

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

Revision	Revised By	Date	Summary of Change
3.0 R0C0	C. J. Lee	12/4/2009	<ul style="list-style-type: none">• Removed “Sensitive But Unclassified // For Official Use Only” markings in the document.• Removed references to “Controlled Unclassified Information”.• Updated Version 1.2b to Version 3.0 to mark the changes.

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.

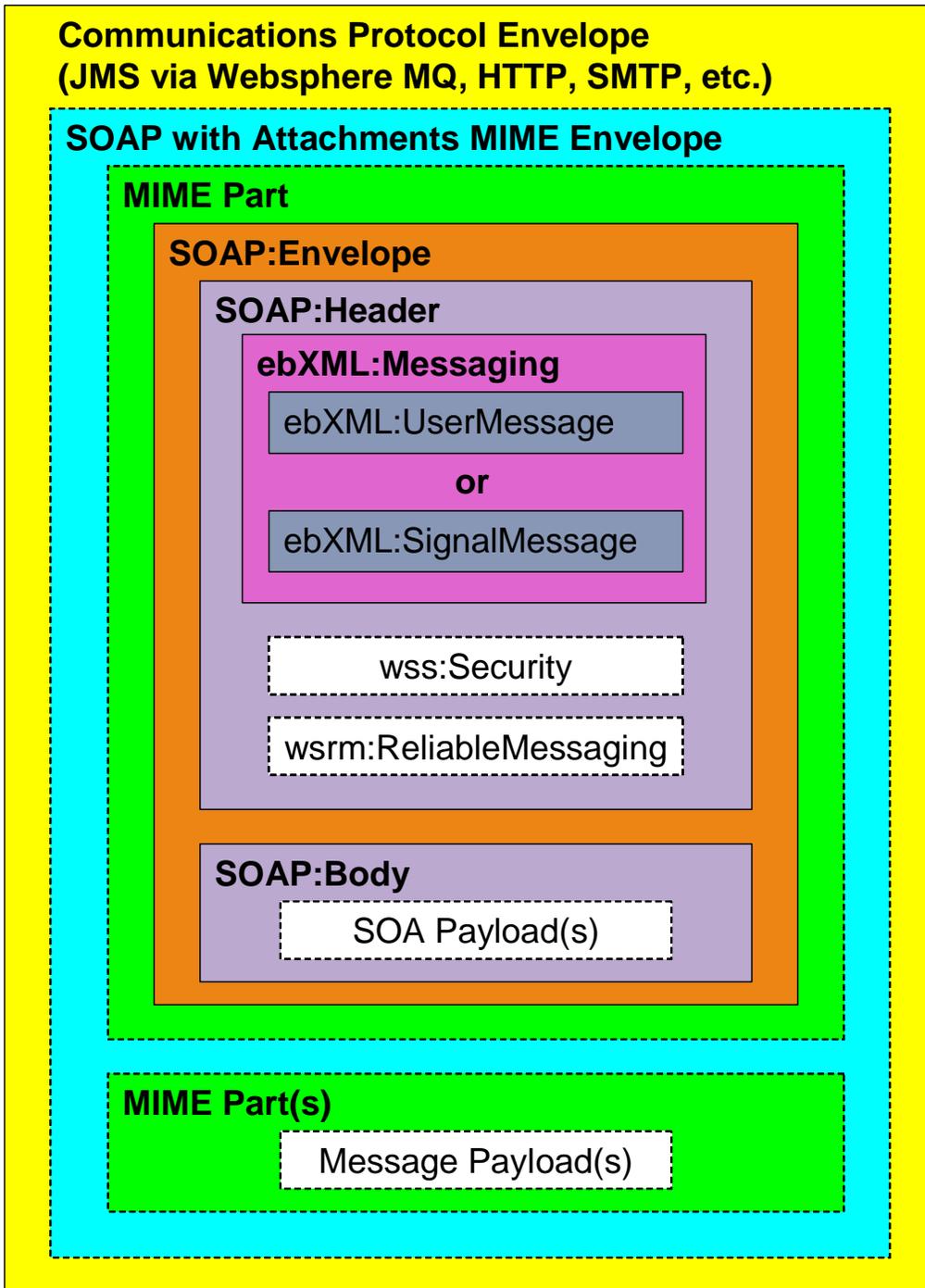


Figure 1 – Canonical ebXML Message Structure

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.

