Digital Imaging and Communications in Medicine (DICOM)

Supplement 148: Web Access to DICOM Persistent Objects by Means of Web Services

Extension of the Retrieve Service (WADO Web Service)

Prepared by:

DICOM Standards Committee, Working Group 27 Web Technology

1300 N. 17th Street, Suite 1752

Rosslyn, Virginia 22209 USA

Developed in accordance with: DICOM Workitem 2008-04-B, and New Work Item Proposal ISO/TC215/SC WG2 N631

Contact: svastagh@medicalimaging.org

VERSION: Final Text, 26 August 2011

Table of Contents

Scope and Field of Application

This Supplement defines Web Services for providing DICOM images and other persistent objects to an Electronic Medical Record/Electronic Health Record (EMR/EHR) system. Access to DICOM normalized objects is not defined within this supplement.

This supplement deals only with retrieval, corresponding to the evolution of the existing WADO to Web Services. Query and notification mechanisms are not defined within this supplement. Both native DICOM and rendered images can be retrieved as well as total or partial metadata of the object without the image pixels.

Security aspects are out of the scope of this supplement. However the proposed mechanism is fully compatible with generic security mechanisms used in Web Services.

The healthcare world has need for Service Oriented Architecture (SOA) employing Web Services (WS) to provide access to image management systems from point of service (POS) systems. There is a need to standardize such Web Services, and harmonize with other uses of Web Services in imaging, such as by IHE XDS-I.b.

Since this document proposes changes to existing Parts of DICOM the reader should have a working understanding of the Standard.

Add Annex XX to Part 17

Annex XX - Evolution of WADO to Web Services (Informative)

This annex discusses the design considerations that went into the definition of the WADO extension to Web services.

XX.1 REQUEST AND RESPONSE PARAMETERS

XX.1.1 Request Parameters

The new service based on WS should continue to support all the request parameters defined by WADO, for maintaining backward compatibility with the present URI based WADO, including the options to return either native DICOM objects or a rendered object (JPEG, PDF etc.). These are summarized as below:

Table XX.1-1 Summary of DICOM/Rendered URI based WADO Parameters

Parameter	Allowed for	Requirement in Request
requestType	DICOM & Rendered	Required
studyUID	DICOM & Rendered	Required
seriesUID	DICOM & Rendered	Required
objectUID	DICOM & Rendered	Required
contentType	DICOM & Rendered	Optional
charset	DICOM & Rendered	Optional
anonymize	DICOM	Optional
annotation	Rendered	Optional
Rows, columns	Rendered	Optional
region	Rendered	Optional
windowCenter, windowWidth	Rendered	Optional
imageQuality	DICOM & Rendered	Optional
presentationUID	Rendered	Optional
presentationSeriesUID	Rendered	Optional
transferSyntax	DICOM	Optional
frameNumber	DICOM & Rendered	Optional

For the WS “DICOM Requester” transaction, the parameters will be the following:

Table XX.1-2 Summary of “DICOM Requester” WADO-WS Parameters

Parameter	Requirement in Request	Multiplicity
StudyRequest	Required	One
>SeriesRequest	Required	One or more
>>DocumentRequest	Required	One or more
>>>RepositoryUniqueId	Optional	One
>>>DocumentUniqueId	Required	One
>>>HomeCommunityId	Optional	One
>>>FrameNumber	Optional	One
>>>Anonymize	Optional	One
>>>TransferSyntaxUIDList	Optional	One
>>>>TransferSyntaxUID	Required	One or more

178

Table XX.1-3 Summary of "Rendered Requester" WADO-WS Parameters

Parameter	Requirement in Request	Multiplicity
StudyRequest	Required	One
>SeriesRequest	Required	One or more
>>DocumentRequest	Required	One or more
>>>RepositoryUniqueId	Optional	One
>>>DocumentUniqueId	Required	One
>>>HomeCommunityId	Optional	One
>>>Annotation	Optional	One
>>>Rows / Columns	Optional	One
>>>Region	Optional	One
>>>WindowCenter/ WindowWidth	Optional	One
>>>ImageQuality	Optional	One
>>>PresentationUID	Optional	One
>>>PresentationSeriesUID	Optional	One
>>>FrameNumber	Optional	One
>>>Anonymize	Optional	One
>>>ContentTypeList	Required	One
>>>>ContentType	Required	One or more
>>>CharsetList	Optional	One
>>>>Charset	Required	One or more

180

Table XX.1-4 Summary of "Metadata Requester" WADO-WS Parameters

182

Parameter	Requirement in Request	Multiplicity
StudyRequest	Required	One
>SeriesRequest	Required	One or more
>>DocumentRequest	Required	One or more
>>>RepositoryUniqueId	Optional	One
>>>DocumentUniqueId	Required	One
>>>HomeCommunityId	Optional	One
>>>Anonymize	Optional	One
>>>XPath	Required	One

XX.1.2 Response parameters

In the URI based WADO, the response is the single payload returned in the HTTP Get response. It may be the DICOM object in a DICOM format or in a rendered format.

