

Integrating the Healthcare Enterprise



5

**IHE IT Infrastructure
Technical Framework Supplement**

10

**Extensions to the
Document Metadata Subscription
Profile**

15

Rev. 1.1 – Trial Implementation

20

Date: August 5, 2016
Author: IHE IT Infrastructure Technical Committee
Email: iti@ihe.net

25

Please verify you have the most recent version of this document. See [here](#) for Trial Implementation and Final Text versions and [here](#) for Public Comment versions.

Foreword

30 This is a supplement to the IHE IT Infrastructure Technical Framework V13.0. Each supplement undergoes a process of public comment and trial implementation before being incorporated into the volumes of the Technical Frameworks.

This supplement is published on August 5, 2016 for trial implementation and may be available for testing at subsequent IHE Connectathons. The supplement may be amended based on the results of testing. Following successful testing it will be incorporated into the IT Infrastructure
35 Technical Framework. Comments are invited and may be submitted at http://www.ihe.net/ITI_Public_Comments.

This supplement describes changes to the existing technical framework documents.

“Boxed” instructions like the sample below indicate to the Volume Editor how to integrate the relevant section(s) into the relevant Technical Framework volume.

40

<i>Amend Section X.X by the following:</i>
--

Where the amendment adds text, make the added text **bold underline**. Where the amendment removes text, make the removed text **~~bold strikethrough~~**. When entire new sections are added, introduce with editor’s instructions to “add new text” or similar, which for readability are not bolded or underlined.

45

General information about IHE can be found at <http://www.ihe.net>.

Information about the IHE IT Infrastructure domain can be found at http://www.ihe.net/IHE_Domains.

50 Information about the structure of IHE Technical Frameworks and Supplements can be found at http://www.ihe.net/IHE_Process and <http://www.ihe.net/Profiles>.

The current version of the IHE Technical Framework can be found at http://www.ihe.net/Technical_Frameworks.

55 **CONTENTS**

	Introduction to this Supplement.....	6
	Open Issues and Questions	6
	Closed Issues	6
60	Volume 1 – Profiles	7
	Copyright Permission.....	7
	Domain-specific additions	7
	26 Document Metadata Subscription Integration Profile	8
	26.1.1.5 Notification Puller	10
65	26.1.1.6 Notification Pull Point	11
	26.2.2 Folder Subscription Option	12
	26.2.3 Patient-Independent Subscription Option	12
	26.4.1 Concepts	13
	26.4.2.6 Use Case #6: Folder subscription	14
70	26.4.2.6.1 Folder subscription Use Case Description.....	14
	26.4.2.6.2. Folder subscription Process Flow	15
	26.4.2.7 Use Case #7: GP’s EHR notification.....	15
	26.4.2.7.1 GP’s EHR notification Use Case Description	16
	26.4.2.7.2 GP’s EHR notification Process Flow	16
75	26.4.2.8 Use Case #8: Patient-independent tele-consultant notification	17
	26.4.2.8.1 Patient-independent tele-consultant scenario	17
	26.4.2.8.2 Tele-Consultant patient-independent notification Process Flow	18
	26.5 DSUB Security Considerations.....	18
	Volume 2b – Transactions.....	20
80	3.52.4.1.3.1 Folder Subscription Option.....	20
	3.52.5.1.4 ihe:FolderMetadata	20
	3.52.5.2 Building Filter Expressions	20
	3.52.5.2.3 Subscriptions for folders metadata	21
	3.52.5.2.4 Patient-Independent Subscriptions for Document metadata.....	22
85	3.52.6 Security Considerations.....	26
	3.53.4.1.2 Message Semantics.....	27
	3.53.4.1.4.4 Folder Notification Example (ihe:FolderMetadata).....	29
	3.54.4.1.1 Trigger	31
	3.54.4.1.2 Message Semantics.....	31
90	Volume 2c – Transactions	33
	3.69 Create Destroy Pull Point.....	33
	3.69.1 Scope	33
	3.69.2 Actor Roles.....	33
	3.69.3 Referenced Standards.....	34
95	3.69.4 Interaction Diagram.....	34
	3.69.4.1 CreatePullPoint Request message	34

	3.69.4.1.1 Trigger Events	36
	3.69.4.1.2 Message Semantics	36
	3.69.4.1.3 Expected Actions	37
100	3.69.4.1.4 Example SOAP Encoding of the CreatePullPoint Request Message	37
	3.69.4.2 CreatePullPoint Response message	37
	3.69.4.2.1 Trigger Events	37
	3.69.4.2.2 Message Semantics	38
	3.69.4.2.3 Expected Actions	38
105	3.69.4.2.4 Example SOAP Encoding of the CreatePullPoint Response Message	38
	3.69.4.3 DestroyPullPoint Request message	39
	3.69.4.3.1 Trigger Events	39
	3.69.4.3.2 Message Semantics	39
	3.69.4.3.3 Expected Actions	39
110	3.69.4.3.4 Example SOAP Encoding of the DestroyPullPoint Request Message	40
	3.69.4.4 DestroyPullPoint Response message	40
	3.69.4.4.1 Trigger Events	40
	3.69.4.4.2 Message Semantics	40
	3.69.4.4.3 Expected Actions	40
115	3.69.4.4.4 Example SOAP Encoding of the DestroyPullPoint Response Message ..	40
	3.69.5 Security Considerations	41
	3.69.5.1 Security Audit Considerations	41
	3.69.5.1.1 Notification Pull Point audit message	41
	3.69.5.1.2 Notification Puller audit message	43
120	3.69.5.1.3 Notification Pull Point Actor Specific Security Considerations	45
	3.70 Pull Notification	45
	3.70.1 Scope	45
	3.70.2 Actor Roles	45
	3.70.3 Referenced Standards	46
125	3.70.4 Interaction Diagram	46
	3.70.4.1 GetMessages Request message	46
	3.70.4.1.1 Trigger Events	46
	3.70.4.1.2 Message Semantics	47
	3.70.4.1.3 Expected Actions	47
130	3.70.4.1.4 Example SOAP Encoding of the GetMessages Request message	48
	3.70.4.2 GetMessages Response message	48
	3.70.4.2.1 Trigger Events	48
	3.70.4.2.2 Message Semantics	48
	3.70.4.2.3 Expected Actions	49
135	3.70.4.2.4 Example SOAP Encoding of the GetMessage Response Message	49
	3.70.5 Security Considerations	49
	3.70.5.1 Security Audit Considerations	50
	3.70.5.1.1 Notification Puller audit message	50

140 3.70.5.1.2 Notification Pull Point audit message..... 52
 3.70.5.1.3 Metadata Notification Pull Point Specific Security Considerations..... 53

Introduction to this Supplement

145 This Trial Implementation supplement extends the notification infrastructure defined in the DSUB Profile (integrated in Final Text in September 2016) adding the following functionalities:

1. Subscription to Folders objects and Folder's updates (ITI TF-1: 26.2.2 see "Folder Subscription Option").
2. Patient independent subscriptions for DocumentEntry and SubmissionSet objects (see ITI TF-1: 26.2.3 "Patient-Independent Subscription Option").
- 150 3. Pull-style notification approach (as defined in WS-BaseNotification standard Section 5 "Pull-Style Notification"). There are certain circumstances in which the basic "push-style" of notification message delivery is not appropriate. For example, certain Document Metadata Notification Recipients are behind a firewall such that the Document Metadata Notification Broker cannot initiate a message exchange to send the notification. A similar
155 circumstance exists for Document Metadata Notification Recipient that is unable or unwilling to provide an endpoint to which the Notification Broker can send notification messages. In other situations, the Notification Recipient prefers to control the timing of receipt of notification messages, instead of receiving notification messages at unpredictable intervals, it may prefer to "pull" (retrieve) the notification messages at
160 a time of its own choosing. In the "Pull-style" method, a Notification Puller creates a Pull Point resource able to store notification generated by the Document Metadata Notification Broker. This Pull Point resource is a resource managed by the Pull Point that allows the storing of notification targeted to a specific recipient. Notifications stored in the Pull Point can be retrieved by the Notification Puller using a specific transaction.

165 **Open Issues and Questions**

None

Closed Issues

None

170

Volume 1 – Profiles

Copyright Permission

None

Domain-specific additions

175 None

Editor: Add the following actor definitions in Appendix A

Actor	Definition
Notification Pull Point	The Notification Pull Point is the actor that stores notifications targeted to a specific Document Metadata Notification Recipient that cannot be directly notified. This actor delivers notifications to the Notification Puller when requested.
Notification Puller	The Notification Puller is the actor that can create a pull point resource for the storing of notifications. It pulls notifications stored in a Notification Pull Point when requested.

180

Editor: Add the following transaction definitions in Appendix B

Transaction	Definition
[ITI-69] Create Destroy Pull Point	This transaction is used to create a pull point resource. This resource is used for the creation of subscriptions and for the pulling of the notifications stored. This transaction is also used to destroy the pull point resource when it is no longer needed.
[ITI-70] Pull Notification	This transaction is used to retrieve pending notifications.

185

Editor: Add the following terms in the Glossary

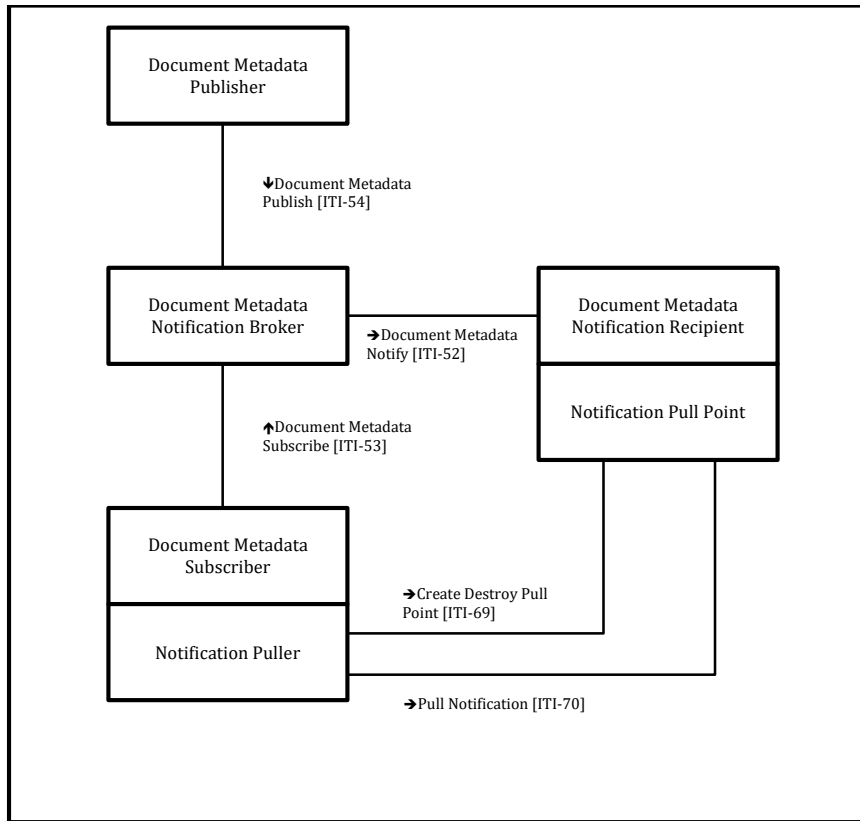
Glossary Term	Definition
Pull Point resource	A resource managed by the Pull Point that allows the storing of notification targeted to a specific recipient.

Editor: make the following changes in Section 26

26 Document Metadata Subscription Integration Profile

- 190 This profile describes the use of subscription and notification mechanism for use within an XDS Affinity Domain and across communities. The subscription allows for the matching of metadata during the publication of a new document for a given patient, and results in the delivery of a notification. This profile is based on the OASIS WS-BaseNotification standard and **defines two a methods of subscription and notification:**
- 195 1. **In the defines a “Push-style” method for notification., a Using a push-style method of notification, the** Document Metadata Subscriber may subscribe on behalf of the Document Metadata Notification Recipient to receive notifications about the availability of documents based on specific criteria. A Document Metadata Notification Broker keeps track of the subscriptions and sends the appropriate notifications based on the registration
- 200 of objects in an XDS Document Registry. Subscriptions exist for a certain period of time and can be cancelled.
- 205 2. **In the “Pull-style” method, a Notification Puller creates a Pull Point resource able to store notification generated by the Document Metadata Notification Broker. This Pull Point resource is a resource managed by the Notification Pull Point Actor that allows the storing of notification targeted to a specific recipient. Notifications stored in the Notification Pull Point can be retrieved by the Notification Puller using a specific transaction.**

Editor: Replace Figure 26.1-1 with the following:



210

Editor: Apply the following updates to Table 26.1-1:

Table 26.1-1: Document Metadata Subscription Integration Profile - Actors and Transactions

215

Actors	Transactions	Optionality	Reference
Document Metadata Notification Broker	Document Metadata Subscribe	R	ITI TF-2b:3.52
	Document Metadata Notify	R	ITI TF-2b:3.53
	Document Metadata Publish	O	ITI TF-2b:3.54
Document Metadata Subscriber	Document Metadata Subscribe	R	ITI TF-2b:3.52
Document Metadata Publisher	Document Metadata Publish	R	ITI TF-2b:3.54
Document Metadata Notification Recipient	Document Metadata Notify	R	ITI TF-2b:3.53

Actors	Transactions	Optionality	Reference
<u>Notification Puller</u>	<u>Pull Notification</u>	<u>R</u>	<u>ITI TF-2c:3.70</u>
	<u>Create Destroy Pull Point</u>	<u>O</u>	<u>ITI TF-2c:3.69</u>
<u>Notification Pull Point</u>	<u>Pull Notification</u>	<u>R</u>	<u>ITI TF-2c:3.70</u>
	<u>Create Destroy Pull Point</u>	<u>O</u>	<u>ITI TF-2c:3.69</u>

Editor: Add sections 26.1.1.5 and 26.1.1.6

26.1.1.5 Notification Puller

220 The Notification Puller is the actor that is involved in a notification system using a pull-style approach. This actor “activates” the Notification Pull Point using the Create Destroy Pull Point [ITI-69] transaction to create (and destroy) the Pull Point resource.

