Terrorist Watchlist Person Data Exchange Standard (TWPDES)
Version 3.0

Message Transaction Specification (MTS)

Information Exchange Package Documentation (IEPD)

December 2009
# Table of Contents

List of Figures ......................................................................................................................................................... 3  
List of Tables .......................................................................................................................................................... 3  
Document Revision History .................................................................................................................................... 4  

1 Introduction ..................................................................................................................................................... 5  
1.1 Purpose ..................................................................................................................................................... 5  
1.2 Principles .................................................................................................................................................. 5  
1.3 Requirements ............................................................................................................................................ 5  
1.4 Base Transports ........................................................................................................................................ 5  

2 Message Construction Using ebXML ............................................................................................................. 6  
2.1 Packaging Specification ........................................................................................................................... 6  
2.2 Message Delivery ..................................................................................................................................... 8  
2.3 Message Exchange Patterns ..................................................................................................................... 8  
2.4 Architectural Simplifications and Assumptions ....................................................................................... 9  
2.5 Watchlist Designation or Encounter Nomination Request Envelope .................................................... 10  
2.5.1 Request Envelope Structure ............................................................................................................ 10  
2.5.2 Request Envelope Elements and Attributes .................................................................................... 12  
2.5.3 Request Sample Message Content .................................................................................................. 15  
2.6 Watchlist Designation or Encounter Disposition Response Envelope ................................................... 17  
2.6.1 Response Envelope Structure ......................................................................................................... 17  
2.6.2 Response Envelope Elements and Attributes ................................................................................. 19  
2.6.3 Response Sample Message Content ................................................................................................ 23  
2.7 Error Response Envelope ....................................................................................................................... 25  
2.7.1 Error Response Envelope Structure ................................................................................................ 25  
2.7.2 Error Response Envelope Elements and Attributes ....................................................................... 26  
2.7.3 Error Response Sample Message Content ...................................................................................... 28  

3 Message Construction Using LEXS PD ....................................................................................................... 29  
3.1 High Level Message Structure ................................................................................................................. 29  
3.2 LEXS Metadata Elements and Attributes .............................................................................................. 30  
3.3 TWPDES-LEXS Sample Message Content ........................................................................................... 31  

4 Glossary ........................................................................................................................................................ 35  

5 References ..................................................................................................................................................... 37
List of Figures
Figure 1 – Canonical ebXML Message Structure .............................................................................................. 7
Figure 2 – Nominal ebXML Messaging Model ..................................................................................................... 8
Figure 3 – Watchlist Designation Request Envelope Structure .......................................................................... 10
Figure 4 – Watchlist Designation Response Envelope Structure ......................................................................... 17
Figure 5 – Error Response Envelope Structure .................................................................................................. 25
Figure 6 – High level view of a TWPDES-LEXS message .................................................................................. 29

List of Tables
Table 1: Elements and Attributes for Request Envelope ...................................................................................... 14
Table 2: Elements and Attributes for Response Envelope .................................................................................... 22
Table 3: Elements and Attributes for Error Response .......................................................................................... 27
Table 4: Elements and Attributes for LEXS Metadata ......................................................................................... 31
## Document Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Revised By</th>
<th>Date</th>
<th>Summary of Change</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.0 R0C0</td>
<td>C. J. Lee</td>
<td>12/4/2009</td>
<td>• Removed “Sensitive But Unclassified // For Official Use Only” markings in the document.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Removed references to “Controlled Unclassified Information”.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>• Updated Version 1.2b to Version 3.0 to mark the changes.</td>
</tr>
</tbody>
</table>
1 Introduction

1.1 Purpose
This document describes the Message Transaction Specification (MTS) of the Terrorist Watchlist Person Data Exchange Standard (TWPDES), version 3.0. The goal of the MTS is to provide a flexible standard for the electronic exchange of TWPDES 3.0 messages and related content between agencies and transactional systems deployed across the law enforcement and intelligence communities. Watchlist nominations, designations, encounters, and encounter dispositions are examples of content that can be conveyed via the MTS.

1.2 Principles
In developing the MTS, the primary goal was to satisfy the TWPDES conveyance requirements, while leveraging existing standards as much as possible. As such, the TWPDES MTS introduces a new standard, along with a prescription for leveraging three established information technology standards: the Simple Object Access protocol (SOAP), which is maintained by the World Wide Web Consortium (W3C); the e-business XML (ebXML) Messaging Service (ebMS), which is maintained by the Organization for the Advancement of Structured Information Standards (OASIS); and the Law Enforcement Information Sharing Program (LEISP) Exchange Specification (LEXS) standard, which is endorsed and supported by the U.S. Department of Justice. Additional message exchange standards may be considered for inclusion in future versions of the MTS.

1.3 Requirements
The MTS was defined with the goal of supporting the electronic exchange of TWPDES messages, regardless of payload size or composition. Initially, this MTS specification document prescribes the use of SOAP and ebMS standards in support of two-way request/reply conveyance semantics, where the reply may be a confirmation, custom response structure or an error. However, these underlying standards are fully capable of supporting additional semantics, including one and two way push and pull. Future versions of this document will describe how SOAP and ebMS can be leveraged to convey TWPDES content using these additional semantics.

TWPDES messages can also be conveyed using the LEXS Publish and Discovery (LEXS-PD) framework. LEXS contains a statically defined “Digest” section that sets a standard for the base level of data sharing among all of the systems that are LEXS-enabled. LEXS Digest includes common data objects such as person, location, vehicle, activity, etc. in NIEM content. By copying data to the “Digest”, the TWPDES-LEXS messages allow all of the LEXS-enabled systems to share the data in the Digest without requiring these systems to understand the specifics of TWPDES.

