Document filename: ITK 2.2 TMS Transport Requirements			
Directorate / Programme :	HSCIC - Architecture	**Project**	Interoperability
Document Reference :		HSCIC-ITK-ARCH-106-1	
Project Manager :	Keith Naylor	**Status :**	Final
Owner :	George Hope	**Document Version :**	1.0 Final
Author :	George Hope	**Version issue date :**	01/11/2015

ITK2.2 TMS Transport Requirements

Document Management

Revision History

Version	Date	Summary of Changes
1.0	November 2015	First version of ITK version 2.2

Reviewers

This document was reviewed by the following people:

Reviewer name	Title / Responsibility	Date	Version
George Hope	ITK Architecture Lead	November 2015	1.0
Richard Kavanagh	ITK Messaging Lead	November 2015	1.0
Richard Dobson	ITK Accreditation Manager	November 2015	1.0
Nigel Saville	ITK Accreditation	November 2015	1.0

Approved by

This document was approved by the following people:

Name	Signature	Title	Date	Version
Shaun Fletcher		Head of Architecture	November 2015	1.0

Reference Documents

Ref no	Doc Reference Number	Title	Version
1.	2087 EIS11.6--Part 2--MHS.doc	External Interface Specification: Part 2 MHS	
2.			
3.			
4.			

Document Control:

The controlled copy of this document is maintained in the HSCIC corporate network. Any copies of this document held outside of that area, in whatever format (e.g. paper, email attachment), are considered to have passed out of control and should be checked for currency and validity.

Contents

1 Introduction

This document forms part of the overall document set for ITK Architecture.

1.1 Purpose of Document

This document defines a set of requirements for ITK TMS Transport Accreditation.

1.2 ITK Architecture Documentation Set

The position of this document in relation to the document set is shown below.

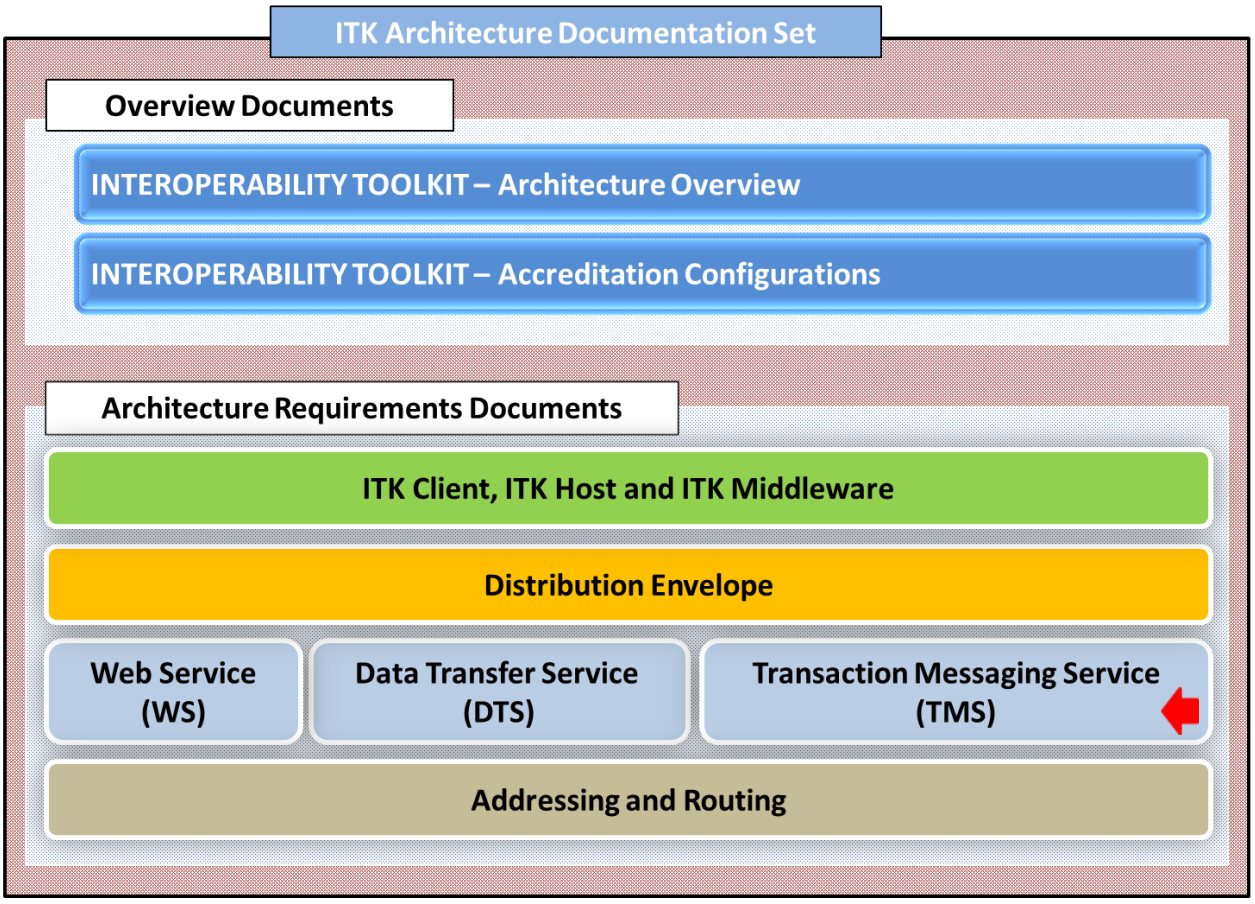


Figure 1 – ITK Architecture Documentation Set

1.3 Audience

The primary audience are supplier technical and product development staff who are interested in developing a Toolkit Implementation.

1.4 Document Scope

The document covers the ITK TMS Transport Interoperability requirements required for accreditation.

1.5 Document Overview

The rest of this document covers a number of areas of functionality. Within each area the functionality is described, and a number of formal requirements are listed in bold type, with additional detail provided in smaller type below this.

1.6 Requirements Presentation

The requirements are presented in the format given below:

Ref (1)	Description (2)	Client (3)	Host (4)	MW (5)	SMSP (6)
COR-REL-03	**Toolkit Implementations MUST retain responsibility for processing until a request completes**	**Y**	**N**	**Y**	**N**
NB (7)	Specifically, any response returned from the initial part of the asynchronous invocation does NOT indicate a transfer of responsibility. It is only a transport acknowledgement, and it does NOT imply that the message has necessarily been persisted, nor does it indicate a transfer of responsibility, nor promise that subsequent application processing will be completed.				

Clarification Notes

(1) The requirement reference

(2) The Description of the requirement

(3), (4), (5) and **(6)** Shows the requirements applicability for accreditation

(7) Provides further details relating to the requirement and supplementary notes

1.7 Reference Implementation

A reference implementation is available as a training aid and contains for example, code samples for typical Healthcare Interoperability scenarios.

http://developer.nhs.uk/library/interoperability/nhs-interoperability-framework/

1.8 Reference Code

Reference code for integrating with TMS is available for both Java and Microsoft .NET.

http://developer.nhs.uk/downloads-data/

2 Overview of TMS use in ITK

2.1 TMS

The Transaction Messaging Service is a central message transfer service to allow messages from users of NHS SPINE Connected computer systems and services to be securely routed (nationally) to the service they are requesting and to manage the response to that request.

Predominantly TMS is used to access central services. Depending on the type of message (e.g. the Personal Demographics Service), TMS automatically identifies where the message needs to be sent.

TMS can route messages between any connected systems. The ITK TMS specification is exploiting its any-to-any messaging capability, using a single message interaction called the ITK Trunk message. Note that there are many message types categorised within the MiM (HSCIC Message Implementation Manual).

