Integrating the Healthcare Enterprise

IHE IT Infrastructure
Technical Framework Supplement

Mobile access to Health Documents (MHD)

Trial Implementation

Date: August 31, 2012

Author: IHE ITI Technical Committee

Email: iti@ihe.net

Foreword

This is a supplement to the IHE IT Infrastructure (ITI) Technical Framework V9.0. 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 for Trial Implementation on August 31, 2012 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 ITI Technical Framework. Comments are invited and may be submitted at http://www.ihe.net/iti/iticomments.cfm.

This supplement describes changes to the existing technical framework documents and where indicated amends text by addition (**bold underline**) or removal (~~**bold strikethrough**~~), as well as addition of new sections introduced by editor's instructions to "add new text" or similar, which for readability are not bolded or underlined.

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

Replace Section X.X by the following:

General information about IHE can be found at: www.ihe.net

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

Information about the structure of IHE Technical Frameworks and Supplements can be found at: http://www.ihe.net/About/process.cfm and http://www.ihe.net/profiles/index.cfm

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

CONTENTS

Introduction to this Supplement

The Mobile access to Health Documents (MHD) profile defines a simple HTTP interface to an XDS like environment. It defines transactions to a) submit a new document and metadata from the mobile device to a document receiver, b) get the metadata for an identified document, c) find document entries containing metadata based on query parameters, and d) retrieve a copy of a specific document.

These transactions leverage the document content and format agnostic metadata concepts from XDS, but simplify them for access by constrained environments such as mobile devices. The MHD profile does not replace XDS. It can be used to allow mobile devices constrained access to an XDS health information exchange. The following figure shows one possible way to implement MHD with a document sharing environment (that may, but is not necessarily, XDS based). This implementation choice is not mandatory and we recognize other architectures will be implemented. An Implementation Guide is being maintained at **http://wiki.ihe.net/index.php?title=mHealthDossier_Guide**

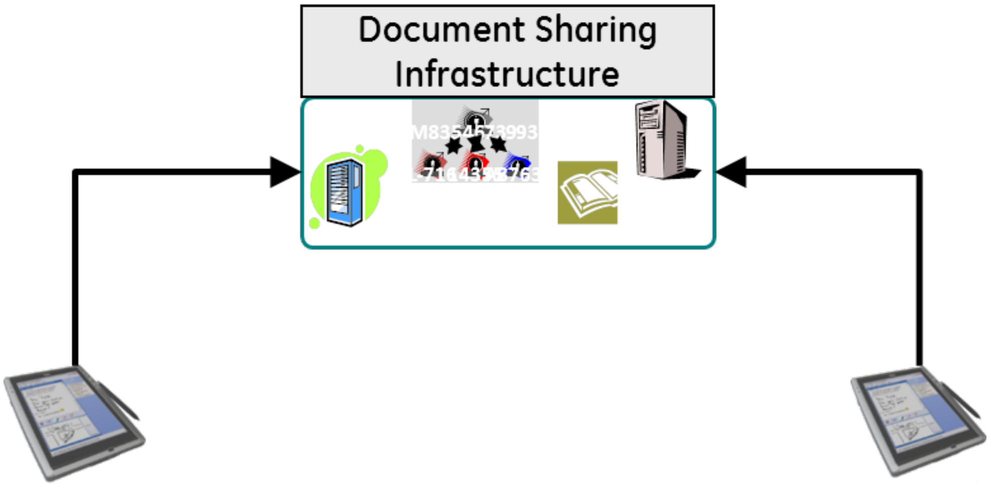


Figure 1: Mobile access to a Document Sharing environment.

The XDS profile is specifically designed to support the needs of Cross-Enterprise security, privacy, interoperability, and includes characteristics to support this level of policy and operational needs. The MHD profile has simplified the interactions in ways that are more consistent with a single policy domain use. The MHD transactions are not specifically tied to XDS, and some of the system implementations envisioned would interface directly to an organizational EHR, or a multi-national PHR.

The following lists a few examples of the environments which might choose to use the MHD profile instead of the XDS profile. The MHD profile supports a broad set of the XDS use cases and functionality while keeping the technology as simple as possible. The MHD profile is focused on a useful subset of the use-cases that XDS supports and does not try to reproduce the full scalability, flexibility, privacy, or security supported by the more robust XDS infrastructure.

- Medical devices such as those targeted by the Patient Care Devices (PCD) domain or Continua organization, submitting data in the form of documents.
- Kiosks used by patients in hospital registration departments, where it is anticipated that a hospital staff member will review, edit, and approve the document before it is allowed into the hospital system.
- PHR publishing into a staging area for subsequent import into an EHR or HIE.
- Patient or provider application that is configured to securely connect to a PHR in order to submit a medical history document.
- Electronic measurement device participating in an XDW workflow and pulling medical history documents from an HIE.
- A General Practitioner physician's office with minimal IT capabilities using a mobile application to connect to an HIE or EHR.

Open Issues and Questions

- MHD_020: This supplement offers only an OR relationship in queries/searches with multiple values of search parameter. There have been use-cases where it is desirable to sometimes use AND between multiple values especially with the EventCodeList. Where the client wants only entries where the EventCodeList contains both value A and value B, but not just one or the other. This supplement would force the client to receive all and filter out unnecessary items locally.
- MHD_021: The Find Document Dossiers transaction returns a list of entries that match the search criteria. This requires the client to do additional HTTP GET transactions to get the complete document metadata. There is concern that this forces many round-trips to get the metadata to the client.

Closed Issues

- MHD_001: Standards selection is to define simple HTTP RESTful transactions acting upon the resource defined by a hybrid of the Document Entry metadata with submission set, folders and associations. This hybrid is named in this supplement the Document Dossier. This metadata is described in JSON format only. We choose to focus only on JSON to have clear separation of this simplified encoding from the more flexible encoding of the existing XDS metadata. Thus not having two XML encodings.
- MHD_002: Security model is undefined as there are plenty of HTTP based security models that layer in between the low level transport (TCP) and the HTTP encoding. These security models can be layered in without modifying the characteristics of this profile. The use of TLS will be encouraged, specifically the use of ATNA, but will not be mandated. A companion Implementers Guide will include guidance on the use of the current common implementations of OpenID and OAuth. Formal specification of OpenID or OAuth should be done as a modular specification similar to XUA. This would be an effort outside the scope of MHD, but would also need to wait until OAuth and OpenID to be fully standardized and

matured to a profile ready state. Security Standards applicable and transparently usable that are known to the editing team are:

1. Kerberos (as used in IHE RID + EUA)
2. SAML (modified XUA for non SOAP – Likely SAML SSO Profile)
3. OpenID
4. OAuth 2.0 or later
5. Hybrid Auth -- http://hybridauth.sourceforge.net/index.html
6. TLS Client Authentication

- MHD_003: The Document List transaction is constrained from the XDS Queries to a hybrid of FindDocuments, FindSubmissionSets, and FindFolders.
- MHD 005: We know that the length of the URL encoding could become long. We believe that is still within URL length limits and can be supported. The HTTP RFCs do not specify a limit, and current servers are prepared for very long URLs.
- MHD_006: We chose JSON encoding because it is native to many mobile platforms. The choice also makes a clear distinction from the current XML encoding, thus limiting our XML encodings to only the SOAP transports. The expectation is that we will change XML encoded values into JSON encoded values as well, so that the mobile device doesn't need to have both JSON and XML processing. HL7 v2 values will be converted as needed.
- MHD_008: There has been a request for further simplification of encoding of codes and identifiers omit the assigning authority. We have relaxed the rules yet require the service to only support fully specified Affinity Domain Patient ID.
- MHD_010: PatientID is fundamental to the documentDossier object, yet we would like to allow a Patient Identity to have a potentially flexible service side. Thus we have moved the PatientID into a mandatory parameter, where this parameter can be a fully specified PatientID in the Affinity Domain, or if the service has the ability (e.g. is grouped with PIX actors) then the PatientID can contain the Patient Identity in other domains. The conformance criteria will still be tied to a fully specified PatientID, meaning that this must be supported; but conformance will not be tied to specifically a fully cross-referenced PatientID. This will allow the mobile device to provide the patient id in the domain that it knows, and allow the service side to cross-reference.
- MHD_011: The Document Source response to a GET on the base URL will return all Document Entries as scoped by any parameters also provided on the GET. The return results is simply a list of URLs to the Document Entries. We allow for the client to request the result in JSON list, or ATOM feed. The Service side must support both formats.
- MHD_012: This supplement forbids the use of HTTP “Conditional GET” because there is concern that the real meaning of metadata values (for example the Document Entry metadata value for creationTime) is slightly different than what HTTP “Conditional GET” would require for consistent processing.

