

OTA Architecture Review

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

Background

- OTA Architecture subCommittee
- Chair → David Morley (Marriott)
- Support Staff → Greg Wilson (OTA)
- From the 2005 Advisory Forum...
 - “Implementation support needed !”
 - Several Arch project teams developed
 - OTA Implementers Forum created

OTA
The OpenTravel™ Alliance

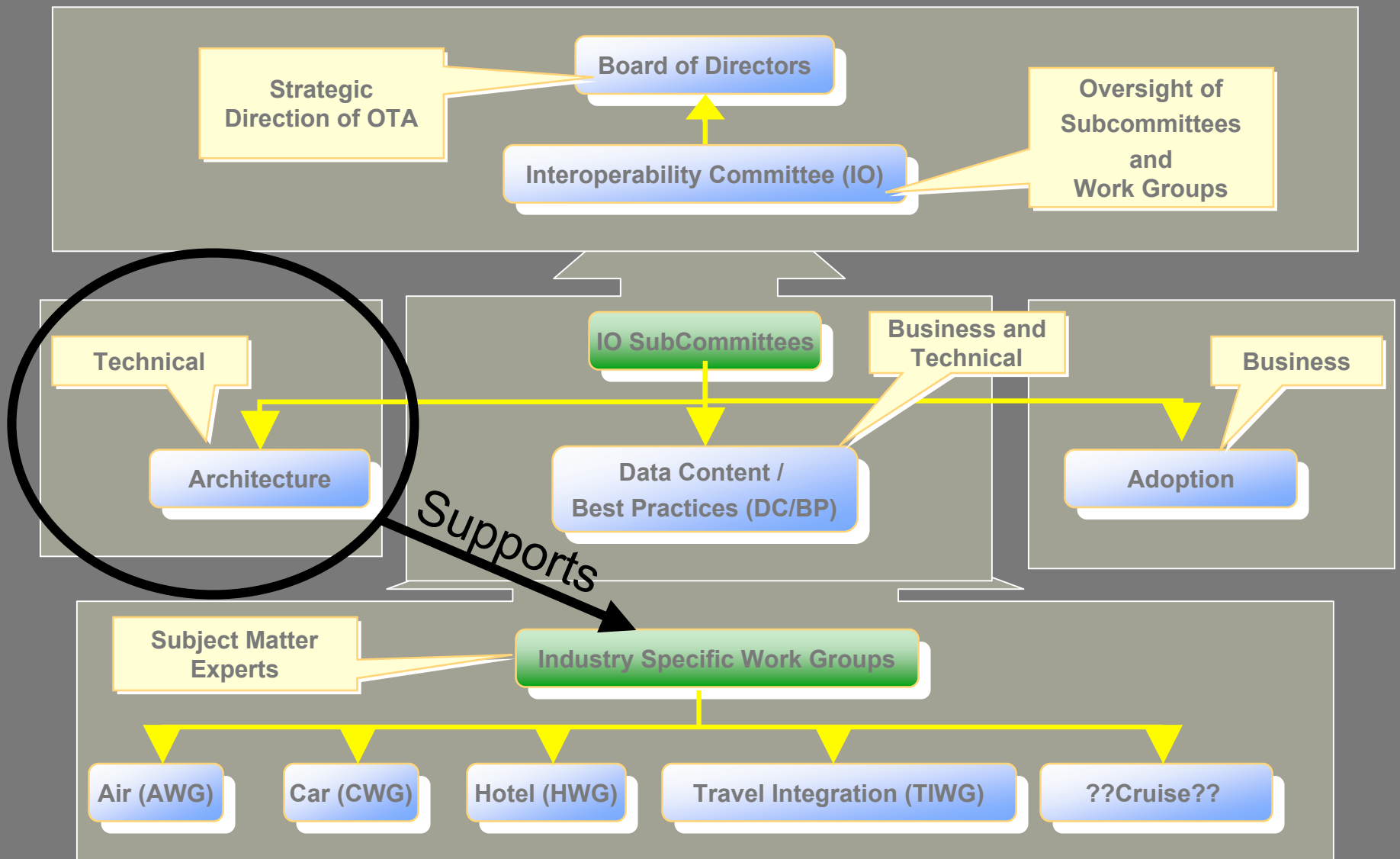
**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating **THE**
Total Travel Solution



Background

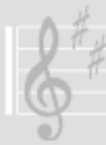


Project Overview

- HTTP Transport Protocol Reference
 - 2005B
 - Stephen Adkins (The Rubicon Group)
- SOAP Transport Protocol Reference
 - 2006A
 - John Lambe (OpenJaw Technologies)

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating **THE**
Total Travel Solution

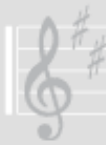
type="SailingCategoryInfoType"><xs:annotation base="SailingCategoryInfoType" namespace="http://www.opentravel.org/OTA/2005/OTA/StandardInterface.xsd" type="SailingCategoryInfoType"/></xs:annotation><xs:sequence><xs:element name="VesRentalDesc" type="VesRentalDescType"/></xs:sequence></xs:complexType></xs:element></xs:annotation><xs:element name="MeetingProfile" type="MeetingProfileType"/></xs:element></xs:schema>