1.4 Base Transports
The MTS, and the standards it leverages, are capable of supporting the electronic exchange of content via multiple protocols (e.g. Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP)) and infrastructures, including message-oriented middleware (MOM) products. Initially, a MOM product that provides guaranteed message delivery, such as IBM’s Websphere MQ, is assumed to be the choice of transport. However, future versions of the MTS will describe how additional web service standards, such as WS-ReliableMessaging and WS-Addressing, can be leveraged to provide guaranteed message delivery between parties hosting Service-Oriented Architecture (SOA) and Enterprise Service Bus (ESB) infrastructures.
2 Message Construction Using ebXML

2.1 Packaging Specification

An ebXML message is a communication-protocol independent, Multipurpose Internet Mail Extensions (MIME) multipart message envelope, structured in compliance with the SOAP Messages with Attachments (see http://www.w3.org/TR/SOAP-attachments) specification, referred to as a Message Package.

There are two logical MIME parts within the Message Package:

- A MIME section, referred to as the Header Container, containing one SOAP-compliant message. This XML document is referred to as a SOAP Message for the remainder of this specification.

- Zero or more MIME sections, referred to as Payload Containers, containing application level payloads.

The SOAP Message is an XML document that contains a SOAP Envelope element. This is the root element of the XML document representing the SOAP Message. In this document, the term “element” refers to a unit of information which may be simple (individual) or complex (consist of other elements). The SOAP Envelope element consists of the following:

- One SOAP Header element. This is a generic mechanism for adding features to a SOAP Message, including ebXML specific header elements. This is a container for message service handler control data and information related to the payload parts of the message.

- A SOAP Body element that is used to convey TWPDES 3.0 SOA payloads between agencies and systems. Watchlist nominations and designations, encounters, and encounter dispositions are some of the TWPDES 3.0 payloads that will be conveyed within the bodies of SOAP messages.

The ebXML Message Service Specification uses two types of messages. Application User Messages convey custom application payloads, while Signal Messages convey errors, message receipts, and recipient-driven pull requests. However, this document – and the TWPDES 3.0 standard as currently envisioned – specifically leverages user messages for custom application payloads, and signal messages for conveying one or more infrastructure or application-specific errors.

The canonical structure and composition of an ebXML User Message is described in the following figure.
Figure 1 illustrates optional message components using dashed lines, while required message components are denoted with solid lines. While ebXML and SOAP are designed to work together and with additional web service standards for security (WS-Security) and guaranteed delivery (WS-ReliableMessaging), the current version of MTS assumes such capabilities will be provided by an underlying MOM infrastructure. Hence,
unlike the figure, current MTS messages will not contain any WS-Security or WS-ReliableMessaging constructs within their SOAP headers.

2.2 Message Delivery

The ebMS standard prescribes several roles and infrastructure components that are involved in the exchange of messages. A producer and consumer exchange messages via a pair of proxy components acting on their behalf. Producers and consumers may be implemented or represented as applications, queuing systems, or additional SOAP processors.

These proxy components, called message service handlers (MSHs), generate and convey messages that conform to the ebMS specification. As such, the sending and receiving MSH components are usually implemented as one or more SOAP processors, which are designed to understand the Messaging headers contained within the messages being processed. Additionally, these same components are responsible for exchanging requests, responses, notifications, and errors with producers and consumers.

![Figure 2 – Nominal ebXML Messaging Model](image)

The MSH components shown in Figure 2 are typically implemented and provided as components within a commercial SOA/ESB infrastructure. Several application server and MOM vendors have either included support for the ebXML messaging model within their core product offerings, or provide such functionality in additional service packs. As such, it is envisioned that the MTS’s reliance on this model will have a positive influence on the adoption of SOA/ESB technology across the law enforcement and intelligence communities.

2.3 Message Exchange Patterns

Currently, the MTS only prescribes support for the request/response message exchange pattern (MEP), where the initiating connection has an application payload to send reliably to the responder and can receive an application payload as the response, or an error message in the process.

Initially, the MTS Request/Response MEP prescribes the transmission of watchlist designation requests and responses between WLDesignationSender and WLDesignationRecipient actors. Future versions of MTS documentation will prescribe the conveyance of additional domain content, such as encounters. However, for now, it is presumed that these actors are systems that are engaged in the transactional conveyance of watchlist content. WLDesignationSender actors would initiate the conveyance conversation by transmitting a service request containing one or more WatchListDesignation elements to a WLDesignationRecipient system. The
service request would essentially ask a \textit{WLDesignationIngestService} (hosted by the \textit{WLDesignationRecipient} system) to ingest the watchlist payload(s) using the \textit{IngestWLDesignation} action (or service endpoint). The \textit{WLDesignationRecipient} would then reply with one or more response elements, by asking a \textit{WLDesignationResponseIngestService} (hosted by the \textit{WLDesignationSender} system) to ingest the watchlist response payload(s) using the \textit{IngestWLDesignationResponse} action (or service endpoint).

At present, an underlying MOM product is expected to provide reliable message delivery. However, future versions of the MTS will prescribe the use of additional standards, such as WS-ReliableMessaging, in order to provide reliable delivery without a traditional MOM infrastructure.

\section*{2.4 Architectural Simplifications and Assumptions}

- While ebMS supports both SOAP 1.1 and SOAP 1.2, several prominent commercial SOA infrastructure vendors currently only support interoperability with other vendors’ products at the SOAP 1.1 level. Hence, this specification requires SOAP 1.1 in order to maximize interoperability between these vendors’ implementations.

- While the ebMS specification allows for either plain SOAP message or multipart MIME, for consistency only multipart MIME messages are supported. However, it is envisioned that TWPDES 3.0 payloads will be contained within SOAP bodies – not MIME parts (i.e. payload containers).