- MHD_013: This supplement supports creation of only one document entry at a time through the documentDossier and, for simplicity sake, does not require the submitter to specify attributes of a Submission Set. This means that a Source can only publish one document at a time, and the service side will need to create the submission set based on the optional submission set values, document entry and local configuration.
- MHD_014: This supplement is focused on Document entry, but provides search to and copies of the related Submission Set, Folders, or Associations. This is not in the same format as XDS, but does provide access to the same access to the information. This hybrid object is called a documentDossier. As a result the search mechanism is a combined search thus putting functional responsibility on the service side. For example the client can through one search request searches across document entries, folders and submission sets; where the service side that is a proxy using Document Consumer grouping would need to do multiple types of XDS stored queries and combine the results.
- MHD_015: This supplement does not include specific hData use but is designed in harmony with hData. This will make easy integration with hData both at the operational level and in the future at the standards level.
- MHD_017: Error codes have been enhanced. Both in better mapping of XDS specific error codes as well as more appropriate use of HTTP error codes.
- MHD_018: We recognize a useful synergy with the ITI-RID profile. We have aligned closer to the URL format, specifically the same PatientID handling. This will allow Apps to utilize the MHD profile to get computable formats, while using RID to get displayable transforms. Note that RID may be more useful for specific use-cases, such as asking for a only the most recent discharge summary in display ready format, or asking only for the list of allergies. This grouping behavior will be most fully explored in the Implementers Guide.
- MHD_019: The JSON encoding uses non-anonymous objects.

Volume 1 – Profiles

Add to Section 33

33 Mobile access to Health Documents (MHD) Profile

Applications specific to mobile device is an emerging platform for healthcare enhancing software. These devices are resource and platform constrained, which drives the implementer to use simpler network interface technology. There are numerous uses of documents, for example hosted by a Health Information Exchange (HIE), large health provider electronic health record (EHR), or personal health record (PHR).

The Mobile access to Health Documents (MHD) profile defines one standardized interface to health documents for use by mobile devices so that deployment of mobile applications is more consistent and reusable. In this context, mobile devices include tablets and smart-phones, plus embedded devices like home-health devices. This profile is also applicable to larger systems where the needs are simple, such pulling the latest summary for display. The critical aspects of the 'mobile device' are that it is resource constrained, has a simple programming environment (e.g., JSON, javascript), simple network stack (e.g., HTTP), and simple display functionality (e.g., HTML browser). The goal is to limit the additional libraries that are necessary to process SOAP, WSSE, MIME-Multipart, MTOM/XOP, ebRIM, and multi-depth XML.

The Mobile access to Health Documents (MHD) profile defines actors and transactions. There is one set of actors and a transaction used to submit a single new document entry from the mobile device to a receiving system. The other set of actors and transactions is used to get a list of document entries containing metadata, and to retrieve a copy of a specific document.

These transactions leverage the metadata concepts from XDS, but simplify the technology requirements for access by mobile devices. The MHD profile defines a Document Dossier that is focused on the Document Entry as defined by XDS with all the related metadata including Submission Sets, Folders, and Associations. The MHD profile does not replace XDS. It enables simplified access by mobile devices to an XDS (or a similar) document management environment containing health information.

33.1 MHD Actors, Transactions, and Content Modules

Figure 33.1-1 shows the actors directly involved in the MHD Profile and the relevant transactions between them.

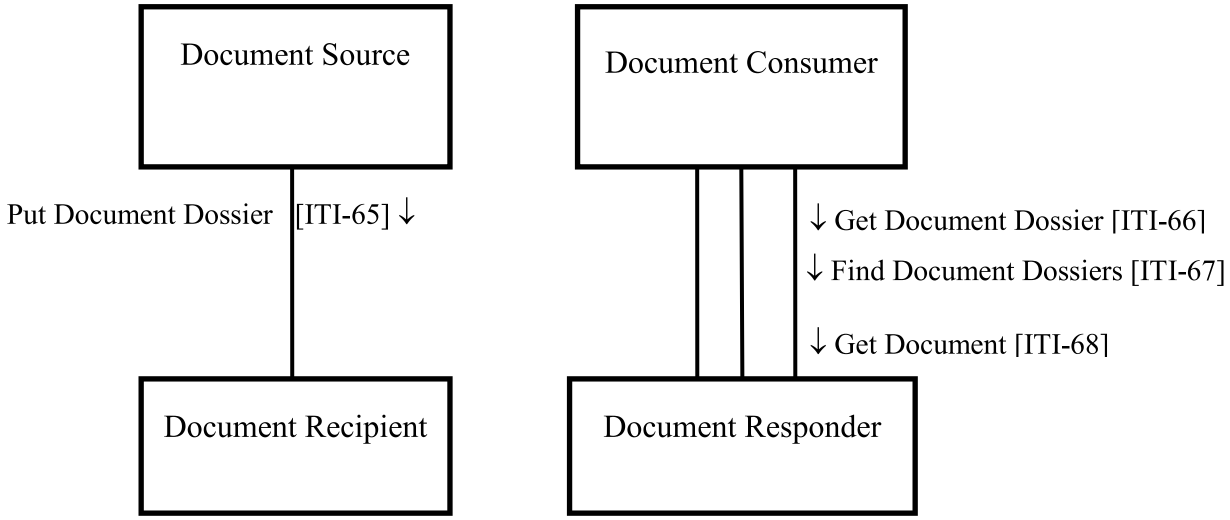

Document Source
Document Consumer
Put Document Dossier [ITI-65] ↓
↓ Get Document Dossier [ITI-66]
↓ Find Document Dossiers [ITI-67]
↓ Get Document [ITI-68]
Document Recipient
Document Responder

Figure 33.1-1: MHD Actor Diagram

Table 33.1-1 lists the transactions for each actor directly involved in the MHD Profile. In order 265 to claim support of this Profile, an implementation of an actor must perform the required transactions (labeled "R") and may support the optional transactions (labeled "O"). Actor groupings are further described in Section 33.3.

Table 33.1-1: MHD - Actors and Transactions

Actors	Transactions	Optionality	Section in Vol. 2
Document Source	Put Document Dossier [ITI-65]	R	ITI TF-2b:3.65
Document Recipient	Put Document Dossier [ITI-65]	R	ITI TF-2b:3.65
Document Consumer	Get Document Dossier [ITI-66]	O (Note 1)	ITI TF-2b:3.66
	Find Document Dossiers [ITI-67]	O (Note 1)	ITI TF-2b:3.67
	Get Document [ITI-68]	O (Note 1)	ITI TF-2b:3.68
Document Responder	Get Document Dossier [ITI-66]	R	ITI TF-2b:3.66
	Find Document Dossiers [ITI-67]	R	ITI TF-2b:3.67
	Get Document [ITI-68]	R	ITI TF-2b:3.68

270 Note 1: Document Consumer shall implement at least one transaction: Get Document Dossier, Find Document Dossiers, or Get Document.

33.1.1 Actor Descriptions and Actor Profile Requirements

The Document Source and Document Consumer actors are designed so that they can easily be implemented on a mobile device, and yet have sufficient functionality to support a wide range of mobile applications and use cases.

The Document Recipient and Document Responder are expected to be implemented in a service environment and thus do not have the mobile device constrained environment.

The transactions in the MHD Profile correspond to the following equivalent transactions used in XDS.

- MHD Put Document Dossier→ XDS Provide and Register
- MHD Get Document Dossier → XDS Registry Stored Query – GetDocuments
- MHD Find Document Dossiers → XDS Registry Stored Query – FindDocuments+FindSubmissionSets+FindFolders
- MHD Get Document → XDS Retrieve Document Set

The MHD transactions are not precisely equal to the XDS transactions as the MHD profile provides less functionality. These limitations are:

- the MHD PutDocumentDossier can only publish one new document at a time into a new SubmissionSet.
- the MHD Put Document Dossier cannot be used to replace an existing document or provide a transform
- the MHD Get Document Dossier can get only metadata about one document at a time.
- the MHD Get Document can only pull one document at a time.
- the MHD Find Document Dossiers supports only the OR operator within parameters.
- the MHD Find Document Dossiers returns only references to Document Entries, requiring a MHD Get Document Dossier to retrieve the metadata
- the MHD Find Document Dossiers does not support the XDS Registry Stored Query GetRelatedDocuments stored query.

In XDS, the Document Registry and Document Repository actors are independent to enable the widest possible deployment architectures. In contrast, the MHD profile combines the Registry and Repository functionality in one MHD Document Responder. This is expected to ease configuration needs on the mobile health application and mobile health application deployment, and reduce the overall solution complexity. The MHD Document Recipient and the MHD Document Responder actors are independent because there are use cases where only one is needed, such as supporting a mobile medical measuring device that simply creates and submits new documents. More general-purpose systems would likely implement both of these actors to provide a complete service definition for the hosting organization.

Due to these simplifying constraints, the MHD profile can be used as an interface to an XDS environment, but as discussed above, the MHD profile does not support all of the functionality supported by the XDS Document Source and XDS Document Consumer.

33.2 MHD Actor Options

Options that may be selected for this Profile are listed in the table 33.2-1 along with the Actors to which they apply. Dependencies between options when applicable are specified in notes.

Table 33.2-1: MHD - Actors and Options

Actor	Options	Volume & Section
Document Source	*No options defined*	- -
Document Recipient	*No options defined*	- -
Document Consumer	*No options defined*	- -
Document Responder	*No options defined*	- -

33.3 MHD Actor Required Groupings

Actor(s) which are required to be grouped with another Actor(s) are listed in this section. The grouped Actor may be from this profile or a different domain/profile. These mandatory required groupings, plus further descriptions if necessary, are given in the table below.

An Actor from this profile (Column 1) must implement all of the required transactions in this profile in addition to all of the required transactions for the grouped profile/actor listed (Column 2).