Project Overview

- WSDL Implementation Guide
 - 2006A
 - Kevin Camenzuli (Cendant Car Rental Group)
- Asynchronous Exchange
 - 2006B
 - Alain Leveillé (Expedia)

OTA
The OpenTravel™ Alliance

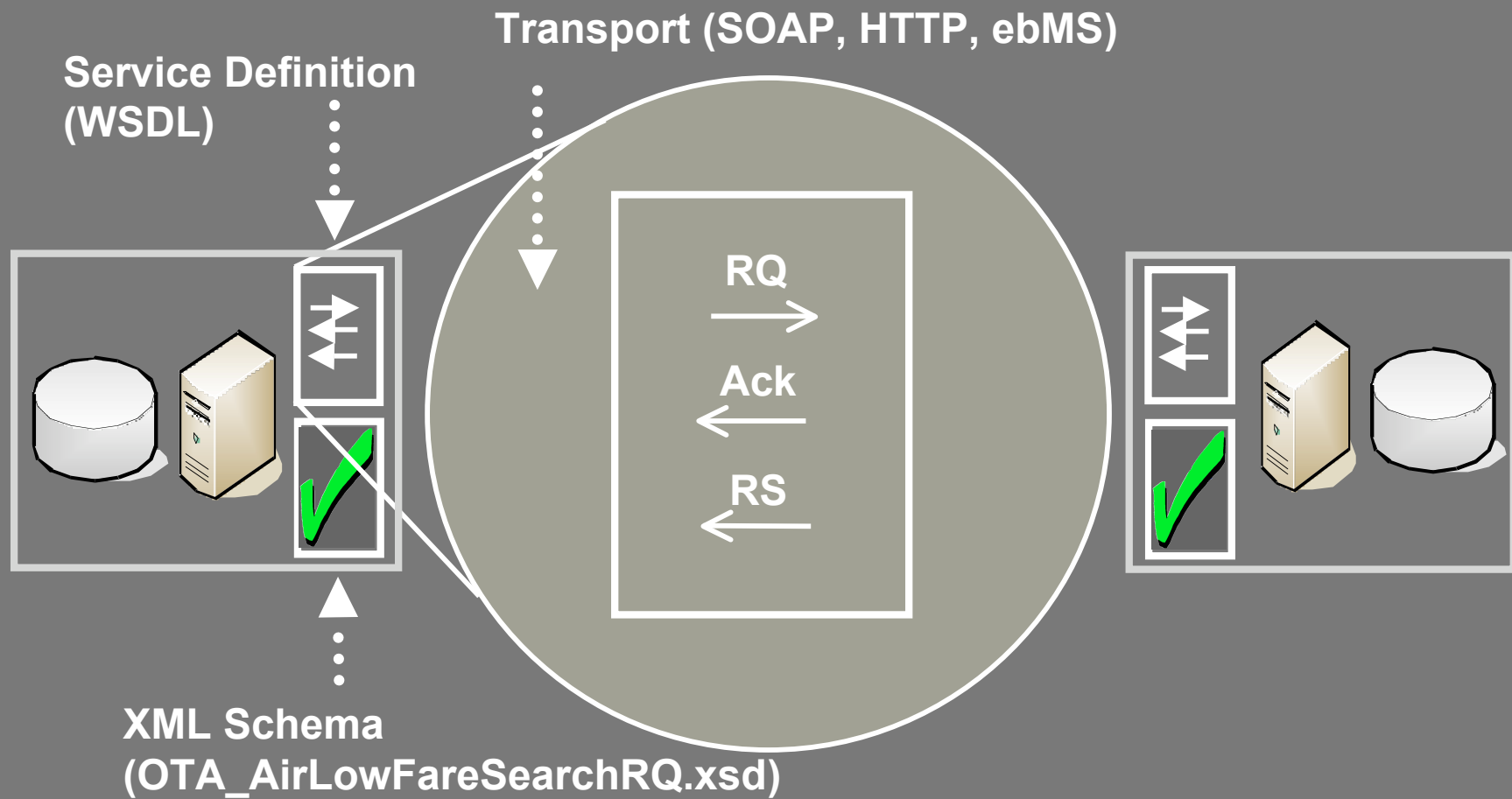
**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution



A Simple Web Service



OpenTravel Alliance - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites History Mail Print Edit

Address <http://www.opentravel.org/> Go

Google Search 111 blocked Check AutoLink AutoFill

OTA

The OpenTravel™ Alliance

Home Members Only Site Map Contact Us

About Specific Technical Resources Documents Archives Updates Governance

OTA Implementers Forum

or Public Review

[Comment on current OTA Specification](#)

Register Now for the 2006 Advisory Forum

2006 OTA ADVISORY FORUM

THE NEXT STEPS —
Orchestrating THE
Total Travel Solution

APRIL 25-26 • ORANGE COUNTY, CALIFORNIA

Attend this annual two-day conference to hear compelling speakers, discover current workgroup projects, and help determine the next twelve month's work. Members and non-members are welcome to attend.