- While the ebMS specification allows for multiple lower level transports, such as HTTP(s) and SMTP, this specification currently assumes the use of a MOM product for conveying asynchronous requests and responses. Additionally, it is assumed that MOM or HTTP(s) will be leverage for the synchronous exchange of TWPDES 3.0 messages.

- While the ebMS specification allows for many different MEPs (i.e.: One-Way/Push, One-Way/Pull, Two-Way and potentially others), this specification only provides support for: Two-Way request-response message exchanges.

- This specification \textbf{REQUIRES} the SOAP message to have an XML Prolog, and it \textbf{MUST} have the encoding UTF-8.

- The current MTS relies on commercial MOM products to provide secure and reliable message delivery. Hence, the use of WS-Security and WS-Reliability will not be prescribed until a future MTS specification release.
2.5 Watchlist Designation or Encounter Nomination Request Envelope

2.5.1 Request Envelope Structure

An MTS request message consists of an ebMS UserMessage element that is contained within the header of a SOAP envelope. Information uniquely identifying the message, the conversing parties, and the desired service action are all expressed as child elements within the UserMessage. Although request information can be conveyed via name-value properties within a MessageProperties element inside the UserMessage, this information will most likely be conveyed within a custom payload element contained within the SOAP message body. Currently, MTS prescribes that such payloads be encapsulated within an MTS Request element that is uniquely identified by its request identifier (i.e. RequestID attribute). Zero or more Request elements are contained within an MTS RequestList envelope element. Hence, sending watchlist designations to downstream customers would involve the generation of one or more physical MTS messages – each containing a SOAP
header and body – where each body would, in turn, contain a single MTS RequestList element containing zero or more MTS Request elements. Each Request would contain a single TWPDES 3.0 watchlist designation, as shown in Figure 3.

Similarly, sending encounter requests to the watchlisting agency or sending encounter information forward requests would be done in the same manner. Each Request would contain a single TWPDES 3.0 encounter request. See TWPDES 3.0 Encounter Information Exchange Package Documentation (IEPD).
### 2.5.2 Request Envelope Elements and Attributes

Table 1 describes the elements and attributes that comprise the watchlist designation request message. The “Source” column in the table indicates whether the element or attribute is populated by the sending business application or the sending agency’s underlying ebXML MSH.

<table>
<thead>
<tr>
<th>Element / Attribute</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:Message/eb:MessageInfo/eb:Timestamp</td>
<td>MSH</td>
<td>The timestamp for the message determined by the ebXML MSH when the actual message to be sent is created.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:MessageInfo/eb:MessageId</td>
<td>App</td>
<td>A unique message identifier generated by the sender, either a concatenation of message elements, to create a globally unique identifier, or a single message element if that element is globally unique.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PartyInfo/eb:From/eb:Role</td>
<td>App</td>
<td>A description of the role this party assumes with respect to a particular conversation or exchange of messages with a target party.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PartyInfo/eb:To/eb:Role</td>
<td>App</td>
<td>A description of the role this party assumes with respect to a particular conversation or exchange of messages with a source party.</td>
</tr>
<tr>
<td>Element / Attribute</td>
<td>Source</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PayloadInfo/eb:PartInfo@href</td>
<td>MSH</td>
<td>References the identifier (i.e. “#idref”) of the SOAP Body element which contains the SOA payload (ex: WatchlistDesignation, transaction Response, encounter/disposition).</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PayloadInfo/eb:PartInfo /eb:Schema</td>
<td>App</td>
<td>OPTIONAL. References the schema associated with the reference SOAP body element, enabling handlers to determine their ability to validate/process the payload without actually having to do so in full.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PayloadInfo/eb:PartInfo /eb:PartProperties/eb:Property</td>
<td>App</td>
<td>OPTIONAL. One or more name/value pairs of information that is relevant/useful to – or required by – the desired action of the targeted service.</td>
</tr>
</tbody>
</table>

**Payload**

<table>
<thead>
<tr>
<th>Element / Attribute</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/S11:Envelope/S11:Body/tx:RequestList</td>
<td>App</td>
<td>Contains zero or more requests for service.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:RequestList /tx:Request</td>
<td>App</td>
<td>Contains a request to be processed by the action of the targeted service. This element contains a unique identifier that can (optionally be followed by one or more elements of any type. By convention, MTS prescribes that these elements are application payloads, representing the full content of the request. A <em>wl:WatchlistDesignation</em> element is an example of an application request payload.</td>
</tr>
<tr>
<td>Element / Attribute</td>
<td>Source</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------------------------------------------</td>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:RequestList/tx:Request/tx:RequestID</td>
<td>App</td>
<td>Identifies the request, and is unique to the sending entity (agency).</td>
</tr>
</tbody>
</table>

Table 1: Elements and Attributes for Request Envelope
2.5.3 Request Sample Message Content