Table 33.3-1: MHD - Actors Required Groups

MHD Actor	Actor to be grouped with	Technical Framework Reference	Content Binding Reference
Document Source	None		
Document Recipient	None		
Document Consumer	None		
Document Responder	None		

33.4 MHD Overview

This profile assumes that the prime resource that is being operated on via RESTful operations is the XDS Document Dossier, which is the XDS Document Entry that describes a document, submission sets and folders it is contained in, and associations to other Document Entries. Thus the profile's prime focus is on actions upon the Document entry, rather than the document itself. But the profile does also provide access to the document.

The MHD Profile defines a base URL pattern with a mandatory patient identifier argument. This is a typical HTTP RESTful pattern and has the advantage of making it clearer that these are patient-centric transactions. Where this mobile device gets the patient ID is out of scope for the MHD profile with expectations that this could come from a previous browser session, some service call, or be configured. The mandatory inclusion of the Patient Identity on the URL should make the enforcement of privacy and security more straightforward (See section 33.5 Security Considerations).

33.4.1 Concepts

The MHD profile supports a broad set of the XDS use cases and functionality while keeping the technology as simple as possible. The MHD profile is focused on a subset of the use cases that XDS supports and does not try to reproduce the full scalability, flexibility, privacy, or security supported by the more robust XDS infrastructure. Example Use cases are:

- Medical devices such as those targeted by the Patient Care Devices (PCD) domain or Continua organization, submitting data in the form of documents.
- Kiosks used by patients in hospital registration departments, where it is anticipated that a hospital staff member will review, edit, and approve the document before it is allowed into the hospital system.
- PHR publishing into a staging area for subsequent import into an EHR or HIE.
- Patient or provider application that is configured to securely connect to a PHR in order to submit Recording history document.
- Electronic measurement device participating in an XDW workflow and pulling medical history documents from an HIE.
- A General Practitioner physician's office with minimal IT capabilities using a mobile application to connect to an HIE or EHR.

These specific use cases can be generalized into two general use cases. The first general use case is one where a new document is published from the mobile device. The second general use case is one where the mobile device needs to discover available documents and retrieve documents of interest. There are clearly complex use cases that combine these two general use cases. These are not specifically diagramed.

Where more complex use cases are needed, the one of the more robust XDS family of profiles is a more appropriate interface.

33.4.2 Use Case #1: Publication of new documents

33.4.2.1 Publication of new documents Use Case Description

In this use case there is a single new document that is published from the mobile device. An example might be that the mobile device is a medical device that has acquired new health measurements, or the mobile device has a user-interface used to capture user input such as a Patient Consent. This device created content is formed by the application implementing the Document Source into a Document and is submitted with the metadata.

The use cases presume that the mobile device knows the patient identity, although allows for identity cross-referencing to be implemented in the Document Recipient. The patient identity might be obtained through some IHE transactional method such as PIX/PDQ, might simply be entered via some device interface (RFID, Bar-Code), a user interface, or be specified in a configuration setting (e.g., mobile PHR Application). This use case also presumes that the mobile device knows the location of the URL endpoints, likely through a configuration setting, or a workflow driven by a web interface.

33.4.2.2 Publication of new documents Process Flow

The publication of a new document is done using the Put Document Dossier transaction, which carries both the document entry metadata and the document.

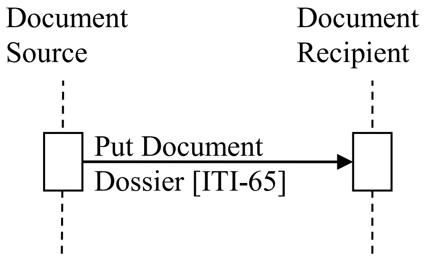


Figure 33.4.2.2-1: Basic Process Flow in MHD Profile

33.4.3 Use Case #2: Discovery and Retrieval of existing documents

33.4.3.1 Discovery and Retrieval of existing documents Use Case Description

In this use case the mobile device needs access to existing documents. An example is a mobile device involved in a workflow that needs to determine the current state of the workflow, or where the mobile device needs to discover the most current medical summary.

33.4.3.2 Discovery and Retrieval of existing documents Process Flow

The Find Document Dossiers transaction is used to provide parameterized queries that result in a set of pointers to Document Entries. The results can be returned as either a JSON structure or as an Atom feed.

The Get Document Dossier transaction is used to get the metadata for a specific Document Entry including the related submission set, folders, and associations.

The Get Document transaction is used to get the document itself.

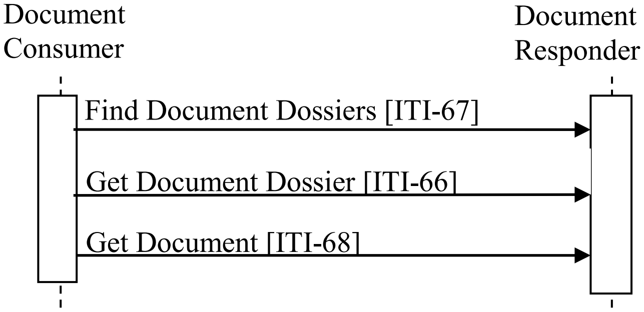

Document Consumer
Document Responder
Find Document Dossiers [ITI-67]
Get Document Dossier [ITI-66]
Get Document [ITI-68]

Figure 33.4.3.2-1: Basic Process Flow in MHD Profile

33.4.4 Mapping to RESTful operators

The MHD profile provides the resources and transactions against those resources. These are summarized in table 33.4.4-1. MHD does not use any additional extended or custom methods.

Table 33.4.4-1 Methods and Resources

HTTP Method	Transactions on Document Dossier	Transactions on Document
GET	Get Document Dossier [ITI-66]	Get Document [ITI-68]
PUT	Prohibited	Prohibited
POST	Put Document Dossier [ITI-65]	
DELETE	Prohibited	Prohibited
UPDATE	Prohibited	Prohibited
HEAD	Not Specified	Not Specified
OPTIONS	Not Specified	Not Specified
TRACE	Not Specified	Not Specified

33.5 MHD Security Considerations

There are many security and privacy concerns with mobile devices, simply because they are harder to physically control. Many common information technology uses of HTTP, including the RESTful pattern, are accessing far less sensitive information than health documents. These factors present an especially difficult challenge for the security model. It is recommended that application developers utilize a Risk Assessment in the design of the applications, and that the operational environment utilize a Risk Assessment in the design and deployment of the operational environment.

There are many reasonable methods of securing the interoperability transactions. These security models can be layered in without modifying the characteristics of the MHD profile transactions. The use of TLS is encouraged, specifically the use of the ATNA profile. User authentication on mobile devices is typically handled by a more lightweight authentication system such as HTTP Authentication, OAuth, or OpenID Connect. IHE does have a good set of profiles for the use of Enterprise User Authentication (EUA) on HTTP-based devices, with bridging to Cross-Enterprise User Assertion (XUA) for the backend. In all of these cases the network communication security, and user authentication are layered in at the HTTP transport layer thus do not modify the interoperability characteristics defined in the MHD profile.

The Security Audit logging (e.g., ATNA) is recommended. Support for ATNA-based audit logging on the mobile health device may be beyond the ability of this constrained environment. This would mean that the operational environment must choose how to mitigate the risk of relying only on the service side audit logging.

The Resource URL pattern defined in this profile does include the Patient ID as a mandatory argument. The advantage of this is to place clear distinction of the patient identity on each transaction, thus enabling strong patient-centric privacy and security controls. This URL pattern does present a risk when using typical web server audit logging of URL requests, and browser history. In both of these cases the URL with the patient identity is clearly visible. These risks need to be mitigated in system or operational design.

33.6 MHD Cross Profile Considerations

33.6.1 MHD Actor grouped with XDS infrastructure

When the MHD Document Recipient actor is acting as a proxy for an XDS environment, it could be grouped with an XDS Document Source or an XDS Integrated Document Source/Repository. In this way, the Put Document Dossier transaction would be converted by the grouped system into an XDS Provide and Register Document Set-b transaction. It is expected that this system would be configured to support only a designated set of mobile devices authorized by the hosting organization and use the security model defined by that hosting organization. The proxy would be expected to fill in any necessary missing information, convert any user authentication credentials, and implement fully the IHE ATNA Secure Node or Secure Application actors.

When the MHD Document Responder actor is acting as a proxy for an XDS environment, it could be grouped with an XDS Document Consumer. In this way the Get Document Dossier, Find Document Dossiers, and Get Document transactions will be supported in the system through the use of the XDS Registry Stored Query and XDS Retrieve Document Set-b transactions as needed. It is expected that this proxy would be configured to support a designated set of mobile devices and the security model defined by the hosting organization. The proxy would be expected to fill in any necessary missing information, convert any user authentication credentials, and implement fully the IHE ATNA Secure Node or Secure Application actors.

