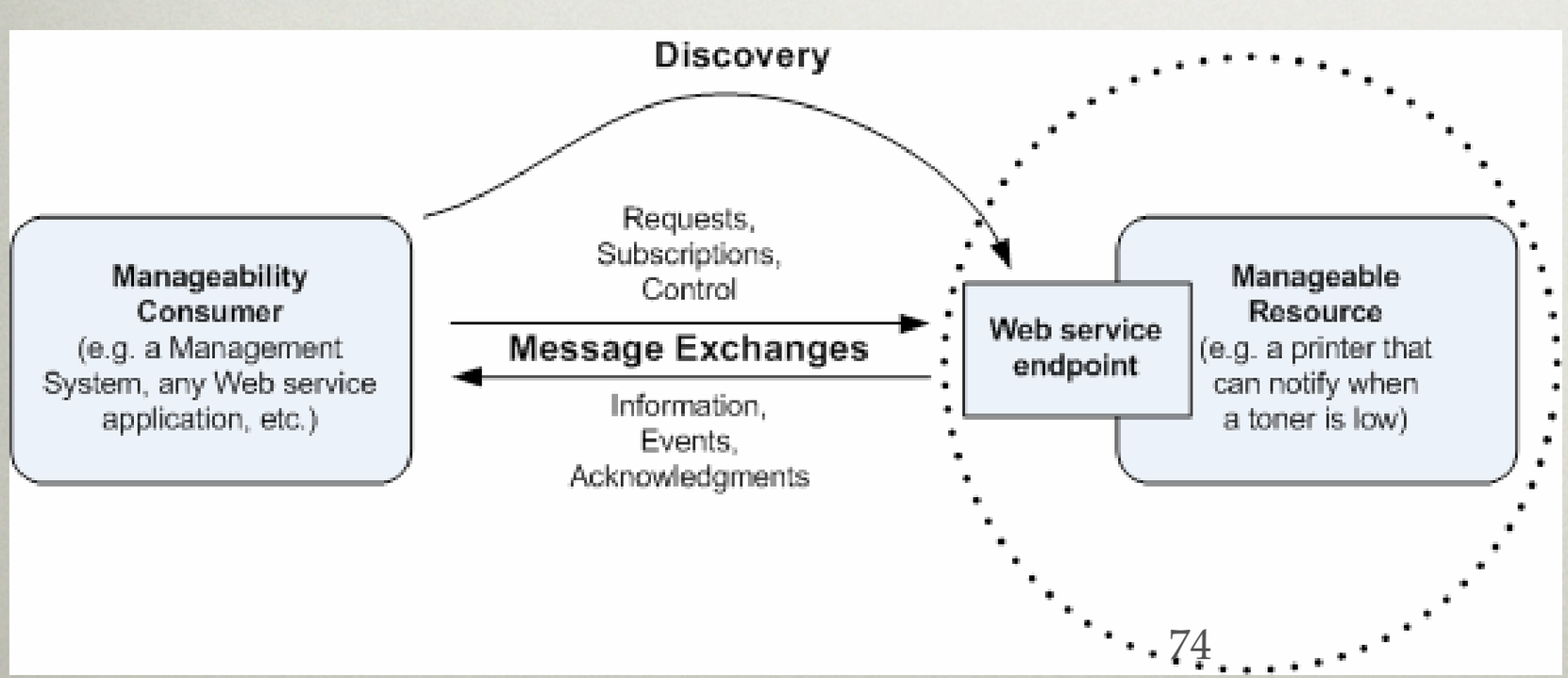
Management

- Management using Web Services (MUWS)
- Management of Web Services (MOWS)
- Web Services Management Framework (WSMF) - HP
- WS-Management
- WS-Events

