



Specialists in air transport communications and IT solutions

ATI XML Reliable Messaging TypeX

Mansour Rezaei Mazinani

**AXIM / MET Information Exchange Conference
Washington DC, May 12-14, 2009**

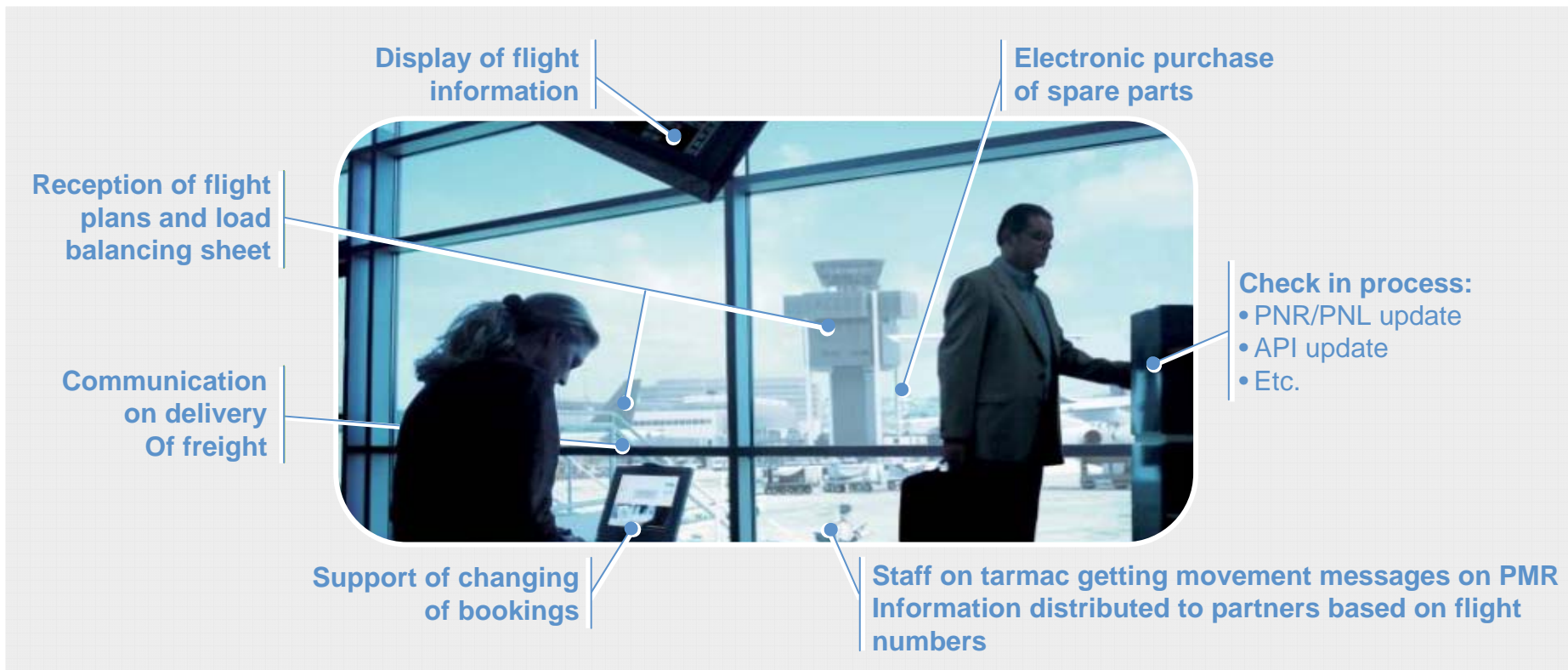


Agenda

- TypeX Background
- TypeX Characteristics
- TypeX Exchange Patterns
- TypeX Implementation Stacks
- Deployment Architectures
- TypeX role in SWIM

Reliable Messaging

A fundamental requirement for data exchange between different stakeholders of the air transport industry