<http://groups.google.com/group/OTA-Impl-Forum> Internet

Architecture Review

OTA Transport Protocol Reference: HTTP

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

type="SailingCategoryInfoType"><xs:annotation
base="OpenTravel:BaseType" namespace="http://www.opentravel.org/OTA/2002/07/01/XMLSchema.xsd"/><xs:sequence><xs:element name="VesRentalDesc" type="VesRentalDescType"/></xs:sequence></xs:complexType></xs:element></xs:annotation><xs:element name="MeetingProfile" type="MeetingProfileType"/></xs:schema>

A Vision of Interoperability

- Write Once. Connect Everywhere.
- When two travel companies decide to do business together, there should be no technical barriers to them connecting their systems to do so.
- Companies developing OTA-conformant systems should have sufficient guidance to build systems which are highly interoperable with systems built by other companies.

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

<xs:sequence><xs:element name="VendorInfo" type="VendorInfoType"/></xs:sequence></xs:complexType></xs:element>
</xs:annotation><xs:element name="MeetingProfile" type="MeetingProfileType"/></xs:element>

History and Background

- The Travel Industry and the OTA in 2001
- 2001C Specification: ebXML

OTA Design Goals: 2001

- Openness
- Flexibility
- Platform Independence
- Security
- Extensibility
- International Scope

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating **THE**
Total Travel Solution

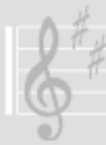


Additional Design Goals: 2005

- Interoperability
- Simplicity
- Performance
- De-Facto Acceptance

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

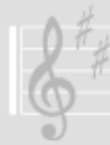


Requirements for Interoperability

- Common Use Cases and Scenarios (i.e. Usage Profiles) (TBD)
- Transport Protocol Reference (Done!)
- Software Validation Suite (TBD)

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

type="SailingCategoryInfoType"><xs:annotation base="SailingCategoryInfoType" namespace="http://www.opentravel.org/OTA/2008/01/01/XMLSchema.xsd" type="SailingCategoryInfoType"/></xs:annotation><xs:sequence><xs:element name="VesRentalDesc" type="VesRentalDescType"/></xs:sequence></xs:complexType></xs:element></xs:annotation><xs:element name="MeetingProfile" type="MeetingProfileType"/></xs:element></xs:schema>

Why HTTP?

- Simple to get Right, Difficult to get Wrong (just Headers + Content)
- Simple enough to implement directly yourself
- Implements Request/Response of Text Messages (just what OTA needs)
- Provides Authentication and Encryption
- Can Test in a Web Browser
- Companies are using it ! (as registered with OTA)

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating **THE**
Total Travel Solution

type="SailingCategoryInfoType"><xs:annotation base="SailingCategoryInfoType" namespace="http://www.opentravel.org/OTA/2002/11/OTA.xsd" type="SailingCategoryInfoType"/></xs:annotation><xs:sequence><xs:element name="VesRentalDesc" type="VesRentalDescType"/></xs:sequence></xs:complexType></xs:element></xs:annotation><xs:element name="MeetingProfile" type="MeetingProfileType"/></xs:element>

Transport Protocol Reference: HTTP

- Plain HTTP POST
 - optionally with Basic Authentication and SSL/HTTPS Encryption
- Content-Length Header mandatory
 - Facilitates ease of implementation
- The Content in HTTP Request and Response **are** the OTA messages
- This is the way web servers, web browsers, and HTTP support libraries already work!

What's Next?

- 2006A Transport Protocol Reference: SOAP
- 2006B: Interoperability Testing
 - Choose some common use cases and scenarios to test
 - Develop an initial test suite as a Free/Open Source Software project
 - Self-Testing: Testers test themselves with the Test Suite
 - Member-to-Member Pairwise Testing: Testers try to connect to each other

Architecture Review

OTA Transport Protocol Reference: SOAP

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

type="SailingCategoryInfoType"><xs:annotation
base="OpenTravel:BaseType" namespace="OpenTravel:Types"><xs:sequence><xs:element name="VesRentalDesc
type="VesRentalDescType"/></xs:sequence></xs:complexType></xs:element>
</xs:annotation><xs:element name="MeetingProfile" type="MeetingProfileType"><xs:annotation

What is SOAP?