The following XML fragment provides a sample of a TWPDES watchlist designation request message.

```xml
<?xml version="1.0" encoding="UTF-8"?>
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <S11:Header>
        <eb:Messaging S11:MustUnderstand="1">
            <eb:UserMessage>
                <eb:MessageInfo>
                    <eb:MessageId>16215d60-30ed-11dc-8d58-e7af10974858</eb:MessageId>
                </eb:MessageInfo>
                <eb:PartyInfo>
                    <eb:From>
                        <eb:PartyId>Watchlisting_Agency</eb:PartyId>
                    </eb:From>
                    <eb:To>
                        <eb:PartyId>Screening_Agency</eb:PartyId>
                    </eb:To>
                </eb:PartyInfo>
                <eb:CollaborationInfo>
                    <eb:Service type="Receiver">WLDesignationIngestService</eb:Service>
                    <eb:Action>IngestWLDesignation</eb:Action>
                    <eb:ConversationId>4321</eb:ConversationId>
                </eb:CollaborationInfo>
                <eb:PayloadInfo>
                    <eb:PartInfo href="IDREQ190">
                        <eb:Schema location="http://registry.niem.gov/niem/domains/watchlisting/1.0/watchlisting-v1.0.xsd" version="1.0"/>
                        <eb:Description xml:lang="en-us">Watchlist Designation</eb:Description>
                    </eb:PartInfo>
                </eb:PayloadInfo>
            </eb:UserMessage>
        </eb:Messaging>
    </S11:Header>
</S11:Envelope>
```
<S11:Envelope>
  <S11:Header>
    <eb:PartProperties>
      <eb:Property name="MimeType">application/xml</eb:Property>
    </eb:PartProperties>
    <eb:PayloadInfo>
      <eb:UserMessage>
      </eb:UserMessage>
    </eb:PayloadInfo>
    <eb:Messaging>
    </eb:Messaging>
  </S11:Header>
  <S11:Body>
    <tx:RequestList ism:ownerProducer="USA" ism:classification="U"
    xsi:schemaLocation="http://twpdes.gov/transaction/1.1 ../schemas/transaction/1.1/transaction.xsd"
    xmlns:s="http://niem.gov/niem/structures/2.0" xmlns:tx="http://twpdes.gov/transaction/1.1"
      <tx:Request ism:ownerProducer="USA" ism:classification="U" s:id="IDREQ190">
        <tx:RequestID>190</tx:RequestID>
        <wl:WatchlistDesignation s:id="WLD_1"
        xsi:schemaLocation="http://twpdes.gov/twpdes/watchlisting/2.0 ../schemas/twpdes/watchlisting/2.0/watchlisting.xsd"
        xmlns:xlink="http://www.w3.org/TR/xlink">
          <wl:IdentityID>2009433</wl:IdentityID>
        </wl:WatchlistDesignation>
      </tx:Request>
    </tx:RequestList>
  </S11:Body>
</S11:Envelope>
2.6 Watchlist Designation or Encounter Disposition Response Envelope

2.6.1 Response Envelope Structure

An MTS response message consists of an ebMS *UserMessage* element that is contained within the header of a SOAP envelope. Information uniquely identifying the message, the conversing parties, and the desired service action are all expressed as child elements within the *UserMessage*. Although response information can be conveyed via name-value properties within a *MessageProperties* element inside the *UserMessage*, this information will most likely be conveyed within a custom payload contained within the SOAP message body. Currently, MTS prescribes that such payloads be encapsulated within an MTS *Response* element that specifically references (via its *ReferencedRequestID* attribute) the request to which the response refers. Zero or
more Response elements are contained within an MTS ResponseList envelope element. Hence, sending
watchlist designation responses would involve the generation of one or more physical MTS messages – each
containing a SOAP header and body – where each body would, in turn, contain a single MTS ResponseList
element containing zero or more MTS Response elements. Each Response would contain a single response
payload (ex: transaction:BasicResponsePayload), as shown in Figure 4.

Since it is assumed that MTS messages will be conveyed reliably via a MOM infrastructure, there is no need to
use explicit response messages for positive acknowledgement of message receipt. Instead, MTS responses are
reserved for cases where specific responses are required under nominal request/reply messaging conditions –
such as conveying application-level informational messages, warnings, and errors.
### 2.6.2 Response Envelope Elements and Attributes

Table 2 describes the elements and attributes that comprise the watchlist designation response message. The “Source” column in the table indicates whether the element or attribute is populated by the sending business application or the sending agency’s underlying ebXML MSH.

<table>
<thead>
<tr>
<th>Element / Attribute</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:Message/eb:MessageInfo/eb:Timestamp</td>
<td>MSH</td>
<td>The timestamp for the message determined by the ebXML MSH when the actual message to be sent is created.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:Message/eb:MessageInfo/eb:MessageId</td>
<td>App</td>
<td>A unique message identifier generated by the sender, either a concatenation of message elements, to create a globally unique identifier, or a single message element if that element is globally unique.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:Message/eb:MessageInfo/eb:RefToMessageId</td>
<td>App</td>
<td>Used only in response messages, this attribute references the unique identifier of the original request message to which this message is responding.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:Message/eb:PartyInfo/eb:From/eb:Role</td>
<td>App</td>
<td>A description of the role this party assumes with respect to a particular conversation or exchange of messages with a target party.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:Message/eb:PartyInfo/eb:To/eb:Role</td>
<td>App</td>
<td>A description of the role this party assumes with respect to a particular conversation or exchange of messages with a source party.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage</td>
<td>Msg</td>
<td>Identification of the specific, desired action being requested of the</td>
</tr>
<tr>
<td>Element / Attribute</td>
<td>Source</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>message/eb:CollaborationInfo/eb:Action</td>
<td></td>
<td>message processing service on the receiver’s end of the conversation.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:MessageProperties/Property</td>
<td>App</td>
<td>OPTIONAL. One or more name/value pairs of information that is relevant/useful to – or required by – the desired action of the targeted service.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PayloadInfo/eb:PartInfo@href</td>
<td>MSH</td>
<td>References the identifier (i.e. “#idref”) of the SOAP Body element which contains the SOA payload (ex: watchlist designation/nomination, encounter/disposition).</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PayloadInfo/eb:PartInfo /eb:Schema</td>
<td>Msg</td>
<td>OPTIONAL. References the schema associated with the reference SOAP body element, enabling handlers to determine their ability to validate/process the payload without actually having to do so in full.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PayloadInfo/eb:PartInfo /eb:PartProperties/Property</td>
<td>App</td>
<td>OPTIONAL. One or more name/value pairs of information that is relevant/useful to – or required by – the desired action of the targeted service.</td>
</tr>
<tr>
<td><strong>Payload</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:ResponseList</td>
<td>App</td>
<td>Contains zero or more responses to prior requests for service.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:ResponseList /tx:Response</td>
<td>App</td>
<td>Contains the response to a prior service request. Response is an MTS element that can be extended to provide custom response structures for applications by embedding a custom response payload. BasicResponsePayload is an MTS element that provides a simple,</td>
</tr>
<tr>
<td><strong>Element / Attribute</strong></td>
<td><strong>Source</strong></td>
<td><strong>Description</strong></td>
</tr>
<tr>
<td>------------------------</td>
<td>------------</td>
<td>-----------------</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:ResponseList /tx:BasicResponsePayload/tx:ActionTaken</td>
<td>App</td>
<td>An enumerated value that codifies the summarizing action taken by the service in processing a prior request.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/ActionTaken</td>
<td>App</td>
<td>An enumerated value (i.e. <em>Rejected</em>, <em>Proceeded-With-Errors</em>, <em>Accepted</em>) that codifies the action taken by the service in processing a prior request.</td>
</tr>
<tr>
<td>Element / Attribute</td>
<td>Source</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------------------------------------------------------------</td>
<td>--------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:ResponseList/tx:Response/tx:BasicResponsePayload/tx:ResponseItem/tx:SystemIdentifier/intel:SystemName</td>
<td>App</td>
<td>References the name of the targeted system of record that hosts the relevant entity and scopes its (presumably) unique identifier.</td>
</tr>
<tr>
<td>/S11:Envelope/S11:Body/tx:ResponseList/tx:Response/tx:BasicResponsePayload/tx:ResponseItem/tx:Note</td>
<td>App</td>
<td>A convenience structure that currently holds a textual element, but could be expanded to include additional information.</td>
</tr>
</tbody>
</table>