XML - responds to ATI messaging development needs

XML is the future format

- Dynamic
- Flexible
- Supportive of much more information rich messaging

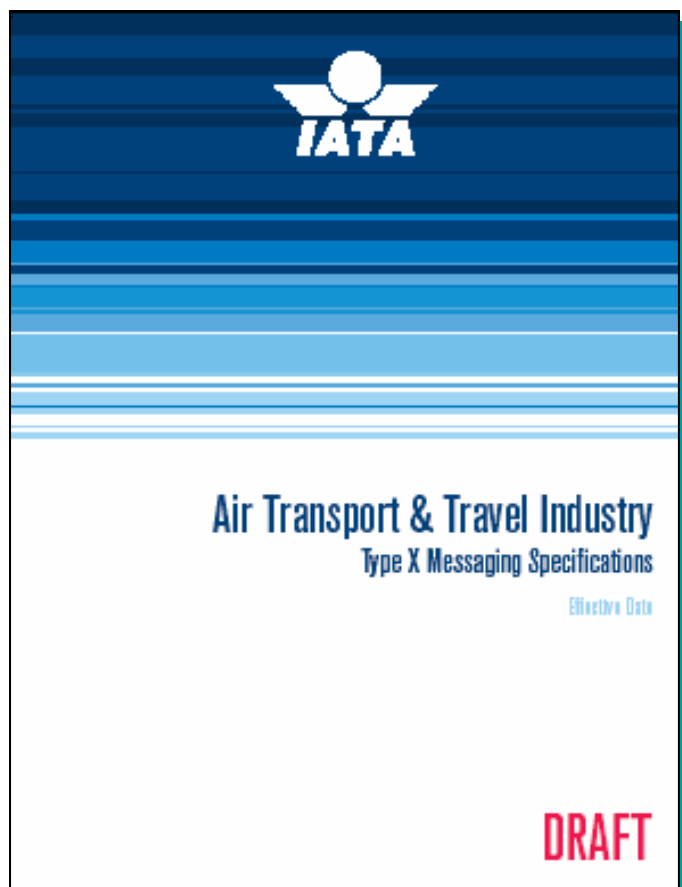
Use of XML will reduce

- Development cost
- Development time
- Dependency on hard-to-get expertise

And increase available performance

TypeX standard is developed to provide a solution for high performance reliable messaging with XML and Web services technologies compatible with air transport business practices

Available Documentation - TypeX Specification



- ✓ Specifies TypeX Message (TXM) structure and necessary headers
- ✓ Specifies TXM – SOAP mapping
- ✓ Specifies TypeX addressing; supports IATA and ICAO addressing
- ✓ Specifies reliability protocol
- ✓ Specifies Session Management
- ✓ Specifies all TypeX related Schema
- ✓ Initial implementations and testing completed

TypeX is planned to be published as IATA SCR Volume 7 in June 2009

TypeX -overview

- In its simplest form, the protocol consists of sending a payload with a message header. The transport layer will send some form of acknowledgement
- If reliability is required, then a XATAP header is joined to the message header. The receiver will return a XATAP acknowledgement
- If session management is required, then an XSM session must be established before messages may be sent
- Note that XSM and XATAP are independent in that either or both may be used
- In addition , regardless of the scenario, Command headers may be sent in order to control flow or repeat messages
- Finally, the sender may request a report (delivery or non-delivery)