TMS in its role as a central messaging hub supports a number of message handling behaviours. For example, there is a message handling behaviour which simply forwards to an endpoint, without TMS taking responsibility for delivery, whereas a different behaviour will result in TMS assuming responsibility for message delivery. Handling behaviour selection happens in TMS automatically and maps to the interaction being used. TMS Message handling behaviours is fully detailed within the EIS Document (Part II).

Note:

Although this document uses the example of the ITK Trunk Interaction (using the urn:nhs:names:services:itk service and the COPC_IN000001GB01 action identifiers), the specification equally applies to other ‘action’ identifiers using the same the urn:nhs:names:services:itk service.

2.2 ITK TMS Transport Overview

As mentioned each SPINE message has a unique interaction identifier. The ITK Trunk Message identifier is: `<eb:Service>urn:nhs:names:services:itk` and `<eb:Action>COPC_IN000001GB01`.

Typically ITK messages use “End-Party Reliability” pattern as defined in the EIS document.

After sending the initial message, TMS as a message forwarder will respond with HTTP 202 accepted status code. The final spine endpoint responds with HTTP 200 because in terms of the ITK message, processing is successful.

This simple example is given below to show what happens when a Business Acknowledgement is requested (as defined in the Domain Message Specification).

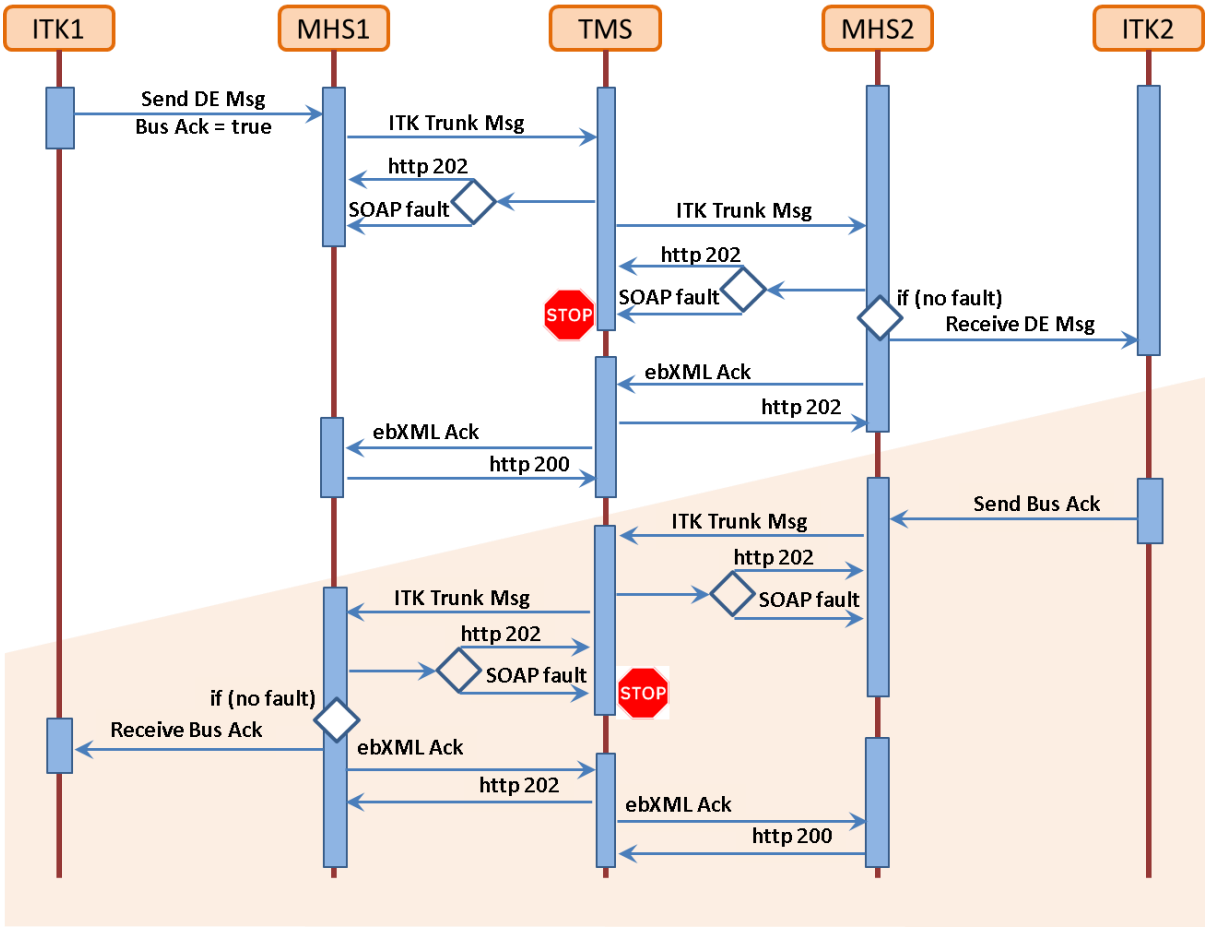


Figure 2 - ITK TMS Messaging and Business Acknowledgements

(Note: Infrastructure Acknowledgements are also handled in the same way)

Reliability features such as message de-duplication are the responsibility of the sending and receiving endpoints.

2.3 Constructing the ITK TMS Message

When a ITK message is to be sent over TMS, it requires the following information to construct the message.

Data	Source
TMS service URL	Published in the “Authority Service Names” document.
ITK CPAid	SDS (from nhsMHS object).
Remote ITK endpoint ASID	SDS (from nhsAS object) (obtained from SPINE LDAP Lookup or query local cache)
Remote ITK endpoint Party Key	SDS (obtained from SPINE LDAP Lookup or query local cache)
Local endpoint ASID	Local configuration, established during the endpoint registration process required for accreditation.
Local endpoint Party Key	Local configuration, established during endpoint registration.

Table 1: Data Sources to construct ITK TMS Message

When a message is successfully posted, TMS will respond with HTTP 202 over an empty response body. The maximum size of the complete ITK message must be within the limit defined in the latest EIS Specifications.

2.4 Processing the ITK TMS Message

On acceptance of the TMS transmission, the receiving MHS extracts and decompresses the distribution envelope attachment and passes on to the ITK DE processor.

The Distribution Envelope `<itk:handlingSpecification>` element is checked to find the message configurations. Please refer to the Distribution Envelope Requirements documents for details.

- If a DE receiver, receives the message on behalf of a recipient and infra ack is set to true will return either ITK infrastructure ACK (if it accepts the message), or NACK (if it rejects).

In every case, any kind of acknowledgement returned is routed to the address specified in the sender address element of the received Distribution Envelope.

3 The ITK TMS Accreditation Requirements

The ITK TMS Accreditation requirements associated with the above are:

3.1 TMS Message Construct Requirements

Ref	Description	Client	Host	MW	SMSP
TMS-MSG-01	**Every ITK TMS Message MUST be identified by eb:Service and eb:Action elements**	**Y**	**N**	**N**	**N**
NB	MHS Sender must use the following values `eb:Service="urn:nhs:names:services:itk"` and `eb:Action="COPC_IN000001GB01".` Depending on this pair of values, TMS will Authorise and Authenticate the pair of MHS to interact. Section 2.5.3 in EIS.				

TMS-MSG-02	**Every ITK TMS Message MUST use eb:CPAId and [eb:PartyId + ASID] pair for message addressing**	**Y**	**N**	**N**	**N**
1	ITK TMS Accredited Systems MUST be able to resolve ODS address into PartyId by looking up into SDS using LDAP query onto nhsMHS object.				
2	Alternatively, the systems SHOULD cache the data locally.				
NB	• ASID (Accredited System Identifier) which is declared by OID `"1.2.826.0.1285.0.2.0.107"` • HL7 elements `hl7:communicationFunctionRcv and hl7:communicationFunctionSnd` must be populated • Synchronisation of local cache with SDS Data is out of scope of this requirement.				