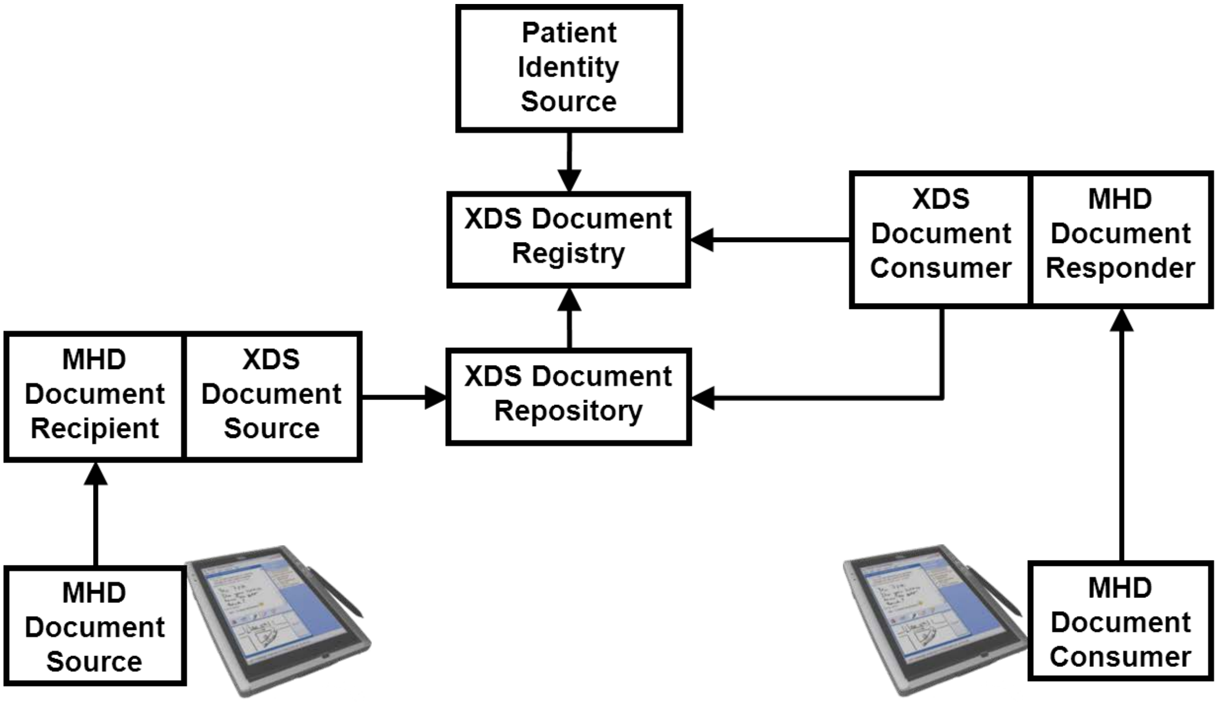


Figure 33.6.1-1: MHD Actors grouped with XDS

33.6.2 MHD Actor grouped with XCA infrastructure

When a MHD Document Responder acts as a proxy into an XCA environment, it could be grouped with an XCA Initiating Gateway. This type of MHD Document Responder will support the Find Document Dossiers and Get Document transactions by utilizing the XCA Cross Gateway Query and XCA Cross Gateway Retrieve transactions as necessary. This type of proxy would be configured to support a designated set of mobile devices and enable a security model as defined by the hosting organization. The proxy would be required to fill in any necessary missing information, convert any user authentication credentials, and implement fully the IHE-ATNA Secure Node or Secure Application.

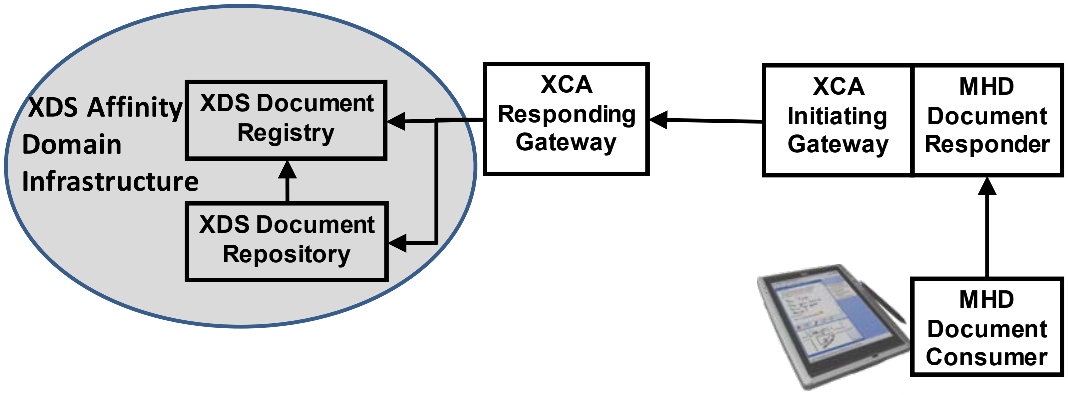
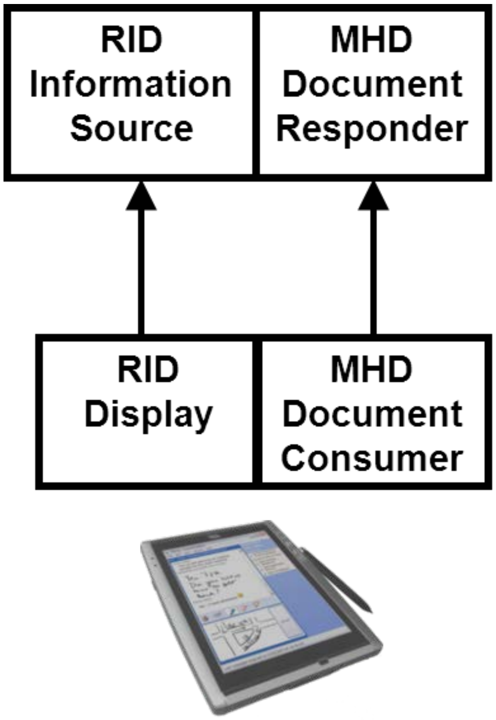


Figure 33.6.2-1: MHD Actors grouped with XCA

33.6.3 MHD Actor grouped with Retrieve Information for Display (RID) Profile

The Retrieve Information for Display (RID) profile includes a similar set of transactions to those defined in the MHD profile for Document Consumer. The RID profile is focused more on delivering display-ready health information that may or may not be document based, whereas the MHD profile is providing access to Documents and the metadata about the document. By grouping the RID "Information Source" actor with a MHD "Document Responder" actor will provide both access to the metadata and document content, with also access to display-ready information.



475 **Figure 33.6.2-1: MHD Actors grouped with RID**

Appendices

Actor Summary Definitions

Update (and add) the following terms to the IHE TF General Introduction Namespace list of Actors:

Document Source - The Document Source Actor is the producer and publisher of documents **and metadata**~~.~~ ~~**It is responsible for sending documents to a Document Repository Actor. It also supplies metadata to the Document Repository Actor for subsequent registration of the documents with the Document Registry Actor.**~~

Document Consumer - The Document Consumer Actor queries for document metadata meeting certain criteria, and may retrieve selected documents.

Document Recipient: ~~**This**~~ **The Document Recipient** actor receives ~~**a set of**~~ documents **and metadata** sent by another actor. ~~**Typically this document set will be made available to the intended recipient who will choose to either view it or integrate it into a Health Record.**~~

Document Responder – The Document Responder actor is receiver of and responder to requests for document entries and documents.

Transaction Summary Definitions

Add the following terms to the IHE TF General Introduction Namespace list of Transactions:

Put Document Dossier This transaction is used to transfer a document and metadata, equivalent to a Provide and Register Document Set-b transaction.

Get Document Dossier – This transaction is used to get the metadata related to a particular Document Entry.

Find Document Dossiers – This transaction is used to provide parameterized queries that result in a list of Document Entries.

Get Document – This transaction is used to get a single document.

Volume 2c – Transactions

Add sections 3.65, 3.66, 3.67 and 3.68

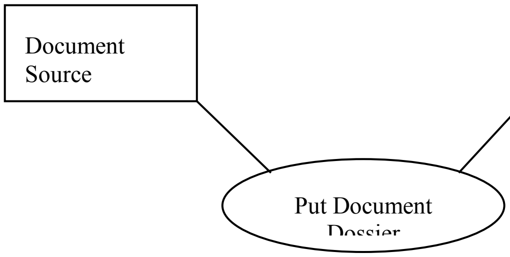
3.65 Put Document Dossier ITI-65

This section corresponds to Transaction ITI-65 of the IHE Technical Framework. Transaction ITI-65 is used by the Document Source and Document Recipient actors.

3.65.1 Scope

This transaction is used to publish a new document entry and the document.

3.65.2 Use Case Roles



Actor: Document Source

Role: Sends Document Entry and Document to the Receiver for publication.

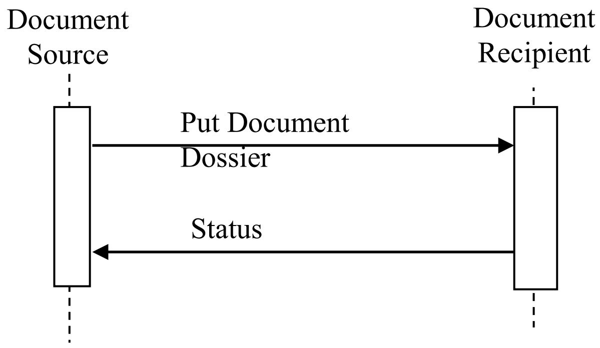
Actor: Document Receiver

Role: Accepts the document and metadata sent from the Source.