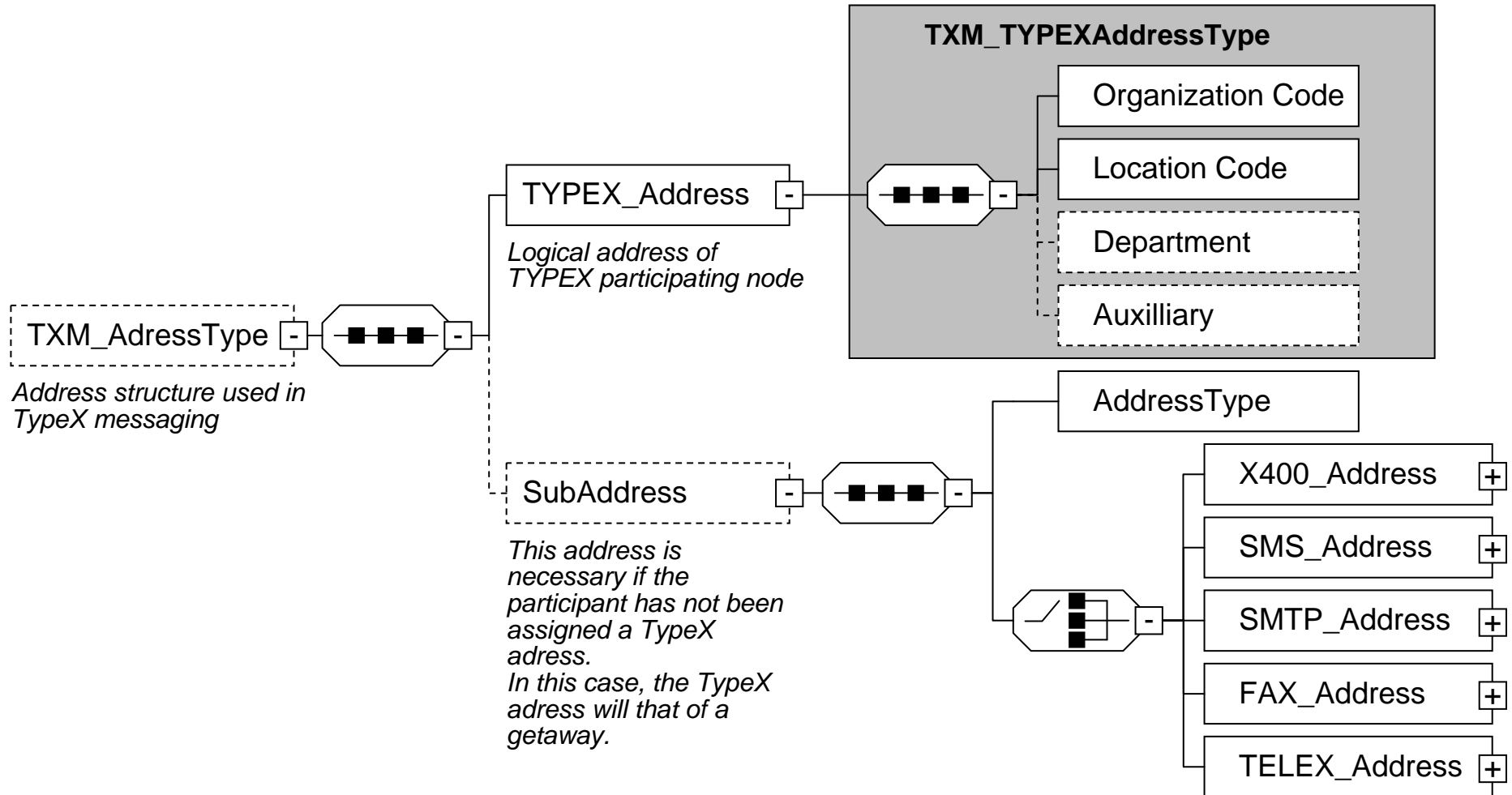
Type X vs- current standards

	Currently available XML capable standards	New TypeX
Addressing multiple recipients	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Addressing compatibility with ATI standards (IATA Type B, ICAO etc.)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Ability to address recipients over various protocols (e.g. X400, SMTP)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
End-to-End reporting (e.g. delivery notification)	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Application to Application Reliability	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Session management	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Grouping of messages	<input type="checkbox"/>	<input checked="" type="checkbox"/>