The Notification Puller SHALL be grouped with a Document Metadata Subscriber.

When using the “pull-style” method of notification, the order of the transactions SHALL be:

- 225 1. The Notification Puller (grouped with a Document Metadata Subscriber) sends the Create Destroy Pull Point [ITI-69] transaction. In response, the Notification Pull Point (grouped with the Document Metadata Notification Recipient) returns the endpoint of the Pull Point resource. The Notification Puller/Document Metadata Subscriber now knows from where to “pull” notifications.
- 230 2. The Document Metadata Subscriber/Notification Puller sends the Document Metadata Subscribe [ITI-52] transaction to the Document Metadata Notification Broker, identifying the endpoint where the notification is to be sent (the Notification Pull Point/Document Metadata Notification Recipient).
- 235 3. Over time, the Notification Pull Point/Document Metadata Notification Recipient will receive notifications via Document Metadata Notify [ITI-53].
4. Later, the Notification Puller sends the Pull Notification [ITI-70] transaction to the Notification Pull Point/Document Metadata Notification Recipient to retrieve notifications it subscribed to.

The order of the transaction for the destroy of the Pull Point resource SHALL be:

- 240 5. The Notification Puller (grouped with a Document Metadata Subscriber) sends unsubscribe Requests (Document Metadata Subscribe [ITI-52] transaction) to delete all

subscriptions created using the endpoint of the Pull Point resource that needs to be destroyed.

- 245 6. The Notification Puller can proceed with the destroying of the Pull Point resource using Create Destroy Pull Point [ITI-69] transaction.

If the Notification Puller does not support the optional Create Destroy Pull Point [ITI-69] transaction, it SHALL be able to support the following configuration requirements:

- It SHALL be configurable with the endpoints for the Pull Point resources already created for it.
- 250 • It SHALL configure the endpoint for Pull Notification [ITI-70] transaction.

26.1.1.6 Notification Pull Point

255 The Notification Pull Point is the actor that stores notifications targeted to systems which cannot be directly notified. The intended recipient for the notifications received by the Notification Pull Point is the Notification Puller that creates a Pull Point resource using the Create Destroy Pull Point transaction. A Pull Point resource is created in response to each Create Pull Point request (it is possible to have many Pull Point resources for the same Notification Puller) and is used to collect all notifications destined for the requesting Notification Puller.

The Notification Pull Point can manage Pull Point resources created by many different Notification Puller actors.

- 260 The Notification Pull Point SHALL be grouped with a Document Metadata Notification Recipient in order to receive notifications from a Document Metadata Notification Broker.

In order to not lose nor duplicate notifications:

- 265 • After the creation of a Pull Point resource, the Notification Pull Point receives and stores all notifications in the target Pull Point resource. The Notification Puller can then retrieve the notifications stored in a target Pull Point resource.
- Notifications returned to the Notification Puller are deleted from the Pull Point resource in accordance to the WS-BaseNotification standard.

If the Notification Pull Point does not support the optional Create Destroy Pull Point [ITI-69] transaction, it SHALL be able to support the following configuration requirements:

- 270 • At least one Pull Point resource needs to be pre-created for each Notification Puller involved in the notification system.
- The endpoints of these Pull Point resources need to be disclosed to the correct Notification Puller.

275 *Editor: apply the following changes in Table 26.2-1*

Table 26.2-1: Document Netadata Subscription – Actors and Options

Actor	Option Name	Reference
Document Metadata Notification Broker	Document Metadata Publish Recipient	ITI TF-1: 26.2.1
	<u>Folder Subscription</u>	<u>ITI TF-1: 26.2.2</u>
	<u>Patient-Independent Subscription Option</u>	<u>ITI TF-1: 26.2.3</u>
Document Metadata Subscriber	No options defined <u>Patient-Independent Subscription Option</u>	<u>ITI TF-1: 26.2.3</u>
Document Metadata Publisher	No options defined	--
Document Metadata Notification Recipient	No options defined	--
<u>Notification Pull Point</u>	No options defined	<u>--</u>
<u>Notification Puller</u>	No options defined	<u>--</u>

280 *Editor: add sections 26.2.2 and 26.2.3*

26.2.2 Folder Subscription Option

285 The Document Metadata Notification Broker that supports this option shall accept and process subscriptions that use Folder metadata as filter parameters and shall be able to send notifications when the content of the folder changes. A notification is sent if a new document is added to an existing folder or if a document in an existing folder is replaced. Refer to ITI TF-2b:3.52.4.1.3.1 and 3.52.5.2.2 and 3.53.4.1.2 for details. It is likely that the Document Metadata Notification Broker will maintain status of existing folders rather than making GetFolders queries from the Register Stored Query [ITI-18] transaction to determine folder status.

290 26.2.3 Patient-Independent Subscription Option

A Patient-Independent Subscription does not specify a patientId parameter. This type of subscription may be applied to DocumentEntry objects or SubmissionSets.

A Document Metadata Subscriber that supports this option shall be able to create patient-independent subscription filters.

295 A Document Metadata Notification Broker that supports this option shall be able to accept patient-independent subscriptions.

See ITI TF-2b:3.52.5.2.4 and 3.52.5.2.5 for use cases related to patient-independent subscriptions.

300 *Editor: apply the following changes to Table 26.3-1*

Table 26.3-1: DSUB - Required Actor Groupings

DSUB Actor	Profile/Actor to be grouped with	Reference
Document Metadata Notification Broker	ATNA / Secure Node or Secure Application	ITI TF-1:9.4
	CT / Time Client	ITI TF-1:7.1
Document Metadata Subscriber	ATNA / Secure Node or Secure Application	ITI TF-1:9.4
	CT / Time Client	ITI TF-1:7.1
Document Metadata Publisher	ATNA / Secure Node or Secure Application	ITI TF-1:9.4
	CT / Time Client	ITI TF-1:7.1
Document Metadata Notification Recipient	ATNA / Secure Node or Secure Application	ITI TF-1:9.4
	CT / Time Client	ITI TF-1:7.1
<u>Notification Pull Point</u>	<u>DSUB / Document Metadata Notification Recipient</u>	<u>ITI TF-1:26.1</u>
<u>Notification Puller</u>	<u>DSUB / Document Metadata Subscriber</u>	<u>ITI TF-1:26.1</u>

Editor: make the following changes to Section 26.4.1

305 **26.4.1 Concepts**

This profile describes the use of subscription and notification mechanisms for use within an XDS Affinity Domain and across communities. The subscription allows for the matching of metadata during the publication of a new document for a given patient, and results in the delivery of a notification.

310 If a system can implement the Document Metadata Notification Recipient, it can be directly notified using a push-style method.

In other scenarios, a system that cannot be notified using the push-style delivery approach implements the pull-style approach because, for example,

- **a system that receives notifications is behind a firewall**
- 315 • **a system is unable or unwilling to provide an endpoint to which the Document Metadata Notification Broker can send notifications.**

- **a system doesn't want to be notified at unpredictable times but rather at a time of its own choosing.**

The use-cases below describe both of these scenarios.

320

<i>Editor: add Sections 26.4.2.6, 24.4.2.7 and 24.2.4.8</i>

26.4.2.6 Use Case #6: Folder subscription

325 This use case recognizes that it is often not possible to identify in advance the type of document that will be produced during a clinical event, so a subscription using the findDocuments filter expression is useless. In many cases, it is only possible to identify the specific clinical event of interest (e.g., Hospitalization, Clinical Day Service, etc.).

26.4.2.6.1 Folder subscription Use Case Description

330 Mr. White is admitted in hospital for a complex diagnostic study pathway. Dr. Brown is the clinician responsible for the “Day Service” of this patient. Any document produced by LIS, RIS, and ward Informative Systems should be collected in a Folder object used for keeping and for managing the evolution of the clinical pathway. Dr. Brown wants to be notified of any content published in this folder. The Dr. Brown’s system can create a subscription selecting as filter parameter the XDSFolder.uniqueId of the folder just created.

335 During the “Day Service”, a Laboratory Report and a Radiology Report are produced. Both the documents are submitted into the folder created for the clinical event. Each publishing event results in a match with the subscription created by Dr. Brown’s system. The Document Metadata Notification Broker creates two notifications and they are sent to the Document Metadata Notification Recipient referenced in the subscription (i.e., Dr. Brown’s system).
340 The Dr. Brown is kept up to date during the clinical processes.

26.4.2.6.2. Folder subscription Process Flow

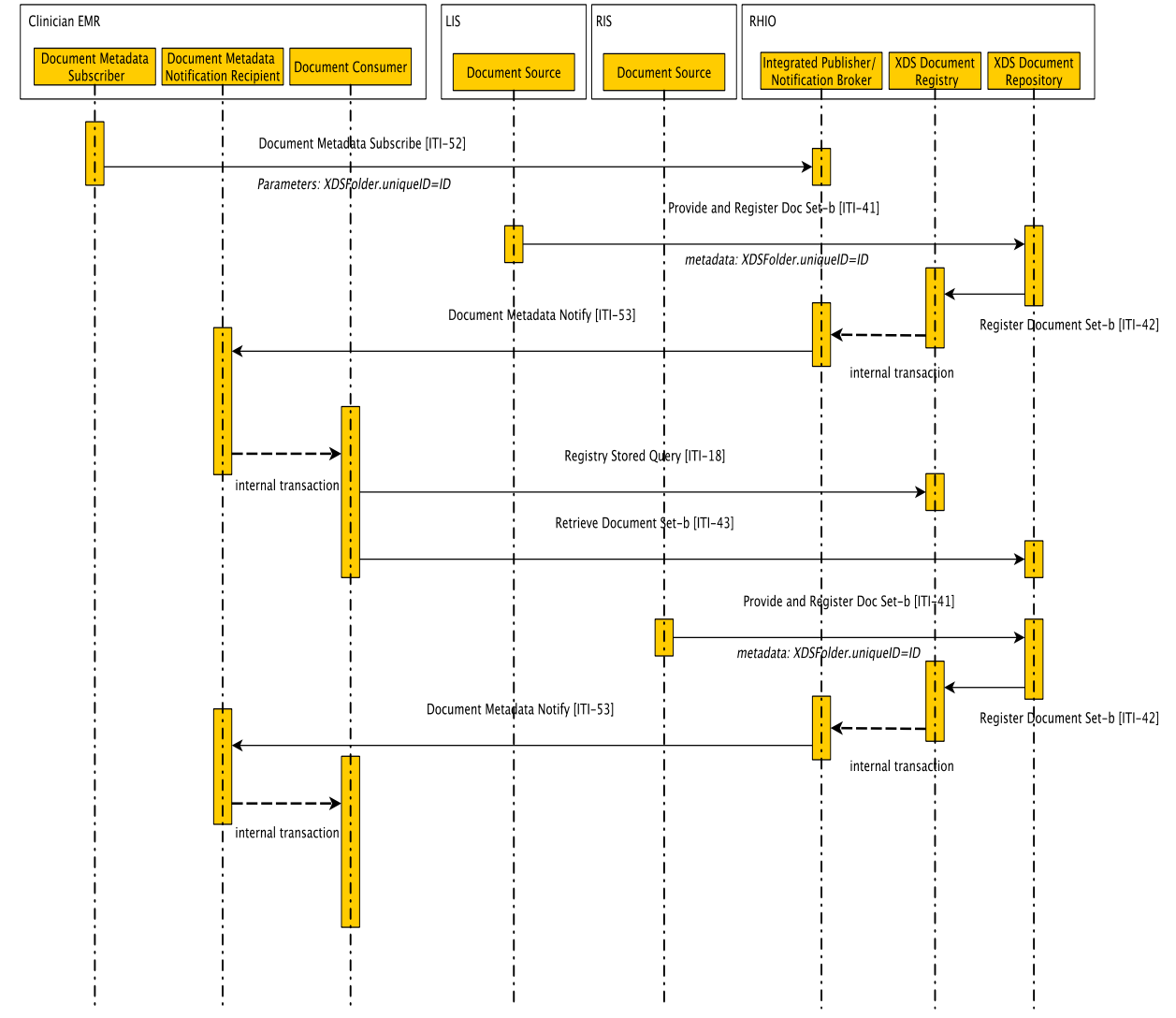


Figure 26.4.2.6.2-1: Interaction Diagram for Folder subscription Use Case

345 26.4.2.7 Use Case #7: GP's EHR notification

This use-case describes the scenario in which a General Practitioner (GP) would like to be notified for hospitalizations of patients assisted, even though the GP's EHR system is on-line only for a restricted time interval.