- **Simple Object Access Protocol**
- Specification Work started in 1998
- Submitted to **W3C** (Specification Body) in 2000
- Initially intended as a replacement for existing RPC protocols
- Also developed into a document exchange transport mechanism
- SOAP provides a simple and extensible vehicle for interchanging data and invoking remote services using XML
- **WS-I** is an open industry organization chartered to promote Web services interoperability across platforms

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating **THE**
Total Travel Solution

`<xs:sequence>
<xs:element name="SailingCategoryInfoType" type="SailingCategoryInfoType"/>
<xs:element name="VesRentalInfoType" type="VesRentalInfoType"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:annotation>
<xs:element name="MeetingProfile" type="MeetingProfileType"/>
</xs:element>
</xs:schema>`

Need for Interoperability

- SOAP is already being widely used for transporting OTA documents.
- However, due to the dual purpose of SOAP (RPC vs. messaging) and SOAP's flexibility with regards to structure, a whole range of SOAP structures are being used to transport OTA-compliant data.

SOAP Structure

```
<?xml version="1.0" encoding="UTF-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Header>
    <!-- Routing, security or other control data. -->
  </soap:Header>
  <soap:Body>
    <!-- RPC method call or document data. -->
  </soap:Body>
</soap:Envelope>
```

- SOAP Body element
 - must be a valid XML document:
 - a remote service/method
 - root element of an XML document
- SOAP Header element
 - Optional
 - Used to carry information apart from the actual envelope payload
 - E.g. routing or security information

SOAP Versions

- SOAP version 1.1 was released as a W3C “Note” in 2000
- SOAP version 1.2 was released as a W3C “Recommendation” in June 2003
- Main changes:
 - change in the namespace
 - change in how the SOAP Action value was transmitted over HTTP.
- Most SOAP toolkits/stacks support both versions
- OTA SOAP implementations **MUST** support either SOAP 1.1 *or* SOAP 1.2 ✓
- OTA SOAP implementations **SHOULD** support SOAP 1.1 *and* SOAP 1.2 for clients and services ✓

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution




SOAP Messaging

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <OTA_CancelRQ xmlns="http://www.opentravel.org/OTA/2003/05" Version="2.001">
      <POS>
        <Source ISOCountry="US" ISOCurrency="NOK" PseudoCityCode="HUR">
          <RequestorID ID="abc.123" URL="abcde..." />
        </Source>
      </POS>
      <UniqueID ID="DGNJ6012990-389" Type="14" />
    </OTA_CancelRQ>
  </soap:Body>
</soap:Envelope>
```

- OTA payload is the only and immediate child of the SOAP Body element
- This is the simplest and most efficient means of transporting OTA messages over SOAP
- Recommended by OTA ✓

SOAP RPC

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <acme:otaServiceCancel xmlns:acme="http://www.acme-travel.com">
      <OTA_CancelRQ xmlns="http://www.opentravel.org/OTA/2003/05" Version="2.001">
        <POS>
          <Source ISOCountry="US" ISOCurrency="NOK" PseudoCityCode="HUR">
            <RequestorID ID="abc.123" URL="abcde..." />
          </Source>
        </POS>
        <UniqueID ID="DGNJ6012990-389" Type="14" />
      </OTA_CancelRQ>
    </acme:otaServiceCancel>
  </soap:Body>
</soap:Envelope>
```

- SOAP Body element is an XML element that describes the method or function that is being invoked, in this case “otaServiceCancel”
- namespace of this element is most often defined by the service application (it is not the OTA namespace)
- Not recommended by OTA 

SOAP Action URI

- Transmitted with a SOAP Envelope, apart from the envelope's XML content
- Carries the *intent* of the SOAP message
- Valuable to intermediaries:
 - SOAP routers, gateways, proxies etc.
- Services **SHOULD NOT** require a SOAP Action URI ❌
- Services and intermediaries that require a SOAP Action URI, **SHOULD** use the OTA request tag root as the SOAP Action URI. ✅
- Clients **SHOULD** support any SOAP Action URI for transmitting any OTA message to a Web service. ✅

SOAP Message Structure

- SOAP Envelope, Header and Body **SHOULD** conform to the SOAP 1.1 and SOAP 1.2 specifications, as clarified in WS-I basic profile section 4.1 ✓
- The SOAP Header **SHOULD NOT** contain OTA data ✗
- All content within the SOAP Body element **SHOULD** be in the OTA namespace ✓
 - Except XML-Encryption tokens
- Content within a SOAP Body element **SHOULD** be valid, well-formed XML that conforms to an OTA schema ✓
- only immediate child element of the SOAP Body element **SHOULD** be the root element of a document that is defined in an OTA schema ✓

SOAP Attachments

- SOAP clients **SHOULD** support SOAP Attachments. ✓
- SOAP services **SHOULD** limit the use of SOAP Attachments to images, such as hotel and vehicle images. ✓

Error Handling

- SOAP Fault element is a vehicle for transporting error information from a SOAP service to a SOAP client.
 - intended to return application-level errors as well as errors that originate in the SOAP stack
- OTA also defines elements for returning errors and warnings to a SOAP client.
 - intended to carry application level errors and warnings only.
- OTA SOAP services **SHOULD** use SOAP Fault for SOAP-level errors ✓
- OTA SOAP services **SHOULD** use OTA Errors for application-level errors. ✓
- OTA SOAP clients **SHOULD** support both SOAP Fault and OTA Error handling. ✓

Future

- Ongoing updates to further clarify usage
- Security recommendations?
 - Based on WS-I?
- Attachments ?
 - Further definition and samples

Architecture Review

Implementation Guide: WSDL

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

What is WSDL?