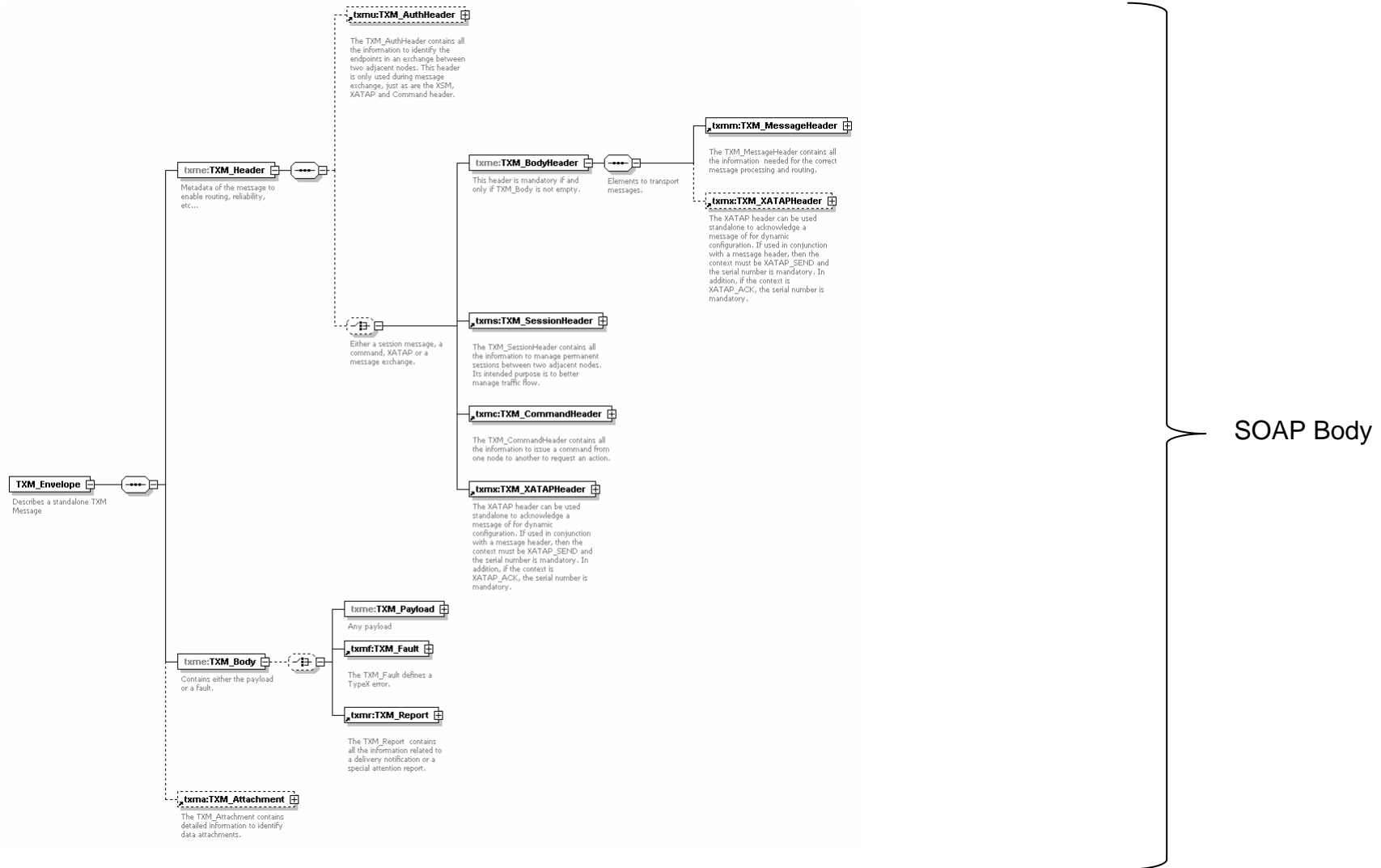
TypeX - properties

- IATA TypeB and ICAO AFTN message compatibility
- Support for all standard ATI store and forward message exchange patterns
- Full message assurance support for application to application delivery
- Permits detection of duplicate messages
- Permits messaging ordering
- Session management
- SOAP binding
- Security options
- Messaging priority
- Grouping of messages
- Multicast (one message to many recipients)
- End-to-End addressing (originator to recipient)
- Rich addressing scheme backward compatible with current ATI addressing
- End-to-End delivery notification (ultimate receiver or recipient to originator)
- Service command messages
- Openness by providing free fields
- Support for attachments

TypeX - address



TypeX Message to SOAP Mapping



TypeX - sample message over SOAP

```
<soap:Envelope xmlns:soap="http://www.w3.org/2003/05/soap-envelope"
  xmlns:txme="http://schema.iata.org/txm/envelope">
  <soap:Header>
    ...
  </soap:Header>
  <soap:Body>
    <txme:TXM_Envelope> ... </txme:TXM_Envelope>
  </soap:Body>
</soap:Envelope>
```

TypeX- security

- TypeX security capabilities include:
 - ✓ Content Integrity
 - ✓ Confidentiality
 - ✓ Authentication
 - ✓ Non repudiation
- Functionality enabled by the use of PKI for encryption and digital signature mainly by the end users
- Uses standard OASIS Web Services Security framework defined as SOAP extensions i.e. WS-Security with W3C XML Encryption & XML Digital Signature, WS-Trust, WS-Federation, WS-SecureConversation, WS-SecurityPolicy, SAML
- [Implementation guidelines for TypeX](#) is part of the work group documents

TypeX - exchange patterns

TypeX Message Exchange Patterns (MEP) govern end to end exchanges

TypeX supports the following MEPs:

- ✓ Asynchronous Request/Response via call back, conversation
- ✓ Send Only, one way exchange with required asynchronous response
- ✓ Fire & Forget, send but no response is expected