26.4.2.7.1 GP’s EHR notification Use Case Description

350 Dr. Brown is a GP. He assists many patients and he is very interested in receiving notifications of their eventual hospitalization. This ready acknowledgment and the direct communication between the GP and the hospital can be fundamental for granting a complete and secure clinical approach in solving the health problems of the patient.

355 The GP’s EHR is not on-line when a clinical event occurs such as a hospitalization. When a pull-style EHR goes on-line, it can immediately retrieve outstanding notifications. When a push-style EHR goes on-line it, must wait until the Document Metadata Notification Broker retries a push. This retry might not be in time before the EHR goes off-line again.

360 In this use-case, the GP’s EHR creates a Pull Point resource able to store notifications on behalf of his EHR. The EHR receives in the Response message the endpoint of the pull point resource just created. The EHR system can use this endpoint for any of the supported subscription.

365 Mr. White is one of the Dr. Brown patients. During the night, he becomes sick and he is hospitalized. A diagnosis for the admission of the patient is formulated, and a document is created by the Emergency Department Information System and registered in XDS Document Registry. The Document Metadata Notification Broker discovers a match with a subscription created by the Dr. Brown’s EHR, and sends a notification to the Document Metadata Notification Recipient grouped with the Notification Pull Point referenced in the subscription. The Notification Pull Point stores this and other notifications. The next morning Dr. Brown’s EHR pulls all pending notifications. The EHR receives the notification that was created after the publication of the Admission Document. Dr. Brown reads the Admission Document and
 370 analyzes notes gathered during the last weeks and discovers some symptoms or findings that can be useful for focusing the diagnostic and therapeutic phases during the current hospitalization.

26.4.2.7.2 GP’s EHR notification Process Flow

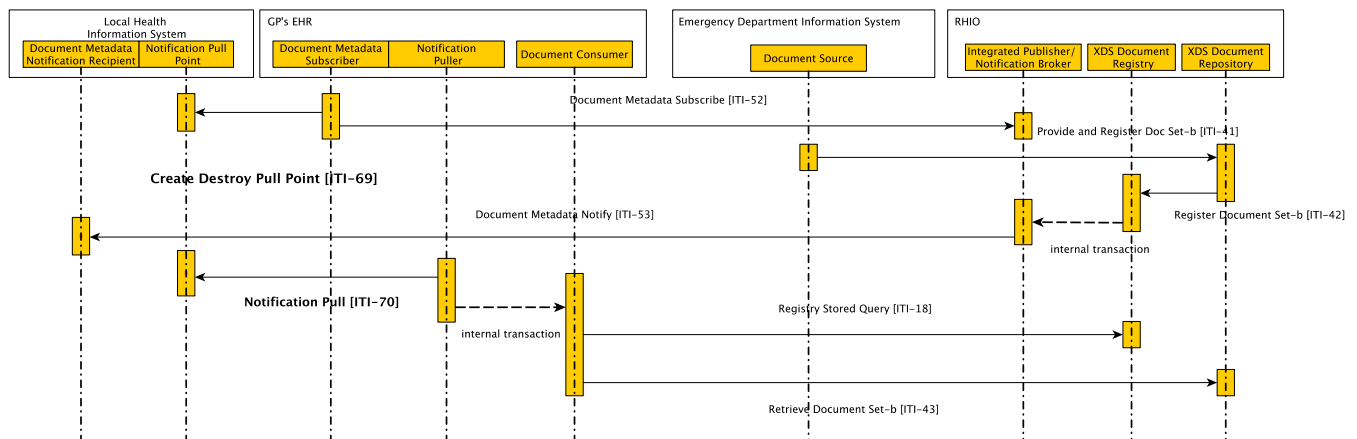


Figure 26.4.2.7.2-1: Sequence Diagram for GP’s EHR notification

375

The Document Metadata Subscriber that is grouped with the Notification Puller creates the pull point resource by the Create Destroy Pull Point [ITI-69] transaction. The response message of this transaction contains the endpoint of the Document Metadata Notification Recipient grouped with the Notification Pull Point.

380 The Document Metadata Subscriber creates a subscription for a specific patient with the Document Metadata Subscribe [ITI-52] transaction identifying the Document Metadata Notification Recipient/Notification Pull Point as target for notifications created.

385 A document published in the XDS environment (using the transactions Provide and Register Document Set-b [ITI-41] and Register Document Set-b [ITI-42]) matches with a subscription already created. The Document Metadata Notification Broker creates and sends a notification to the Document Metadata Notification Recipient/Notification Pull Point using the Document Metadata Notify transaction [ITI-53].

Without any specific trigger event, the Notification Puller can pull notifications stored in the Notification Pull Point using a Pull Notification [ITI-70] transaction.

390 The notification payload might then be used for querying and retrieving documents using the XDS transactions Register Stored Query [ITI-18] and Retrieve Document Set-b [ITI-43].

26.4.2.8 Use Case #8: Patient-independent tele-consultant notification

395 In this use case, a system used to provide tele-consulting services would be notified about Consult Requests published by Consult Requester systems. It is not possible to know in advance the patient for whom the consulting process is started and there are many consultants that can participate in the workflow. The consulting system can create just one patient-independent subscription for Consult Request documents.

26.4.2.8.1 Patient-independent tele-consultant scenario

400 Dr. Green is the clinician in charge of the consulting process. Dr. Green submits a subscription for documents with a document type of "Consult Request".

Dr. Brown is a Clinician who works for the Hope Clinic, a regional hospital specializing in neurological surgery treatment. This hospital and some other clinics provide tele-consulting services to many local hospitals.

405 Mr. White, after a car accident, is admitted to the Emergency Department in a local hospital. This hospital is not equipped with a Neurological ward so the ER physician, Dr. Young, decides to ask for a consult by a specialist. Using a Consult Requester system, Dr. Young publishes a Consult Request, looking for an available Consultant. This is done by publishing a Subscription for Consult Request document with a subscription expiration time that covers the whole work shift of the clinician and that has a "Consult Request" documentType.

410 When Dr. Young's Consult Request is published, the Notification Broker identifies a match with a patient-independent subscription and sends a notification to Dr. Green.

26.4.2.8.2 Tele-Consultant patient-independent notification Process Flow

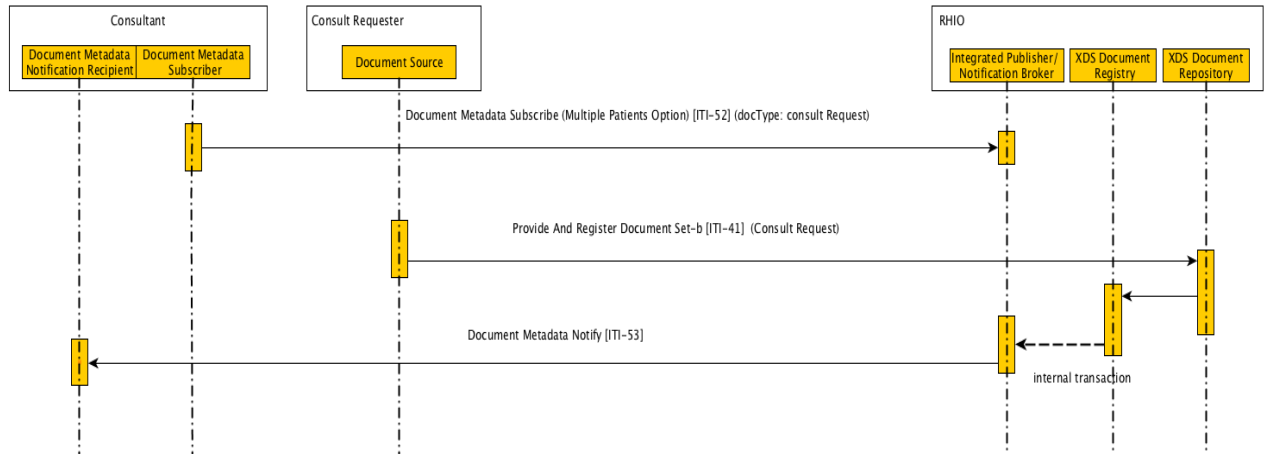


Figure 26.4.2.8.2-1: Interaction Diagram for patient-independent subscription

415

Editor: apply the following changes to Section 26.5

26.5 DSUB Security Considerations

420 The risk analysis for this profile enumerates assets, threats, and mitigations. The risk assessment spreadsheet is stored and available from IHE at http://wiki.ihe.net/images/4/46/DSUB_risk_assesment.xls.

425 The purpose of this risk assessment is to notify implementers of some of the risks that they need to consider in implementing DSUB actors. For general IHE risks and threats please see ITI TF-1: Appendix L. The implementers are also advised that many risks cannot be mitigated by the IHE profile and instead the responsibility for mitigation is transferred to the implementer, and occasionally to the XDS Affinity Domain and enterprises. In these instances, IHE's responsibility to notify affected parties is fulfilled through the following section.

430 A policy decision can be made during the Subscribe transaction whether the subscription is an authorized subscription and whether a notification/type of notification is authorized. (This could be based on the XUA identity, the consumer address value, etc.)

435 This profile does not include the solution to changes of policy between the subscribe time and notify time (which can be substantial). The recommendation is that the policy is enforced conservatively (i.e., the length of subscription can be determined by the Document Metadata Notification Broker). **The need to convey submissionSet metadata or Folder metadata can be related to access policies to content published.** An approach allows the access of content published in accordance to consent given by the patient. The consent is dynamic and can change during time. The availability of content can be discovered only asking the document-sharing

440 infrastructure. The creation of subscription is not dependent to access policies rules. If the Document Metadata Notification Broker sends the references, than the control of access policies is in query/retrieve transactions of the Document Metadata Notification Recipient. **It is suggested to use ihe:FolderMetadata topic or ihe:SubmissionSetMetadata topics when it is not desirable to convey more sensible content (wrapped in documentEntry metadata) using notification.**

445 Specific security considerations are presented in the Security Considerations section of each transaction in Volume 2.

Volume 2b – Transactions

Editor: Add Section 3.52.4.1.3.1

450

3.52.4.1.3.1 Folder Subscription Option

A Document Metadata Notification Broker supporting the Folder Subscription Option shall accept and understand a subscription created for an existing folder.

455

Editor: Add Section 3.52.5.1.4

3.52.5.1.4 ihe:FolderMetadata

460

This topic indicates that the events for which the subscription is made shall be creating or updating a Folder, and that the notification shall contain the full metadata describing each matching Folder object, as described in the Document Metadata Notify transaction in ITI TF-2b: 3.53.4.1.2.

Editor: Apply the following changes in Section 3.52.5.2

465

3.52.5.2 Building Filter Expressions

470

The XDS metadata, specified in ITI TF-3: 4.1, describes the objects which are used in a document registration. The Registry Stored Query transaction [ITI-18] uses a subset of the metadata to build a list of queries available to a XDS Document Consumer to search for documents with specific characteristics. The list of queries is in ITI TF-2a: 3.18.4.1.2.3.7. The transaction Document Metadata Subscribe uses the syntax of the Registry Stored Query [ITI-18] transaction for the creation of the filtering expression.

475

Filters can be created using the parameters of the FindDocuments, FindDocumentsByReferenceId, **GetFolders**, **FindFolders**, FindSubmissionSet queries defined within the Registry Stored Query transaction and use the syntax of the FindDocuments FindDocumentsByReferenceId, ~~or~~ FindSubmissionSets, **GetFolders or FindFolders** queries to express the filter.

The evaluation of filter expressions is based on the XDS metadata model. In this transaction, the stream of events for which subscriptions are possible is limited to events representing the

480 existence of **Folder**, SubmissionSet and documentEntry Objects. The Document Metadata Notification Broker becomes aware of such events either via a Document Metadata Publish transaction [ITI-54], or via other mechanisms not specified by IHE. The Document Metadata Notification Broker shall determine if there is a subscription which matches any of the Document Entry Objects, **Folder Objects** or SubmissionSet Object in an event.

485 A match means that if a Registry Stored Query, with the same parameters as the filter expression in the subscription, were sent to a XDS Document Registry containing the Document Entry Objects, ~~or~~ SubmissionSet Object **or Folder Object** from the event, the result of this Stored Query would contain one or more of these Objects.

490 In an XDS Affinity Domain context, the applicable events are likely to be Register Document Set [ITI-42] transaction containing one or more Document Entry objects. In this case, the Document Metadata Notification Broker may have to map between the model within which the events took place, and the XDS metadata model.