In the Web Services implementation, for the “DICOM Requester” and the “Rendered Requester” transactions, one or more DICOM objects are returned using the MTOM/XOP mechanism as well as associated metadata.

For the “Metadata Requester” transaction, the response will contain the an XML encoded part containing the information selected from the retrieved objects header using the “XPath” filter as described in the Native DICOM Model defined in PS3.19.

XX.2 WEB SERVICES IMPLEMENTATION

The implementation architecture has to maximize interoperability, preserve or improve performance and minimize storage overhead.

The Web Services technologies have been selected to:

a. be firewall friendly and supporting security,
b. be supported by and interoperable between multiple development environments, and
c. have sufficient performance for both large and small text and for binary data.

The XML implementation of the messages uses the CamelCase parameter style used in SOAP 1.2 (element names starting with an upper case character, e.g., ElementOne, attribute names starting with a lower case character e.g. attributeOne).

The response will be provided as list of instances in MTOM/XOP (“DICOM” or “Rendered” Requesters), XML encoded additional information resulting from the XPath filters applied on every objects selected (“Information Requester”)

XX.3 USES FOR WADO WEB SERVICES

XX.3.1 General requirements

Imaging information is important in the context of EMR/EHR. But EMR/EHR systems often do not support the DICOM protocol. The EMR/EHR vendors need access using web and web service technologies to satisfy their users.

XX.3.2 Analysis of use cases

Examples of use cases / clinical scenarios, as the basis to develop the requirements, include:

1. Providing access to images and reports from a point-of-service application e.g., EMR.
2. Following references to significant images used to create an imaging report and displaying those images.
3. Following references / links to relevant images and imaging reports in email correspondence or clinical reports e.g., clinical summary.
4. Providing access to anonymized DICOM images and reports for clinical research and teaching purposes.
5. Providing access to a DICOM encoded imaging report associated with the DICOM IE (patient/study/series/objects) to support remote diagnostic workflows e.g., urgent medical incidents, remote consultation, clinical training, teleradiology/telemedicine applications.
6. Providing access to summary or selected information from DICOM objects.

Examples of the use cases described in 1 above are:

a. The EMR displays in JPEG one image with annotations on it (patient and/or technique related), based upon information provided in a report.
b. The EMR retrieves from a “Manifest” document all the referenced objects in DICOM and launches a DICOM viewer for displaying them (use case addressed by the IHE XDS-I.b profile).
c. The EMR displays in JPEG one image per series with information describing every series (e.g. series description).
d. The EMR displays in JPEG all the images of a series with information describing the series as well as every image (e.g. instance number and slice location for scanner images).
e. The EMR populates in its database for all the instances referred in a manifest (KOS) the relevant information (study ID/UID/AccessionNumber/Description/DateTime, series UID/Modality/Description/DateTime, instance UID/InstanceNumber/SliceLocation).

As an example, the 1c use case is decomposed in the following steps (all the other use cases can be implemented through a similar sequence of basic transactions):

A. The EMR sends to the DICOM server the list of the objects (“selection”), asking for the object content.

B. The DICOM server sends back the JPEG images corresponding to the listed objects.

C. The EMR sends to the DICOM server the “selection” information for obtaining the relevant information about the objects retrieved.

D. The DICOM server sends back the corresponding information in the form of a “metadata” document, converted in XML.

XX.3.3 Description of the Use Cases

The use cases described above in terms of clinical scenarios correspond to the following technical implementation scenarios. In each case the use is distinguished by the capabilities of the requesting system:

- Does it prefer the URI based requests, or the web-services based requests.
- Does it have the ability to decode and utilize the DICOM PS 3.10 format or not.
- Does it need the metadata describing the image and its acquisition, and/or does it need an image to be displayed.

These then become the following technical use cases.

XX.3.3.1 URI based WADO Use Case

A. The requesting system is Web Browser or other application that can make simple HTTP/HTTPS requests,

B. Reference information is provided as URL or similar information that can be easily converted into a URL.

C. The request specifies:

1. Individual SOP Instance
2. Desired format and subset selection for information to be returned

D. The Response provides

1. SOP instance, reformatted and subset as requested. This may be encoded as a DICOM PS 3.10 instance, or rendered into a generic image format such as JPEG.