Table 2: Elements and Attributes for Response Envelope
2.6.3 Response Sample Message Content

The following XML fragment provides a sample of a TWPDES watchlist designation response message.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<S11:Envelope xsi:schemaLocation="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/
../schemas/transaction/1.1/ebms-header-3_0-200704.xsd" xmlns:eb="http://docs.oasis-open.org/ebxml-
msg/v3.0/ns/core/200704/" xmlns:S11="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <S11:Header>
        <eb:Messaging S11:MustUnderstand="1">
            <eb:UserMessage>
                <eb:MessageInfo>
                    <eb:Timestamp>2007-07-12T23:01:31</eb:Timestamp>
                    <eb:MessageId>024d6040-30ed-11dc-a870-b5f244be8fc9</eb:MessageId>
                    <eb:RefToMessageId>16215d60-30ed-11dc-8d58-e7af10974858</eb:RefToMessageId>
                </eb:MessageInfo>
                <eb:PartyInfo>
                    <eb:From>
                        <eb:PartyId>Screening_Agency</eb:PartyId>
                        <eb:Role>http://niem.gov/niem/domains/watchlisting/roles/WLDesignationRecipient</eb:Role>
                    </eb:From>
                    <eb:To>
                        <eb:PartyId>Watchlisting_Agency</eb:PartyId>
                        <eb:Role>http://niem.gov/niem/domains/watchlisting/roles/WLDesignationSender</eb:Role>
                    </eb:To>
                    <eb:CollaborationInfo>
                        <eb:Service type="Receiver">WLDesignationResponseIngestService</eb:Service>
                        <eb:Action>IngestWLDesignationResponse</eb:Action>
                        <eb:ConversationId>4321</eb:ConversationId>
                    </eb:CollaborationInfo>
                </eb:PartyInfo>
                <eb:MessageProperties>
                    <eb:Property name="ServiceActionTaken">Accepted</eb:Property>
                </eb:MessageProperties>
                <eb:PayloadInfo>
                    <eb:PartInfo href="IDWLDR200">
```

---

2.6.3 Response Sample Message Content

The following XML fragment provides a sample of a TWPDES watchlist designation response message.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<S11:Envelope xsi:schemaLocation="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/
../schemas/transaction/1.1/ebms-header-3_0-200704.xsd" xmlns:eb="http://docs.oasis-open.org/ebxml-
msg/v3.0/ns/core/200704/" xmlns:S11="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <S11:Header>
        <eb:Messaging S11:MustUnderstand="1">
            <eb:UserMessage>
                <eb:MessageInfo>
                    <eb:Timestamp>2007-07-12T23:01:31</eb:Timestamp>
                    <eb:MessageId>024d6040-30ed-11dc-a870-b5f244be8fc9</eb:MessageId>
                    <eb:RefToMessageId>16215d60-30ed-11dc-8d58-e7af10974858</eb:RefToMessageId>
                </eb:MessageInfo>
                <eb:PartyInfo>
                    <eb:From>
                        <eb:PartyId>Screening_Agency</eb:PartyId>
                        <eb:Role>http://niem.gov/niem/domains/watchlisting/roles/WLDesignationRecipient</eb:Role>
                    </eb:From>
                    <eb:To>
                        <eb:PartyId>Watchlisting_Agency</eb:PartyId>
                        <eb:Role>http://niem.gov/niem/domains/watchlisting/roles/WLDesignationSender</eb:Role>
                    </eb:To>
                    <eb:CollaborationInfo>
                        <eb:Service type="Receiver">WLDesignationResponseIngestService</eb:Service>
                        <eb:Action>IngestWLDesignationResponse</eb:Action>
                        <eb:ConversationId>4321</eb:ConversationId>
                    </eb:CollaborationInfo>
                </eb:PartyInfo>
                <eb:MessageProperties>
                    <eb:Property name="ServiceActionTaken">Accepted</eb:Property>
                </eb:MessageProperties>
                <eb:PayloadInfo>
                    <eb:PartInfo href="IDWLDR200">
```
<eb:Schema
location="http://registry.niem.gov/niem/domains/transaction/1.1/transaction-v1.1.xsd" version="1.1"/>
<eb:Description xml:lang="en-us">Watchlist Designation</eb:Description>
<eb:PartProperties>
  <eb:Property name="MimeType">application/xml</eb:Property>
</eb:PartProperties>
</eb:PayloadInfo>
</eb:UserMessage>
</eb:Messaging>
</S11:Header>
<S11:Body>
<tx:ResponseList ism:ownerProducer="USA" ism:classification="U"
xmlns:s="http://niem.gov/niem/structures/2.0" xmlns:tx="http://twpdes.gov/transaction/1.1"
  <tx:Response ism:ownerProducer="USA" ism:classification="U" s:id="IDWLDR200">
    <tx:ReferencedRequestID>190</tx:ReferencedRequestID>
    <tx:BasicResponsePayload>
      <tx:ResponseItem ism:classification="U" ism:ownerProducer="USA">
        <tx:Severity>Informational</tx:Severity>
        <tx:ActionTaken>Accepted</tx:ActionTaken>
        <tx:SystemIdentifier>
          <nc:IdentificationID>212312312</nc:IdentificationID>
          <intel:SystemName>Screening_Agency_DB</intel:SystemName>
        </tx:SystemIdentifier>
      </tx:ResponseItem>
    </tx:BasicResponsePayload>
  </tx:Response>
</tx:ResponseList>
</S11:Body>
</S11:Envelope>
2.7 Error Response Envelope