73

WSDM-MUWS

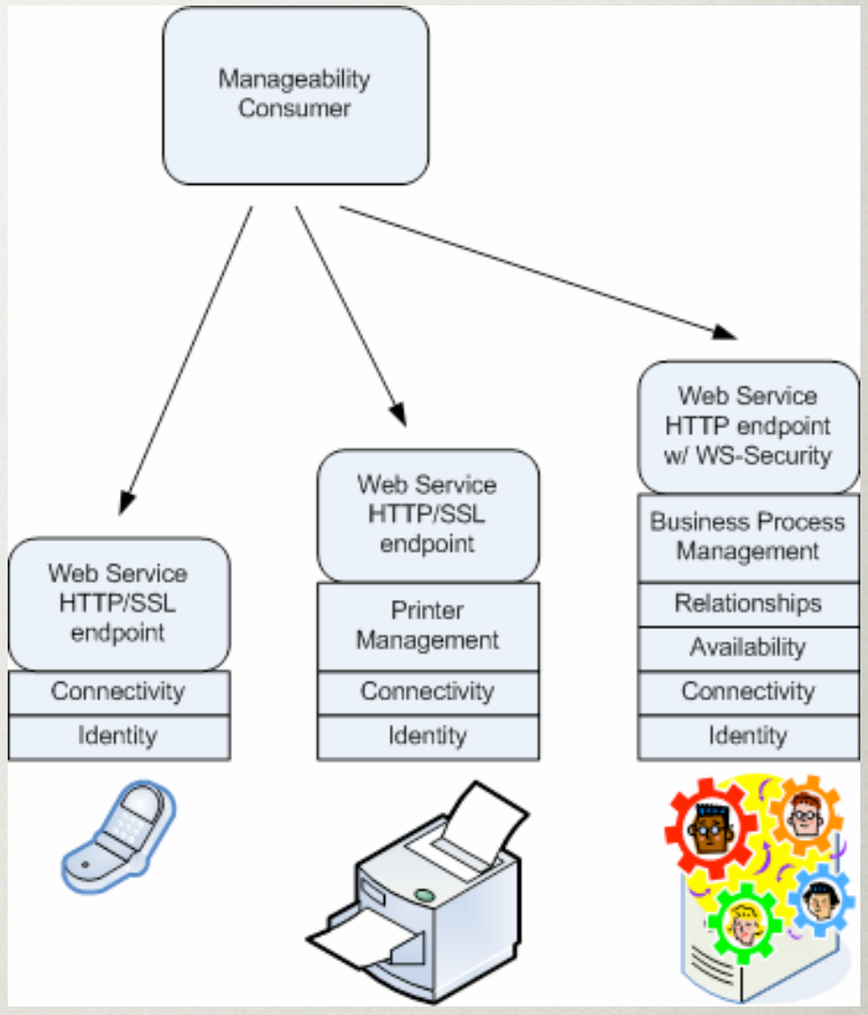
- WSDM - Web Service Distributed Management
- Concept of manageability capabilities
- Management (of IT resources) using Web services



C	HP
S	OASIS Standard
I	Axis

Manageability Capabilities

- Uniquely defined, with specific semantics
- Extensible (DMTF, CIM)
- Composable
- Some predefined values (like identity)



75

WSDM-MOWS

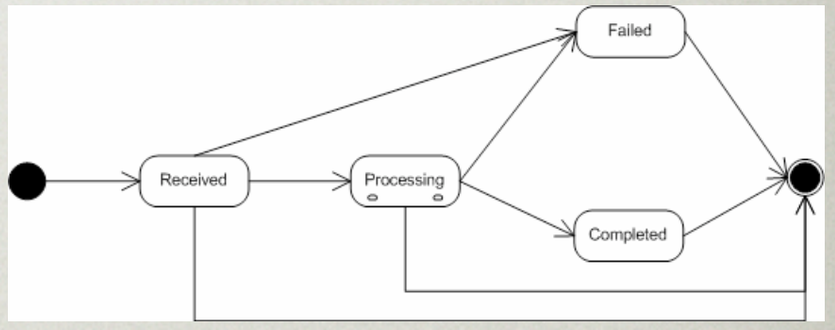
- Management of Web service endpoints
- Through Web services - relies on the MUWS concepts
- Management operation separately, or as part of the functional endpoint
- Includes support for notification