TMS-MSG-05	**MHS Sender and Receiver SHOULD use same TMS Service URL to send messages and acknowledgements**	**Y**	**Y**	**Y**	**N**

TMS-MSG-06	**ITK TMS Accredited Systems MUST ensure that all ITK Messages remain within the maximum allowable total size**	**Y**	**N**	**N**	**N**
1	The systems MUST ensure that the total number of MIME parts does not exceed allowable limit.				
2	The systems MUST ensure that the order of the MIME parts is in sync with the eb:Manifest section.				
NB	External Interface Specifications (Sec 2.5) may be referred for current values and related details.				

TMS-MSG-07	ITK TMS Accredited Systems SHOULD compress the Distribution Envelope using GZIP Algorithm and send as a raw binary octet stream	Y	N	N	N
1	If DE is compressed, MUST use GZIP Algorithm.				
NB	It is allowable to compress the Distribution Envelope even if it is encrypted.				

TMS-MSG-08	ITK TMS Accredited Systems MUST use SOAP 1.1 standards and UTF-8 Encoding	Y	Y	Y	N
1	The systems MUST use document/literal style of SOAP Message				

TMS-MSG-09	ITK TMS Messages MUST use eb:Manifest to reference its MIME Parts	Y	N	N	N
NB	The manifest contains references about the following MIME Parts.				

TMS-MSG-10	MHS Sender of ITK TMS Accredited Systems MAY set hl7:versionCode. If version code is present the MHS Receiver SHOULD populate hl7:versionCode in its response	Y	Y	N	N
NB	The *versionCode* is asserted by the sender to inform the receiver of the version of the MIM used to generate the message.				

TMS-MSG-11	Every ITK TMS Message sent from a MHS MUST receive an ebXML Standard Acknowledgement identified by eb:Service and eb:Action elements	N	Y	N	N
1	A MHS Sender MUST regard a transmission as incomplete until an ebXML acknowledgement has been received and processed because it is End Party Reliable Messaging System (aka Forward Express).				
NB	MHS Receiver must use the following values `eb:Service="urn:oasis:names:tc:ebxml-msg:service"` and `eb:Action="Acknowledgment"`				

3.2 ITK TMS Message Invocation

Ref	Description	Client	Host	MW	SMSP
TMS-INV-01	Every MHS (Sender and Receiver) MUST use HTTP 1.1 standards for ITK Messages	Y	Y	Y	N
1	When in Sender mode each MHS MUST use http POST				
2	When in Sender mode each MHS MUST use Asynchronous Invocation				

3.3 ITK TMS Message Error Handling

Ref	Description	Client	Host	MW	SMSP
TMS-ERR-01	**Every MHS Sender MUST be able to handle all http errors and status codes received**	**Y**	**Y**	**Y**	**N**
NB	MHS Sender MUST implement appropriate behaviour for all http response status code received other than 2xx series				

TMS-ERR-02	**ITK TMS Accredited Systems MUST be capable of Exception Ownership**	**Y**	**Y**	**Y**	**N**
1	If the MHS Receiver accepts the ebXML message as a TMS message, the receiving MHS MUST return an ebXML acknowledgment over TMS, to the sender. (Refer figure 1)				
2	If the MHS Receiver finds some error in processing the message from TMS it SHOULD return an ebXML Message Error				
3	MHS Sender MUST support receipt of a SOAP fault in a response				
NB	• Under normal circumstances, ITK Accredited Systems will send only http status code in its synchronous response. As defined in the EIS specifications • Any errors encountered in DE processing – for example authorisation errors due to checking the sender, or forward routing failures, are ITK exceptions and MUST be signalled by an ITK infrastructure NACK routed to the sender address as carried by the Distribution Envelope.				

TMS-ERR-03	**Both MHS Sender and MHS Receiver MUST implement the common ITK behaviours defined within the Contract Properties bound to a CPAid**	**Y**	**Y**	**Y**	**N**
1	MHS Receiver MUST implement message de-duplication and acknowledgement irrespective of the `eb:DuplicateElimination` and `eb:AckRequested` values				
2	MHS Sender MUST implement Retry and Retry Interval				
3	MHS Sender MUST use the same Message ID for Retry				
4	MHS Sender SHOULD discard a “Failed to Send Response” after being exhausted all its Retries				
5	MHS Sender SHOULD implement Persist Duration requirements defined in the ITK Contract Properties				
NB	Contract Properties related to MHS behaviour are never overridden by MHS Sender				

3.4 ITK TMS Message Audit

Ref	Description	Client	Host	MW	SMSP
TMS-AUD-01	**Every ITK TMS Message Sent and Received by the MHS MUST be logged**	**Y**	**Y**	**Y**	**N**
NB	Requirements for logging can be found in the Section 3.4 of the Client, Host and Middleware Requirements Document				

TMS-AUD-02	ITK TMS Accredited Systems message transmission logs MUST be auditable	Y	Y	Y	N
1	The MHS Sender and Receiver log SHOULD capture all the necessary and sufficient attributes of the message metadata usable to investigate all possible recoverable errors. - Transmission timestamp - ebXML Message ID - Source and destination party key - ITK service, Interaction and profile ID - ITK Tracking ID - Message type - Transmission result with code (status from the transfer report / success, or the details of any failure notified either by TMS, or the destination message handler) - If present the auditIdentity values from the distribution envelope.				

TMS-AUD-03	ITK TMS Accredited Systems Audit Log MUST NOT contain a copy of the messages transmitted / received	Y	Y	Y	N

TMS-AUD-04	ITK TMS Accredited Systems MUST NOT send ebXML acknowledgements along with message payload	Y	Y	N	N

4 ITK TMS Messaging in Practice

A Spine ebXML message is a MIME package consisting of three parts, with the HL7v3 part existing only to satisfy the TMS transmission requirement for a location for the sending and receiving Accredited System Identifiers, or ASIDs:

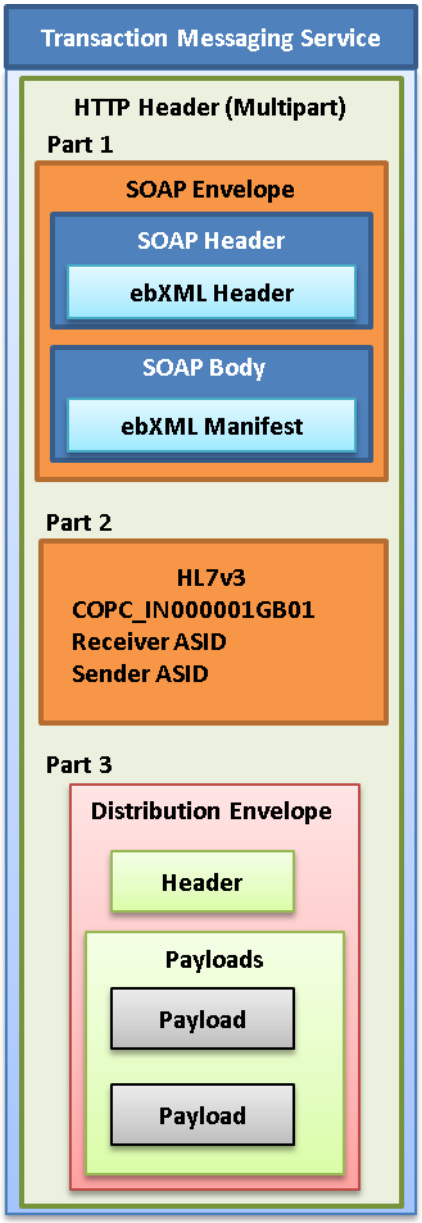


Figure 3 - ITK TMS 3 Part MIME message structure

4.1 TMS Addressing

A TMS address is explicit where it is given as a Spine ASID identified by the OID “1.2.826.0.1285.0.2.0.107”, for example:

```
<itk:addressList>[1]
 <itk:address type=”1.2.826.0.1285.0.2.0.107” uri=”631955299542”/>
</itk:addressList>
```

In this case, the sender has explicitly addressed the message to the Spine endpoint with the ASID “631955299542”. Explicit addressing of this type is only valid where the sender “knows” that a TMS endpoint is available for this use.