Standards

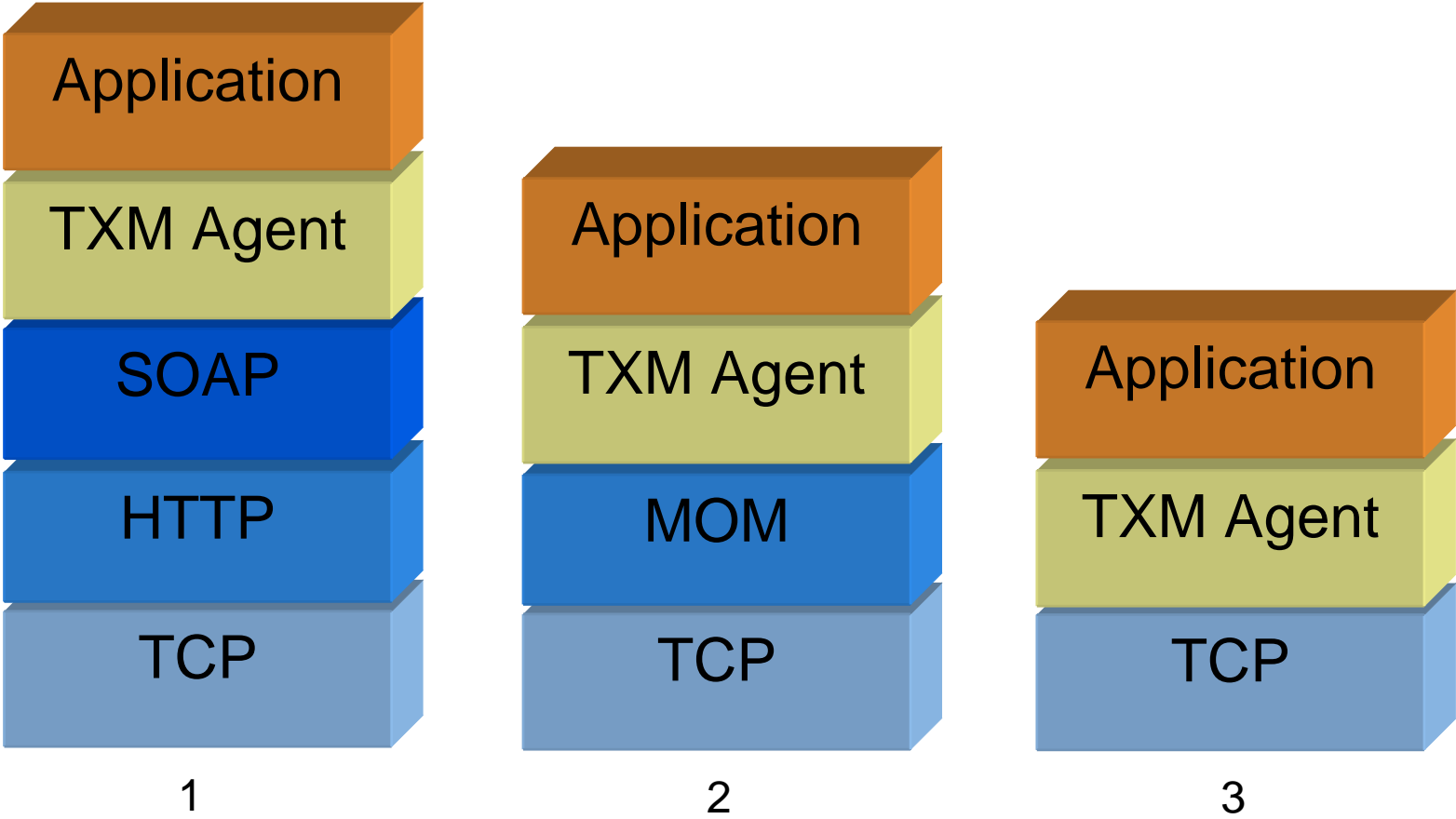


Technology



Services

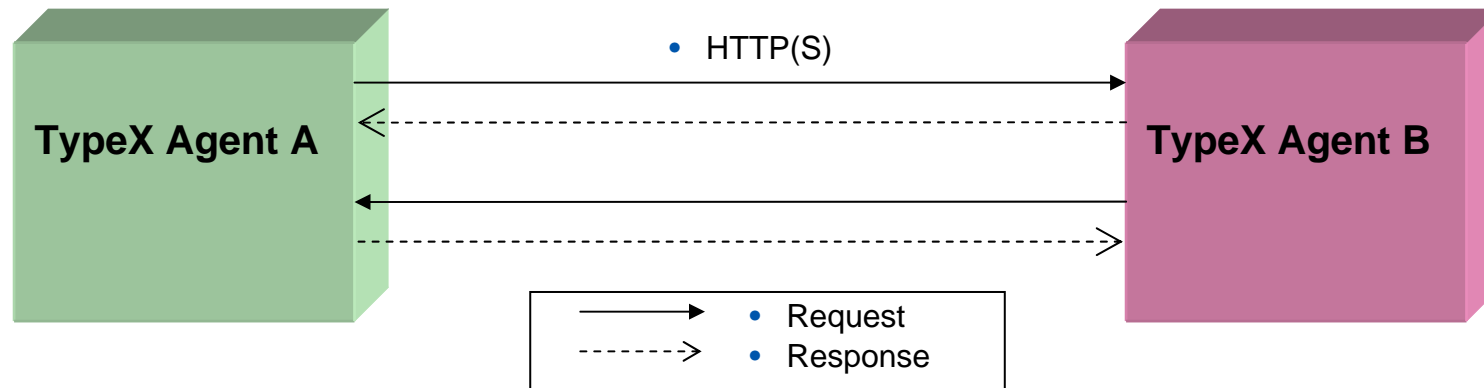
Recommended TypeX Stacks



TypeX over HTTP

Permanent two-way deployment view

Recommended for users with large throughput in both directions

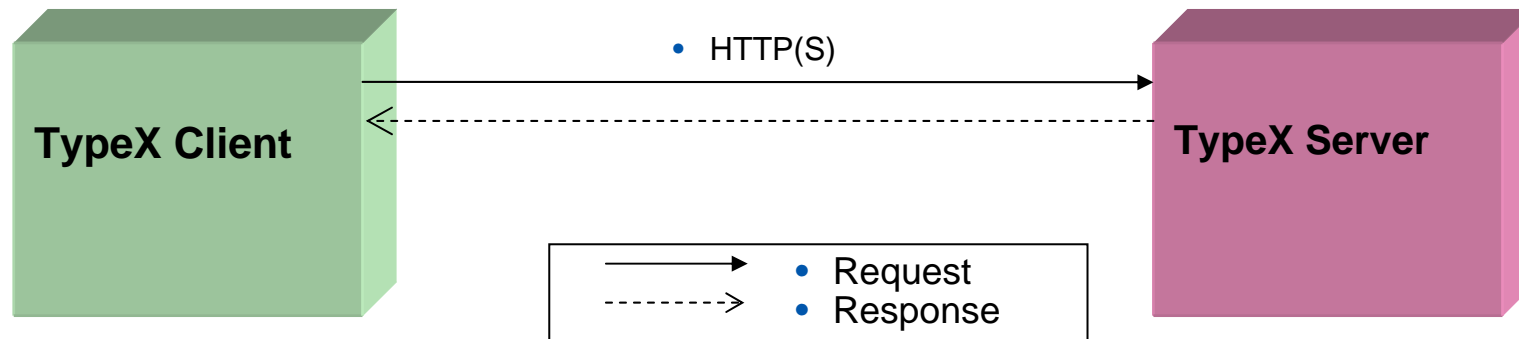


Characteristics:

- Permanent connections
- Dual HTTP connection, XSM & XATAP are enforced
- Service with a single operation: *sendMessage*

TypeX over HTTP

Permanent one-way deployment view



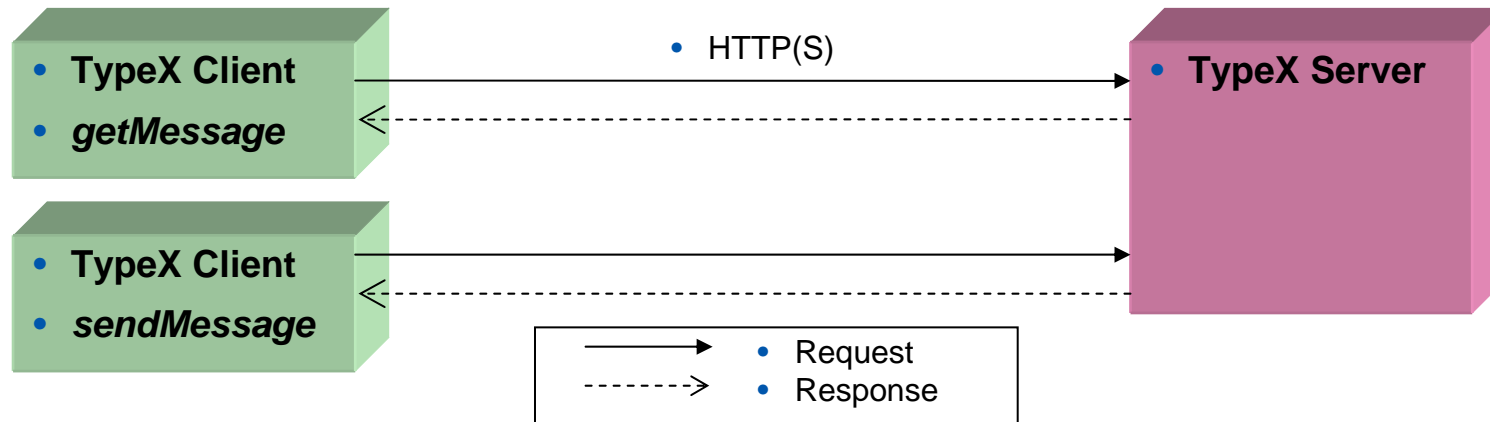
Characteristics:

- Permanent single connection
- XSM & XATAP are enforced
- Service with two operations: *sendMessage* & *getMessage*

TypeX over HTTP

Non-Permanent one-way deployment view

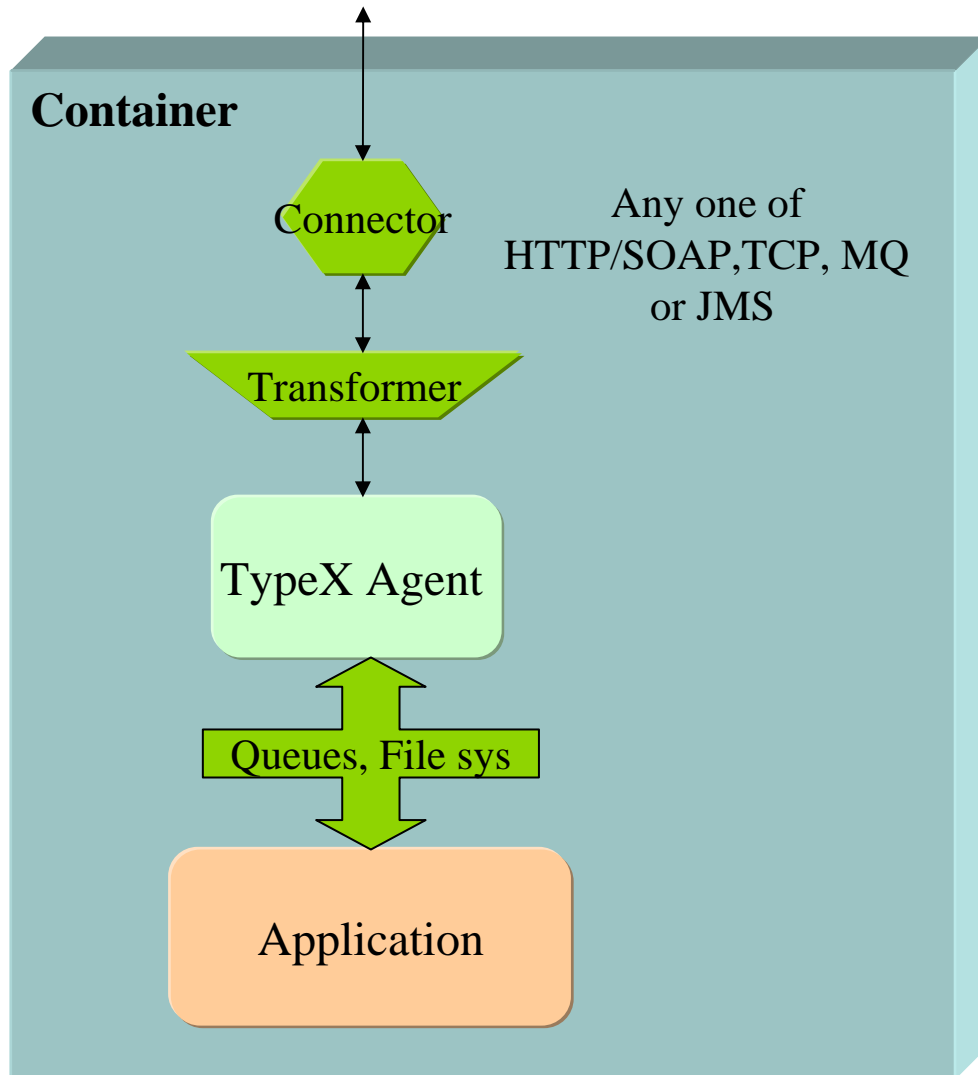
Recommended for users with low throughput; get or send message & disconnect