C	CA
S	OASIS Standard
I	Axis

76

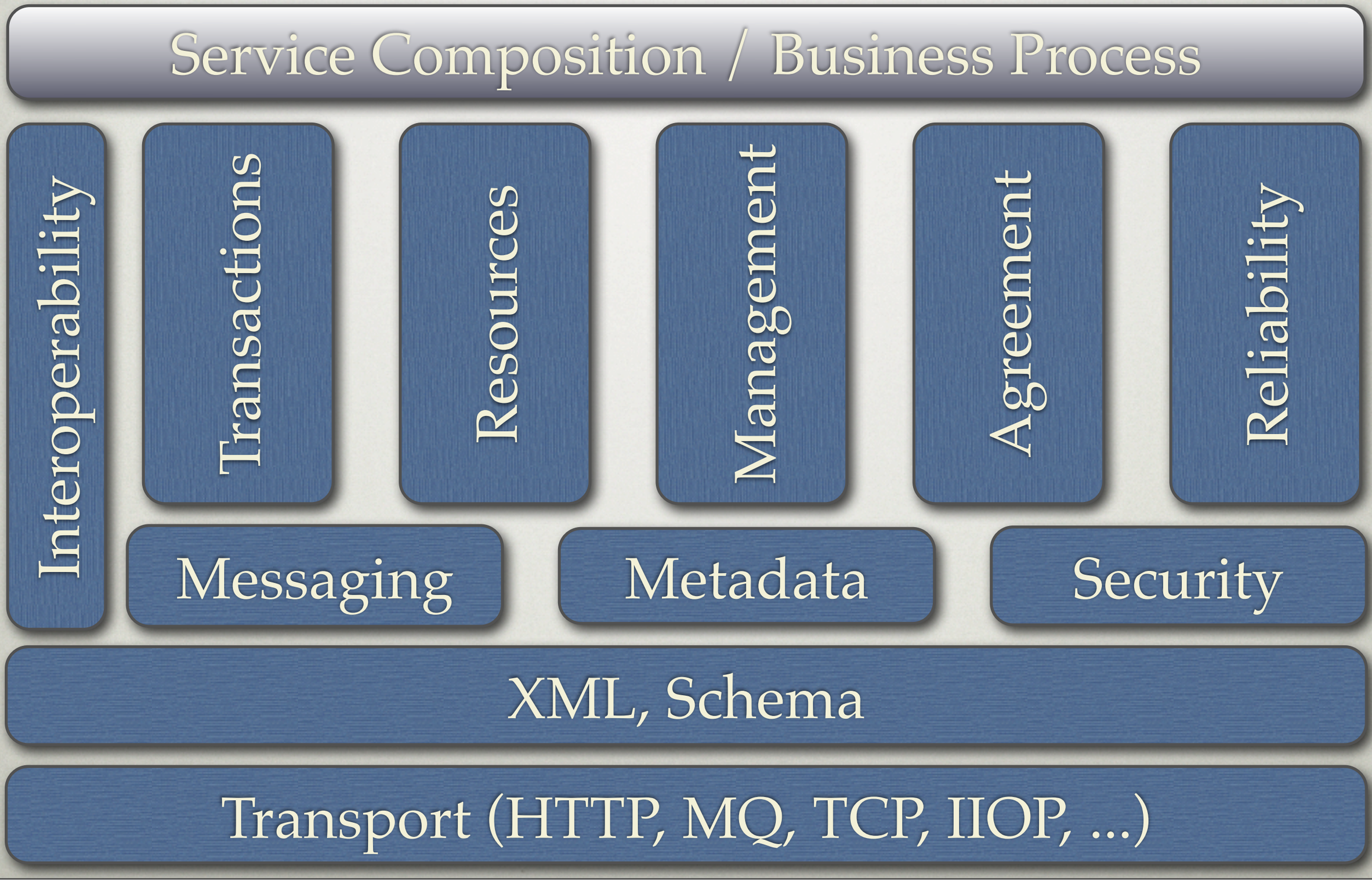
Manageability Capabilities

- Obtaining management interface
- Identity of managed resource
- Endpoint metrics
- Number of requests, failed requests and successful requests
- Max / Last response time
- Operational state (WSLC)
- Service time