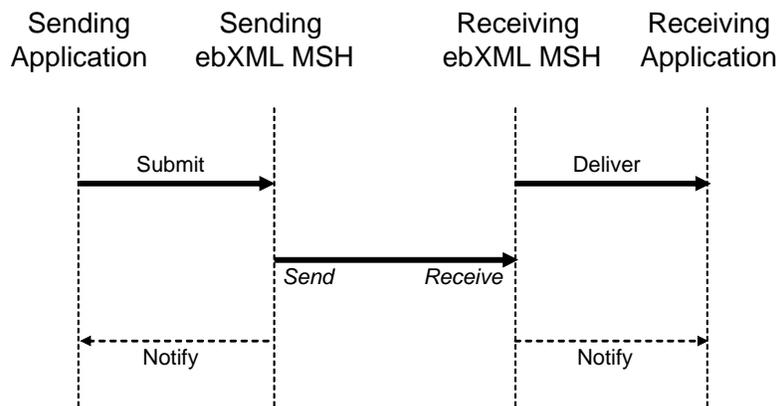


Figure 2 – Nominal ebXML Messaging Model

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 *WLDesignationIngestService* (hosted by the *WLDesignationRecipient* system) to ingest the watchlist payload(s) using the *IngestWLDesignation* action (or service endpoint). The *WLDesignationRecipient* would then reply with one or more response elements, by asking a *WLDesignationResponseIngestService* (hosted by the *WLDesignationSender* system) to ingest the watchlist response payload(s) using the *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.

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 **REQUIRES** the SOAP message to have an XML Prolog, and it **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

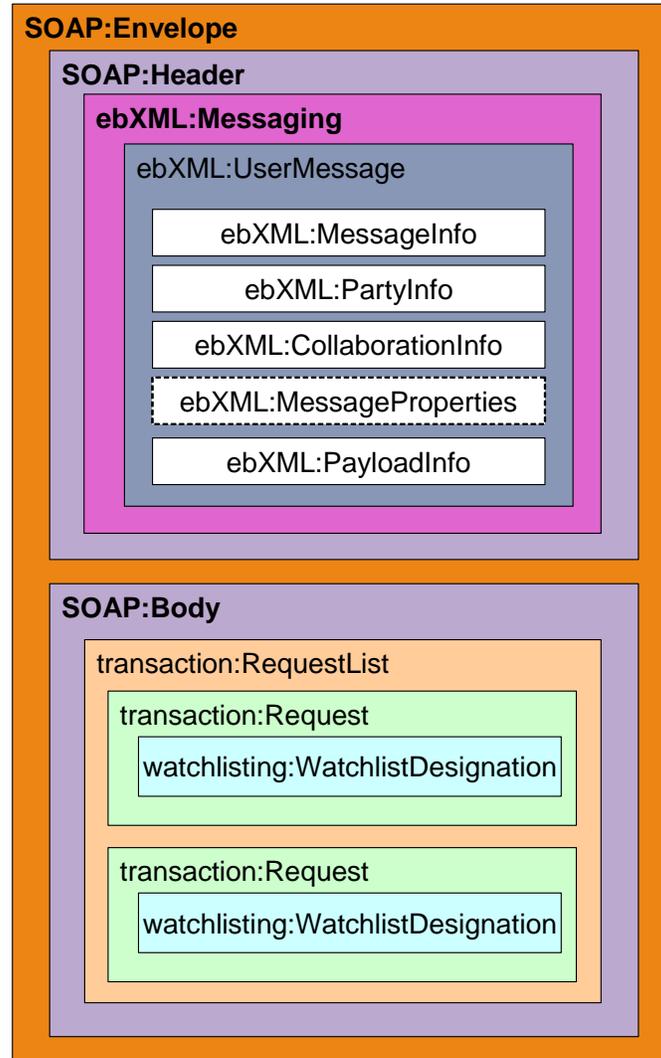


Figure 3 – Watchlist Designation Request Envelope Structure

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.