3.65.3 Referenced Standard

RFC2616	IETF Hypertext Transfer Protocol – HTTP/1.1
RFC3986	IETF Uniform Resource Identifier (URI): Generic Syntax
RFC4627	The application/json Media Type for JavaScript Object Notation (JSON)
RFC6585	IETF Additional HTTP Status Codes

3.65.4 Interaction Diagram



3.65.4.1 Put Document Dossier Message

This message uses the HTTP POST method on the target Document Entry to convey the metadata and the document.

3.65.4.1.1 Trigger Events

This method is sent when the Document Source needs to create a Document Entry.

3.65.4.1.2 Message Semantics

The HTTP POST method is used to create a child resource below the document Dossier Section URL resource. The media type of the HTTP body is a MIME multi-part (i.e., multipart/form-data) which conforms to the following requirements:

1. The first mime part is the metadata encoded using the JSON encoding of the D with the media type set to application/json. Content-Disposition SHALL be set to “form-data”, with name of the “ihe-mhd-metadata”.
2. The second mime part contains the document with the appropriate media type set, and Content-Disposition set to “form-data”, with a name of “content”.

The Put Document Dossier message is sent to the URL defined below, and shall include the PatientID argument. The format for a Document Dossier Section URL is:

documentDossierSectionURL :=
http://<location>/net.ihe/DocumentDossier/?PatientID=<PatientID>

Where:

Location – a locally defined root part of arbitrary path.

patientID – the CX encoded patient ID. See Section 3.65.4.1.2.2 Patient Identity. This value may need to be transformed for URL encoding.

For example:
`http://blah.com/blah/net.ihe/DocumentDossier/?PatientID=144ba3c4aad24e9%5E%5E%5E%261.3.6.1.4.1.21367.2005.3.7%26ISO`

3.65.4.1.2.1 JSON encoding of a documentDossier

The documentDossier is made up of the DocumentEntry, SubmissionSet, Folder, and Association objects. An instance of a documentDossier is identified by the entryUUID for the documentEntry that it contains. The encoding rules for single string JSON value representation can be found in ITI TF-3: Table 4.1-3 Data Types.

For composite attributes datatypes that are Coded Values, such as Coded Value and Author, the representation shall be as shown in Table 3.65.4.1.2.1-1 XDS XML Data Type Encoding Composite datatypes.

Table 3.65.4.1.2.1-1: XDS XML Data Type Encoding for Composite datatypes

Composite datatype	JSON representation	JSON example

Coded Value	*nameofattribute*:{
	code:*string*,
	codingScheme:*string*,
	codeName:*string*
	}
	Note: attribute names are found in the table below.
		classCode: {
		code:"2345-3",
		codingScheme:"OINK",
		codeName:"Author garbage"
		}

Author	Author: {
	authorInstitution:*XON*,
	authorPerson:*XCN*,
	authorRole:*string*,
	authorSpecialty:*string*
	}
		Author:{
		authorInstitution:"Hospital^^^^^^^^^^1.2.3.4.5.6.7.8.9.1789.45",
		authorPerson:"Name of Author",
		authorRole:"name of role",
		authorSpecialty:"specialty of author"
		}

The XDS Document Entry metadata as defined in ITI TF-3: Table 4.1-5 is encoded using JSON according to Table 3.65.4.1.2.1-2 XDS Document Entry JSON encoding. All other encoding and validation rules found in XDS apply.

Table 3.65.4.1.2.1-2: XDS Document Entry Metadata JSON encoding

XDSDocumentEntry Name	JSON value representation
author	Author value
availabilityStatus	String
classCode	Coded value
comments	String
confidentialityCode	Coded value
creationTime	DTM
entryUUID	UUID
eventCodeList	Coded Value
formatCode	Coded value
hash	String
healthcareFacilityTypeCode	Coded Value
homeCommunityId	anyURI
languageCode	String
legalAuthenticator	XCN
mimeType	String
patientId	CX
practiceSettingCode	Coded Value
repositoryUniqueId	String
serviceStartTime	DTM
serviceStopTime	DTM
size	String
sourcePatientId	CX
sourcePatientInfo	String
title	String
typeCode	Coded value
uniqueId	String

Note: Not all of these attributes are valid for the Put Document Dossier transaction. See the Provide and Register Document Set-b [ITI-41] transaction for rules for use with XDS.

Table 3.65.4.1.2.1-3: XDS Submission Set Metadata JSON representation

XDSSubmissionSet Name	JSON value representation
author	Author value
comments	String
contentTypeCode	Coded value
entryUUID	UUID

XDSSubmissionSet Name	JSON value representation
homeCommunityId	anyURI
intendedRecipient	XON/XCN
patientId	CX
sourceID	OID
submissionTime	DTM
title	String
uniqueId	OID

Table 3.65.4.1.2.1-4: XDS Folder Metadata JSON representation

XDS Folder Metadata Name	JSON value representation
availabilityStatus	String
codeList	Coded value
comments	String
entryUUID	UUID
homeCommunityId	anyURI
lastUpdateTime	DTM
patientId	CX
title	String
uniqueId	OID

Table 3.65.4.1.2.1-5: XDS associations Metadata JSON representation

XDS associations Name	JSON value representation
toUUID	UUID
toURL	anyURI
fromUUID	UUID
fromURL	anyURI
type	String

Associations shall include a toUUID and a fromUUID. The entryUUID for the Document Entry will be equal to either the toUUID or the fromUUID; indicating the direction of the association. Associations to other Document Entries shall include both toURL and fromURL. Associations with submissionSets or folders will only have the UUID field populated.

All values shall be encoded using RFC-1738 rules.

Any attribute with a single value shall be encoded as the attribute name and value, e.g.,

```
comments: "This is a comment"
```

Any attribute that has multiple values shall be encoded as a JSON array, e.g.,

```
comments: ["This is a comment", "This is also a comment"]
```

Any attribute made up of composite datatype multiple attributes shall be encoded as a JSON object, e.g.,

```
classCode: { code:"2345-3",codingScheme:"OINK", codeName:"garbage"}
```

An attribute with multiple values and multiple attributes would look like:

```
classCode: [{code:"2345-3",codingScheme:"OINK", codeName:"garbage"},
{code:"2345-2",codingScheme:"OINK", codeName:"trash"}]
```

A complete Document Entry would be encoded as a JSON object (Note: the example below does not include all XDS required attributes)

```
documentEntry:{patientID:
"144ba3c4aad24e9^^^&1.3.6.1.4.1.21367.2005.3.7&ISO" ,
classCode: {code:" 34133 -9 ",codingScheme:“2.16.840.1.113883.6.1",
codeName:“Summary of Episode Note"},
confidentialityCode:{code:”N”,codingScheme:”2.16.840.1.113883.5.25”,cod
eName:”Normal sensitivity”},
formatCode:{code:”urn:ihe:lab:xd-lab:2008”,codingScheme:”
1.3.6.1.4.1.19376.1.2.3”,codeName:”XD-Lab”},
typeCode:{code:””,codingScheme:””,codeName:””},
Author:{…},
practiceSettingCodes:{code:" 394802001
",codingScheme:“2.16.840.1.113883.6.96 ", codeName:“General Medicine"}
Title:"document title",
creationTime:“20061224”,
hash:“e543712c0e10501972de13a5bfcbe826c49feb75”,
Size:“350”,
languageCode:“en-us”,
serviceStartTime:“200612230800”,
serviceStopTime:“200612230900”,
sourcePatientId:“89765a87b^^^&3.4.5&ISO”,
mimeType:” text/xml ”,
uniqueId:” 1.2009.0827.08.33.5074”,
entryUUID:”urn:uuid:14a9fdec-0af4-45bb-adf2-d752b49bcc7d “}
```

A documentDossier is a complete document metadata entry that is composed of a documentEntry, and one or more submissionSet(s), zero or more folder(s), and one or more association(s).

```
{
documentEntry:{…},
submissionSet:[{…},
{…},
{…}],
folder:[{…}],
association:[{…},
{…},
  {…},
  {…}]
}
```

3.65.4.1.2.2 Patient Identity

The patientID is included as a parameter to facilitate handling and access control processing easier. The value should be encoded using the CX encoding with the assigning authority included. The Document Recipient may use Patient ID cross-referencing, such as defined in the ITI PIX profile, to provide support for multiple Patient Identity assigning authorities. When the patientID argument is used in an access control decision, it must be confirmed with the patientID found within the documentDossier.

3.65.4.1.3 Expected Actions

The Document Recipient shall verify the Document Dossier attributes for consistency with the requirements as specified for attributes sent through the Provide and Register Document Set-b [ITI-41] transaction when used with XDS. The Document Recipient may accept a documentDossier that is missing SubmissionSet and derive the values according to guidance provided in Table 3.65.4.1.3-1.

When the MHD Document Recipient is grouped with an XDS Document Source, the Document Entry shall be transformed into a proper Provide and Register Document Set-b [ITI-41] transaction when used with XDS. The Document Recipient may need to create appropriate SubmissionSet metadata based on the Document Entry metadata. Some SubmissionSet metadata is not directly derivable; these values are left to the implementer of the Document Recipient. Table 3.65.4.1.3-1 is provided as guidance.