A good understanding of the Registry Stored Query transaction and the XDS metadata is necessary to understand how the filter expressions work. For example, if the filter expression below were implemented as a stored query on the registry

495

<i>Editor: Add Sections 3.52.5.2.3 , 3.52.5.2.4, 3.52.5.2.5</i>

3.52.5.2.3 Subscriptions for folders metadata

500 This type of filter shall be accepted by a Document Metadata Notification Broker that supports the Folder Subscription Option. Document Metadata Notification Broker that accepts a Subscribe Request containing a filter expression based on the GetFolders and FindFolders stored queries shall yield a match as described in Section 3.52.5.2. A Document Metadata Subscriber MAY be able to create a filter expression that includes XDSFolder.uniqueId, XDSFolder.patientId, XDSFolder.codeList. Parameters that can be used for creating the filter expression are described
505 below:

1. **\$XDSFolderPatientId**: this is a required parameter that matches with the metadata XDSFolder.patientId;
2. **\$XDSFolderUniqueId**: this parameter matches with the metadata XDSFolder.uniqueId. This is an optional parameter that contains the identifier defined for the Folder Object
510 subscribed;
3. **\$XDSFolderCodeList**: this parameter matches with the metadata XDSFolder.codeList. This allows creating a filter specifying the type of clinical activity that resulted in placing XDS Documents in an XDSFolder.

515 The AdHocQuery/@id attribute SHALL contain an identifier specific for the type of filter used in creating the subscription. The UUID that identifies subscriptions for Folder’s metadata is:

“urn:uuid:9376254e-da05-41f5-9af3-ac56d63d8ebd”

An example of subscription on a folder filter is presented below:

```
520 <?xml version="1.0" encoding="UTF-8"?>
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
  xmlns:a="http://www.w3.org/2005/08/addressing"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
525  xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
  xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope http://www.w3.org/2003/05/soap-
envelope http://www.w3.org/2005/08/addressing http://www.w3.org/2005/08/addressing/ws-addr.xsd
http://docs.oasis-open.org/wsn/b-2 http://docs.oasis-open.org/wsn/b-2.xsd
530 urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0 ../schema/ebRS/rim.xsd">
  <s:Header>
    <a:Action>http://docs.oasis-open.org/wsn/bw-
2/NotificationProducer/SubscribeRequest</a:Action>
    <a:MessageID>382dc7-8e84-9fdc-8443-48fd83bca938</a:MessageID>
    <a:To s:mustUnderstand="1">http://localhost:8080/services/initiatingGateway/query</a:To>
535 </s:Header>
  <s:Body>
    <wsnt:Subscribe>
      <!-- The Recipient on whose behalf the subscription is requested - the address where
540 the notification is to be sent -->
      <wsnt:ConsumerReference>
        <a:Address>https://NotificationRecipientServer/xdsBnotification</a:Address>
      </wsnt:ConsumerReference>
      <wsnt:Filter>
        <wsnt:TopicExpression Dialect="http://docs.oasis-open.org/wsn/t-
545 1/TopicExpression/Simple">ihe:FolderMetadata</wsnt:TopicExpression>
        <rim:AdhocQuery id="urn:uuid:9376254e-da05-41f5-9af3-ac56d63d8ebd">
          <rim:Slot name="$XDSFolderPatientId">
            <rim:ValueList>
              <rim:Value>'st3498702^^&amp;1.3.6.1.4.1.21367.2005.3.7&amp;ISO'</rim
550 :Value>
            </rim:ValueList>
          </rim:Slot>
          <rim:Slot name="$XDSFolderCodeList">
            <rim:ValueList>
              <rim:Value>('FolderCodeExample^^folderCodeListCodingScheme')</rim:Val
555 ue>
            </rim:ValueList>
          </rim:Slot>
        </rim:AdhocQuery>
      </wsnt:Filter>
      <wsnt:InitialTerminationTime>2010-05-31T00:00:00.00000Z</wsnt:InitialTerminationTime>
      </wsnt:Subscribe>
    </s:Body>
  </s:Envelope>
```

565

3.52.5.2.4 Patient-Independent Subscriptions for Document metadata

A Document Metadata Subscriber that supports the Patient-Independent Subscription Option shall be able to create subscriptions in accordance with the filter semantics defined in this section.

570 A Document Metadata Notification Broker that supports Patient-Independent Subscription Option shall accept subscription filters defined in this section.

This section defines a filter semantics that allow the subscription for patient-independent DocumentEntry metadata. Each filter parameter described below is optional; however, at least one of \$XDSDocumentEntryClassCode, \$XDSDocumentEntryTypeCode,

575 \$XDSDocumentEntryPracticeSettingCode or \$XDSDocumentEntryHealthcareFacilityTypeCode shall be specified.

1. **\$XDSDocumentEntryClassCode:** this parameter is optional and matches against the XDSDocumentEntry.classCode metadata elements in a given registry submission
- 580 2. **\$XDSDocumentEntryTypeCode:** this parameter is optional and matches against the XDSDocumentEntry.typeCode metadata elements in a given registry submission
3. **\$XDSDocumentEntryPracticeSettingCode:** this parameter is optional and matches against the XDSDocumentEntry.practiceSettingCode metadata elements in a given registry submission
- 585 4. **\$XDSDocumentEntryHealthcareFacilityTypeCode:** this parameter is optional and matches against the XDSDocumentEntry.healthcareFacilityTypeCode metadata elements in a given registry submission
5. **\$XDSDocumentEntryEventCodeList:** this parameter is optional and matches against the XDSDocumentEntry.eventCodeList metadata elements in a given registry submission
- 590 6. **\$XDSDocumentEntryConfidentialityCode:** this parameter is optional and matches against the XDSDocumentEntry.confidentialityCode metadata elements in a given registry submission
7. **\$XDSDocumentEntryFormatCode:** this parameter is optional and matches against the XDSDocumentEntry.formatCode metadata elements in a given registry submission
- 595 8. **\$XDSDocumentEntryAuthorPerson:** this parameter is optional and matches against the XDSDocumentEntry.author metadata elements in a given registry submission. All properties of this parameter specified in ITI TF-2a: 3.18.4.1.2.3.7.1 are applicable in this transaction.

600 The AdHocQuery/@id attribute SHALL be “urn:uuid:742790e0-aba6-43d6-9f1f-e43ed9790b79”.

An example of patient-independent subscription for documents is presented below:

```
605 <?xml version="1.0" encoding="UTF-8"?>
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
  xmlns:a="http://www.w3.org/2005/08/addressing"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```

610   xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
        xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
        xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope http://www.w3.org/2003/05/soap-
envelope http://www.w3.org/2005/08/addressing http://www.w3.org/2005/08/addressing/ws-addr.xsd
http://docs.oasis-open.org/wsn/b-2 http://docs.oasis-open.org/wsn/b-2.xsd
urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0 ../schema/ebRS/rim.xsd">
615   <s:Header>
        <a:Action>http://docs.oasis-open.org/wsn/bw-
2/NotificationProducer/SubscribeRequest</a:Action>
        <a:MessageID>382dc7-8e84-9fdc-8443-48fd83bca938</a:MessageID>
        <a:To s:mustUnderstand="1">http://notificationBroker/ad45m6o5493a</a:To>
620   </s:Header>
        <s:Body>
        <wsnt:Subscribe>
        <!-- The Recipient on whose behalf the subscription is requested - the address where
the notification is to be sent -->
625   <wsnt:ConsumerReference>
        <a:Address>https://NotificationRecipientServer/xdsBnotification</a:Address>
        </wsnt:ConsumerReference>
        <wsnt:Filter>
        <wsnt:TopicExpression Dialect="http://docs.oasis-open.org/wsn/t-
1/TopicExpression/Simple">ihe:MinimalDocumentEntry</wsnt:TopicExpression>
630   <rim:AdhocQuery id="urn:uuid:742790e0-aba6-43d6-9f1f-e43ed9790b79">
        <rim:Slot name="$XDSSubmissionSetSourceId">
        <rim:ValueList>
        <rim:Value>('Emergency
635   Department^^healthcareFacilityCodingScheme')</rim:Value>
        </rim:ValueList>
        </rim:Slot>
        </rim:AdhocQuery>
        </wsnt:Filter>
        <wsnt:InitialTerminationTime>2010-05-31T00:00:00.00000Z</wsnt:InitialTerminationTime>
640   </wsnt:Subscribe>
        </s:Body>
    </s:Envelope>

```

3.52.5.2.5 Patient-Independent Subscriptions for SubmissionSet metadata

645 A Document Metadata Subscriber that supports Patient-Independent Subscription Option shall be able to create subscriptions in accordance with the filter semantics defined in this section.

A Document Metadata Notification Broker that supports Patient-Independent Subscription Option shall accept subscription filters defined in this section.

650 This section defines a filter semantic that allows the subscription for patient-independent SubmissionSet metadata. Each filter parameter described below is optional; however, at least one of \$XDSSubmissionSetSourceId, \$XDSSubmissionSetAuthor or \$XDSSubmissionSetIntendedRecipient shall be specified.

1. **\$XDSSubmissionSetSourceId**: this optional parameter identifies the source of the content published by the submission and represent the value of the

655 **XDSSubmissionSets.sourceId** metadata;

2. **\$XDSSubmissionSetAuthor:** this optional parameter identifies the author person of the content published by the submission and represents the value of the XDSSubmissionSets.author metadata. This parameter may be multi-valued;

660 3. **\$XDSSubmissionSetIntendedRecipient:** this is an optional parameter for the subscription. A Document Metadata Subscriber shall be able to subscribe to this parameter in addition to other parameters that have direct correspondence with query parameters in the Registry Stored Query [ITI-18] transaction. This parameter represents the value of the XDSSubmissionSet.intendedRecipient metadata. This parameter may be multi-valued.

665

Note: intendedRecipient attribute is optional. If the parameter is specified, the filter matches only SubmissionSets where the intendedRecipient contains the value conveyed in the parameter.

The AdHocQuery/@id attribute SHALL contain

670 “urn:uuid:868cad3d-ec09-4565-b66c-1be10d034399”.

An example of patient-independent subscription for SubmissionSet metadata is presented below:

```
675 <?xml version="1.0" encoding="UTF-8"?>
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
  xmlns:a="http://www.w3.org/2005/08/addressing"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
  xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
680  xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope http://www.w3.org/2003/05/soap-
envelope http://www.w3.org/2005/08/addressing http://www.w3.org/2005/08/addressing/ws-addr.xsd
http://docs.oasis-open.org/wsn/b-2 http://docs.oasis-open.org/wsn/b-2.xsd
urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0 ../schema/ebRS/rim.xsd">
  <s:Header>
685    <a:Action>http://docs.oasis-open.org/wsn/bw-
2/NotificationProducer/SubscribeRequest</a:Action>
    <a:MessageID>382dc7-8e84-9fdc-8443-48fd83bca938</a:MessageID>
    <a:To s:mustUnderstand="1">http://notificationBroker/qiwmn34dekE</a:To>
690  </s:Header>
  <s:Body>
    <wsnt:Subscribe>
      <!-- The Recipient on whose behalf the subscription is requested - the address where
the notification is to be sent -->
      <wsnt:ConsumerReference>
695        <a:Address>https://NotificationRecipientServer/xdsBnotification</a:Address>
      </wsnt:ConsumerReference>
      <wsnt:Filter>
        <wsnt:TopicExpression Dialect="http://docs.oasis-open.org/wsn/t-
1/TopicExpression/Simple">ihe:SubmissionSetMetadata</wsnt:TopicExpression>
700        <rim:AdhocQuery id="urn:uuid:868cad3d-ec09-4565-b66c-1be10d034399">
          <rim:Slot name="$XDSSubmissionSetIntendedRecipient">
            <rim:ValueList>
              <rim:Value>('Some Hospital%')</rim:Value>
              <rim:Value>('|Welby%')</rim:Value>
705            </rim:ValueList>
          </rim:Slot>
        </rim:AdhocQuery>
      </wsnt:Filter>
    </wsnt:Subscribe>
  </s:Body>
</s:Envelope>
```

710

```

        </rim:Slot>
    </rim:AdhocQuery>
    </wsnt:Filter>
    <wsnt:InitialTerminationTime>2010-05-31T00:00:00.000000Z</wsnt:InitialTerminationTime>
</wsnt:Subscribe>
</s:Body>
</s:Envelope>
    
```

Editor: Apply the following changes in Table 3.52.5.3-1

715

Table 3.52.5.3-1: Topics and Filter Expression Combinations

Filter Expression	Topic Expression
subscription for DocumentEntry	ihe:FullDocumentEntry
	ihe:MinimalDocumentEntry
<u>subscription for Folders</u>	<u>ihe:FolderMetadata</u>
subscription for SubmissionSets	ihe:SubmissionSetMetadata
<u>subscription for DocumentEntry (Patient-Independent)</u>	<u>ihe:FullDocumentEntry</u>
	<u>ihe:MinimalDocumentEntry</u>
<u>subscription for SubmissionSet (Patient-Independent)</u>	<u>ihe:SubmissionSetMetadata</u>

Editor: Apply the following changes in Section 3.52.6

720 **3.52.6 Security Considerations**

The risk assessment for the Document Metadata Subscribe transaction is described in the risk assessment spreadsheet for the Document Metadata Subscription Profile, which is available from IHE at http://wiki.ihe.net/images/4/46/DSUB_risk_assesment.xls. The derived mitigations are as follows:

- 725
- Document Metadata Subscriber and Document Metadata Notification Broker shall be grouped with an ATNA Secure Node or Secure Application for Node Authentication and Audit Trails
 - The use of encrypted TLS is recommended when the transmission is not otherwise secured (e.g., transmission over a secure network)

730 As it is possible through the document metadata subscribe transaction to maliciously overload the Document Metadata Notification Recipient actors, it is recommended that a strong authentication be used in combination with access rights enforcement and that authentication data should be conveyed through XUA. This recommendation also addresses the possibility of malicious cancellations of subscriptions.