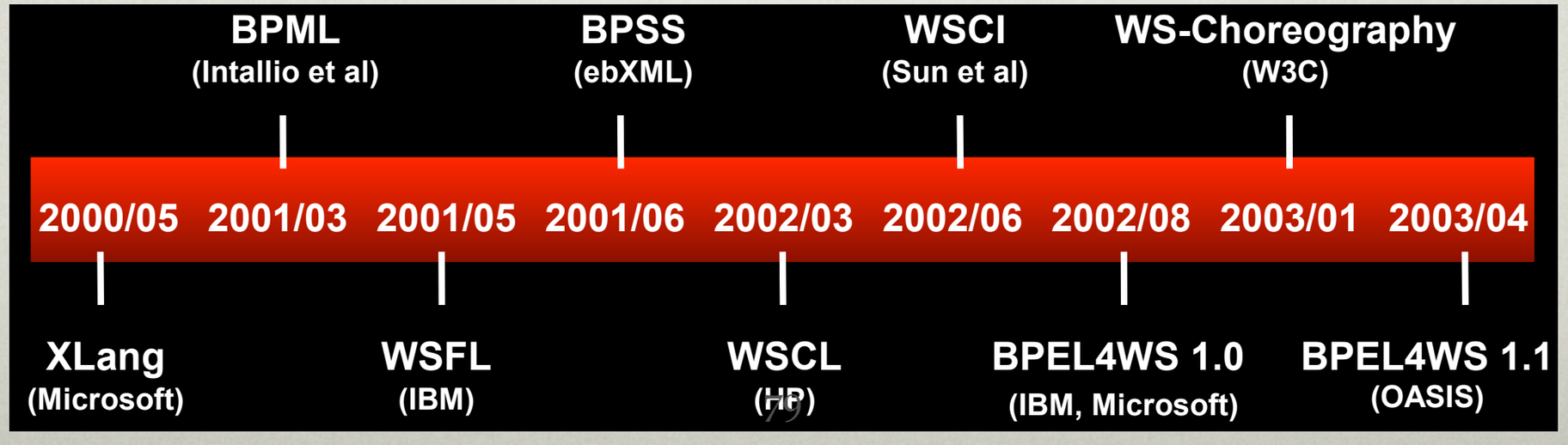
77

Web Service Specification Landscape



Business Processes

- Web Services Business Process Execution Language (BPEL4WS / WS-BPEL) 1.1
  - Describes flow from single endpoint
- WS-Choreography 1.0
  - Rules of interaction with multiple business process endpoints



BPEL4WS

- XML language for describing business processes
- Describes conversation of WSDL message exchanges - Web service composition language
- Process offers coarse-grained Web service for aggregated fine-grained Web services
- Each step as activity
- Partner links (services used by business process)
- Variables for long-running interactions
- Control flow, correlation, scopes