- WSDL 1.1 submitted as a W3C Note in 2001
- The Web Service Description Language (WSDL) defines a service interface or service contract
 - WSDL defines the message format and transport protocol
 - WSDL is a contract defining “*How*” two parties intend to communicate with one another
- Similarly, XML Schema defines an XML message format or data contract
 - OTA Schemas define message payloads and data format only
 - Schema is a data contract defining “*What*” information will be communicated between parties
- Today the OTA defines the “What” not the “How”

Need for Interoperability

- Web Services standards are flexible, which is both a strength and a weakness
- Code First vs. Contract First approach
 - Code First
 - Reliance on tools to generate WSDL
 - Obstacle to interoperability
 - Shields developers from learning proper WSDL creation technique
 - Contract First
 - Results in highly interoperable services
 - Implementers need guidance to understand how to describe OTA based services interfaces via WSDL
 - Document style services are not easily described with WSDL
 - OTA is naturally aligned with Contract First design

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

```
<xs:sequence>  
  <xs:element name="VesRentalDoc" type="SailingCategoryInfoType"/>  
  </xs:sequence>  
</xs:complexType>  
</xs:element>  
</xs:annotation>  
<xs:element name="MeetingProfile" type="MeetingProfileType"/>  
</xs:sequence>
```



WSDL Implementation Team Goals

- Aid in OTA adoption
 - WSDL simplifies service creation\consumption
- WSDL creation guidance
 - “How to” guide for WSDL creation
- Service design best practices
 - Service interoperability
 - Document style services
 - Contract First development
- Sample WSDL files for OTA based services
- Scope defined by WSDL Publication Feasibility Study in 2005B

WSDL Implementation Guide

- WSDL Best Practices
 - Contract First Modular Design
 - Creation of Schema (OTA defined)
 - Creation of Interface Definition WSDL (possibly OTA defined)
 - References schema
 - Creation of Implementation Binding WSDL (implementation specific)
 - References Interface Definition WSDL
 - The OTA can only provide guidance
- WSDL Reference
 - Section by section breakdown of WSDL file
 - Rules for creating highly interoperable document style WSDL files
- Examples

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating **THE**
Total Travel Solution

`<xs:sequence>
<xs:element name="SailingCategoryInfoType" type="SailingCategoryInfoType"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:annotation>
<xs:element name="MeetingProfile" type="MeetingProfileType"/>
</xs:element>`

OTA Schema Reference

```
<wsdl:types>
  <xs:schema>
    <xs:import namespace="http://www.opentravel.org/OTA/2003/05"
      schemaLocation="OTA_VehResRQ.xsd"/>
  </xs:schema>
  <xs:schema>
    <xs:import namespace="http://www.opentravel.org/OTA/2003/05"
      schemaLocation="OTA_VehResRS.xsd"/>
  </xs:schema>
</wsdl:types>
```

- The *wsdl:types* element **MUST** reference the relevant OTA XML Schemas (e.g., request, response, acknowledgement). ✓
- XML Schemas **MUST** be included using the *xs:import* element rather than the *wsdl:import* element (prefixed using the XML Schema namespace rather than the WSDL namespace). ✓
- The *xs:import* element **SHOULD** be referenced from within an *xs:schema* element. ✓
- The *xs:import* element **SHOULD** include a namespace attribute referencing the OTA namespace. ✓
- The *schemaLocation* attribute of the *xs:import* element **MUST** reference the applicable OTA XML Schema by name and **MAY** include the fully qualified file path or URL to the WSDL file. ✓
- If a URL reference is used it **MUST NOT** reference the OTA Online Schemas, which are available for reference only. ✗

Invalid OTA Schema Reference

```
<wsdl:definitions xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"  
  xmlns:ota="http://www.opentravel.org/OTA/2003/05"  
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"  
  xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"  
  xmlns:xs="http://www.w3.org/2001/XMLSchema"  
  targetNamespace="http://www.opentravel.org/OTA/2003/05"  
  name="VehReservationService">
```

```
  <!-- This is a Schema style import not a WSDL style import. -->
```

```
  <xs:import schemaLocation="OTA_VehResRQ.xsd"/>
```

```
  <xs:import schemaLocation="OTA_VehResRS.xsd"/>
```

```
  <!-- Schemas should be imported within the unused Types section below. -->
```

```
</wsdl:types/>
```

- OTA Schemas are referenced via xs:import
- types section of WSDL is unused
- **Not recommended by OTA** ✖

SOAP Binding

```
<wsdl:binding name="VehicleReservationBinding" type="ota:VehicleReservationPortType">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="OTA_VehResRQ">
    <soap:operation soapAction="CreateReservation" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
</wsdl:binding>
```

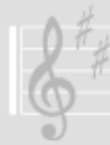
- For SOAP based services, the style attribute of the soap:binding element SHOULD be “document” and not “RPC”. ✓
- For SOAP based services, the use attribute of the soap:body element SHOULD be “literal” and not “encoded”. ✓

What's Next?