- 735 Additionally, it is recommended that organizational measures be taken to avoid:
- overload of a Document Metadata Notification Recipient through subscription which cannot be cancelled because the subscription id has been lost e.g., through an administrative service allowing cancellation of subscription under well-defined circumstances
 - 740 • cancellation of a subscription unnoticed by the intended document metadata notification recipient e.g., through an informative message (out of the scope of this profile) sent to the intended recipient

The Document Metadata Subscriber that supports the Patient-Independent Subscription Option can create a subscription without specifying the patientId subscription parameter.

- 745 **This functionality increases risks connected with policy changes between subscription time and notification event. It is recommended to take organizational/technical measures to reduce this risk.** This profile provides the ihe:MinimalDocumentEntry topic expression to avoid disclosing sensitive information. Using this type of topic expression allows delegation of the access control decisions to the Document Sharing infrastructure.

750

*Editor: Apply the following changes in Section 3.53.4.1.2. Note: **highlighted** text shall be added **bold** in the final text.*

3.53.4.1.2 Message Semantics

- 755 The Notify message shall comply with the requirements in the WS-BaseNotification standard. Note that the value of the WS-Addressing Action element is prescribed in the standard, and differs from the requirements of ITI TF-2x:Appendix V.

The Notify message convey in the *wsnt:Notify/wsnt:NotificationMessage/wsnt:Message* the event that matched with a subscription. Depending on the event which triggered the notification,
760 there may be one or more Document Entry Objects, **Folder Objects**, or SubmissionSet Objects whose metadata matches the filter conditions of any particular subscription. This transaction defines the following structures for conveying a Notify message:

- A Full Notification**, which shall be sent if the subscription request included the topic ihe:FullDocumentEntry (see ITI TF-2b: 3.52.5.1). In this case, the notification
765 shall consist of parts of the payload of a Register Document Set-b Transaction as defined in ITI TF-2b: 3.42.4.1. The <lc:SubmitObjectsRequest> element is the only child of the *wsnt:Notify/wsnt:NotificationMessage/wsnt:Message* element in this message. The <rim:RegistryObjectList> element shall be the only child of the <lc:SubmitObjectsRequest> element. Only <rim:ExtrinsicObject> elements
770 representing Document Entries shall be sent within the <rim:RegistryObjectList> element.

775 **A Minimal Notification**, which shall be sent if the subscription request included the topic `ihe:MinimalDocumentEntry`. In this case, the notification shall consist of parts of the payload of a Register Document Set-b Transaction as defined in ITI TF-2b: 3.42.4.1. The `<lcm:SubmitObjectsRequest>` element is the only child of the `wsnt:Notify/wsnt:NotificationMessage/wsnt:Message` element in this message. The `<rim:RegistryObjectList>` element shall be the only child of the `<lcm:SubmitObjectsRequest>` element. Only `<rim:ObjectRef>` elements representing Document Entries shall be sent within the `<rim:RegistryObjectList>` element.

785 **A Folder Notification**, which shall be sent if the subscription request included the topic `ihe:FolderMetadata`. **A Document Metadata Notification Broker that supports Folder Subscription Option shall be able to create this type of notification. In this case, the response consists of parts of the payload of a Register Document Set-b Transaction as defined in ITI TF-2b: 3.42.4.1. The <lcm:SubmitObjectsRequest> element is the only child of the wsnt:Notify/wsnt:NotificationMessage/wsnt:Message element in this message. The <rim:RegistryObjectList> element shall be the only child of the <lcm:SubmitObjectsRequest> element. Only one <rim:RegistryPackage> element representing the folder object shall be sent within the <rim:RegistryObjectList> element and shall be characterized by classification scheme: classificationScheme="urn:uuid:d9d542f3-6cc4-48b6-8870-ea235fbc94c2" (that represents an object of Folder type).**

795 **A submissionSet Notification**, which shall be sent if the subscription request included the topic `ihe:SubmissionSetMetadata`. In this case the response consists of parts of the payload of a Register Document Set-b Transaction as defined in ITI TF-2b:3.42.4.1. The `<lcm:SubmitObjectsRequest>` element is the only child of the `wsnt:Notify/wsnt:NotificationMessage/wsnt:Message` element in this message. The `<rim:RegistryObjectList>` element shall be the only child of the `<lcm:SubmitObjectsRequest>` element. Only one `<rim:RegistryPackage>` element representing the `submissionSet` object shall be sent within the `<rim:RegistryObjectList>` element and shall be characterized by classification scheme: `classificationScheme="urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd"` (that represents an object of `submissionSet` type).

805 There shall be a single `wsnt:Notify/wsnt:NotificationMessage/wsnt:Message` element in this transaction. If multiple objects need to be represented in a single notification, the WS-BaseNotification standard allows this to be done.

810

Editor: add Section 3.53.4.1.4.4

3.53.4.1.4.4 Folder Notification Example (ihe:FolderMetadata)

```
815 <?xml version="1.0" encoding="UTF-8"?>
      <s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
816         xmlns:a="http://www.w3.org/2005/08/addressing"
817         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
818         xmlns:wsnt="http://docs.oasis-open.org/wsn/b-2"
819         xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0"
820         xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
821         xsi:schemaLocation="http://www.w3.org/2003/05/soap-envelope http://www.w3.org/2003/05/soap-
822 envelope http://www.w3.org/2005/08/addressing http://www.w3.org/2005/08/addressing/ws-addr.xsd
823 http://docs.oasis-open.org/wsn/b-2 http://docs.oasis-open.org/wsn/b-2.xsd
824 urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0 ../schema/ebRS/rim.xsd" urn:oasis:names:tc:ebxml-
825 regrep:xsd:lcm:3.0 ../schema/ebRS/lcm.xsd">
          <s:Header>
            <a:Action>http://docs.oasis-open.org/wsn/bw-2/NotificationConsumer/Notify</a:Action>
            <a:MessageID>382dcdca-8e87-9fdf-8446-48fd83bca93b</a:MessageID>
            <a:To>https://NotificationRecipientServer/xdsBnotification</a:To>
          </s:Header>
          <s:Body>
            <wsnt:Notify>
              <wsnt:NotificationMessage>
                <wsnt:SubscriptionReference>
                  <a:Address>https://NotificationBrokerServer/Subscription/382dcdc7-8e84-9fdc-
835 8443-48fd83bca938</a:Address>
                </wsnt:SubscriptionReference>
                <wsnt:Topic Dialect="http://docs.oasis-open.org/wsn/t-1/TopicExpression/Simple"
                >ihe:FolderMetadata</wsnt:Topic>
                <wsnt:ProducerReference>
                  <a:Address>https://ProducerReference</a:Address>
                </wsnt:ProducerReference>
                <wsnt:Message>
                  <lcm:SubmitObjectsRequest
845                 xsi:schemaLocation="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0
846                 ../schema/ebRS/lcm.xsd"
847                 xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0"
848                 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
849                 xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
850                 xmlns:rs="urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0">
                    <rim:RegistryObjectList>
                      <rim:RegistryPackage id="Folder01">
                        <!-- here all the Folder metadata -->
855
                      </rim:RegistryPackage>
                      <rim:Classification id="Fol" classifiedObject="Folder01"
                      classificationNode="urn:uuid:d9d542f3-6cc4-48b6-8870-
860 ea235fbc94c2"/>
                    </rim:RegistryObjectList>
                  </lcm:SubmitObjectsRequest>
                </wsnt:Message>
              </wsnt:NotificationMessage>
            </wsnt:Notify>
          </s:Body>
        </s:Envelope>
```

Editor: make the following changes in Section 3.53.5.1.1

870

Document Folder SubmissionSet <small>(AuditMessage/ ParticipantObjectIdentification)</small>	ParticipantObjectTypeCode	M	“2” (System)
	ParticipantObjectTypeCodeRole	M	“3” (report)
	<i>ParticipantObjectDataLifeCycle</i>	U	<i>not specialized</i>
	ParticipantObjectTypeCode	M	EV("urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1", IHE, "DocumentEntry") <u>EV("urn:uuid:d9d542f3-6cc4-48b6-8870-ea235fbc94c2", IHE, "Folder")</u> EV(urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd, IHE, "SubmissionSet") The Document Metadata Notification Recipient shall include one of these values in accordance with the specific topic used for the creation of the notification. These codes are already defined by IHE and they are the UUIDs which shall be used in constructing and interpreting XDS objects in a submission request.
	<i>ParticipantObjectSensitivity</i>	U	<i>not specialized</i>
	ParticipantObjectID	M	The value of XDSDocumentEntry.entryUUID, XDSFolder.uniqueId, or XDSSubmissionSet.uniqueId. The Document Metadata Notification Recipient shall include one of these values in accordance with the specific topic used for the creation of the notification
	ParticipantObjectName	C	If known the value of homeCommunityId
	<i>ParticipantObjectQuery</i>	U	<i>not specialized</i>
	<i>ParticipantObjectDetail</i>	U	<i>not specialized</i>

Editor: make the following changes in Section 3.53.5.1.2

875

880

Document Folder SubmissionSet (AuditMessage/ ParticipantObjectIdentification)	ParticipantObjectTypeCode	M	“2” (System)
	ParticipantObjectTypeCodeRole	M	“3” (report)
	<i>ParticipantObjectDataLifeCycle</i>	<i>U</i>	<i>not specialized</i>
	ParticipantObjectTypeCode	M	EV("urn:uuid:7edca82f-054d-47f2-a032-9b2a5b5186c1", IHE, "DocumentEntry") <u>EV("urn:uuid:d9d542f3-6cc4-48b6-8870-ea235fbc94c2", IHE, "Folder")</u> EV(urn:uuid:a54d6aa5-d40d-43f9-88c5-b4633d873bdd, IHE, "SubmissionSet") The Document Metadata Notification Broker shall include one of these values in accordance with the specific topic used for the creation of the notification. These codes are already defined by IHE and they are the UUIDs which shall be used in constructing and interpreting XDS objects in a submission request.
	<i>ParticipantObjectSensitivity</i>	<i>U</i>	<i>not specialized</i>
	ParticipantObjectID	M	The value of XDSDocumentEntry.entryUUID, XDSFolder.uniqueId, or XDSSubmissionSet.uniqueId. The Document Metadata Notification Broker shall include one of these values in accordance with the specific topic used for the creation of the notification
	ParticipantObjectName	C	If known the value of homeCommunityId
	<i>ParticipantObjectQuery</i>	<i>U</i>	<i>not specialized</i>
	<i>ParticipantObjectDetail</i>	<i>U</i>	<i>not specialized</i>

885 *Editor: make the following changes in Section 3.54.4.1.1*

3.54.4.1.1 Trigger

890 When an event occurs for which a subscription may exist, the Document Metadata Publisher will trigger a Notification message to the Document Metadata Notification Broker. Events that could trigger a notification are publication of or update to a DocumentEntry, **Folder** or SubmissionSet.

Editor: make the following changes in Section 3.54.4.1.2

3.54.4.1.2 Message Semantics

895 The Document Metadata Publisher Actor shall use a Notify message to communicate published objects to the Document Metadata Notification Broker.

This message shall have one *<NotificationMessage>* element.

This element SHALL have two child elements:

- *<ProducerReference>* that identifies the source of the data published.
- 900 • *<Message>* that identifies published objects. This element shall have a single child element *<lcm:SubmitObjectsRequest>* that has only one child element *<rim:RegistryObjectList>*. This element conveys a list of SubmissionSet, **Folder**, and DocumentEntry objects.

905 Note: SubmissionSet **and Folder** objects are constructed from *<rim:RegistryObject>* elements and must include the *<rim:Classification>* **that distinguishes SubmissionSet from Folder objects**.

Note that there is no subscription information in the Notify message in the Publish transaction.

Volume 2c – Transactions

910 *Add Section 3.69*

3.69 Create Destroy Pull Point

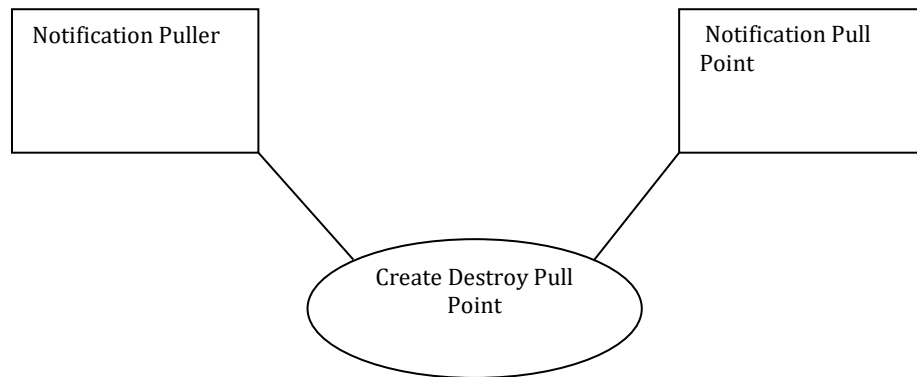
915 This section corresponds to the Transaction ITI-69 of the IHE IT Infrastructure Technical Framework. Transaction ITI-69 is used by the Notification Puller and by the Notification Pull Point.