4.2 ITK Routing

As described in the ITK Addressing and Routing module specification, an ITK routing node may resolve TMS as the physical route appropriate for a particular recipient address. For example:

```
<itk:addressList>
 <itk:address uri=”urn:nhs-uk:addressing:ods:A83045:discharges”/>[2]
</itk:addressList>
```

The ITK message offers a convenient mechanism for national-level routing because it uses SDS to discover the next router. Therefore it can effectively be used as a “back-stop” route for those cases where no other specific route is available:

urn:nhs-uk:addressing:ods:B80310:*	URL http://b80310.gp.nhs.uk/itk
urn:nhs-uk:addressing:ods:RHM:*	URL http://itk.rhm.nhs.uk/internal
urn:nhs-uk:addressing:ods:*	URL for ITK TMS

Table 2: Sample Routing Table

In this example, the organisation “RHM” has an internal router, and a locally-arranged link with a neighbourhood GP practice. But these are the only explicit connections the router is aware of. All other addresses for ODS organisations are determined to use the ITK message over TMS, where the details for the next router are discovered by a lookup on SDS against the ODS code in the address.

[1] In this example, which is a fragment of a distribution envelope, the prefix “itk” would have been bound earlier in the complete message, to the ITK namespace “urn:nhs-itk:ns:201005”.

[2] In this example the OID is omitted, because the distribution envelope supplies the default OID for an ITK address, of “2.16.840.1.113883.2.1.3.2.4.18.22”.

4.3 ITK TMS Configuration

To send or receive ITK TMS Messages an endpoint MUST be a registered endpoint with a certificate signed by the sub-certificate-authority for the environment in use. ITK messaging systems MUST comply with Common Assurance Process (CAP) regarding registered endpoints. Specifically they MUST demonstrate adequate protection of certificate keys and pass-phrases. ITK endpoints MUST demonstrate strict control over access to the message transmission subsystem so as to prevent non-ITK messages being sent.

An ITK endpoint (Distribution Envelope processor) MUST be able both to send and receive messages. Even where an endpoint will only function as a message receiver, the ITK routing specification requires it to be able to return infrastructure, business ACK and NACK messages and business responses.

Typically an MHS Sender requires an SDS nhsAS object listing at least the ITK service and action "`urn:nhs:names:services:itk:COPC_IN000001GB01`" and an nhsMHS object binding that interaction to an endpoint URL to which TMS can forward a message.

4.4 Exception Handling

Figure 2 implies a number of potential points of failure. In each case it is for the sending system to determine whether the exception warrants re-trying the message. Some examples are:

Exception	Processing
Failure to resolve or connect to TMS service URL	Likely to be retry-able temporary failure in DNS or IP routing/connectivity.
TMS connection rejected or access control failure.	As time-to-resolve is likely to be unknown, whether to re-try is a local configuration decision.
TMS SOAP or ebXML fault (HTTP 500).	Not retry-able.
HTTP timeout waiting for HTTP 202 from TMS.	Retry-able.
Timeout waiting for ebXML acknowledgment to be returned from recipient.	Retry-able. The recipient must according to this specification be able to de-duplicate any re-tried messages.
ebXML MessageError[3] returned from recipient.	The recipient should indicate the severity of the problem it encountered, and whether the sender should re-transmit.
Maximum number of retries reached.	Not retry-able.

Table 3: Exception Processing

For those cases where the transfer has failed, after all retry attempts are exhausted by the MHS Sender, an ITK generic infrastructure NACK is routed back to the ITK Message sender.

[3] `urn:oasis:names:tc:ebxml-msg:MessageError` is added to the Spine endpoint's SDS details as part of service registration.

Hence the distribution envelope's "sender address" is required to contain an address usable in the context of the router which makes the ITK message transmission.

4.5 TMS Endpoint Resolution

A major attraction of using TMS as an ITK transport is that the sending router need not have explicit information available about the target endpoint. Typically ITK messages routed over TMS are attached to an "ITK" message urn:nhs:names:services:itk:COPC_IN000001GB01, and all ITK addresses carry the ODS code for the recipient organisation.

A lookup on the Spine Directory Service, for the endpoint for the ITK message on the recipient's ODS code will, if that organisation supports it, resolve all that the sending router needs to know to forward the ITK message, over TMS.

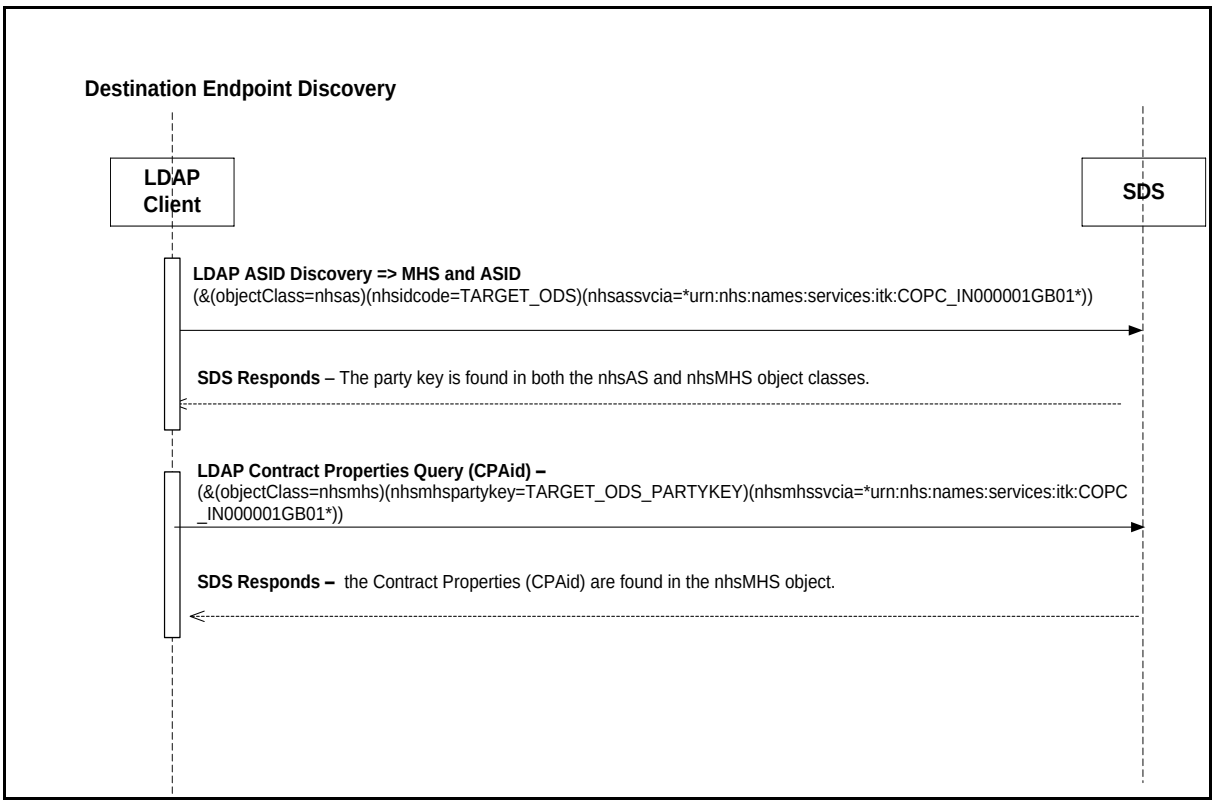


Figure 4 - TMS endpoint discovery

The "SDS lookup" part of this process resolves the details necessary to construct and populate the `COPC_IN000001GB01` "ITK" message. Spine addresses are in two parts, a "party key" which identifies the Spine-connected end point to which TMS will physically forward the message, and an "accredited system identifier" or ASID which represents the collection of Spine interactions that the receiving system can perform.