2.7.1 Error Response Envelope Structure

![Diagram of Error Response Envelope Structure]

An MTS error message consists of a single ebMS `SignalMessage` element that is contained within the header of a SOAP envelope as shown in Figure 5. While `SignalMessage` elements can contain other child elements, such as receipts and pull requests, the MTS currently uses them to convey errors exclusively. Specifically, TWPDES 3.0 error messages can convey one or more errors (and associated `Error` elements) within a single `SignalMessage` element. The errors may have occurred at multiple levels across the SOAP processing and infrastructure stacks. Specifically, an MSH consists of several cooperative subsystems (ex: reliability, security), each of which could encounter an error during the processing of a request message. As such, MTS prescribes that these error messages be reserved for errors that occur outside the sending and receiving applications’ boundaries – such as communications and other infrastructure errors. Such errors are normally raised by the underlying service oriented framework, while application errors would be conveyed within one or more MTS `Response` elements raised by a business application.

Each ebMS `Error` element can optionally contain an origin attribute specifying the subsystem that originally encountered and raised the error condition. Additional `Error` element attributes can be used to categorize errors and specify their relative severities. For additional specifics, see the OASIS ebXML Messaging Services, Version 3.0: Part 1, Core Features document, Section 6 – Error Handling.
### 2.7.2 Error Response Envelope Elements and Attributes

Table 3 describes the elements and attributes that comprise the error response message. The “Source” column in the table indicates whether the element or attribute is populated by the sending business application or the sending agency’s underlying ebXML MSH.

<table>
<thead>
<tr>
<th>Element / Attribute</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:MessageInfo/eb:Timestamp</code></td>
<td>MSH</td>
<td>The timestamp for the message determined by the ebXML MSH when the actual message to be sent is created.</td>
</tr>
<tr>
<td><code>/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:MessageInfo/eb:MessageId</code></td>
<td>MSG</td>
<td>A unique message identifier generated by the ebXML MSH in response to the raising of one or more errors from mechanism(s) executing across the infrastructure (i.e. SOAP, security, reliable messaging) and/or application processing stack.</td>
</tr>
<tr>
<td><code>/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@category</code></td>
<td>App</td>
<td>OPTIONAL. Identifies the type of error related to a specific origin (e.g. content, packaging, unpackaging, communication).</td>
</tr>
<tr>
<td><code>/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@errorCode</code></td>
<td>App</td>
<td>REQUIRED. Contains the unique identifier for a specific error, specified in the format “EBMS:” followed by a unique error code. Blocks of codes that are not referenced by the ebXML message service specification can be used to convey application-specific errors.</td>
</tr>
<tr>
<td><code>/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@severity</code></td>
<td>App</td>
<td>REQUIRED. Indicates the severity of the error. A “warning” indicates a potentially disabling condition, while a “failure” indicates the inability of a service action to process a message.</td>
</tr>
<tr>
<td><code>/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@refToMessageInError</code></td>
<td>App</td>
<td>OPTIONAL. References the unique identifier of the message in error for which this error is being raised.</td>
</tr>
<tr>
<td>Element / Attribute</td>
<td>Source</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------</td>
<td>-------------</td>
</tr>
</tbody>
</table>

Table 3: Elements and Attributes for Error Response
### 2.7.3 Error Response Sample Message Content

The following XML fragment provides a sample of a TWPDES error response message.

```xml
<S11:Envelope xsi:schemaLocation="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/..../schemas/transaction/1.1/ebms-header-3_0-200704.xsd"
xmlns:eb="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/"
xmlns:S11="http://schemas.xmlsoap.org/soap/envelope/
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <S11:Header>
    <eb:Messaging S11:mustUnderstand="1">
      <eb:SignalMessage>
        <eb:MessageInfo>
          <eb:Timestamp>2001-12-17T09:30:47.0Z</eb:Timestamp>
          <eb:MessageId>UUID-2-receiving.agency.gov</eb:MessageId>
        </eb:MessageInfo>
        <eb:Error origin="ebMS" category="Content" errorCode="EBMS:0001" severity="failure"
          refToMessageInError="UUID-1-sending-agency.gov">
          <eb:Description xml:lang="en">Value not recognized</eb:Description>
        </eb:Error>
        <eb:Error origin="ebMS" category="Processing" errorCode="EBMS:0101" severity="failure"
          refToMessageInError="UUID-23-sending-agency.gov">
          <eb:Description xml:lang="en">Failed Authentication</eb:Description>
        </eb:Error>
        <eb:Error origin="receiving.application" category="Processing" errorCode="EBMS:2000" severity="failure"
          refToMessageInError="UUID-45-sending-agency.gov">
          <eb:Description xml:lang="en">Internal Error - Unable to Ingest Watchlist Entry #12345.</eb:Description>
        </eb:Error>
      </eb:SignalMessage>
    </eb:Messaging>
  </S11:Header>
</S11:Envelope>
```
3 Message Construction Using LEXS PD