- WSDL Implementation Guide published in 2006A
- Expand the WSDL Implementation Guide in 2006B
 - Additional examples
 - SOAP header definition
 - Security definition
- WSDL 2.0?
- WSDL 1.1 Binding Extension for SOAP 1.2?
- Multiple messages (operations) within a single service
- UDDI publication of OTA service definitions?
 - Interface Definition WSDL files published as UDDI T-Models?

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

`<xs:sequence>
<xs:element name="SailingCategoryInfoType" type="SailingCategoryInfoType"/>
<xs:element name="VoyageRentalInfoType" type="VoyageRentalInfoType"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:annotation>
<xs:element name="MeetingProfile" type="MeetingProfileType"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:annotation>
</xs:schema>`

Architecture Review

Support of asynchronous exchange of messages
with the OTA specifications

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution



What's “asynchronous”?

- Processing of a request not triggered only by the reception of that request
- When the processing may take a large and indefinite amount of time.
- GDS Type B messages come to mind



When does it occur?

- Bulk data transfer
- Batch processing
- Different priorities
- Manual processing
- Different system sizes

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution

`<xs:sequence>
<xs:element name="VesRentalDoc" type="SailingCategoryInfoType" />
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:annotation>
<xs:element name="MeetingProfile" type="MeetingProfileType" />
</xs:element>
</xs:schema>`

What's the problem?

- OTA specs based on RQ/RS pairs
- Doesn't adapt well to asynchronous mode, because requestor typically waits for response
- Consumes system resources
- When do you stop waiting?
- Need to provide feedback on request delivery



What did we do about it?

- Study project to identify alternatives on how asynchronous messaging could be accomplished
- Study project means no modification to specs, just recommendations

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



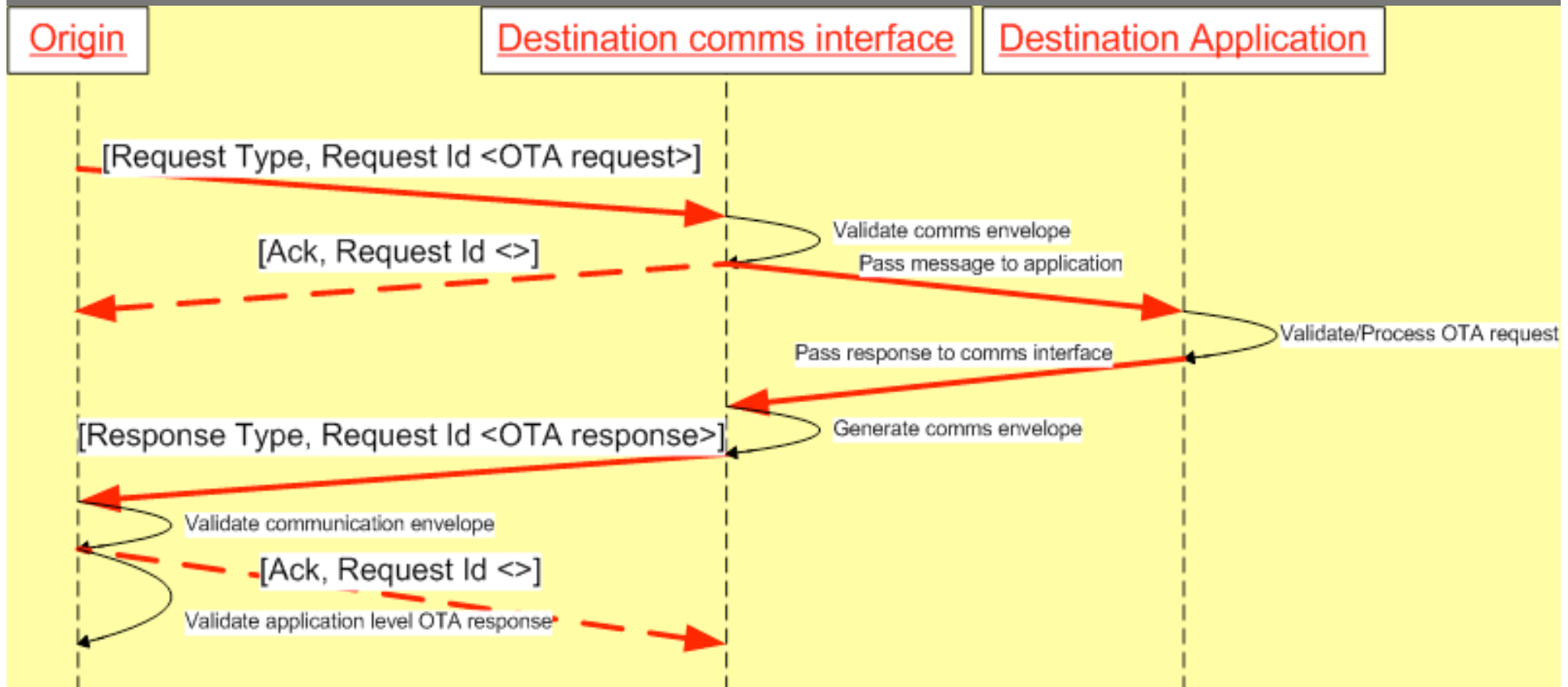
THE NEXT STEPS —
Orchestrating
THE
Total Travel Solution