Element / Attribute	Source	Description
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:MessageInfo/eb:Timestamp	MSH	The timestamp for the message determined by the ebXML MSH when the actual message to be sent is created.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:MessageInfo/eb:MessageId	App	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.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PartyInfo/eb:From/eb:PartyId	App	Identification of the message sender, expressed as a Uniform Resource Identifier (URI).
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PartyInfo/eb:From/eb:Role	App	A description of the role this party assumes with respect to a particular conversation or exchange of messages with a target party.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PartyInfo/eb:To/eb:PartyId	App	Identification of the message receiver, expressed as a URI.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PartyInfo/eb:To/eb:Role	App	A description of the role this party assumes with respect to a particular conversation or exchange of messages with a source party.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:CollaborationInfo/eb:Service	MSH	Identification of the message processing service on the receiver’s end of the conversation.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:CollaborationInfo/eb:Action	App	Identification of the specific, desired action being requested of the message processing service on the receiver’s end of the conversation.

Element / Attribute	Source	Description
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:CollaborationInfo/eb:ConversationId	MSH	Identification of a series of messages (ex: requests, responses), which comprise a conversation (i.e. logically-related exchange of messages) between two parties.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PayloadInfo/eb:PartInfo@href	MSH	References the identifier (i.e. “#idref”) of the SOAP Body element which contains the SOA payload (ex: <i>WatchlistDesignation</i> , transaction <i>Response</i> , encounter/disposition).
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PayloadInfo/eb:PartInfo /eb:Schema	App	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.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PayloadInfo/eb:PartInfo /eb:Description	App	OPTIONAL. A textual description of the referenced payload.
/S11:Envelope/S11:Header/eb:Messaging/eb:User Message/eb:PayloadInfo/eb:PartInfo /eb:PartProperties/eb:Property	App	OPTIONAL. One or more name/value pairs of information that is relevant/useful to – or required by – the desired action of the targeted service.
Payload		
/S11:Envelope/S11:Body/tx:RequestList	App	Contains zero or more requests for service.
/S11:Envelope/S11:Body/tx:RequestList /tx:Request	App	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 <i>wl:WatchlistDesignation</i> element is an example of an application request payload .

Element / Attribute	Source	Description
/S11:Envelope/S11:Body/tx:RequestList /tx:Request/tx:RequestID	App	Identifies the request, and is unique to the sending entity (agency).
/S11:Envelope/S11:Body/tx:RequestList /tx:Request/wl:WatchlistDesignation	App	A single watchlist designation request, containing potentially multiple dispositions, handling codes, and watchlist entry directives.

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 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/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:UserMessage>
        <eb:MessageInfo>
          <eb:Timestamp>2007-07-12T22:50:22</eb:Timestamp>
          <eb:MessageId>16215d60-30ed-11dc-8d58-e7af10974858</eb:MessageId>
        </eb:MessageInfo>
        <eb:PartyInfo>
          <eb:From>
            <eb:PartyId>Watchlisting_Agency</eb:PartyId>
          </eb:From>
          <eb:Role>http://nim.gov/niem/domains/watchlisting/roles/WLDesignationSender</eb:Role>
        </eb:PartyInfo>
        <eb:To>
          <eb:PartyId>Screening_Agency</eb:PartyId>
        </eb:To>
        <eb:Role>http://nim.gov/niem/domains/watchlisting/roles/WLDesignationRecipient</eb:Role>
      </eb:UserMessage>
      <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:Messaging>
  </S11:Header>
</S11:Envelope>
```

```

                <eb:PartProperties>
                    <eb:Property name="MimeType">application/xml</eb:Property>
                </eb:PartProperties>
            </eb:PartInfo>
        </eb:PayloadInfo>
    </eb:UserMessage>
</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"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ism="urn:us:gov:ic:ism:v2"
xmlns:nc="http://niem.gov/niem/niem-core/2.0" xmlns:intel="http://niem.gov/niem/domains/intelligence/2.0">
        <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"
ism:ownerProducer="USA" ism:classification="U" xmlns:wl="urn:us:gov:ic:wl" xmlns:ism="urn:us:gov:ic:ism:v2"
xmlns:intel="http://niem.gov/niem/domains/intelligence/2.0" xmlns:s="http://niem.gov/niem/structures/2.0"
xmlns:iceoi="urn:us:gov:ic:iceoi" 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