The SDS lookup determines these values, plus a third, the "contract property agreement identifier" or CPAid. These, together with some message transmission identifiers and timestamps, are used to construct the "ITK" message.

Address resolution is based on the content of the distribution envelope's "addressList". The "addressList" carries a sequence of "address" elements each of which can carry an address and, optionally, an object type identifier (OID) which declares the type of address.

4.6 Destination Endpoint Discovery

Other forms of ITK transport require that a sending system be in possession of physical details of the receiver. This might be a URL for web service forwarding, a DTS mailbox address. Alternatively, the sender and receiver may be closely related - for example the queue collection mechanism.

For a given ITK TMS transmission an ODS Code and TMS interaction are used to resolve the endpoint address (ASID and Party Key) as shown in Figure 5.

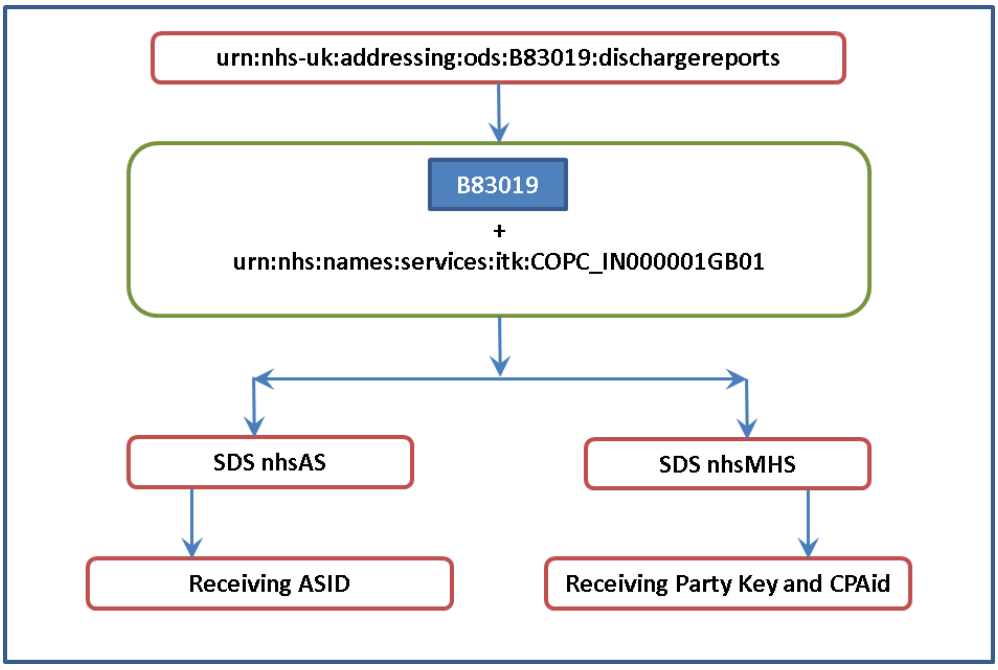


Figure 5 - ITK endpoint discovery

ITK sending MHS's should implement SDS lookup via a local cache – the nature of this cache is the responsibility of the systems developer and/or the sites where the Spine endpoint is deployed.

Examples of LDAP queries for use with SDS, for discovery of remote ASIDs, Party Keys and CPAids are:

ASID discovery:

```
(&(objectClass=nhsas)(nhsidcode=TARGET_ODS)(nhsassvcia=*urn:nhs:names:services:itk:COPC_IN000001GB01*))
```

The party key is found in both the nhsAS and nhsMHS object: This is used to populate the <eb:PartyId> element in the SOAP message.

The ASID is found in the nhsAS object. These are used to populate the <communicationFunctionRcv>, and <communicationFunctionSnd> elements

4.7 Contract Properties

CPAid discovery:

```
(&(objectClass=nhsmhs)(nhsmhspartykey=TARGET_ODS_PARTYKEY)(nhsmhssvcia=*urn:nhs:names:services:itk:COPC_IN000001GB01*))
```

The Contract Properties (CPAid) are found in the nhsMHS object. This is used to populate the <eb:CPAId> element in the SOAP message.

It should be noted that sending systems adhere to the contract properties of the target system, for example:

- nhsMHSRetries: 3
- nhsMHSPersistDuration: PT9M
- nhsMHSRetryInterval: PT2M

4.8 Service Providers

The SDS objects referenced carry an attribute – the "`nhsIDCode`" – which is the ODS code of the organisation owning that object. This is an organisational ownership of the endpoint as a message receiver and is distinct from physical ownership.

The SDS data for a TMS endpoint receiving an ITK message – of object class nhsMHS – is bound to the physical endpoint via the URL on which TMS delivers the message, which is held in the nhsMHSEndPoint attribute. This specification places no constraint on service providers hosting ITK end points for NHS or other organisations.

It is, therefore, open to a service provider to serve the endpoint URL on behalf of some other organisation. In that case each "client" organisation would have its own nhsAS containing, an nhsMHS objects for the ITK message, but with endpoint URLs that resolve to the service provider's platform. The administration of the URL – its DNS entries and mapping to the client systems – are the responsibility of the service provider.

4.9 ebXML

The ITK message is sent using TMS "End-Party Reliable" messaging pattern, so reliability features such as retry, de-duplication are the responsibility of the sending and receiving endpoints: TMS does not provide these services for the message.

The message behaviour pattern for messaging is configured as below.

Contract Properties for End Party Reliable Messaging:

```
syncReplyMode="none"
ackRequested="always"
duplicateElimination="always"
actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"
```

The ebXML header also carries references to the MIME parts containing the HL7 ASID carrier, and the distribution envelope. The MIME header is:

```
Content-Id: <ebXMLHeader@spine.nhs.uk>

Content-Type: text/xml; charset=UTF-8

Content-Transfer-Encoding: 8bit
```