Table 3.65.4.1.3-1: XDS Submission Set potential derivation

XDSSubmissionSet Attribute	Potentially Derived from
author	`DocumentEntry.author`
comments	`DocumentEntry.comment`
contentTypeCode	Chosen from a lookup table based on mobile device ID, or other document metadata like `classCode or formatCode`
intendedRecipient	Configured value, derived from specific use case, or left empty
patientId	`DocumentEntry.patientID`
sourceId	Configured value indicating the identity of the MHD Document Recipient
submissionTime	The current date/time
title	`DocumentEntry.title`

3.65.4.2 Status Message

The Document Recipient returns a HTTP Status code appropriate to the processing.

3.65.4.2.1 Trigger Events

This message shall be sent once the document is received and completely processed.

3.65.4.2.2 Message Semantics

If the Document Recipient has fully processed the POST transaction then the Document Recipient shall return the HTTP response code 201 – Created to indicate success. The Document Recipient shall include an HTTP Location header that points to the URL for the documentDossier resource that was created.

If the Document Recipient cannot recognize the posted data, then the Document Recipient shall return the HTTP response code 400 – Bad Request.

HTTP POST with an If-Unmodified-Since header shall result in HTTP response code 501 – Not implemented.

Note: Other HTTP response codes may be returned by the Document Recipient, indicating conditions outside of the scope of this transaction, for example, 401 – Authentication Failed might be returned if Document Recipient Actor is grouped with the EUA profile Kerberized Server.

The Document Recipient should complement the returned error code with a human readable description of the error condition.

The Document Recipient may return HTTP redirect responses (responses with values of 301, 302, 303 or 307) in response to a request. The Document Source shall follow redirects, but if a loop is detected, it may report an error.

3.65.4.2.3 Expected Actions

The Document Source processes the results according to application-defined rules.

If a Document Source cannot automatically recover from an error condition, at a minimum, it should display the error to the user.

3.65.5 Security Considerations

See the general Security Considerations in ITI TF-1:33.5.

3.65.5.1 Security Audit Considerations

The security audit criteria are similar to those for the Provide and Register Document Set –b transaction [ITI-41] as this transaction does export a document. Grouping a Document Source or Document Recipient with an ATNA Secure Node or Secure Application is recommended, but not mandated. The Document Source may be considered overburdened to fully implement the requirements of Secure Node or Secure Application. The Document Recipient is more full featured and should generate the equivalent to the audit event defined in ITI TF-2b:3.41.7.1.2 Document Repository or Document Recipient audit message.

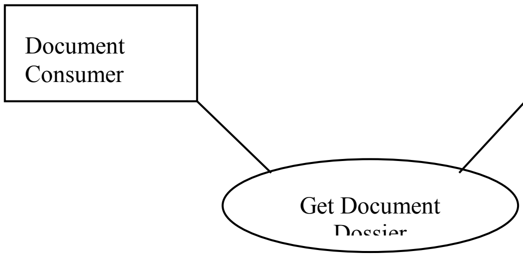
3.66 Get Document Dossier ITI-66

This section corresponds to Transaction ITI-66 of the IHE Technical Framework. Transaction ITI-66 is used by the Document Consumer and Document Responder actors.

3.66.1 Scope

This transaction is used to get a Document Entry, the metadata for a document.

3.66.2 Use Case Roles



Actor: Document Consumer

Role: Requests a Document Dossier from a Document Responder

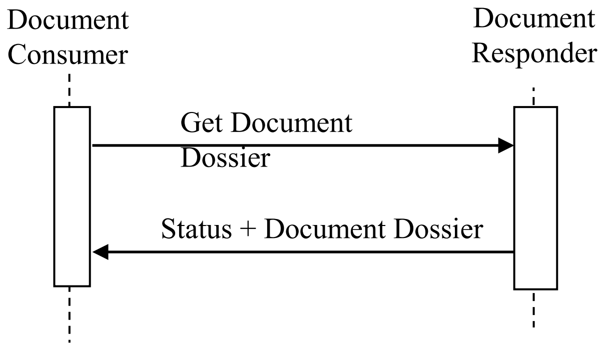
Actor: Document Responder

Role: Provides the Document Dossier.

3.66.3 Referenced Standard

RFC2616	IETF Hypertext Transfer Protocol –HTTP/1.1
RFC3986	IETF Uniform Resource Identifier (URI): Generic Syntax
RFC4627	The application/json Media Type for JavaScript Object Notation (JSON)
RFC6585	IETF Additional HTTP Status Codes

3.66.4 Interaction Diagram



3.66.4.1 Get Document Dossier Message

This message uses the HTTP GET method on the target Document Entry to retrieve the document metadata.

3.66.4.1.1 Trigger Events

This method is sent when the Document Consumer needs to get a specific Document Entry.

3.66.4.1.2 Message Semantics

The HTTP GET method is used to retrieve a copy of the Document Dossier resource. The Get Document Dossier message is sent to the documentDossierSectionURL with the entryUUID of the DocumentEntry as the target, and the PatientID argument. The format for a Document Dossier Section URL is:

documentDossierSectionURL := http://<location>/DocumentDossier/<entryUUID>/?PatientID=<PatientID>

Where:

location – a locally defined root part of arbitrary path

patientID – the CX encoded patient ID. See Section 3.65.4.1.2.2 Patient Identity. This value may need to be transformed URL encoding.

entryUUID – the UUID value for the DocumentEntry

HTTP If-Unmodified-Since header shall not be included in the GET request.

The Accept header should include application/json.

3.66.4.1.3 Expected Actions

The Document Responder shall return the Document Dossier for the Document Entry represented by the patientID and entryUUID. . The Document Responder shall verify that the documentDossier to be returned is consistent with the patientID.

When the Document Responder is grouped with an XDS Document Consumer the Document Entry can be obtained through the use of the GetDocument query in the Registry Stored Query [ITI-18] transaction given the entryUUID

3.66.4.2 Status + Document Dossier Message

The Document Responder shall return a HTTP Status code appropriate to the processing as well as the contents of the requested Document Dossier.

3.66.4.2.1 Trigger Events

This message shall be sent once the document entry is retrieved.

3.66.4.2.2 Message Semantics

The HTTP body shall be the Document Dossier encoded using JSON as defined in Section 3.65.4.1.2.1 JSON encoding of a documentDossier.

If the patientID or entryUUID are missing or malformed, the Document Responder shall return HTTP response code 400 - Bad Request.

If the specified patientID or entryUUID is not known to the Document Responder, it shall return HTTP response-code 404 (not found) with the suggested reason-phrase “Document Entry UUID not found”.

If the requested DocumentEntry is in the deprecated state, then the Document Responder shall return HTTP response-code 410 with the suggested reason-phrase “Document Entry UUID deprecated”.

If the HTTP request specified is otherwise not a legal value according to this transaction, the Document Responder shall return HTTP response-code 403 (forbidden) with the suggested reason-phrase “request type not supported”.

Note: The Document Responder may return other HTTP response codes indicating conditions outside of the scope of this transaction, for example, 401 – Authentication Failed might be returned if Document Responder is grouped with the EUA profile Kerberized Server.

The Document Responder should complement the returned error code with a human readable description of the error condition.

Document Responder may return HTTP redirect responses (responses with values of 301, 302, 303 or 307) in response to a request. Document Consumer shall follow redirects, but if a loop is detected, it may report an error.

3.66.4.2.3 Expected Actions

The Document Consumer shall process the results according to application-defined rules.

If a Document Consumer cannot automatically recover from an error condition, at a minimum, it should display the error to the user.

3.66.5 Security Considerations

See the general Security Considerations in ITI TF-1:33.5.

3.66.5.1 Security Audit Considerations

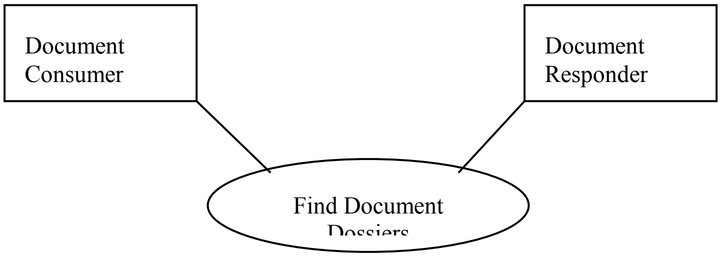
The security audit criteria are similar to those for the Registry Stored Query [ITI-18] transaction as this transaction does import a document entry. Grouping a Document Consumer or Document Responder with an ATNA Secure Node or Secure Application is recommended, but not mandated. The Document Consumer may be considered overburdened to fully implement the requirements of Secure Node or Secure Application. The Document Responder is more full featured and should generate the equivalent of the audit event defined in ITI TF-2a:3.18.5.1.2 Document Registry audit message.

3.67 Find Document Dossiers ITI-67

This section corresponds to Transaction ITI-67 of the IHE Technical Framework. Transaction ITI-67 is used by the Document Consumer and Document Responder actors.

3.67.1 Scope

The Find Document Dossiers transaction is used to get references to the Document Entries that satisfy a number of parameters, equivalent to ITI-18 (Registry Stored Query), FindDocuments, FindSubmissionSets, and FindFolders combined from the XDS stored query catalog from [ITI-18] in TF-2a:3.18.4.1.2.3.7.1:. The references are URLs that are Get Document Dossier compliant, so that the documentDossier can be retrieved using the Get Document Dossier transaction.