3.1 High Level Message Structure

A TWPDES-LEXS message contains a single publish operation, *doPublish*, which is used to share information with a LEXS-enabled data consumer. The *doPublish* operation wraps a single *PublishMessage* in a *PublishMessageContainer*. The *PublishMessage* consists of a *PDMessageMetadata* that describes the message (e.g. message date/time, message receiver, etc.); a *DataSubmitterMetadata* that contains “From” information; and one or more *DataItemPackages*. Each *DataItemPackage* contains a *PackageMetadata* that describes the source (e.g. a person identification number, an encounter identification number, etc.) for which the data in the package belongs to; a *Digest* that contains the basic information (e.g. activity, location, person, etc.) from one or more TWPDES Payloads. The TWPDES Payloads within a *PublishMessage* could be a set of Watchlist Designation Requests, or a set of Encounter Nomination Requests, etc. Figure 6 shows the high-level components in a TWPDES-LEXS message.

![Figure 6 – High level view of a TWPDES-LEXS message](image)

See *LEXS 3.1 User Guide* for more detailed documentation on the structure of a LEXS PD message. See individual *TWPDES 3.0 Watchlist IEPD* and *TWPDES 3.0 Encounter IEPD* for detailed description of the TWPDES-LEXS Watchlist message and TWPDES-LEXS Encounter message, respectively.
### 3.2 LEXS Metadata Elements and Attributes

Table 4 describes the elements and attributes that comprise the LEXS structure.

<table>
<thead>
<tr>
<th>Element / Attribute</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMessageMetadata/ lexs:MessageSequenceNumber</td>
<td>App</td>
<td>REQUIRED. Positive integer. Uniquely identifies a message from a specific application or service provider. This element may be used to indicate the order of processing for PD messages, for auditing purposes, to track messages for troubleshooting, and to tie results to the originating request.</td>
</tr>
</tbody>
</table>
### 3.3 TWPDES-LEXS Sample Message Content

The following XML fragment provides a sample of a TWPDES-LEXS watchlist designation request message:

```xml
<?xml version="1.0" encoding="UTF-8"?>
<lexspd:doPublish xmlns:lexspd="http://usdoj.gov/leisp/lexs/publishdiscover/3.1"
    xmlns:nc="http://niem.gov/niem/niem-core/2.0" xmlns:s="http://niem.gov/niem/structures/2.0"
    xmlns:eb="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/"
    <lexs:PublishMessageContainer>
        <lexs:PublishMessage>
            <lexs:PDMessageMetadata>
                <lexs:LEXSVersion>3.1.2</lexs:LEXSVersion>
                <lexs:MessageDateTime>2007-07-12T22:50:22</lexs:MessageDateTime>
                <lexs:MessageSequenceNumber>1</lexs:MessageSequenceNumber>
                <lexs:DataSensitivity>U</lexs:DataSensitivity>
                <lexs:DomainAttribute>
                    <lexs:AttributeName>MessageID</lexs:AttributeName>
                    <lexs:AttributeValue>e7af10974858</lexs:AttributeValue>
                </lexs:DomainAttribute>
                <lexs:DomainAttribute>
                    <lexs:AttributeName>Domain</lexs:AttributeName>
                    <lexs:AttributeValue>TWPDES</lexs:AttributeValue>
                </lexs:DomainAttribute>
                <eb:PayloadInfo>
                    <eb:PartInfo>

```

Table 4: Elements and Attributes for LEXS Metadata

<table>
<thead>
<tr>
<th>Element / Attribute</th>
<th>Source</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lexs:DomainAttribute</td>
<td></td>
<td></td>
</tr>
<tr>
<td>lejsp:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:DataSubmitterMetadata/</td>
<td>App</td>
<td>REQUIRED. String. Specifies the organization name for the system where the message is being sent. For example, the name of the watchlisting agency.</td>
</tr>
</tbody>
</table>
<lexs:Domain>TWPDES</lexs:Domain>
<lexs:DataSubmitterMetadata><lexs:DataItemPackage><lexs:PackageMetadata>
<lexs:DataItemID>PersonID_for_JS</lexs:DataItemID>
<lexs:DataItemReferenceID>1</lexs:DataItemReferenceID>
<lexs:DataItemStatus>New</lexs:DataItemStatus>
<lexs:DataOwnerMetadata>
<lexs:DataOwnerIdentifier><lexs:ORI>Watchlisting_Agency</lexs:ORI><nc:OrganizationName>Watchlisting_Agency</nc:OrganizationName></lexs:DataOwnerIdentifier>
<lexs:DataOwnerContact><nc:PersonSurName/></lexs:DataOwnerContact>
</lexs:DataOwnerMetadata>
<lexs:DisseminationCriteria>white</lexs:DisseminationCriteria>
<lexs:DataItemCategory><lexs:DataItemCategoryText>Watchlist Designation</lexs:DataItemCategoryText></lexs:DataItemCategory>
</lexs:PackageMetadata>
<lexs:Digest><lexsdigest:EntityPerson s:id="DWLE_1">
<lexsdigest:Metadata s:id="PersonIDLink01">
<nc:SourceIDText>PersonID_for_JS</nc:SourceIDText>
<lexsdigest:LogicalIDText>2009433</lexsdigest:LogicalIDText>
</lexsdigest:Metadata>
<lexsdigest:Person s:metadata="PersonIDLink01">
<nc:PersonAlternateName s:id="DID201011">James</nc:PersonAlternateName>
<nc:PersonGivenName>James</nc:PersonGivenName>
<nc:PersonSurName>Smith</nc:PersonSurName>
<nc:PersonFullName>James Smith</nc:PersonFullName>
</nc:PersonBirthDate s:id="DID101010">1962-12-14</nc:PersonBirthDate>
<nc:PersonBirthDate/>
<nc:PersonPassportIdentification s:id="DID301011">
    <nc:IdentificationID>19929222</nc:IdentificationID>
</nc:PersonPassportIdentification>