Characteristics:

- Non-permanent Connection
- Single HTTP connection, a single request/response
- No support for reliability and session management
- Two services, each with a single operation:
 - *sendMessage*: the client sends an envelope
 - *getMessage*: if a message is available, it is returned in the response

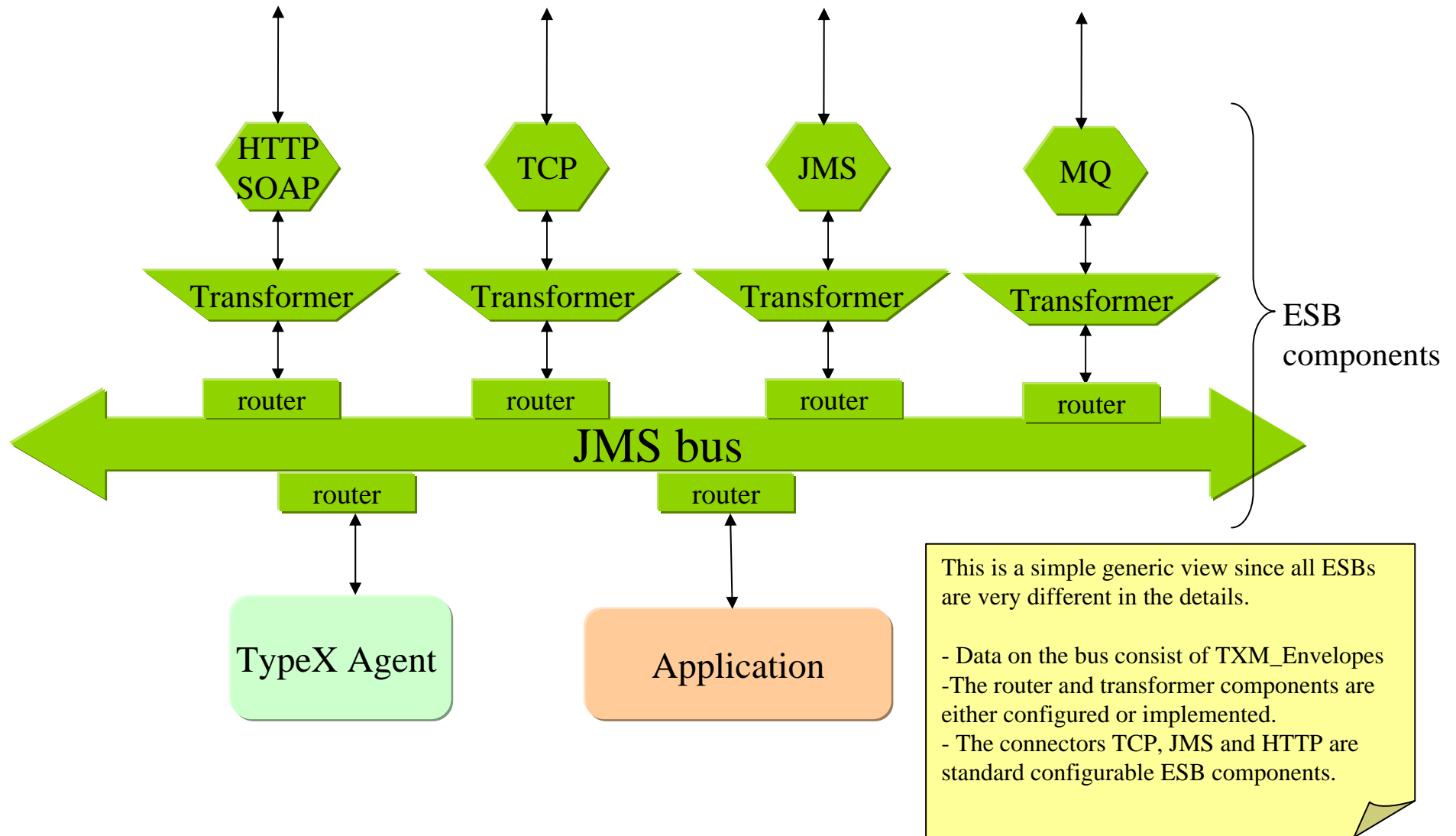
TypeX deployment - client side container