XX.3.3.2 DICOM (Encoded Content) Requestor

A. The requesting system is an application capable of making Web Service requests and able to process data encoded as a DICOM File, per DICOM PS 3.10 encodings.

B. Reference information may come in a wide variety of forms. It is expected to include at least the Study UID, Series UID, and Individual SOP instance information. This may be encoded as part of an HL7 reference within a CDA document, a DICOM SOP Instance reference, or other formats.

C. The request specifies

1. Requested Dataset
   a) Study UID
   b) List of Series UID
   c) List of SOP Instance UIDs
2. Optionally, it may also specify subset information
   a) Instance and Frame Level Retrieve SOP classes subset information for selecting frames
   b) No-pixel data request (using the Transfer Syntax parameter)
   c) Anonymization

D. The response provides

1. SOP Instances, encoded per DICOM PS 3.10.

XX.3.3.3 Rendered (JPEG/PDF) Requestor

A. The requesting system: application capable of making Web Service requests. System is not capable of decoding DICOM PS 3.10 formats. The system is capable of processing images in JPEG or other more generic formats.

B. Reference information may come in a wide variety of forms. It is expected to include at least the Study UID, Series UID, and Individual SOP instance information. This may be encoded as part of an HL7 reference within a CDA document, a DICOM SOP Instance reference, or other formats.

C. Request information

1. Requested Dataset
   a) Study UID
   b) List of Series UID

c) List of SOP Instance UIDs

2. Desired format and subset information
   a) JPEG/PDF/ etc selection, subset area, presentation information
   b) Frame selection for subsets of multi-frame objects
   c) What should be done for requests where image shapes and SOP classes vary and a subset is requested?
   d) Anonymize or not.

D. Response information
   1. JPEGs
      a) Should JPEGs include tag information within the JPEG? If so, what information?
      b) How will JPEGs be related to multi-frame and multi-instance requests? Order? Tag?
   2. PDFs
      a) How will PDFs be related to multi-frame and multi-instance requests? One per frame? One per instance? One for entire set?
   3. Other encodings?

XX.3.3.4 Metadata (XML without pixel data, waveform data, etc) Requestor

A. The requesting system: application capable of making Web Service requests. The requesting System is not capable of decoding DICOM PS 3.10 formats. The system is capable of processing metadata that describes the image, provided that the metadata is encoded in an XML format. The system can be programmed based upon the DICOM definitions for XML encoding and attribute meanings.

B. Reference information may come in a wide variety of forms. It is expected to include at least the Study UID, Series UID, and Individual SOP instance information. This may be encoded as part of an HL7 reference within a CDA document, a DICOM SOP Instance reference, or other formats.

C. Request information
   1. Requested Dataset
      a) Study UID
      b) List of Series UID
      c) List of SOP Instance UIDs
   2. Desired format and subset information
      a) XPath definition for subset or total metadata selection
      b) What should be done when SOP classes vary and a subset is requested? The XPath will fail.
      c) Frame selection for subsets of multi-frame objects
      d) Anonymize or not.
      e) Response information

D. Response information
   1. XML encoded metadata.

XX.4 IHE ITI COMPATIBILITY

There is a strong desire that the ITI Transaction RAD-69 be a proper implementation of the DICOM WS-* transaction. Note that RAD-69 is not the entire suite of XD* transactions. It is the “Retrieve Imaging Document Set” transaction.

The RAD-69 transaction is quite simple, can be difficult to find all the parts of the ITI documentation. In summary, the RAD-69 transaction is a WS request to the IHE "RequestDocumentSet" action and related endpoints. The request is a list of "DocumentRequest", each "DocumentRequest" has three elements: required OID, required RepositoryID, and optional CommunityID. The response is a list of "DocumentResponse". Each "DocumentResponse" has four elements: required OID, required RepositoryID, required Document, and optional CommunityID.

The mapping to DICOM for OID would be SOP Instance UID, and Document the DICOM contents. RepositoryID is analogous to the AE Title. It is not a perfect mapping. IHE considers the configuration where one system acts as a front end for multiple other systems, each identified by a RepositoryID. The CommunityID is an extension of this to "communities" that exchange data through gateways. The gateways will use the RepositoryID to identify internal repository systems.

RAD-69 requires no understanding of document contents. They are binary blobs that are identified by an OID.

XX.5 PROXY AGENT FOR NON-WS DICOM ARCHIVE

Rapid acceptance will be enhanced if a proxy system that automatically converts between the WS notation and the older DICOM C-FIND/etc transaction can be defined; and if this conversion can be simple. Proxy systems can also simplify security configuration.