The ebXML MIME part is:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:xsi="http://www.w3c.org/2001/XML-Schema-Instance"
xmlns:SOAP="http://schemas.xmlsoap.org/soap/envelope/" xmlns:eb="http://www.oasis-
open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd" xmlns:hl7ebxml="urn:hl7-
org:transport/ebxml/DSTUv1.0" xmlns:xlink="http://www.w3.org/1999/xlink">
<SOAP:Header>
  <eb:MessageHeader SOAP:mustUnderstand="1" eb:version="2.0">
    <eb:From>
      <eb:PartyId eb:type="urn:nhs:names:partyType:ocs+serviceInstance">RHM-
801710</eb:PartyId>
    </eb:From>
    <eb:To>
      <eb:PartyId eb:type="urn:nhs:names:partyType:ocs+serviceInstance">RHM-
803229</eb:PartyId>
    </eb:To>
    <eb:CPAId>S2012178A2061869</eb:CPAId>
    <eb:ConversationId>DC3BA663-7224-11DF-A0D3-A34D0675B68F</eb:ConversationId>
    <eb:Service>urn:nhs:names:services:itk</eb:Service>
    <eb:Action>COPC_IN000001GB01</eb:Action>
    <eb:MessageData>
      <eb:MessageId>DC3BA663-7224-11DF-A0D3-A34D0675B68F</eb:MessageId>
      <eb:Timestamp>2010-06-07T12:07:28Z</eb:Timestamp>
    </eb:MessageData>
    <eb:DuplicateElimination/>
  </eb:MessageHeader>
  <eb:AckRequested SOAP:mustUnderstand="1" eb:version="2.0" eb:signed="false"
SOAP:actor="urn:oasis:names:tc:ebxml-msg:actor:toPartyMSH"/>
  </SOAP:Header>
<SOAP:Body>
  <eb:Manifest SOAP:mustUnderstand="1" eb:version="2.0">
    <eb:Reference xlink:href="cid:DC3BA663-7224-11DF-A0D3-
A34D0675B68F@spine.nhs.uk">
      <eb:Schema eb:location="http://www.nhsia.nhs.uk/schemas/HL7-Message.xsd"
eb:version="1.0"/>
      <eb:Description xml:lang="en">HL7 payload</eb:Description>
      <hl7ebxml:Payload style="HL7" encoding="XML" version="3.0"/>
    </eb:Reference>
    <eb:Reference xlink:href="cid:DC3BA663-7224-11DF-A0D3-A34D0675B68F">
      <eb:Description xml:lang="en">ITK Tunnelled Message
Attachment</eb:Description>
    </eb:Reference>
  </eb:Manifest>
</SOAP:Body>
</SOAP:Envelope>
```

Note that the party ids, and CPAid will vary depending on sending and receiving MHS.

4.10 HL7

ITK TMS is a transport service and whilst it may ship clinical content between ITK routers, it carries no specific clinical content itself. As such the principal reason for the use of an HL7 part in messaging is to satisfy the TMS requirement for sender and recipient Accredited System Identifiers (ASIDs)[4].

The MIME header is:

```
Content-Id: <DC3BA663-7224-11DF-A0D3-A34D0675B68F@spine.nhs.uk>

Content-Type: application/xml; charset=UTF-8

Content-Transfer-Encoding: 8bit
```

The HL7 MIME part is:

```
<?xml version="1.0" encoding="UTF-8"?>
<COPC_IN000001GB01 xmlns="urn:hl7-org:v3">
  <id root="DC3BA663-7224-11DF-A0D3-A34D0675B68F"/>
  <creationTime value="20100607120728"/>
  <versionCode code="V3NPfIT3.0"/>
  <interactionId root="2.16.840.1.113883.2.1.3.2.4.12"
extension="COPC_IN000001GB01"/>
  <processingCode code="P"/>
  <processingModeCode code="T"/>
  <acceptAckCode code="NE"/>
  <communicationFunctionRcv>
    <device>
      <id root="1.2.826.0.1285.0.2.0.107" extension="276827251543"/>
    </device>
  </communicationFunctionRcv>
  <communicationFunctionSnd>
    <device>
      <id root="1.2.826.0.1285.0.2.0.107" extension="715373337545"/>
    </device>
  </communicationFunctionSnd>
  <ControlActEvent>
    <author1>
      <AgentSystemSDS>
        <agentSystemSDS>
          <id root="1.2.826.0.1285.0.2.0.107" extension="715373337545"/>
        </agentSystemSDS>
      </AgentSystemSDS>
    </author1>
  </ControlActEvent>
</COPC_IN000001GB01>
```

Note: **communicationFunctionRcv**, and **communicationFunctionSnd** contain the end point ASIDs.

The MIME header is:

```
Content-Id: <DC3BA663-7224-11DF-A0D3-A34D0675B68F>
```

[4] ASIDs are also carried in order to allow existing Spine message handlers, which are built to work with HL7v3 messages that carry ASIDs, to receive message which is presented in the same way as those for which they are already proven.

```
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary
```

Note that the content id value is that referenced in the ebXML manifest.

4.11 Timestamps

Timestamps exist within both the ebXML and HL7 mime parts. General guidance:

- ebXML Timestamp will be UTC
- HL7 creationTime will be the local event time

Timestamps will remain the same for all retries within the target system retry count value e.g. nhsMHSRetries: 3, hence for each of three retries the timestamps are identical to the original message sent.

5 ITK TMS Worked Example

5.1 Point to Point Information Exchange

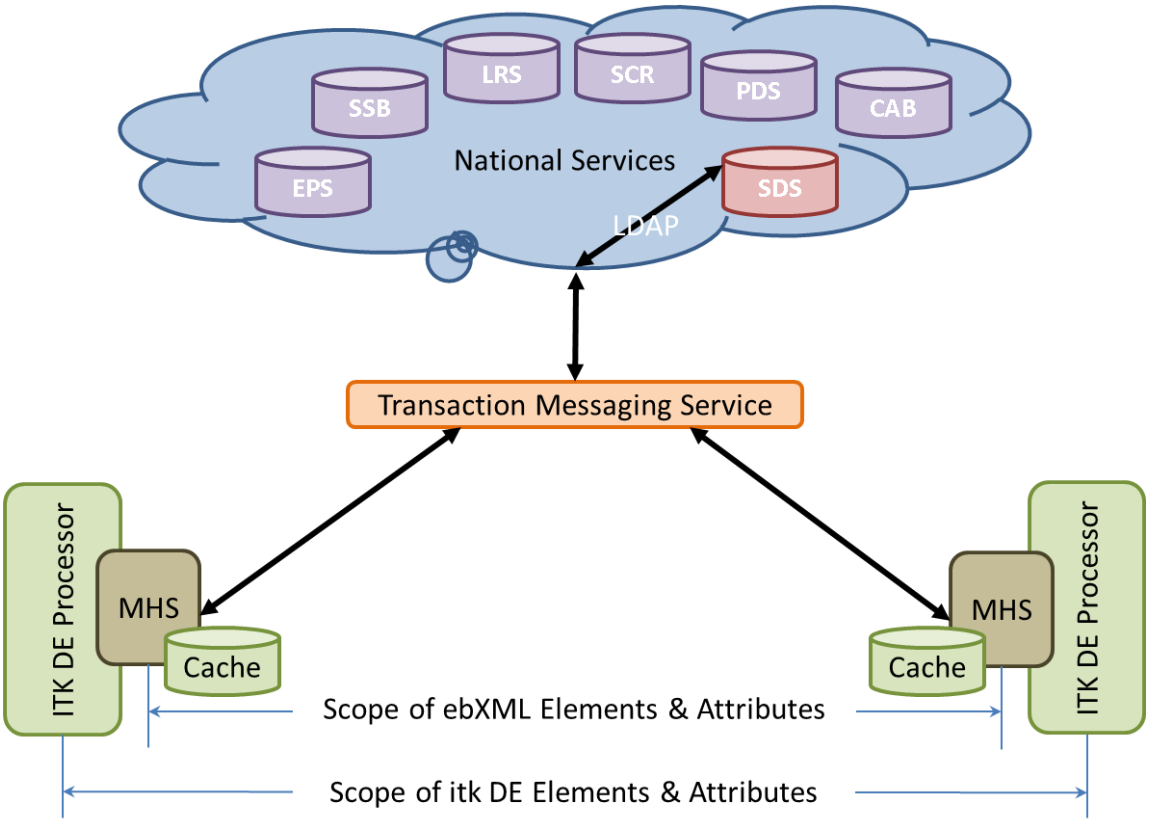


Figure 6 - A schematic diagram of TMS

6 Anatomy of TMS ITK Trunk Message

The Anatomy of the TMS ITK Trunk Message is broken down in to the three MIME parts. The key elements and data values used to enable routing between Message Handling Systems are identified.

6.1 HTTP Header and Overall Structure