3.69.1 Scope

This transaction involves a Request by the Notification Puller to the Notification Pull Point to create a Pull Point resource, and a Response to convey the information that the Request was successfully processed. This Response identifies the endpoint where notifications are delivered.

920 The Notification Puller also uses this transaction to destroy a Pull Point resource which is no longer needed.

3.69.2 Actor Roles



925

Actor:	Notification Puller
Role:	Sends a request to create (or delete) a Pull Point resource,

Actor:	Notification Pull Point
Role:	Manages the creation or the destruction of the Pull Point resource.

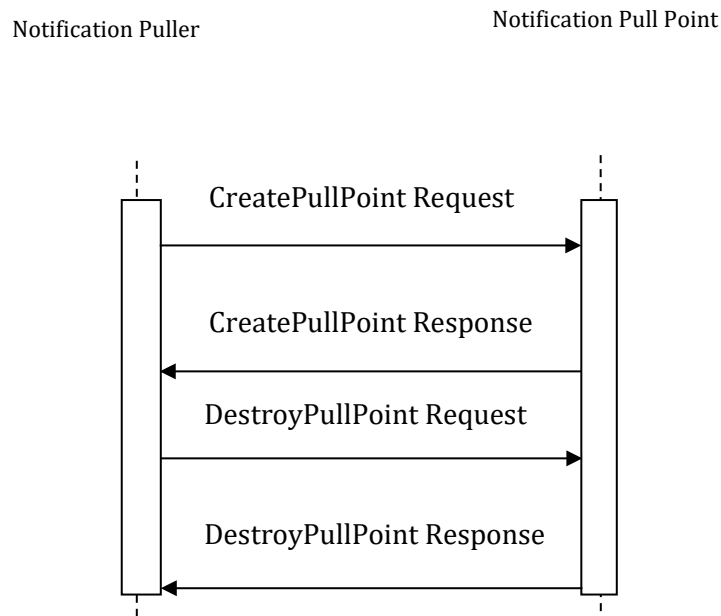
3.69.3 Referenced Standards

- OASIS Web Services Notification Family of standards
- WS-BaseNotification 1.3 OASIS standard
- WS-BrokeredNotification 1.3 OASIS Standard
- WS-Topics 1.3 OASIS Standard
- WS-BaseFaults
- IHE ITI TF-2x: Appendix V
- WS-Addressing OASIS Standard

930

935

3.69.4 Interaction Diagram



3.69.4.1 CreatePullPoint Request message

A Notification Pull Point creates a Pull Point resource in response to each CreatePullPoint Request and collects all notifications destined for the requesting Notification Puller.

940

Within the Notification Pull Point, each Pull Point resource allows the storing and managing of notifications.

945 A Pull Point resource is associated with a Notification Puller. A Pull Point resource is an abstract concept that creates a relationship between a Notification Puller and notifications stored for that actor in the Pull Point.

The Notification Pull Point serves as a Pull Point resource “factory” in processing CreatePullPoint Request messages. It can be asked to create Pull Point resources by many Notification Puller actors. The Notification Pull Point can manage many Pull Point resources for each Notification Puller.

950 The creation of a Pull Point resource requires grouping the Notification Pull Point with a Document Metadata Notification Recipient for receiving notifications sent by the Document Metadata Notification Broker.

955 If many Notification Puller actors are involved in the notification system, the Notification Pull Point is grouped with many Document Metadata Notification Recipient actors (see Figure 3.69.4.1-1). When a Notification Puller sends a CreatePullPoint Request message, the Notification Pull Point returns an endpoint in the CreatePullPoint Response message. This endpoint is associated with a Document Metadata Notification Recipient. The Document Metadata Notification Recipient SHALL store in the Pull Point resource the notifications received. This is an additional requirement for a Document Metadata Notification Recipient that is grouped with a Notification Pull Point.

960 The Notification Puller uses this endpoint for subsequent transactions (subscription requests, pulling of notifications and destroying of the Pull Point resource itself).

The way to store notifications and how to associate notifications to the specific Pull Point resources are not described and are out of scope of this transaction.

965

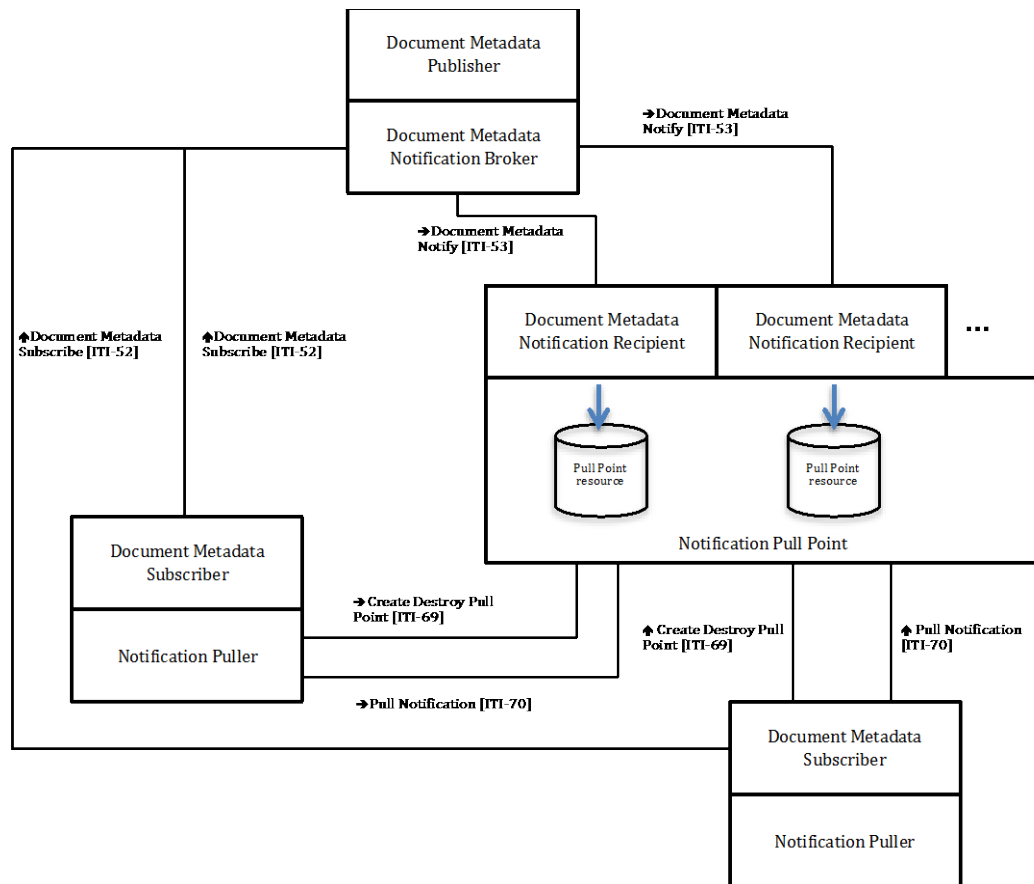


Figure 3.69.4.1-1: Pull-style Notification Framework

3.69.4.1.1 Trigger Events

970 When the Notification Puller wishes to create a new Pull Point resource, it SHALL send a CreatePullPoint Request to a Notification Pull Point.

3.69.4.1.2 Message Semantics

The CreatePullPoint Request shall comply with the requirements in the WS-BaseNotification standard.

975 This message does not convey information to the Notification Pull Point, but is used only as trigger for internal subsequent actions.

The WS-Addressing [action] Message Addressing Property MUST contain the URI <http://docs.oasis-open.org/wsn/bw-2/CreatePullPoint/CreatePullPointRequest>.

3.69.4.1.3 Expected Actions

980 Upon receipt of the CreatePullPoint Request, the Notification Pull Point SHALL create a new Pull Point resource. The Document Metadata Notification Recipient/Notification Pull Point can act in one of two ways:

- 985 1. If the CreatePullPoint Request message is successfully processed, the Notification Pull Point SHALL respond with CreatePullPoint Response message. The behavior of the Notification Pull Point when it is no longer capable of accumulating notifications is out of scope for this transaction.
- 990 2. If the Notification Pull Point cannot respond to the CreatePullPoint Request message with the CreatePullPoint Response message for an application-level fault, then it SHALL send a SOAP fault in accordance to the WS-BaseFault specification. The WS-BaseNotification specification defines the following fault associated with failure to process the CreatePullPoint Request message:
 - UnableToCreatePullPointFault.

3.69.4.1.4 Example SOAP Encoding of the CreatePullPoint Request Message

995

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw2/PullPoint/CreatePullPointRequest
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:CreatePullPoint/>
  </s:Body>
</s:Envelope>
```

3.69.4.2 CreatePullPoint Response message

1000 If the Notification Pull Point can create a Pull Point resource dedicated to the specific Notification Puller, the Notification Pull Point SHALL respond to the request received with a CreatePullPoint Response, as described in the WS-BaseNotification standard.

3.69.4.2.1 Trigger Events

This message is created in response to a request of creation of a Pull Point resource that is successfully processed. The Response message can be created once the endpoint of the Pull Point resource is identified.

1005 **3.69.4.2.2 Message Semantics**

The CreatePullPoint Response shall comply with the requirements in the WS-BaseNotification standard.

This message, “The WS-Addressing [action] Message Addressing Property”, SHALL contain the URI:

1010 <http://docs.oasis-open.org/wsn/bw-2/CreatePullPoint/CreatePullPointResponse>.

The CreatePullPoint response message SHALL contain the attribute:
/wsnt:CreatePullPointResponse/wsnt:PullPoint.

- This component is an EndpointReference, as defined by WS-Addressing, which is a reference to the Pull Point resource created during the processing of the CreatePullPoint Request message.
- This SHALL be the Web-Service endpoint for the Document Metadata Notify [ITI-53] transaction on the Document Metadata Notification Recipient that is grouped with the Notification Pull Point.

3.69.4.2.3 Expected Actions

1020 The CreatePullPoint Response message provides to the Notification Puller the endpoint needed for creating subscriptions to the Document Metadata Notification Broker. The same endpoint SHALL be used for the pulling of the notification stored by the Notification Pull Point and/or the destroying of the Pull Point resource itself as needed using this transaction.

3.69.4.2.4 Example SOAP Encoding of the CreatePullPoint Response Message

1025

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-2/PullPoint/CreatePullPointResponse
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:CreatePullPointResponse>
      <wsnt:PullPoint>
        <wsa:Address>...</wsa:Address>
        ...
      </wsnt:PullPoint>
    </wsnt:CreatePullPointResponse>
  </s:Body>
</s:Envelope>
```

3.69.4.3 DestroyPullPoint Request message

1030 If the Notification Puller wants to terminate the Pull Point resource it SHALL send a DestroyPullPoint Request message. The request of destruction is directly targeted to the endpoint of reference that identifies the Pull Point resource and the grouped Notification Pull Point/Document Metadata Notification Recipient.

3.69.4.3.1 Trigger Events

1035 This message is created when the Notification Puller does not want to be involved in a notification system, or when it is necessary to remove a Pull Point resource for organizational reasons. The Document Metadata Subscriber grouped with the Notification Puller starts the process for unsubscribe filters created using the Pull Point resource endpoint as target for notifications created before to start the destruction.

3.69.4.3.2 Message Semantics

1040 The DestroyPullPoint Request shall comply with the requirements in the WS-BaseNotification standard. The WS-Addressing [action] Message Addressing Property SHALL contain the URI: <http://docs.oasis-open.org/wsn/bw-2/PullPoint/DestroyPullPointRequest>.

3.69.4.3.3 Expected Actions

1045 If the DestroyPullPoint Request is successfully processed, once this message is received by the Notification Pull Point, the Pull Point resource SHALL attempt to destroy itself, responding with the DestroyPullPoint Response message. The Pull Point SHALL discard all queued notifications.

If the Notification Pull Point does not respond to the DestroyPullPoint Request message with the DestroyPullPointResponse message, then it SHALL send a fault. The WS-BaseNotification specification defines the following faults associated with failure to process the DestroyPullPoint Request message:

- 1050
1. If the Pull Point resource identified in the DestroyPullPoint Request message is not known to the Notification Pull Point, it SHALL send a fault specified by the WS-Resource [WS-Resource] specification:
 - ResourceUnknownFault
 - 1055 2. If the Notification Pull Point was unable to destroy the Pull Point resource for some reason, it SHALL send a fault specified by the WS-BaseNotification specification:
 - UnableToDestroyPullPointFault.