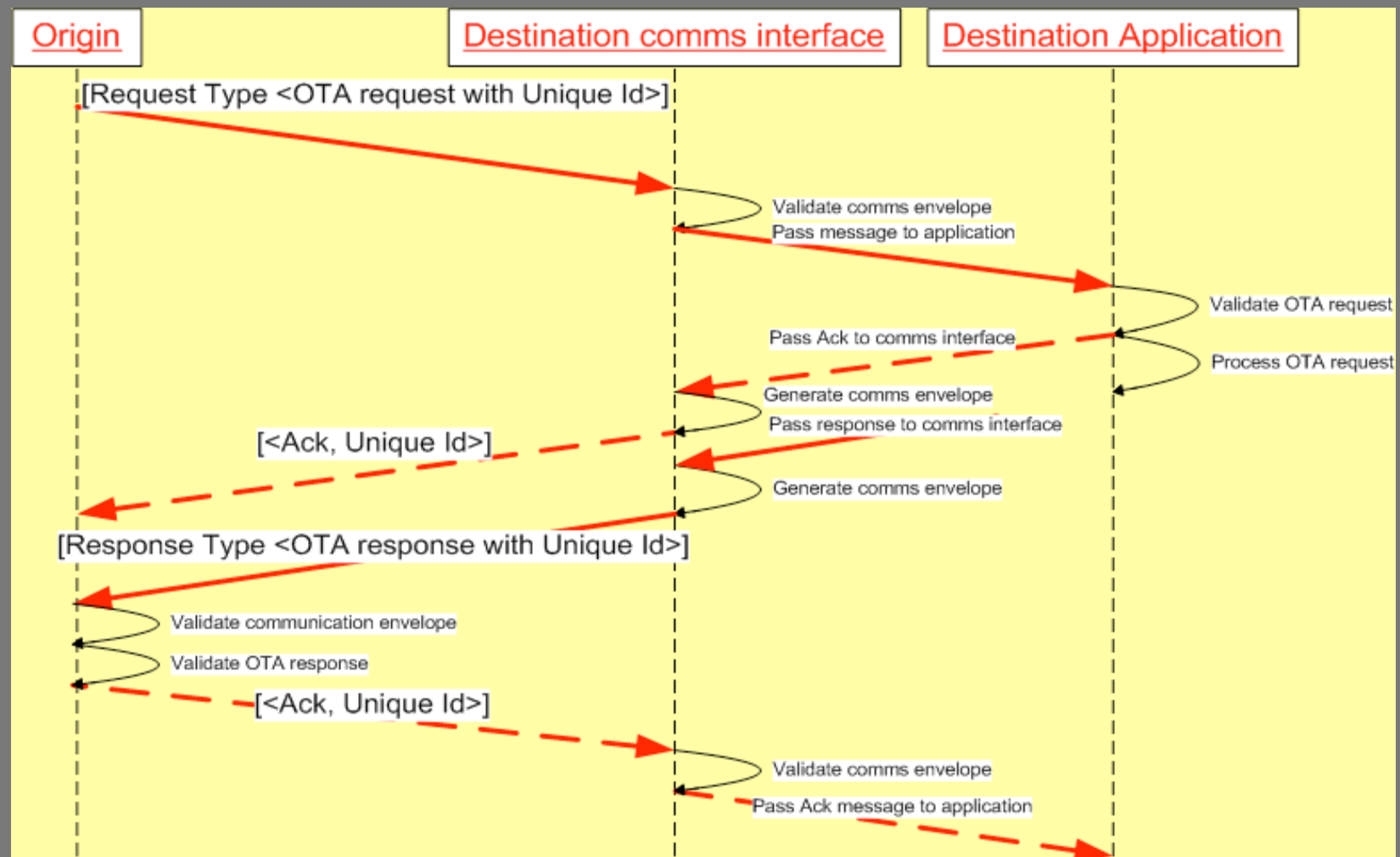
What did we study?