```
POST /proxy HTTP/1.1
host: rmebxmlout.vn3.national.ncrs.nhs.uk
SOAPAction: "urn:nhs:names:services:itk/COPC_IN000001GB01"
Content-Length: 13021
Content-Type: multipart/related;boundary="--=_MIME-Boundary";type="text/xml";start="<16fc2804-b08b-

----=_MIME-Boundary
Content-Id: <16fc2804-b08b-4813-9f1c-b23ed54c2953>
Content-Type: text/xml
Content-Transfer-Encoding: 8bit

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:xsi="http://www.w3c.org/2001/XML-Schema-Instance" xmlns:SOAP="http://schemas.xm
http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd" xmlns:hl7ebxml="urn:hl7-o
<SOAP:Header>
<SOAP:Body>
</SOAP:Envelope>

----=_MIME-Boundary
Content-Id: <03a92ac6-c901-4330-8b09-f29b78b107e4>
Content-Type: application/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit

<?xml version="1.0" encoding="UTF-8"?><COPC_IN000001GB01 xmlns="urn:hl7-org:v3">

----=_MIME-Boundary
Content-Id: <059abcdb-0b30-4911-bc87-05ddeca61b68>
Content-Type: text/xml
Content-Transfer-Encoding: 8bit

<itk:DistributionEnvelope xmlns:itk="urn:nhs-itk:ns:201005">
----=_MIME-Boundary--
```

Points of note:

- The POST uri and Host are both obtained from the Authority Service Names document.
- The content type must be multipart/related and must specify the MIME boundary.
- The type clause refers to the ebXML part and must be text/xml.
- The “start” clause SHOULD be given with the content id of the ebXML MIME part, but MAY be omitted (because it defaults to the first physical MIME part).
- The SOAPAction is the service uri being accessed.
- Content-Length MUST be specified (no chunking support).

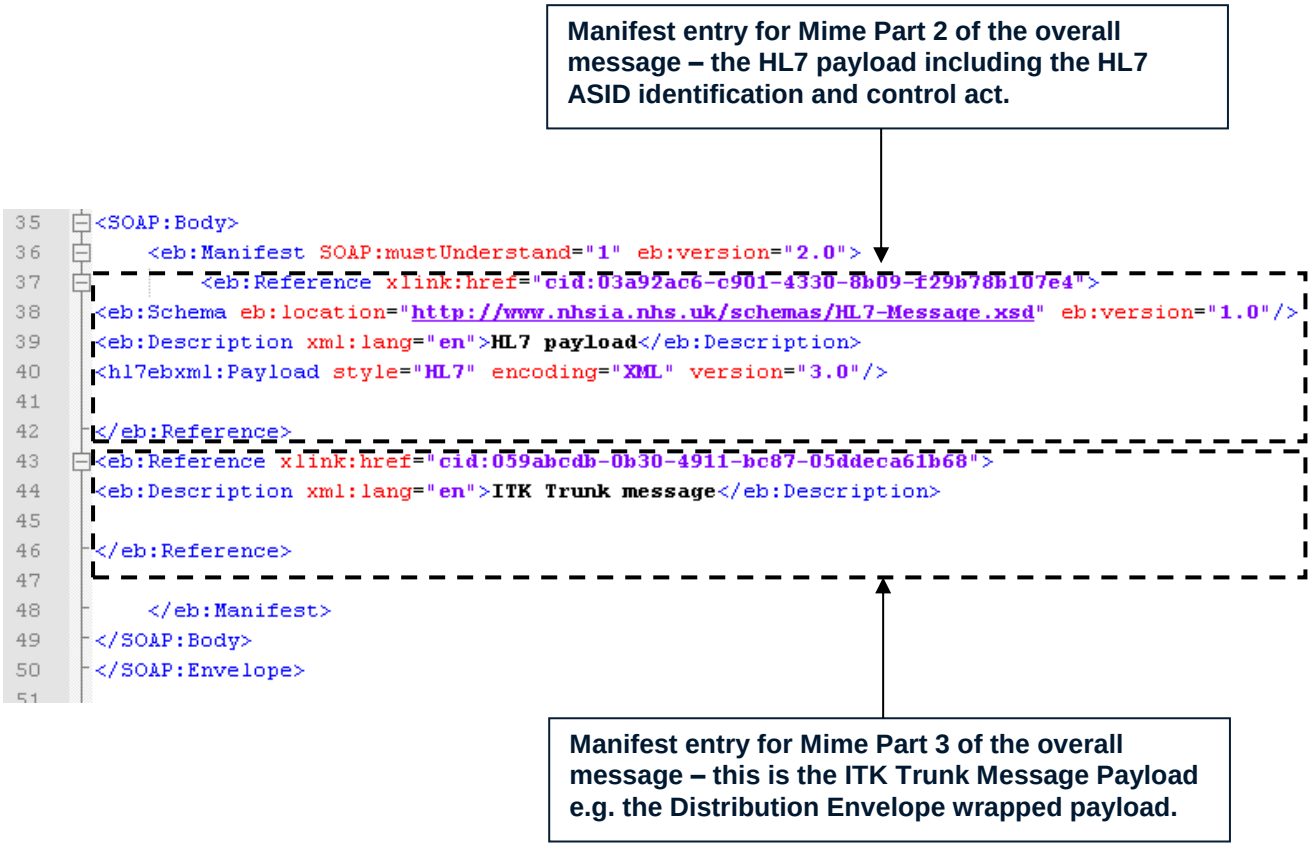
6.2 Mime Part 1 – SOAP Header


```
----=_MIME-Boundary
Content-Id: <16fc2804-b08b-4813-9f1c-b23ed54c2953>
Content-Type: text/xml
Content-Transfer-Encoding: 8bit

<?xml version="1.0" encoding="UTF-8"?>
<SOAP:Envelope xmlns:xsi="http://www.w3c.org/2001/XML-Schema-Instance" xmlns:SOAP="http://schemas.xmlsoap
http://www.oasis-open.org/committees/ebxml-msg/schema/msg-header-2_0.xsd" xmlns:hl7ebxml="urn:hl7-org:tra
<SOAP:Header>
    <eb:MessageHeader SOAP:mustUnderstand="1" eb:version="2.0">
        <eb:From>
            <eb:PartyId eb:type="urn:nhs:names:partyType:ocs+serviceInstance">RHM-1808409</eb:PartyId>
        </eb:From>
        <eb:To>
            <eb:PartyId eb:type="urn:nhs:names:partyType:ocs+serviceInstance">RHM-801710</eb:PartyId>
        </eb:To>
        <eb:CPAId>S2030789A2138483</eb:CPAId>
        <eb:ConversationId>FD5CA0E3-DFF1-4E8B-9BB2-9D1C39E96DBC</eb:ConversationId>
        <eb:Service>urn:nhs:names:services:itk</eb:Service>
        <eb:Action>COPC_IN000001GB01</eb:Action>
        <eb:MessageData>
            <eb:MessageId>FD5CA0E3-DFF1-4E8B-9BB2-9D1C39E96DBC</eb:MessageId>
            <eb:Timestamp>2013-09-26T20:47:10</eb:Timestamp>
        </eb:MessageData>
        <eb:DuplicateElimination/>
    </eb:MessageHeader>
    <eb:AckRequested eb:version="2.0" SOAP:mustUnderstand="1" SOAP:actor="urn:oasis:names:tc:ebxml-msg:ac