3.67.2 Use Case Roles

Actor: Document Consumer

Role: Requests references to document dossiers satisfying a set of key/value pair encoded metadata attributes

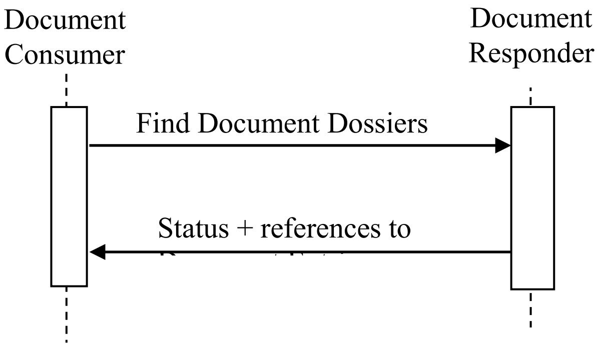
Actor: Document Responder

Role: Services the query and returns one or more references to Document Entries.

3.67.3 Referenced Standard

RFC2616	IETF Hypertext Transfer Protocol –HTTP/1.1
RFC3986	IETF Uniform Resource Identifier (URI): Generic Syntax
RFC4627	The application/json Media Type for JavaScript Object Notation (JSON)
RFC6585	IETF Additional HTTP Status Codes
RFC4287	The Atom Syndication Format

3.67.4 Interaction Diagram



3.67.4.1 Find Document Dossiers Message

This is an HTTP GET parameterized query from a Document Consumer to a Document Responder

3.67.4.1.1 Trigger Events

This method is sent when the Document Consumer needs to discover Document Entries matching various XDS metadata parameters.

3.67.4.1.2 Message Semantics

The Find Document Dossiers message is an HTTP GET request that can be sent to the FindDocuments URL on the Document Responder. This is the “search” target within the documentDossierSectionURL formatted as:

FindDocumentsURL := http://<location>/net.ihe/DocumentDossier/search?PatientID=<PatientID>&<parameters>

Where:

location – a locally defined root part of arbitrary path.

patientID – the CX encoded patient ID. See Section 3.65.4.1.2.2 Patient Identity. This value may need to be transformed for URL encoding.

Parameters – additional search arguments according to 3.67.4.1.2.1 Additional documentDossier search parameters.

The Document Consumer shall indicate in the Accept a preference for JSON (i.e., application/json) or Atom (i.e. application/xml+atom). In the absence of an Accept preference, JSON shall be used. The Document Responder shall support both JSON and Atom return formats.

3.67.4.1.2.1 Additional documentDossier search parameters

The Document Consumer may include additional key/value pairs to further scope the search. They are encoded in accordance with RFC2616 for encoding GET queries. The query string (i.e., the string after the '?' and before the '#' in the HTTP action) is created as a list of key/value pairs, using the following table (from Table 3.67.4.1.2-1 FindDocumentEntries Metadata Parameters) to identify the key names (which correspond to the control names in HTML) and to determine the multiplicity rules. The values are encoded using the encoding methods defined in ITI TF-3: Table 4.1-3: Data Types.

CE is the HL7 v2.5 data type for encoding coded values and is described in ITI TF-3a:3.18.4.1.2.3.4 "Coding of Code/Code-Scheme". Matching rules for coded values are the same as those defined in ITI-18 Registry Stored Query in that section.

Note that HTTP convention allows for multiple instances of a parameter to be requested with different values; multiple values of the same parameter name shall be treated in an OR relationship. There is no support in this transaction for the query mechanism in ITI-18 for query parameters of the same name in an AND relationship.

If the "status" parameter is not specified, the value "urn:oasis:names:tc:ebxml-regrep:StatusType:Approved" shall be assumed.

Table 3.67.4.1.2-1 FindDocumentEntries Metadata Parameters

XDS - Parameter Name	MHD Parameter Name	Encoded	Multiplicity
$XDSDocumentEntryClassCode	classCode	CE	M
$XDSDocumentEntryTypeCode	typeCode	CE	M
$XDSDocumentEntryPracticeSettingCode	practiceSettingCode	CE	M
$XDSDocumentEntryCreationTimeFrom	creationTimeFrom	DTM	S
$XDSDocumentEntryCreationTimeTo	creationTimeTo	DTM	S
$XDSDocumentEntryServiceStartTimeFrom	serviceStartTimeFrom	DTM	S
$XDSDocumentEntryServiceStartTimeTo	serviceStartTimeTo	DTM	S
$XDSDocumentEntryServiceStopTimeFrom	serviceStopTimeFrom	DTM	S
$XDSDocumentEntryServiceStopTimeTo	serviceStopTimeTo	DTM	S
$XDSDocumentEntryHealthcareFacilityTypeCode	healthcareFacilityTypeCode	CE	M
$XDSDocumentEntryEventCodeList	eventCodeList	CE	M
$XDSDocumentEntryConfidentialityCode	confidentialityCode	CE	M
$XDSDocumentEntryAuthorPerson	authorPerson	XCN	M
$XDSDocumentEntryFormatCode	formatCode	CE	M
$XDSDocumentEntryStatus	status	String	M

XDS - Parameter Name	MHD Parameter Name	Encoded	Multiplicity
$XDSSubmissionSetSourceId	sSourceID	OID	M
$XDSSubmissionSetSubmissionTimeFrom	sSubmissionTimeFrom	DTM	S
$XDSSubmissionSetSubmissionTimeTo	sSubmissionTimeTo	DTM	S
$XDSSubmissionSetAuthorPerson	sAuthorPerson	XCN	S
$XDSSubmissionSetContentType	sContentType	CE	M
$XDSSubmissionSetStatus	sStatus	String	M
$XDSFolderLastUpdateTimeFrom	fLastUpdateTimeFrom	DTM	S
$XDSFolderLastUpdateTimeTo	fLastUpdateTimeTo	DTM	S
$XDSFolderCodeList	fCodeList	CE	M
$XDSFolderStatus	fStatus	String	M

For example:

```
http://blah.com/blah/net.ihe/DocumentDossier/search?PatientID=144ba3c4aad24e9%5E%5E%5E
%261.3.6.1.4.1.21367.2005.3.7%26ISO&classCode=2345-
3%5EGarbage%5EOINK&serviceStartTimeFrom=200501020304
```

3.67.4.1.3 Expected Actions

The Document Responder shall process the query using the same rules as defined for ITI-18 Registry Stored Query for FindDocuments, FindSubmissionSets, and/or FindFolders as needed. This may be accomplished through grouping the Document Responder with an XDS Document Consumer, and transforming the parameters and combining the returned metadata entries.

3.67.4.2 Status + References to Document Entries Message

The Document Responder returns a HTTP Status code appropriate to the processing as well as a list of the matching document entries.

3.67.4.2.1 Trigger Events

The Document Responder found Document Entries using the query parameters.

3.67.4.2.2 Message Semantics

When the query is successful, the Document Responder shall return the list of Get Document Dossier compliant URLs, one for each matching result. There are two formats required by this transaction; the Accept header indicates which format the Document Consumer desires. See section 3.67.4.2.2.1 for JSON encoding, and section 3.67.4.2.2.2 for Atom encoding.

If the patientID is missing, the Document Responder shall return HTTP response code 400 - Bad Request.

If the specified parameters do not result in matching Document Entries, the Document Responder shall return either HTTP response-code 404 (not found) with the suggested reason-phrase "No Document Entries found", or an empty Atom feed or JSON entries array. The choice of which return method is used is based on policy and not defined by this transaction.

If the requested media type is not supported by the server, the Document Responder shall return a status code of 415.

If the HTTP request specified is otherwise not a legal value according to this transaction, the Document Responder shall return HTTP response-code 403 (forbidden) with the suggested reason-phrase "request not supported".

Note: Other HTTP response codes may be returned by the Document Responder, indicating conditions outside of the scope of this transaction, for example, 401 – Authentication Failed might be returned if Document Responder is grouped with the EUA profile Kerberized Server.

The Document Responder should complement the returned error code with a human readable description of the error condition.

Document Responder may return HTTP redirect responses (responses with values of 301, 302, 303 or 307) in response to a request. Document Consumers must follow redirects, but if a loop is detected, it may report an error.

3.67.4.2.2.1 JSON encoded list of Get Document Dossier URLs

When the query is successful, the message shall contains a HTTP response status code of 200, and a body containing a JSON Array of URLs compliant with Get Document Dossier transaction

For the JSON representation of the query results, the encoding is:

- The object SHOULD have an attribute "updated" with the time of the completion of the query rendered by calling the JavaScript Date.toString method.
- The object SHOULD include an attribute "self" with a URL pointing to the URL corresponding to the query.
- The object SHALL include an array called "entries" that contains objects corresponding to the results of the query.

The objects contained in the entries array SHALL contain the following attributes:

- An attribute called "id" that is set to a string representation of the *entryUUID* (see 3.66.4.1.2).
- An attribute called "self" that SHALL contain a URL which points to the documentDossierURL, as defined in 3.66.4.1.2.
- An attribute called "related" that SHALL contain a URL that points to the documentURL for the Document that this Entry references (see 3.68.4.1.2).
- An attribute called "updated" that SHALL contain the CreatedDateTime rendered by calling the JavaScript Date.toString method.