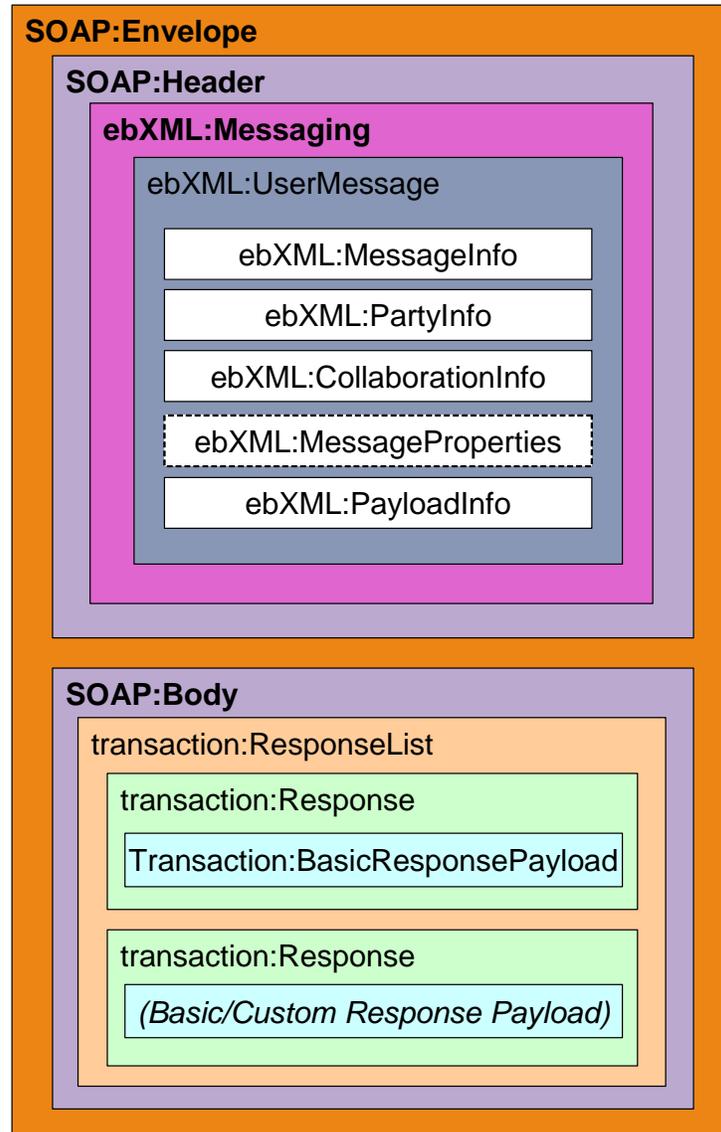


Figure 4 – Watchlist Designation Response Envelope Structure

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.

Element / Attribute	Source	Description
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:MessageInfo/eb:Timestamp	MSH	The timestamp for the message determined by the ebXML MSH when the actual message to be sent is created.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:MessageInfo/eb:MessageId	App	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.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:MessageInfo/eb:RefToMessageId	App	Used only in response messages, this attribute references the unique identifier of the original request message to which this message is responding.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PartyInfo/eb:From/eb:PartyId	App	Identification of the message sender, expressed as a URI.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PartyInfo/eb:From/eb:Role	App	A description of the role this party assumes with respect to a particular conversation or exchange of messages with a target party.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PartyInfo/eb:To/eb:PartyId	App	Identification of the message receiver, expressed as a URI.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PartyInfo/eb:To/eb:Role	App	A description of the role this party assumes with respect to a particular conversation or exchange of messages with a source party.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:CollaborationInfo/eb:Service	MSH	Identification of the message processing service on the receiver’s end of the conversation.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage	Msg	Identification of the specific, desired action being requested of the

Element / Attribute	Source	Description
message/eb:CollaborationInfo/eb:Action		message processing service on the receiver's end of the conversation.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:CollaborationInfo/eb:ConversationId	MSH	Identification of a series of messages (ex: requests, responses), which comprise a conversation (i.e. logically-related exchange of messages) between two parties.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:MessageProperties/Property	App	OPTIONAL. One or more name/value pairs of information that is relevant/useful to – or required by – the desired action of the targeted service.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PayloadInfo/eb:PartInfo@href	MSH	References the identifier (i.e. “#idref”)of the SOAP Body element which contains the SOA payload (ex: watchlist designation/nomination, encounter/disposition).
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PayloadInfo/eb:PartInfo /eb:Schema	Msg	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.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PayloadInfo/eb:PartInfo /eb:Description	App	OPTIONAL A textual description of the referenced payload.
/S11:Envelope/S11:Header/eb:Messaging/eb:UserMessage/eb:PayloadInfo/eb:PartInfo /eb:PartProperties/Property	App	OPTIONAL. One or more name/value pairs of information that is relevant/useful to – or required by – the desired action of the targeted service.
Payload		
/S11:Envelope/S11:Body/tx:ResponseList	App	Contains zero or more responses to prior requests for service.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response	App	Contains the response to a prior service request. <i>Response</i> is an MTS element that can be extended to provide custom response structures for applications by embedding a custom response payload. <i>BasicResponsePayload</i> is an MTS element that provides a simple,