</SOAP:Header>
```

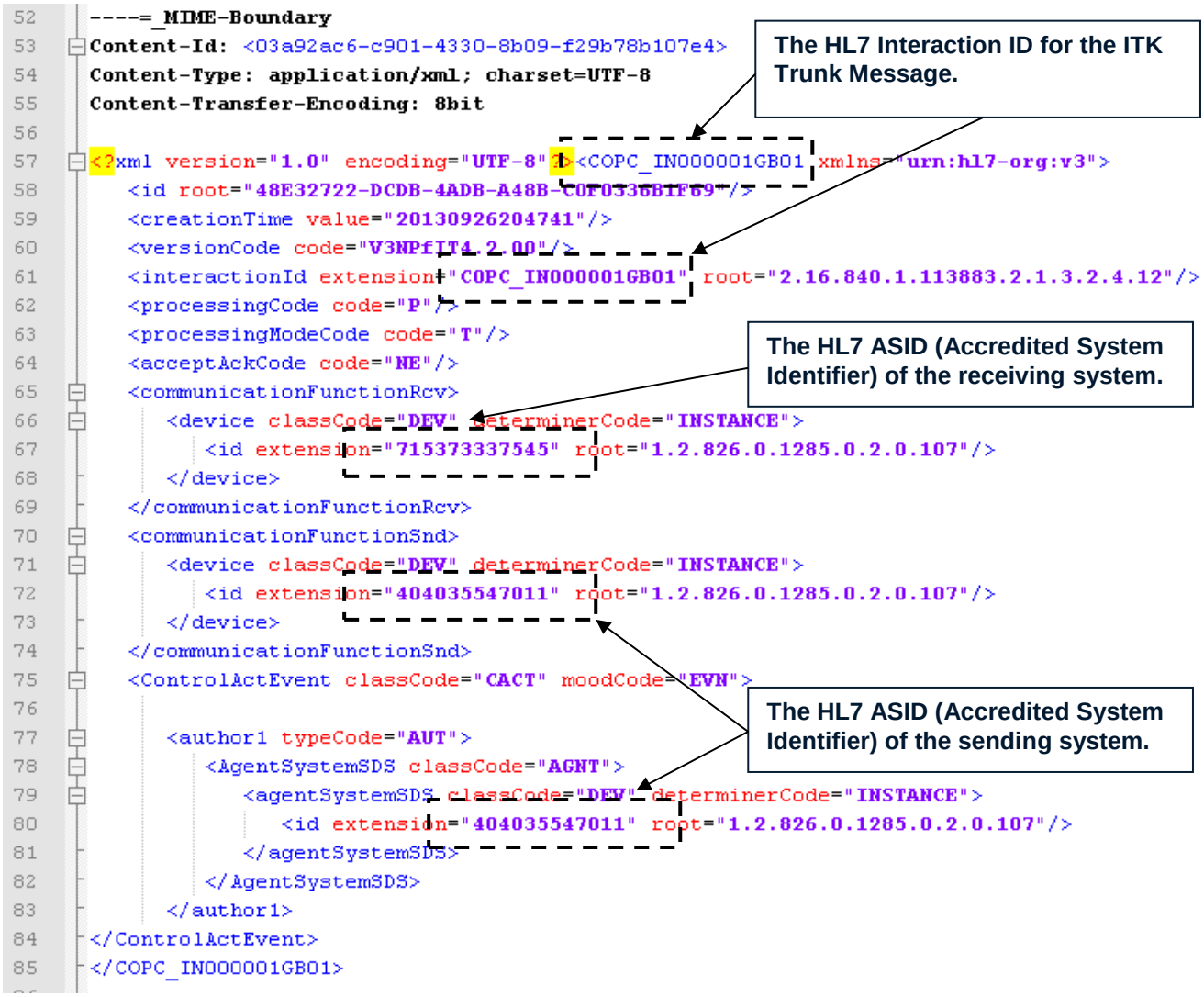
Points of Note:

- The ITK TMS Trunk message uses the Spine End-Party Reliable messaging (also known as Forward Express) which mandates ebXML use, with an asynchronous ebXML acknowledgment.

6.3 Mime Part 1 – SOAP Body



6.4 Mime Part 2 – HL7 Payload




```
----=_MIME-Boundary
Content-Id: <03a92ac6-c901-4330-8b09-f29b78b107e4>
Content-Type: application/xml; charset=UTF-8
Content-Transfer-Encoding: 8bit

<?xml version="1.0" encoding="UTF-8"?><COPC_IN000001GB01 xmlns="urn:hl7-org:v3">
   <id root="48E32722-DCDB-4ADB-A48B-C0F0336B1F69"/>
   <creationTime value="20130926204741"/>
   <versionCode code="V3NPfLT4.2.00"/>
   <interactionId extension="COPC_IN000001GB01" root="2.16.840.1.113883.2.1.3.2.4.12"/>
   <processingCode code="P"/>
   <processingModeCode code="T"/>
   <acceptAckCode code="NE"/>
   <communicationFunctionRcv>
      <device classCode="DEV" determinerCode="INSTANCE">
         <id extension="715373337545" root="1.2.826.0.1285.0.2.0.107"/>
      </device>
   </communicationFunctionRcv>
   <communicationFunctionSnd>
      <device classCode="DEV" determinerCode="INSTANCE">
         <id extension="404035547011" root="1.2.826.0.1285.0.2.0.107"/>
      </device>
   </communicationFunctionSnd>
   <ControlActEvent classCode="CACT" moodCode="EVN">

      <author1 typeCode="AUT">
         <AgentSystemSDS classCode="AGNT">
            <agentSystemSDS classCode="DEV" determinerCode="INSTANCE">
               <id extension="404035547011" root="1.2.826.0.1285.0.2.0.107"/>
            </agentSystemSDS>
         </AgentSystemSDS>
      </author1>
</ControlActEvent>
</COPC_IN000001GB01>
```

6.5 Mime Part 3 – ITK Trunk Payload / Compressed DE

```
----=_MIME-Boundary
Content-Id: <059abcdb-0b30-4911-bc87-05ddeca61b68>
Content-Type: text/xml
Content-Transfer-Encoding: 8bit

<itk:DistributionEnvelope xmlns:itk="urn:nhs-itk:ns:201005">
    <itk:header service="urn:nhs-itk:services:201005:SendCDADocument-v2-0"
                trackingid="9f6aa44b-17f3-401a-afe6-2e00e4358c79">
        <itk:addresslist>
            <itk:address type="2.16.840.1.113883.2.1.3.2.4.18.22"
                         uri="urn:nhs-uk:addressing:ods:rhm:team1:A" />
            <itk:address type="1.2.826.0.1285.0.2.0.107"
                         uri="874567095" />
        </itk:addresslist>
        <itk:auditIdentity>
            <itk:id type="1.2.826.0.1285.0.2.0.107"
                    uri="99999999999" />
        </itk:auditIdentity>
        <itk:manifest count="1">
            <itk:manifestitem mimetype="text/xml" base64="false" compressed="true"
        </itk:manifest>
        <itk:senderAddress type="2.16.840.1.113883.2.1.3.2.4.18.22"
                           uri="urn:nhs-uk:addressing:ods:rhm:team1:B" />
        <itk:handlingSpecification>
            <itk:spec value="true" key="urn:nhs-itk:ns:201005:ackrequested"/>
        </itk:handlingSpecification>
    </itk:header>
    <itk:payloads count="1">

    </itk:payloads>
</itk:DistributionEnvelope>
```

Both ASID and ODS Address can go into the DE

Standard ITK Distribution Envelope. DE SHOULD be compressed using GZIP

* * * End of Document * * *