... More EntityPerson constructs...
</lexsdigest:Persons>

... More SameAsPayloadAssociation constructs...
</lexsdigest:Associations>
</lexs:Digest>

<lexs:StructuredPayload s:id="Payload1">
    <lexs:StructuredPayloadMetadata>
        <lexs:CommunityURI>http://twpdes.gov/twpdes/watchlisting/2.0</lexs:CommunityURI>
        <lexs:CommunityVersion>TWPDES 3.0</lexs:CommunityVersion>
    </lexs:StructuredPayloadMetadata>
        ... TWPDES Watchlist Designation Payload ...
    </tx:Request>
</lexs:StructuredPayload>
## 4 Glossary

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ebMS</td>
<td>ebXML Messaging Service</td>
</tr>
<tr>
<td>ebXML</td>
<td>e-business XML</td>
</tr>
<tr>
<td>ESB</td>
<td>Enterprise Service Bus</td>
</tr>
<tr>
<td>HTTP</td>
<td>Hypertext Transfer Protocol</td>
</tr>
<tr>
<td>LEISP</td>
<td>Law Enforcement Information Sharing Program</td>
</tr>
<tr>
<td>LEXS</td>
<td>LEISP Exchange Specifications</td>
</tr>
<tr>
<td>LEXS PD</td>
<td>LESS Publish and Discovery</td>
</tr>
<tr>
<td>IEPD</td>
<td>Information Exchange Package Documentation</td>
</tr>
<tr>
<td>JMS</td>
<td>Java Messaging Service</td>
</tr>
<tr>
<td>MEP</td>
<td>Message Exchange Pattern</td>
</tr>
<tr>
<td>MIME</td>
<td>Multipurpose Internet Mail Extensions</td>
</tr>
<tr>
<td>MOM</td>
<td>Message Oriented Middleware</td>
</tr>
<tr>
<td>MSH</td>
<td>Message Service Handler</td>
</tr>
<tr>
<td>MTS</td>
<td>Message Transaction Specification</td>
</tr>
<tr>
<td>OASIS</td>
<td>Organization for the Advancement of Structured Information Standards</td>
</tr>
<tr>
<td>SMTP</td>
<td>Simple Mail Transfer Protocol</td>
</tr>
<tr>
<td>SOA</td>
<td>Service Oriented Architecture</td>
</tr>
<tr>
<td>SOAP</td>
<td>Simple Object Access Protocol</td>
</tr>
<tr>
<td>TWPDES</td>
<td>Terrorist Watchlist Person Data Exchange Standard</td>
</tr>
<tr>
<td>UTF-8</td>
<td>8-bit UCS/Unicode Transformation Format</td>
</tr>
<tr>
<td>W3C</td>
<td>World Wide Web Consortium</td>
</tr>
<tr>
<td>WL</td>
<td>Watchlist</td>
</tr>
<tr>
<td>WS</td>
<td>Web Services</td>
</tr>
<tr>
<td>Acronym</td>
<td>Definition</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------</td>
</tr>
<tr>
<td>XML</td>
<td>eXtensible Markup Language</td>
</tr>
</tbody>
</table>
5 References

**ebXML/ebMS – Electronic Business Extensible Markup Language / ebXML Message Service**

The ebXML Message Service (ebMS), developed by the Organization for the Advancement of Structured Information Standards (OASIS), defines the message envelope schema used to exchange ebXML messages over communications protocols, such as HTTP and SMTP, or over message-oriented middleware (MOM) products, such as IBM’s Websphere MQ. ebMS is defined as a set of layered extensions to the base Simple Object Access Protocol (SOAP) and SOAP Messages with Attachments specifications. While MOM-specific features and expanded capabilities can be leveraged to provide guaranteed message delivery and security, additional web service standards, such as WS-ReliableMessaging, WS-Addressing, and WS-Security can be used as well.

ebXML Specification documents may be found at:

WS Reliability 1.1 documents may be found at:

WS Security: SOAP Message Security documents may be found at:

ebXML IIC Technical Committee web site may be found at
http://www.oasis-open.org/committees/ebxml-iic/

**SOAP – Simple Object Access Protocol**

SOAP is an XML-based protocol for exchange of information in a decentralized, distributed environment.

SOAP documents may be found at:
http://www.w3.org/TR/SOAP/

SOAP with Attachments documents may be found at:
http://www.w3.org/TR/SOAP-attachments

MIME (Multipurpose Internet Mail Extensions) documents may be found at http://www.ietf.org/rfc/rfc2045.txt

Internet Message Format documents may be found at:
http://www.ietf.org/rfc/rfc2822.txt

IETF RFC 2392: Content-Id (cid:) and Message-Id (mid:) Uniform Resource Locators scheme document may be found at:
http://www.ietf.org/rfc/rfc2392.txt
LEXS Documentation

http://www.it.ojp.gov/jsr/common/viewDetail.jsp?sub_id=256&view=yes&keyword=1