C	BEA, IBM, Microsoft
S	OASIS Standard
I	Axis

80

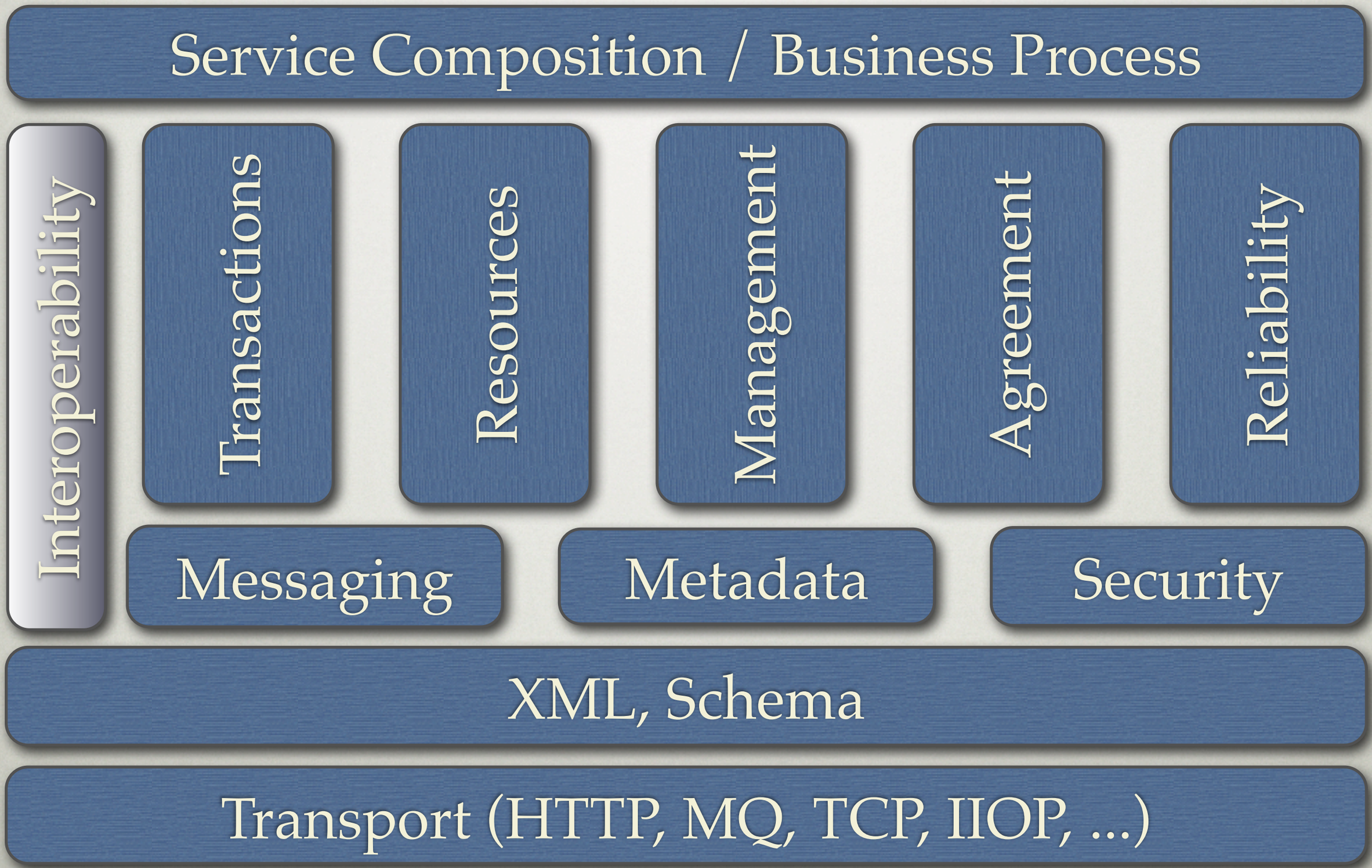
WS-Choreography

- Define sequence and conditions for SOAP message exchange
- Information model for Choreography Definition Language

	Abstract	Portable	Concrete
Types of Messages	Identified	Identified	Identified
Message Structure	Not Defined	Defined	Defined
Conditions	Identified	Identified	Identified
Condition evaluation rules	Not defined	Defined as far as possible	Defined as far as possible
Technology used	Not defined	Defined	Defined
Message Endpoint Data	Not defined	Not Defined	Defined

81

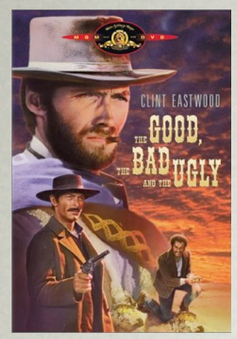
Web Service Specification Landscape



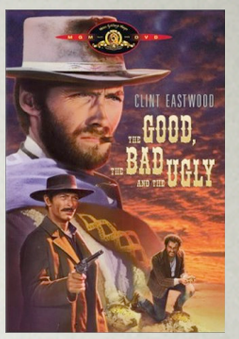
Interoperability

- WS-I profiles
  - Basic Profile Working Group
  - Basic Security Profile Working Group
    - Requirements gathering, sample scenarios, testing tools, XML schema issues
- Clarifications: Missing details, interop problems, attachments, SOAP binding, security token, ...
- Sun: “Shadow government for standards”