3.69.4.3.4 Example SOAP Encoding of the DestroyPullPoint Request Message

1060

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>http://docs.oasis-open.org/wsn/bw-
2/PullPoint/DestroyPullPointRequest
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:DestroyPullPoint/>
  </s:Body>
</s:Envelope>
```

3.69.4.4 DestroyPullPoint Response message

The Notification Pull Point responds to the Notification Puller creating a DestroyPullPoint Response that attests to the success of the destruction process.

1065 3.69.4.4.1 Trigger Events

If the DestroyPullPoint Request message is successfully processed, the Notification Pull Point SHALL respond with the DestroyPullPoint Response message.

3.69.4.4.2 Message Semantics

1070 The DestroyPullPoint Response shall comply with the requirements in the WS-BaseNotification standard. The WS-Addressing [action] Message Addressing Property SHALL contain the URI: <http://docs.oasis-open.org/wsn/bw-2/PullPoint/DestroyPullPointResponse>.

3.69.4.4.3 Expected Actions

The Notification Puller should discard the endpoint of the Pull Point resources.

3.69.4.4.4 Example SOAP Encoding of the DestroyPullPoint Response Message

1075

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-
2/PullPoint/DestroyPullPointResponse
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:DestroyPullPointResponse/>
  </s:Body>
</s:Envelope>
```



```
</s:Body>  
</s:Envelope>
```

3.69.5 Security Considerations

This section addresses security considerations related to the Create Destroy Pull Point transaction.

1080 The risks connected to this transaction are:

Authentication of the Notification Puller is required. This avoids requests made by not reputable actors. The endpoint has to be used by the Puller for the creation of subscriptions that can be sent to a recognized Notification Recipient. If the authentication of the Puller was not required, the notifications will be sent anyway to the grouped Notification Pull Point, because the Broker recognizes as a reputable actor the Document Metadata Notification Recipient.

1085

- Mitigation:

- Node Authentication using ATNA: to assure that the requesting system is an authorized requesting system.

1090

- User authentication using XUA: when it is necessary to know who the user is.

3.69.5.1 Security Audit Considerations

If a Request of creation of Pull Point or a Request of destruction is processed, the Notification Pull Point and Notification Puller SHALL create an Audit Record in accordance to the structure defined below. These events are of type “Application Activity” as defined in ITI TF-2a: Table 3.20.4.1.1.1-1. The actors involved in the transaction SHALL create audit data in conformance with DICOM¹ Part 15 “Application Activity”.

1095

3.69.5.1.1 Notification Pull Point audit message

1100

¹ DICOM is the registered trademark of the National Electrical Manufacturers Association for its standards publications relating to digital communications of medical information.

IHE IT Infrastructure Technical Framework Supplement – Extensions to the Document Metadata Subscription Profile

Real World Entities	Field Name	Opt.	Value Constraints
Event	EventID	M	EV (110100, DCM, "Application Activity")
	EventActionCode	M	Enumerated Value C = Create (in case of creation of the Pull Point) or D= Delete (in case of deletion of the Pull Point)
	EventDateTime	M	time of creation or deletion of the Pull Point resource
	<i>EventOutcomeIndicator</i>	<i>M</i>	<i>not specialized</i>
	EventTypeCode	M	EV("ITI-69", "IHE Transactions", "Create Destroy Pull Point")
Source (Notification Puller) (1)			
Destination (Notification Pull Point) (1)			
Pull Point (1)			
Human Requestor (0..1)			

1105

Source: AuditMessage/ ActiveParticipant (1)	<i>UserID</i>	<i>U</i>	<i>not specialized</i>
	<i>AlternativeUserID</i>	<i>U</i>	<i>not specialized</i>
	<i>UserName</i>	<i>U</i>	<i>not specialized</i>
	<i>UserIsRequestor</i>	<i>U</i>	<i>not specialized</i>
	RoleIDCode	M	EV (110153, DCM, "Source")
	NetworkAccessPointType Code	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address

Destination: AuditMessage/ ActiveParticipant (1)	UserID	M	Metadata Notification Pull Point SOAP URI
	AlternativeUserID	U	the process ID as used within the local operating system in the local system of logs
	<i>UserName</i>	<i>U</i>	<i>not specialized</i>
	<i>UserIsRequestor</i>	<i>U</i>	<i>not specialized</i>
	RoleIDCode	M	EV (110152, DCM, "Destination")
	NetworkAccessPointType Code	U	"1" for machine (DNS) name "2" for IP address
	NetworkAccessPointID	U	The machine name or IP address

1110

Human Requestor (if known) AuditMessage/Active Participant	UserID	M	The person who wants to create (or destroying) a Pull Point resource
	AlternativeUserID	U	not specialized
	UserName	U	not specialized
	UserIsRequestor	U	not specialized
	RoleIDCode	M	not specialized
	NetworkAccessPointTypeCode	U	not specialized
	NetworkAccessPointID	U	not specialized

Pull Point (AuditMessage/ParticipantObjectIdentification)	ParticipantObjectTypeCode	M	“2” (SYSTEM)
	ParticipantObjectTypeCodeRole	M	“4” (Resource)
	ParticipantObjectDataLifeCycle	U	not specialized
	ParticipantObjectIDTypeCode	M	not specialized
	ParticipantObjectSensitivity	U	not specialized
	ParticipantObjectID	M	The URL of the Pull Point resource
	ParticipantObjectName	U	not specialized
	ParticipantObjectQuery	U	not specialized
ParticipantObjectDetail	U	not specialized	

1115

3.69.5.1.2 Notification Puller audit message

Real World Entities	Field Name	Opt.	Value Constraints
Event	EventID	M	EV (110100, DCM, "Application Activity")
	EventActionCode	M	Enumerated Value C = Create (in case of creation of the Pull Point) or D= Delete (in case of deletion of the Pull Point)
	EventDateTime	M	time of creation or deletion of the Pull Point resource
	EventOutcomeIndicator	M	not specialized
	EventTypeCode	M	EV("ITI-69", "IHE Transactions", "Create Destroy Pull Point")
Source (Notification Puller) (1)			
Destination (Notification Pull Point) (1)			
Pull Point (1)			
Human Requestor (0..1)			

IHE IT Infrastructure Technical Framework Supplement – Extensions to the Document Metadata Subscription Profile

Source: AuditMessage/ ActiveParticipant	<i>UserID</i>	<i>U</i>	<i>not specialized</i>
	<i>AlternativeUserID</i>	<i>U</i>	<i>not specialized</i>
	<i>UserName</i>	<i>U</i>	<i>not specialized</i>
	<i>UserIsRequestor</i>	<i>U</i>	<i>not specialized</i>
	<i>RoleIDCode</i>	<i>M</i>	EV (110153, DCM, “Source”)
	<i>NetworkAccessPointTypeCode</i>	<i>U</i>	“1” for machine (DNS) name “2” for IP address
	<i>NetworkAccessPointID</i>	<i>U</i>	The machine name or IP address

Destination: AuditMessage/ ActiveParticipant (1)	<i>UserID</i>	<i>M</i>	Metadata Notification Pull Point SOAP URI
	<i>AlternativeUserID</i>	<i>U</i>	<i>not specialized</i>
	<i>UserName</i>	<i>U</i>	<i>not specialized</i>
	<i>UserIsRequestor</i>	<i>U</i>	<i>not specialized</i>
	<i>RoleIDCode</i>	<i>M</i>	EV (110152, DCM, “Destination”)
	<i>NetworkAccessPointTypeCode</i>	<i>U</i>	“1” for machine (DNS) name “2” for IP address
	<i>NetworkAccessPointID</i>	<i>U</i>	The machine name or IP address

1120

Human Requestor (if known) AuditMessage/Active Participant	<i>UserID</i>	<i>M</i>	The person that wants to create (or destroying) a Pull Point resource
	<i>AlternativeUserID</i>	<i>U</i>	<i>not specialized</i>
	<i>UserName</i>	<i>U</i>	<i>not specialized</i>
	<i>UserIsRequestor</i>	<i>U</i>	<i>not specialized</i>
	<i>RoleIDCode</i>	<i>M</i>	<i>not specialized</i>
	<i>NetworkAccessPointTypeCode</i>	<i>U</i>	<i>not specialized</i>
	<i>NetworkAccessPointID</i>	<i>U</i>	<i>not specialized</i>

Pull Point (AuditMessage/ParticipantObjectIdentification)	<i>ParticipantObjectTypeCode</i>	<i>M</i>	“2” (SYSTEM)
	<i>ParticipantObjectTypeCodeRole</i>	<i>M</i>	“4” (Resource)
	<i>ParticipantObjectDataLifeCycle</i>	<i>U</i>	<i>not specialized</i>
	<i>ParticipantObjectIDTypeCode</i>	<i>M</i>	<i>not specialized</i>
	<i>ParticipantObjectSensitivity</i>	<i>U</i>	<i>not specialized</i>
	<i>ParticipantObjectID</i>	<i>M</i>	The URL of the Pull Point resource
	<i>ParticipantObjectName</i>	<i>U</i>	<i>not specialized</i>
	<i>ParticipantObjectQuery</i>	<i>U</i>	<i>not specialized</i>
<i>ParticipantObjectDetail</i>	<i>U</i>	<i>not specialized</i>	

3.69.5.1.3 Notification Pull Point Actor Specific Security Considerations

1125 For the receiving of notifications pushed by the Document Metadata Notification Broker, the grouped Document Metadata Notification Recipient/Notification Pull Point has to grant the requirement of Synchronous Web Services described in ITI TF-2x: Appendix V.

Add Section 3.70

1130 3.70 Pull Notification

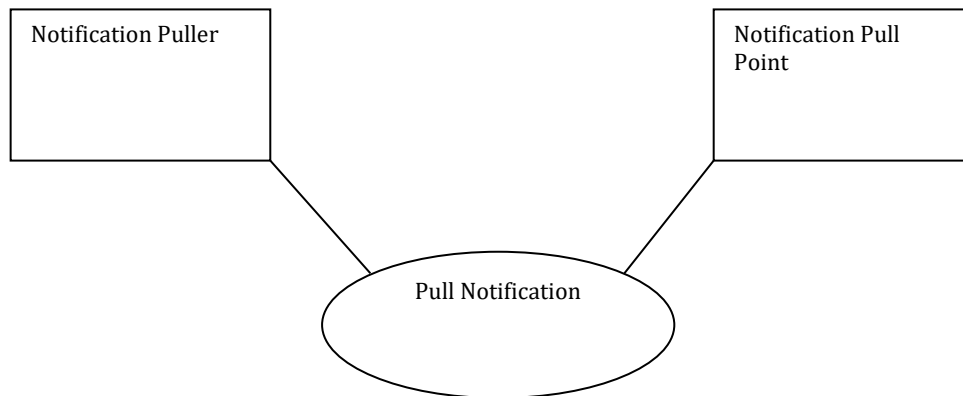
This section corresponds to the Transaction ITI-70 of the IHE IT Infrastructure Technical Framework. Transaction ITI-70 is used by the Notification Puller and by the Notification Pull Point actors.

3.70.1 Scope

1135 The Pull Notification transaction provides a message exchange to allow a Notification Puller to retrieve (or pull) notification messages from a Notification Pull Point. This transaction involves a Request by the Notification Puller for the retrieving of messages from Notification Pull Point and a response conveying pending notifications targeted to the Notification Puller.

3.70.2 Actor Roles

1140



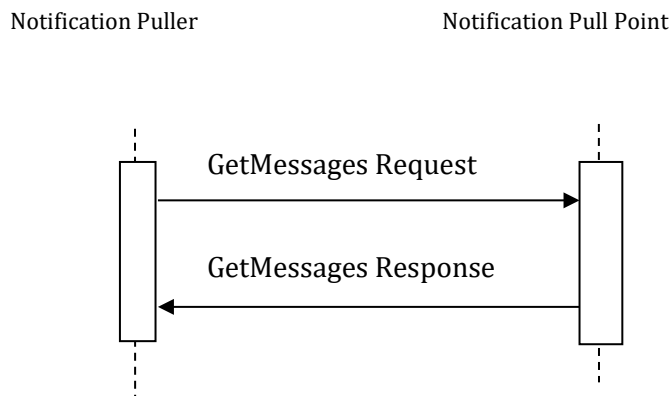
Actor:	Notification Puller
Role:	Sends a request to the Notification Pull Point for pending notifications stored in a Pull Point resource.

Actor:	Notification Pull Point
Role:	Responds to the request to deliver pending messages for the Notification Puller.

3.70.3 Referenced Standards

- OASIS Web Services Notification Family of standards
- 1145 • WS-BaseNotification 1.3 OASIS standard
- WS-BrokeredNotification 1.3 OASIS Standard
- WS-BaseFaults
- WS-Topics 1.3 OASIS Standard
- IHE ITI TF-2x: Appendix V
- 1150 • WS-Addressing OASIS Standard

3.70.4 Interaction Diagram



3.70.4.1 GetMessages Request message

1155 The Notification Puller asks for pending notifications on the Notification Pull Point using the endpoint obtained after the creation of the Pull Point resource.

3.70.4.1.1 Trigger Events

If the Notification Puller wishes to retrieve pending notifications from the Pull Point resource, it SHALL send a GetMessages request to the endpoint returned in the response to the Create Destroy Pull Point [ITI-69] transaction.