- Communication level handling of “asynchronicity”
- Application level handling of “asynchronicity” via
 - Generic Ack payload
 - Generic request for reporting purposes
 - Distinct RQ/RS pairs for reporting purposes
 - Synchronous handling of “asynchronicity” (Type C)

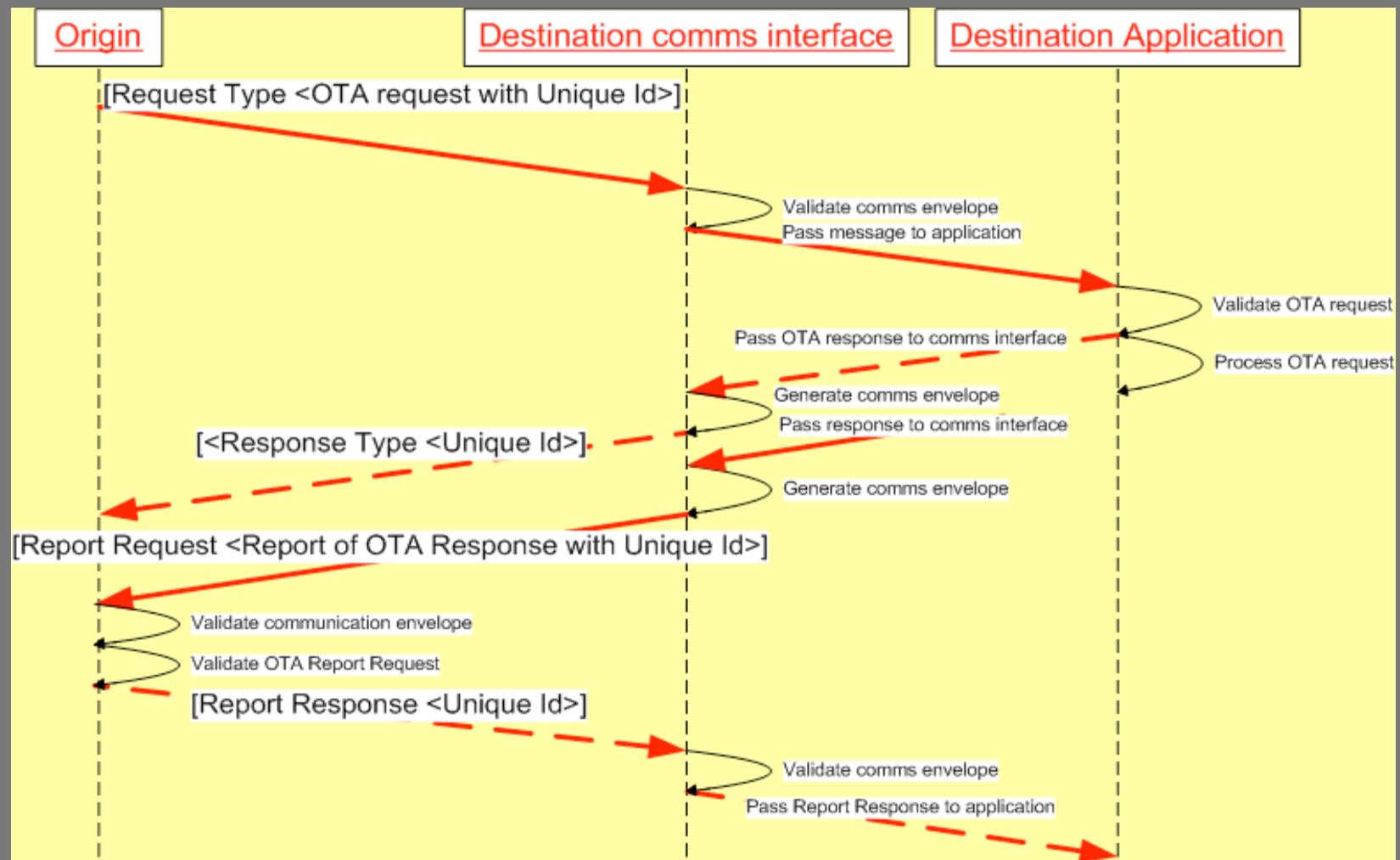
Communication level handling



Generic Ack payload



Generic request for reporting



What did we conclude?

- The OTA specs should recognize 2 methods to support asynchronous exchange of OTA messages:
 - at the communication level
 - at the application level using a generic OTA_Ack message
- Each reference transport protocol covered by OTA specs should include a section on asynchronous messaging and how it should be supported with this transport protocol

Next steps (2006B)

- Add a section in the http and SOAP Reference Transport Protocol specs to cover asynchronous messaging
- Define the OTA_Ack schema
- Determine which element/attribute should be used as a Unique Identifier
 - TransactionIdentifier, EchoToken, UniqueID
- Add of a new section in the MUG covering asynchronous messaging

Thank You !

David Morley (Marriott) *(chair)*
Stephen Adkins (Rubicon Group)
John Lambe (OpenJaw)
Kevin Camenzuli (Cendant CRG)
Alain Leveillé (Expedia)



Questions / Comments

OTA
The OpenTravel™ Alliance

**2006
ADVISORY
FORUM**



THE NEXT STEPS —
Orchestrating **THE**
Total Travel Solution