Element / Attribute	Source	Description
		predefined response structure for convenience.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response/tx:ReferencedRequestID	App	Contains the unique identifier of the request to which this response refers.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ActionTaken	App	An enumerated value that codifies the summarizing action taken by the service in processing a prior request.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem	App	Contains service response information regarding the processing of a prior request.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/tx:Severity	App	An enumerated value (i.e. <i>Informational</i> , <i>Warning</i> , or <i>Error</i>) that codifies the severity of the response.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/ActionTaken	App	An enumerated value (i.e. <i>Rejected</i> , <i>Proceeded-With-Errors</i> , <i>Accepted</i>) that codifies the action taken by the service in processing a prior request.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/tx:ResponseCode	App	An enumerated code that further indicates the action taken by the service in processing a prior request.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/tx:SystemIdentifier	App	A structure that identifies an entity within a targeted system of record.

Element / Attribute	Source	Description
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/tx:SystemIdentifier/nc:IdentificationID	App	References the (presumably) unique identifier of an entity that resides in a targeted system of record.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/tx:SystemIdentifier/intel:SystemName	App	References the name of the targeted system of record that hosts the relevant entity and scopes its (presumably) unique identifier.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/tx:Note	App	A convenience structure that currently holds a textual element, but could be expanded to include additional information.
/S11:Envelope/S11:Body/tx:ResponseList /tx:Response /tx:BasicResponsePayload/tx:ResponseItem/tx:Note/ tx:Text	App	Contains a textual description of the response to a prior service request.

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 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/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: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:PartyInfo>
        <eb:CollaborationInfo>
          <eb:Service type="Receiver">WLDesignationResponseIngestService</eb:Service>
          <eb:Action>IngestWLDesignationResponse</eb:Action>
          <eb:ConversationId>4321</eb:ConversationId>
        </eb:CollaborationInfo>
        <eb:MessageProperties>
          <eb:Property name="ServiceActionTaken">Accepted</eb:Property>
        </eb:MessageProperties>
        <eb:PayloadInfo>
          <eb:PartInfo href="IDWLDLDR200">
```

```

        <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:PartInfo>
    </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"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ism="urn:us:gov:ic:ism:v2"
xmlns:nc="http://niem.gov/niem/niem-core/2.0" xmlns:intel="http://niem.gov/niem/domains/intelligence/2.0">
        <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

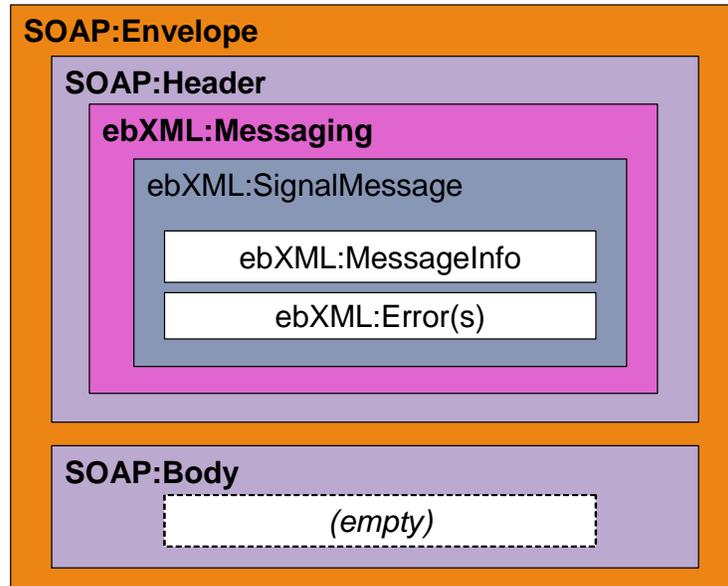


Figure 5 – Error Response Envelope Structure

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.