1160 **3.70.4.1.2 Message Semantics**

The GetMessages Request SHALL comply with the requirements in the WS-BaseNotification standard.

This is a request message to the Notification Pull Point that allows the pulling of pending notifications.

1165 The WS-Addressing [action] Message Addressing Property MUST contain the URI:

<http://docs.oasis-open.org/wsn/bw-2/PullPoint/GetMessagesRequest>.

The components of the GetMessages request are:

- 1170 • **/wsnt:GetMessages:** This component, which is required, requests that notifications held by the Pull Point resource be returned. Once the GetMessages message is received by the Metadata Notification Pull Point the request message is processed. The Pull Point shall respond to the Notification Puller immediately when the request is processed, returning content stored at that time without waiting for other notifications to be accumulated.
- 1175 • **/wsnt:GetMessages/wsnt:MaximumNumber:** This non-negative integer that indicates the maximum number of accumulated Notification Messages to be returned in the response message. This element is default set to “1”. This requirement is related to auditing reasons and allows that Response messages convey data related to only one patient.

3.70.4.1.3 Expected Actions

1180 Once the GetMessages Request message is received by the Notification Pull Point, it processes the request message. There are three possibilities:

1. ***The Notification Pull Point resource has no notifications messages stored.*** The Notification Pull Point SHALL respond with a GetMessages Response message containing zero Notification Response Messages
- 1185 2. ***The Notification Pull Point resource has one notification message stored.*** The Notification Pull Point SHALL respond with a GetMessage Response containing only one Notification Message into a GetMessages Response message.
3. ***The Notification Pull Point resource stores more than one message.*** The Response message convey only one notification message but the Notification Puller is required to start another Notification Pull transaction in order to retrieve all notifications.
- 1190 4. ***The Pull Point resource is unable to respond to the request.*** The Notification Pull Point SHALL send one of these faults:
 - ResourceUnknownFault - The Pull Point resource is acting as a WS-Resource, and the resource identified in the request message is not known to the Web service. This fault is specified by the WS-Resource [WS-Resource] specification.

1195 UnableToGetMessagesFault -The Notification Pull Point cannot return notifications messages for some unspecified reasons

3.70.4.1.4 Example SOAP Encoding of the GetMessages Request message

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-2/PullPoint/GetMessagesRequest
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:GetMessages>
      <wsnt:MaximumNumber>1</wsnt:MaximumNumber>
    </wsnt:GetMessages>
  </s:Body>
</s:Envelope>
```

1200

3.70.4.2 GetMessages Response message

If the Notification Pull Point can process the request, it SHALL respond to the request received with a GetMessages Response, as described in the WS-BaseNotification standard.

3.70.4.2.1 Trigger Events

1205 This message is created in response to a request of retrieving of notification messages stored in a specific Pull Point resource. A Pull Point resource is related to one Notification Puller.

3.70.4.2.2 Message Semantics

The WS-Addressing [action] element of the response of the GetMessages request message MUST contain the URI:

1210 <http://docs.oasis-open.org/wsn/bw-2/PullPoint/GetMessagesResponse>.

The contents of the GetMessages response message are further described as follows:

- **/wsnt:GetMessagesResponse**: This component contains one Notification Message. The number of messages appearing is limited by the wsnt:MaximumNumber component of the GetMessages request message (that SHALL be equal to 1). The Notification Message appearing in a GetMessagesResponse is “removed” from the PullPoint and SHALL NOT appear in the response message of subsequent GetMessages requests.
- **/wsnt:GetMessagesResponse/wsnt:NotificationMessage**: The content of this component is a Notification Message. The Notification Message component is described

1220 as part of the Notify message defined in the Document Metadata Notify [ITI-53] transaction. The GetMessagesResponse message does not define additional constraints on the Notification Message component. The content of the Notification Message is exactly the content of the Notification Message component of the accumulated Notify messages using the Document Metadata Notify [ITI-53] transaction.

1225 There shall be a single wsnt:Notify/wsnt:NotificationMessage/wsnt:Message element in this transaction. If multiple objects need to be represented in a single notification, the WS-BaseNotification standard allows this to be done.

3.70.4.2.3 Expected Actions

1230 The Notification Puller SHALL accept the GetMessages Response message and SHALL be able to manage the Notify Message contained in the response according to the configuration and business logic of the actor. Possibilities include conveying the notification information to other systems and/or users.

3.70.4.2.4 Example SOAP Encoding of the GetMessage Response Message

```
<s:Envelope ... >
  <s:Header>
    <wsa:Action>
      http://docs.oasis-open.org/wsn/bw-2/PullPoint/GetMessagesResponse
    </wsa:Action>
    ...
  </s:Header>
  <s:Body>
    <wsnt:GetMessagesResponse>
      <wsnt:NotificationMessage>
        ...
      </wsnt:NotificationMessage>
    </wsnt:GetMessagesResponse>
  </s:Body>
</s:Envelope>
```

1235 3.70.5 Security Considerations

Notification Puller and Notification Pull Point actors are required to log a "query" event associated to the Pull Notification transaction. The use of encrypted TLS is recommended when the transmission is not otherwise secured (e.g., transmission over a secure network)

1240 Additionally, it is recommended that the Notification Pull transaction be associated with a SAML assertion so that the Notification Puller can outline authorizations to access the notification content (for example, see the XUA Profile ITI TF-1: 13). Only the Notification Puller that has created the Pull Point resource can ask it for pulling notification messages stored.

3.70.5.1 Security Audit Considerations

1245 The Pull Notification transaction is a Query event, as defined in ITI TF-2a: Table 3.20.4.1.1.1-1. The actors involved in the transaction SHALL create audit data in conformance with DICOM Part 15 “Query” because the GetMessages Request message allows the Notification Puller to query for instances stored in the Pull Point resource.

3.70.5.1.1 Notification Puller audit message

1250 The Notification Puller MUST send an audit message for each document conveyed in the notificationMessage element of the GetMessage Response message.

	Field Name	Opt	Value Constraints
Event AuditMessage/ EventIdentification	EventID	M	EV(110112, DCM, “Query”)
	EventActionCode	M	“E” (Execute)
	EventDateTime	M	not specialized
	EventOutcomeIndicator	M	not specialized
	EventTypeCode	M	EV(“ITI-70”, “IHE Transactions”, “Pull Notification”)
Source (Notification Puller) (1)			
Human Requestor (0..1)			
Destination (Notification Pull Point) (1)			
Patient (0..1)			
document (1)			

Where:

Source AuditMessage/ ActiveParticipant	UserID	U	not specialized
	AlternativeUserID	M	The process ID as used within the local operating system in the local system logs.
	UserName	U	not specialized
	UserIsRequestor	U	not specialized
	RoleIDCode	M	EV(110153, DCM, “Source”)
	NetworkAccessPointTypeCode	M	“1” for machine (DNS) name, “2” for IP address
	NetworkAccessPointID	M	The machine name or IP address

1255

IHE IT Infrastructure Technical Framework Supplement – Extensions to the Document Metadata Subscription Profile

Human Requestor (f known) AuditMessage/ ActiveParticipant	UserID	M	Identity of the human that initiated the transaction.
	AlternativeUserID	U	not specialized
	UserName	U	not specialized
	UserIsRequestor	U	not specialized
	RoleIDCode	U	Access Control role(s) the user holds that allows this transaction.
	NetworkAccessPointTypeCode	U	not specialized
	NetworkAccessPointID	U	not specialized

Destination AuditMessage/ ActiveParticipant	UserID	M	SOAP endpoint URI.
	AlternativeUserID	U	not specialized
	UserName	U	not specialized
	UserIsRequestor	U	not specialized
	RoleIDCode	M	EV(110152, DCM, “Destination”)
	NetworkAccessPointTypeCode	M	“1” for machine (DNS) name, “2” for IP address
	NetworkAccessPointID	M	The machine name or IP address

1260

Patient AuditMessage/ ActiveParticipant	ParticipantObjectTypeCode	M	“1” (Person)
	ParticipantObjectTypeCodeRole	M	“1” (Patient)
	ParticipantObjectDataLifeCycle	U	not specialized
	ParticipantObjectIDTypeCode	M	not specialized
	ParticipantObjectSensitivity	U	not specialized
	ParticipantObjectID	M	The patient ID in HL7 ^{®2} CX format.
	ParticipantObjectName	U	not specialized
	ParticipantObjectQuery	U	not specialized
	ParticipantObjectDetail	U	not specialized

Document AuditMessage/ ActiveParticipant	ParticipantObjectTypeCode	M	“2” (system object)
	ParticipantObjectTypeCodeRole	M	“1” (Patient)
	ParticipantObjectDataLifeCycle	U	not specialized
	ParticipantObjectIDTypeCode	M	EV(“ITI-70”, “IHE Transactions”, Pull Notification”)

² HL7 is the registered trademark of Health Level Seven International.

	<i>ParticipantObjectSensitivity</i>	<i>U</i>	<i>not specialized</i>
	ParticipantObjectID	M	The value of <ihe:DocumentUniqueID>
	ParticipantObjectName	C	If known the value of <ihe:HomeCommunityID/>
	<i>ParticipantObjectQuery</i>	<i>M</i>	<i>not specialized</i>
	<i>ParticipantObjectDetail</i>	<i>C</i>	<i>not specialized</i>

3.70.5.1.2 Notification Pull Point audit message

	Field Name	Opt	Value Constraints
Event AuditMessage/ EventIdentification	EventID	M	EV(110112, DCM, “Query”)
	EventActionCode	M	“E” (Execute)
	<i>EventDateTime</i>	<i>M</i>	<i>not specialized</i>
	<i>EventOutcomeIndicator</i>	<i>M</i>	<i>not specialized</i>
	EventTypeCode	M	EV(“ITI-70”, “IHE Transactions”, “Pull Notification”)
Source (Notification Puller) (1)			
Human Requestor (0..1)			
Destination (Notification Pull Point) (1)			
Patient (0..1)			
document (1)			

1265

Where:

Source AuditMessage/ ActiveParticipant	<i>UserID</i>	<i>U</i>	<i>not specialized</i>
	AlternativeUserID	M	The process ID as used within the local operating system in the local system logs.
	<i>UserName</i>	<i>U</i>	<i>not specialized</i>
	<i>UserIsRequestor</i>	<i>U</i>	<i>not specialized</i>
	RoleIDCode	M	EV(110153, DCM, “Source”)
	NetworkAccessPointTypeCode	M	“1” for machine (DNS) name, “2” for IP address
	NetworkAccessPointID	M	The machine name or IP address

Human Requestor (if known) AuditMessage/ ActiveParticipant	UserID	M	Identity of the human that initiated the transaction.
	<i>AlternativeUserID</i>	<i>U</i>	<i>not specialized</i>
	<i>UserName</i>	<i>U</i>	<i>not specialized</i>
	<i>UserIsRequestor</i>	<i>U</i>	<i>not specialized</i>
	RoleIDCode	U	Access Control role(s) the user holds that allows this transaction.
	<i>NetworkAccessPointTypeCode</i>	<i>U</i>	<i>not specialized</i>
	<i>NetworkAccessPointID</i>	<i>U</i>	<i>not specialized</i>

Destination AuditMessage/ ActiveParticipant	UserID	M	SOAP endpoint URI.
	AlternativeUserID	U	not specialized
	UserName	U	not specialized
	UserIsRequestor	U	not specialized
	RoleIDCode	M	EV(110152, DCM, "Destination")
	NetworkAccessPointTypeCode	M	"1" for machine (DNS) name, "2" for IP address
	NetworkAccessPointID	M	The machine name or IP address

Patient AuditMessage/ ActiveParticipant	ParticipantObjectTypeCode	M	"1" (Person)
	ParticipantObjectTypeCodeRole	M	"1" (Patient)
	ParticipantObjectDataLifeCycle	U	not specialized
	ParticipantObjectIDTypeCode	M	not specialized
	ParticipantObjectSensitivity	U	not specialized
	ParticipantObjectID	M	The patient ID in HL7 CX format.
	ParticipantObjectName	U	not specialized
	ParticipantObjectQuery	U	not specialized
	ParticipantObjectDetail	U	not specialized

Document AuditMessage/ ActiveParticipant	ParticipantObjectTypeCode	M	"2" (system object)
	ParticipantObjectTypeCodeRole	M	"1" (Patient)
	ParticipantObjectDataLifeCycle	U	not specialized
	ParticipantObjectIDTypeCode	M	EV("ITI-70", "IHE Transactions", "Pull Notification")
	ParticipantObjectSensitivity	U	not specialized
	ParticipantObjectID	M	The value of <ihe:DocumentUniqueID>
	ParticipantObjectName	C	If known the value of <ihe:HomeCommunityID/>
	ParticipantObjectQuery	M	not specialized
	ParticipantObjectDetail	C	not specialized

1270

3.70.5.1.3 Metadata Notification Pull Point Specific Security Considerations

Notifications stored in the Pull Point should be managed in a secure way, but modalities to do this are not constrained by this supplement. The Pull Point resource is directly related to one Metadata Notification Puller. A system for the management of access policies can be created over this actor but these topics are not addressed by transactions.

1275