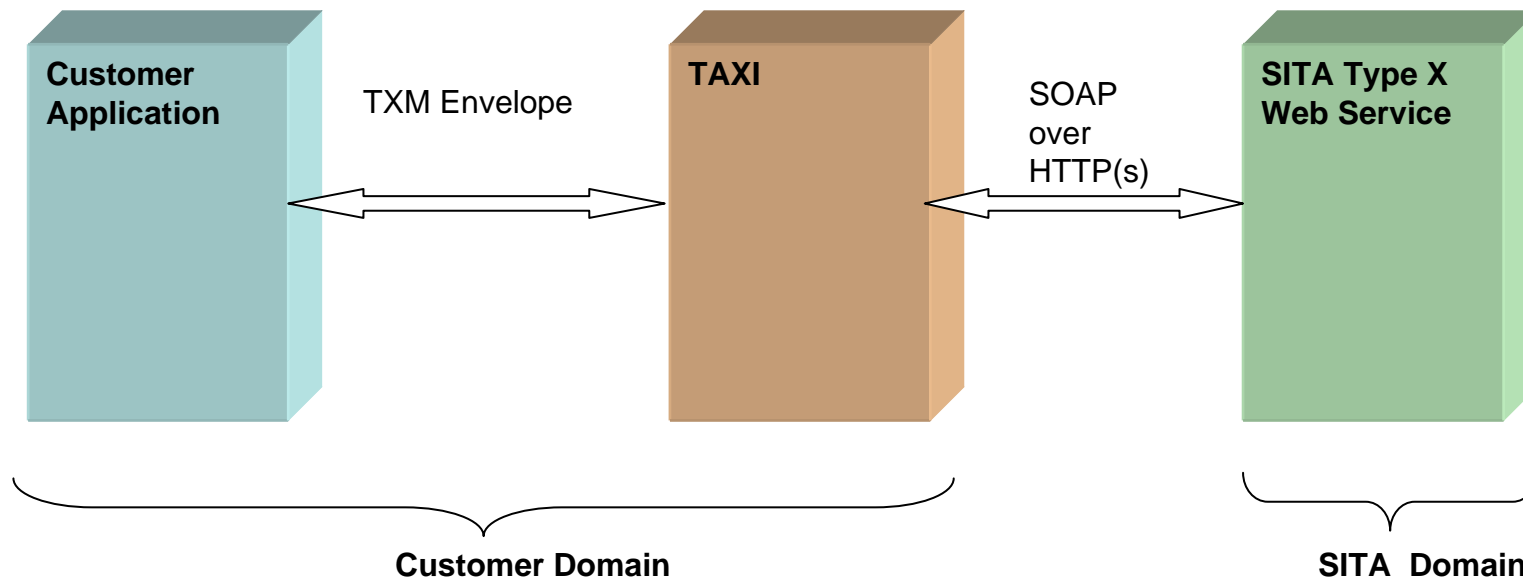
Simple view showing an application sending and receiving over JMS queues or a file system etc.

- Application is decoupled from the communication interface.
- Data on the queues are TXM_Envelopes
- One instance of the TypeX agent per connection.
- Transformer converts TXM_Envelopes into suitable form for the connector.
- Suitable containers include Tomcat, JBoss, etc..

TypeX deployment - client side ESB



TypeX - user package



TypeX's role in SWIM ?

- AIM and SWIM programs sets ambitious objectives to move to rich information availability and new generation of applications
- XML is largely used to describe information along various programs
- Airlines and ATCs exchange information on a daily basis and share similar requirements for reliable and secure messaging
- TypeX meets the new generation messaging requirements of both airline and ATC communities, facilitates interoperability and integration, and delivers cost efficiencies
- TypeX is created to meet ATI requirements for reliable messaging including those of SWIM program
- A prototype for xNOTAM is to be initiated during 2Q09

Thank You

Mansour.rezaei-mazinani@sita.aero