Element / Attribute	Source	Description
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:MessageInfo/eb:Timestamp	MSH	The timestamp for the message determined by the ebXML MSH when the actual message to be sent is created.
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:MessageInfo/eb:MessageId	MSG	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.
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@origin	App	OPTIONAL. Identifies the functional module within which the error occurred (ex: “ebMS”, “reliability”, “security”, “application”).
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@category	App	OPTIONAL. Identifies the type of error related to a specific origin (e.g. content, packaging, unpackaging, communication).
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@errorCode	App	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.
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@severity	App	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.
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error@refToMessageInError	App	OPTIONAL. References the unique identifier of the message in error for which this error is being raised.

Element / Attribute	Source	Description
/S11:Envelope/S11:Header/eb:Messaging/eb:SignalMessage/eb:Error/eb:Description	Msg	OPTIONAL. Provides a narrative description of the error in the language defined by the <i>xml:lang</i> attribute.

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.

```
<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:Body/>
</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.

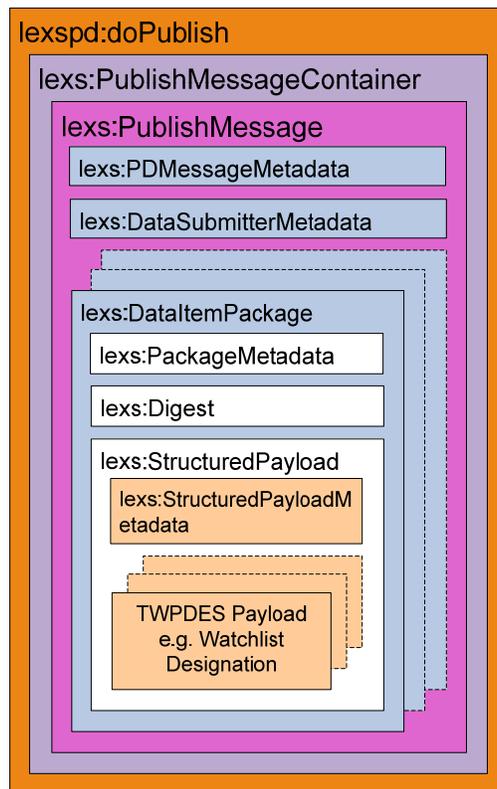


Figure 6 – High level view of a TWPDES-LEXS message

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.

Element / Attribute	Source	Description
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMMessageMetadata/ lexs:LEXSVersion	App	REQUIRED. String. Version of LEXS used to generate this request or response.
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMMessageMetadata/ lexs:MessageDateTime	App	REQUIRED. DateTime. Date and time the message was created. Mapped from eb:Messaging/ eb:UserMessage/ eb:MessageInfo/ eb:Timestamp.
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMMessageMetadata/ lexs:MessageSequenceNumber	App	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.
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMMessageMetadata/ lexs:DataSensitivity	App	REQUIRED. String. Information security classification level. Mapped from ism:classification.
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMMessageMetadata/ lexs:DomainAttribute	App	OPTIONAL: ebMS MessageID and PayloadInfo.
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMMessageMetadata/ lexs:DomainAttribute	App	OPTIONAL: ebMS CollaborationInfo.
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:PDMMessageMetadata/	App	OPTIONAL: ebMS PartyInfoTo.

Element / Attribute	Source	Description
lexs:DomainAttribute		
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:DataSubmitterMetadata/	App	REQUIRED. String. Specifies the organization name for the system where the message is being sent. For example, the name of the watchlisting agency.
lexspd:doPublish/ lexs:PublishMessageContainer/ lexs:PublishMessage/ lexs:DataSubmitterMetadata/	App	OPTIONAL: ebMS Role.

Table 4: Elements and Attributes for LEXS Metadata

3.3 TWPDES-LEXS Sample Message Content

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

```
<?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:lexs="http://usdoj.gov/leisp/lexs/3.1" xmlns:lexslib="http://usdoj.gov/leisp/lexs/library/3.1"
xmlns:lexsdigest="http://usdoj.gov/leisp/lexs/digest/3.1" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:eb="http://docs.oasis-open.org/ebxml-msg/ebms/v3.0/ns/core/200704/"
xsi:schemaLocation="http://usdoj.gov/leisp/lexs/publishdiscover/3.1 ../schemas/LEXS/publish-
discover/3.1/publish-discover.xsd">
  <lexs:PublishMessageContainer>
    <lexs:PublishMessage>
      <lexs:PDMMessageMetadata>
        <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>16215d60-30ed-11dc-8d58-
e7af10974858</lexs:AttributeValue>
          <lexs:Domain>TWPDES</lexs:Domain>
          <eb:PayloadInfo>
            <eb:PartInfo>
```