83



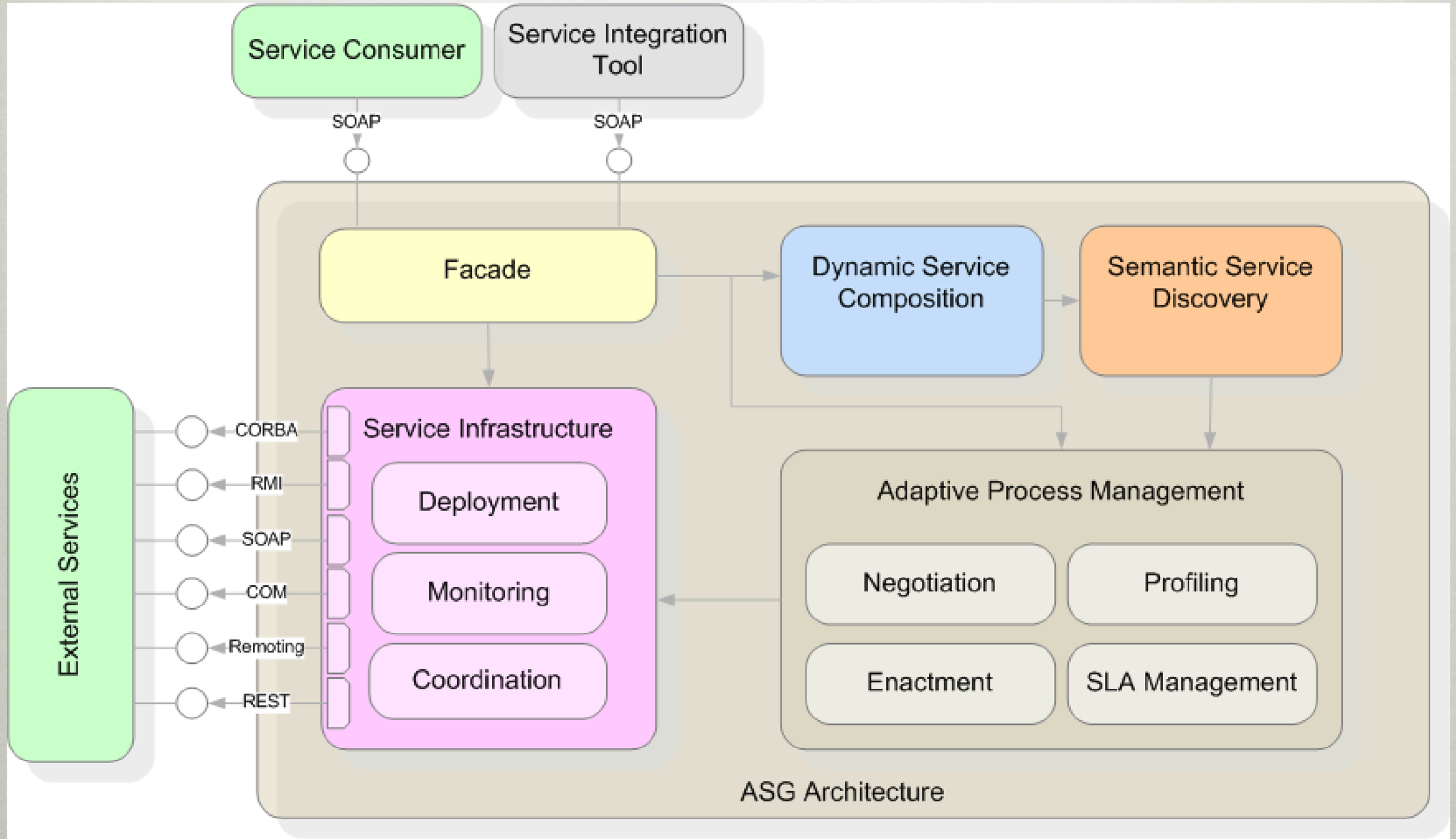
The Good, The Bad, And The Ugly



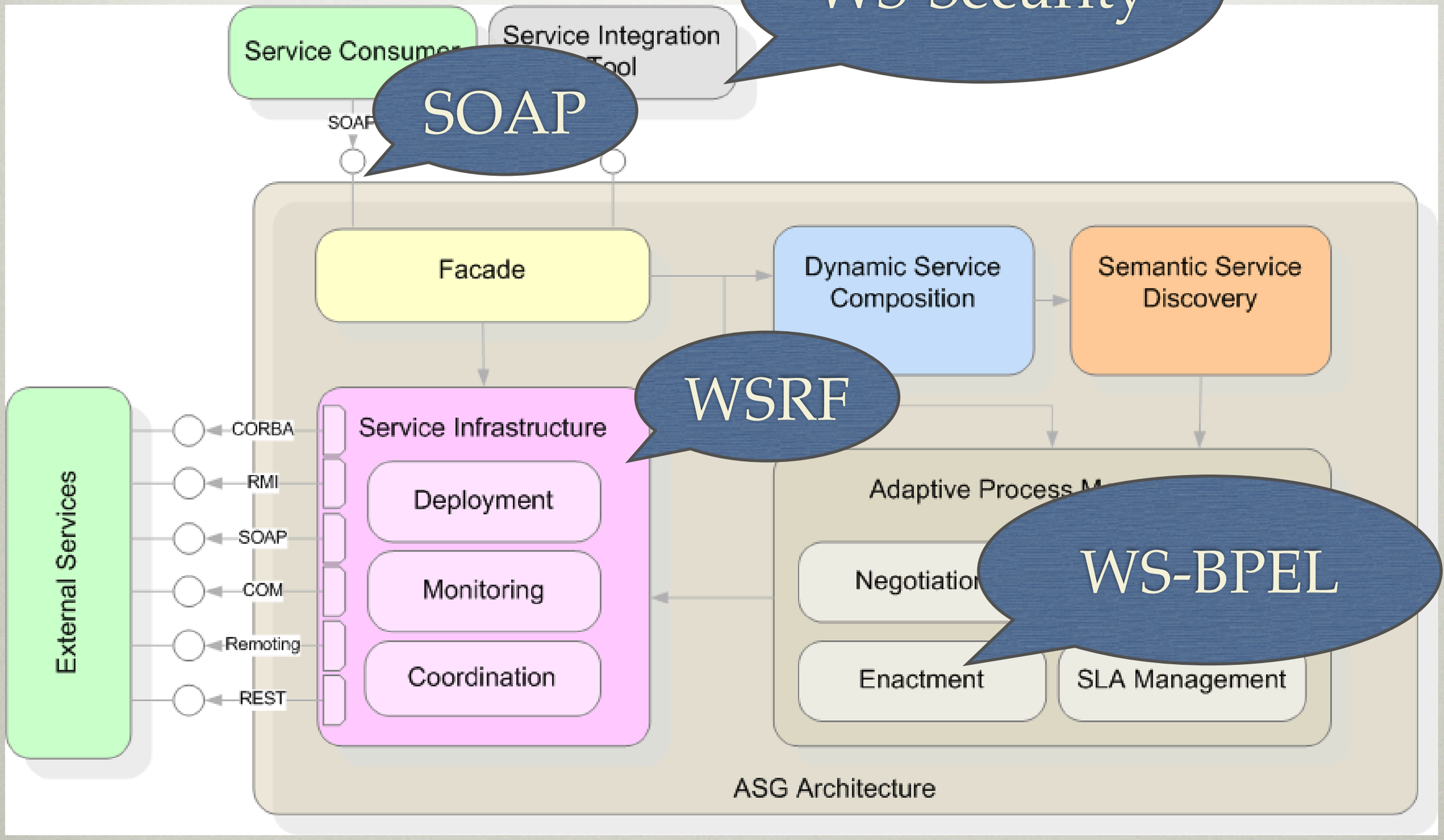
- The Good
  - IBM + MS + somebody else
  - W3C recommendation / OASIS standard, maybe also covered by WS-I
- The Bad
  - Superseded specifications
  - Specs without participation from the big vendors
- The Ugly
  - Proposed standards or company proposals

84

WS Standards In ASG



WS Standards in ASG



Conclusions & Remarks

- Web services are no promise for interoperability
- Web services are no promise for service orientation
- Web services are maybe a ‘bad version’ of CORBA (researchers still fighting)
- Web services are (no longer) bound to the Web
  - But: Google Maps is named a Web service
  - So please: Start talking protocol names (SOAP, HTTP)

86

Conclusions & Remarks

- Don’t fear to read the standard
- Respect standard compliance
- The process is your friend
- Interoperability ALWAYS requires YOUR participation and testing

87

Conclusions & Remarks

- Don’t fear to read the standard
- Respect standard compliance
- The process is your friend
- Interoperability ALWAYS requires YOUR participation and testing

87

Some Sources

- soaprpc.com
- A. Gokhale et. al. Reinventing the Wheel? CORBA vs. Web Services. www202.org
- Scott Mitchell. An Extensive Examination of Web Services: Part 1. 4guysfromrolla.com
- Jordi Albornoz. Finding your way through Web service standards, Part 1. IBM DeveloperWorks
- David Booth et.al. Web Services Architecture. W3C Web Services Architecture Working Group Note

88