Example:

```
{
      “updated”:” Sun Oct 21 2011 12:34:28 GMT-0700”,
      “self”:http://example.com/foo/bar/net.ihe/DocumentDossier/search?
key=value&foo=bar,
      “entries”:[
            {
                  “id”:”123456”,

      “self”:http://example.com/foo/bar/net.ihe/DocumentDossier/123456”
,

      “related”:http://example.com/foo/bar/net.ihe/Document/abcxyz”,
                  “updated”:” Sun Oct 21 2011 12:34:28 GMT-0700”
            },
{
                  “id”:”9876”,

      “self”:http://example.com/foo/bar/net.ihe/DocumentDossier/9876”,

      “related”:http://example.com/foo/bar/net.ihe/Document/werwer”,
                  “updated”:” Sun Oct 28 2011 08:34:28 GMT-0700”
            }
      ]
}
```

3.67.4.2.2.2 Atom encoded list of DocumentDossier URLs

When the query is successful, the message shall contain a HTTP response status code of 200, and a body containing an Atom Array of URLs compliant with Get Document Dossier transaction.

The overall return type is an Atom feed, i.e. <atom:feed> document. Each result shall be represented by a distinct <atom:entry> node. It is recommended that the <atom:feed> includes

- <atom:updated> node with the time of the completion of the query, and
- <atom:link> node with attributes
    - rel=”self”
    - type=”application/atom+xml”
    - href set to the URL representing this query

Each <atom:enty> represents a reference to a documentDossier, using the following values for the child elements:

- <atom:id> shall be set to the documentDossierId, as specified in 3.66.4.1.2.
- <atom:link rel=”self” href=*documentDossierURL*>shall point to the documentDossierURL, as defined in 3.66.4.1.2.
- <atom:link rel=”related” href=*documentURL*> shall contain the documentURL for the Document that this Entry references (see 3.68.4.1.2).
- <atom:updated> shall contain the CreatedDateTime in W3C Data representation.

All other elements that are required by the Atom specification should have reasonable content, but they shall be ignored for the purpose of this transaction.

Example:

```
<?xml version="1.0" encoding="utf-8"?>
 <feed xmlns="http://www.w3.org/2005/Atom">
   <title>Example Query</title>
   <link rel="self"

href="http://example.com/foo/bar/net.ihe/DocumentDossier/search?key=val
ue&foo=bar"/>
   <updated>2012-10-21T12:34:28Z</updated>
   <entry>
     <title>123456</title>
     <link rel="self"
href="http://example.com/foo/bar/net.ihe/DocumentDossier/123456"/>
     <link rel="related"
href="http://example.com/foo/bar/net.ihe/Document/abcxyz"/>
     <id>123456</id>
     <updated>2012-10-21T12:34:28Z</updated>
   </entry>
  <entry>
     <title>9876</title>
     <link rel="self"
href="http://example.com/foo/bar/net.ihe/DocumentDossier/9876"/>
     <link rel="related"
href="http://example.com/foo/bar/net.ihe/Document/werwer"/>
     <id>9876</id>
     <updated>2012-10-28T08:34:28Z</updated>
   </entry>
 </feed>
```

3.67.4.2.3 Expected Actions

The return result shall be processed according to application behavior.

If a Document Consumer cannot automatically recover from an error condition, at a minimum, it should display the error to the user.

3.67.5 Security Considerations

See the general Security Considerations in ITI TF-1:33.5.

3.67.5.1 Security Audit Considerations

The Security audit criteria are similar to those for the Registry Stored Query transaction [ITI-18] as this transaction does import a Document Entry. Grouping the Document Consumer or Document Responder with an ATNA Secure Node or Secure Application is recommended, but not mandated. The Document Consumer may be considered overburdened to fully implement the requirements of Secure Node or Secure Application. The Document Responder is more full featured and should generate an equivalent event to the audit event defined in TF-2a:3.18.5.1.2 Document Registry audit message.

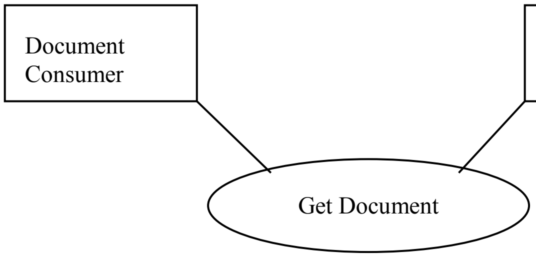
3.68 Get Document ITI-68

This section corresponds to Transaction ITI-68 of the IHE Technical Framework. Transaction ITI-68 is used by the Document Consumer and Document Responder actors.

3.68.1 Scope

The Get Document transaction is used by the Document Consumer to retrieve a document from the Document Responder.

3.68.2 Use Case Roles



Actor: Document Consumer

Role: Requests a document identified by URL from the Document Responder

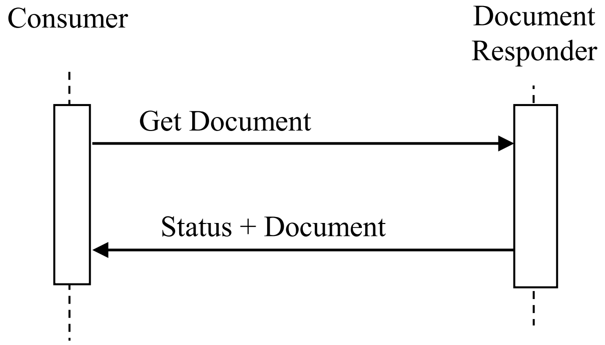
Actor: Document Responder

Role: Serves the document at the provided resource URL to the Document Consumer

3.68.3 Referenced Standard

RFC2616	IETF Hypertext Transfer Protocol – HTTP/1.1
RFC6585	IETF Additional HTTP Status Codes

3.68.4 Interaction Diagram



3.68.4.1 GET Document request Message

This message is a HTTP GET request to retrieve the document.

3.68.4.1.1 Trigger Events

The Document Consumer needs a copy of an identified document.

3.68.4.1.2 Message Semantics

The Document Consumer sends a HTTP GET request to the server. The Document Consumer may use content negotiation by providing a HTTP Accept header, according to the semantics of the HTTP protocols (see RFC 2616, section 14.1). The only MIME type assured to be returned is the MIME type indicated in the Document Entry.

http://<location>/net.ihe/Document/<entryUUID>/?PatientID=<PatientID>

Where:

location – a locally defined root part of arbitrary path.

patientID – the CX encoded patient. See Section 3.65.4.1.2.2 Patient Identity. This value may need to be transformed for URL encoding.

entryUUID – the UUID value.

HTTP If-Unmodified-Since header shall not be included in the GET request.

3.68.4.1.3 Expected Actions

The Document Responder shall provide the document in the requested MIME type, or reply with an HTTP status code indicating the error condition. The Document Responder is not required to transform the document.

3.68.4.2 Status + Document Message

This is the return message sent by the Document Responder.

3.68.4.2.1 Trigger Events

The HTTP Response message is sent when completing the Get Document Request.

3.68.4.2.2 Message Semantics

This message complies with the HTTP response message, as required by RFC 2616. The HTTP body contains the Document requested.

If the patientID or entryUUID are missing, the Document Responder shall return HTTP response code 400 - Bad Request.

If the specified entryUUID is not known to the Document Responder or it doesn’t correlate to the provided patientID, the Document Responder shall return HTTP response-code 404 (not found) with the suggested reason-phrase “Document Entry UUID not found”.

If the Document was deprecated, the Document Responder may send a status code of 410, if this is acceptable by privacy and security policy. Otherwise a 404 shall be sent.

If the Document Responder is not able to format the document in any content types listed in the 'Accept' field, it shall return HTTP response code 406 – Not Acceptable.

If the HTTP request specified is otherwise not a legal value according to this transaction, the Document Responder shall return HTTP response-code 403 (forbidden) with the suggested reason-phrase “request type not supported”.

Note: Other HTTP response codes may be returned by the Document Responder, indicating conditions outside of the scope of this transaction, for example, 401 – Authentication Failed might be returned if Document Responder is grouped with the EUA profile Kerberized Server.

The Document Responder should complement the returned error code with a human readable description of the error condition.

The Document Responder may return HTTP redirect responses (responses with values of 301, 302, 303 or 307) in response to a request. Document Consumer must follow redirects, but if a loop is detected, it may report an error.

3.68.4.2.3 Expected Actions

The Document Consumer is expected to continue its workflow upon receiving the document.

If a Document Consumer cannot automatically recover from an error condition, at a minimum, it should display the error to the user.

3.68.5 Security Considerations

See the general Security Considerations in ITI TF-1:33.5.

3.68.5.1 Security Audit Considerations

The Security audit criteria are similar to those for the Retrieve Document Set – b transaction [ITI-43] as this transaction does import a Document Entry. Grouping the Document Consumer or Document Responder with an ATNA Secure Node or Secure Application is recommended, but not mandated. The Document Consumer may be considered overburdened to fully implement the requirements of Secure Node or Secure Application. The Document Responder is more full featured and should generate an equivalent event to the audit event defined in ITI TF-2b:3.43.6.1.2 Document Repository audit message.