```

                                <eb:Schema
location="../schemas/twpdes/watchlisting/2.0/watchlisting.xsd" version="TWPDES 3.0"
namespace="http://twpdes.gov/twpdes/watchlisting/2.0/watchlisting.xsd"/>
                                <eb:Description xml:lang="en-us">Watchlist
Designation</eb:Description>
                                <eb:PartProperties>
                                    <eb:Property name="MimeType">application/xml</eb:Property>
                                </eb:PartProperties>
                                </eb:PartInfo>
                                </eb:PayloadInfo>
</lexs:DomainAttribute>
<lexs:DomainAttribute>
    <lexs:AttributeName>CollaborationInfo</lexs:AttributeName>
    <lexs:Domain>TWPDES</lexs:Domain>
    <eb:CollaborationInfo>
        <eb:Service type="Reciever">WLDesignationIngestService</eb:Service>
        <eb:Action>IngestWLDesignation</eb:Action>
        <eb:ConversationId>4321</eb:ConversationId>
    </eb:CollaborationInfo>
</lexs:DomainAttribute>
<lexs:DomainAttribute>
    <lexs:AttributeName>PartyInfoTo</lexs:AttributeName>
    <lexs:Domain>TWPDES</lexs:Domain>
    <eb:To>
        <eb:PartyId>Screening_Agency</eb:PartyId>

<eb:Role>http://nim.gov/nim/domains/watchlisting/roles/WLDesignationRecipient</eb:Role>
    </eb:To>
</lexs:DomainAttribute>
</lexs:PDMessageMetadata>
<lexs:DataSubmitterMetadata>
    <lexs:SystemIdentifier>
        <nc:OrganizationName>Watchlisting_Agency</nc:OrganizationName>
    </lexs:SystemIdentifier>
</lexs:SystemIdentifier>
<lexs:SystemContact>
    <nc:PersonSurName/>
</lexs:SystemContact>
</lexs:DomainAttribute>
    <lexs:AttributeName>Role</lexs:AttributeName>

```

```

    <lexs:AttributeValue>http://nim.gov/niem/domains/watchlisting/roles/WLDesignationSender</lexs:Attribute
teValue>
        <lexs:Domain>TWPDES</lexs:Domain>
    </lexs:DomainAttribute>
</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:SystemID/>
            </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">
                <nc:PersonGivenName>James</nc:PersonGivenName>
                <nc:PersonSurName>Smith</nc:PersonSurName>
                <nc:PersonFullName>James Smith</nc:PersonFullName>
            </nc:PersonAlternateName>
            <nc:PersonBirthDate s:id="DID101010">
                <nc>Date>1962-12-14</nc>Date>

```

```

        </nc:PersonBirthDate>
        <nc:PersonPassportIdentification s:id="DID301011">
            <nc:IdentificationID>19929222</nc:IdentificationID>
            <nc:IdentificationJurisdictionFIPS10-
4Code>US</nc:IdentificationJurisdictionFIPS10-4Code>
        </nc:PersonPassportIdentification>
    </lexsdigest:Person>
</lexsdigest:EntityPerson>
... More EntityPerson constructs...
<lexsdigest:Associations>
    <lexslib:SameAsPayloadAssociation>
        <lexslib:ObjectReference s:ref="DWLE_1"/>
        <lexslib:PayloadObjectReference lexslib:ref="WLE_1"/>
        <lexslib:PayloadReference s:ref="Payload1"/>
    </lexslib:SameAsPayloadAssociation>
    ... 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>
        <tx:Request ism:ownerProducer="USA" ism:classification="U" s:id="IDREQ190"
xmlns:ism="urn:us:gov:ic:ism:v2" xmlns:tx="http://twpdes.gov/transaction/1.1">
            ... TWPDES Watchlist Designation Payload ...
        </tx:Request>
    </lexs:StructuredPayload>
</lexs:DataItemPackage>
</lexs:PublishMessage>
</lexs:PublishMessageContainer>
</lexspd:doPublish>

```

4 Glossary

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

Acronym	Definition
XML	eXtensible Markup Language

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:

http://www.oasis-open.org/committees/documents.php?wg_abbrev=ebxml-msg

WS Reliability 1.1 documents may be found at:

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsm

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

http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss

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

LEXS 3.1 User Guide, revision 7, August 7, 2008.

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