Changes to NEMA Standards Publication PS 3.2-2009

Digital Imaging and Communications in Medicine (DICOM)

Part 2:

Modify section A.4.2 of the Conformance Claim Template

A.4.2 AE SPECIFICATIONS:

The next section in the DICOM Conformance Statement is a set of Application Entity Specifications. There shall be one such specification for each Application Entity. Each individual AE Specification has a subsection, A.4.2.x. There are as many of these subsections as there are different AE's in the implementation. That is, if there are two distinct AE's, then there will be two subsections, A.4.2.1, and A.4.2.2.

A.4.2.1 "Application Entity <1>"

Every detail of this specific Application Entity shall be completely specified under this section.

AE's that utilize the DIMSE services shall have the following sections.

Note: AE's that utilize other services are described later, and will re-use this section numbering.

A.4.2.1.1 SOP Classes

The specification for an Application Entity shall contain a statement of the form:

Add new sections A.4.2 for the WADO conformance after the A.4.2 for DICOM Conformance.

A.4.2.1.4.1.3 SOP Specific Conformance for SOP Class(es)

This section includes the SOP specific behavior, i.e. error codes, error and exception handling, time-outs, etc. The information shall be as described in the SOP specific Conformance Statement section of PS 3.4 (or relevant private SOP definition).

The behavior of an Application Entity shall be summarized as shown in Table 4.2.13. Standard as well as the manufacturer specific status codes and their corresponding behavior shall be specified.

Table 4.2 - 13
STORAGE C-STORE RESPONSE STATUS

Service Status	Further Meaning	Error Code	Reason
Success	Success	0000	Explain
Refused	Out of Resources	A700-A7FF	Explain
Error	Data Set does not match SOP Class	A900-A9FF	Explain
Error	Specify	Specify	Explain
Warning	Specify	Specify	Explain

An Application Entity that supports the WADO transport services shall have the following sections:

A.4.2.1 "Application Entity <1>"

Details of this specific Application Entity shall be specified under this section.

A.4.2.1.1 WADO WS Specifications

All WADO WS services that are supported shall be listed. Other WADO WS services that are not supported may be indicated.

For each supported service, the parameters and restrictions on those parameters shall be described.

Any connection policies such as restrictions on the number of connections, support for asynchronous WS requests, etc. shall be described.

A.4.2.1.2 WADO URI Specifications

All WADO URI services that are supported shall be listed. Other WADO URI services that are not supported may be indicated.

For each supported service, the parameters and restrictions on those parameters shall be described.

Any connection policies such as restrictions on the number of connections, support for pipeline requests, etc. shall be described.

Add Annex XX: Sample Conformance Claim for WADO Service

ANNEX X (Informative) CONFORMANCE STATEMENT SAMPLE WADO SERVICE

Disclaimer:

This document is an example DICOM Conformance Statement for a fictional application service called EXAMPLE-WADO-SERVICE produced by a fictional vendor called EXAMPLE-PACS-PRODUCTS.

As stated in the annex title, this document is truly informative, and not normative. A conformance statement of an actual product might implement additional services and options as appropriate for its specific purpose. In addition, an actual product might implement the services described in a different manner and, for example, with different characteristics and/or sequencing of activities. In other words, this conformance statement example does not intend to standardize a particular manner that a product might implement DICOM functionality.

X.0 COVER PAGE

Company Name: EXAMPLE-PACS-PRODUCTS.

Product Name: EXAMPLE-WADO-SERVICE

Version: 1.0-rev. A.1

Internal document number: 4226-xxx-yyy-zzz rev 1

Date: YYYYMMDD

X.1 CONFORMANCE STATEMENT OVERVIEW

436 This fictional product EXAMPLE-WADO-SERVICE implements both the WADO URI services and the WADO WS services for access to DICOM SOP Instances that are stored on an EXAMPLE-PACS-
438 ARCHIVE. The EXAMPLE-WADO-SERVICE is only available as a plug in option for the EXAMPLE-PACS-ARCHIVE. All of the networking, database, and other services are provided by the EXAMPLE-
440 PACS-ARCHIVE. This conformance claim refers to the conformance claim for the EXAMPLE-PACS-ARCHIVE for all such services.

442 Table X.1-1 provides an overview of the network services supported by EXAMPLE-INTEGRATED-MODALITY.

444 **Table X.1-1**
NETWORK SERVICES

Network Service	User of Service (Client)	Provider of Service (Server)
WADO		
WADO – URI – Retrieve Imaging Document	No	Yes
WADO – URI – Retrieve Rendered Imaging Document	No	Yes
WADO – WS – Retrieve Imaging Document Set	No	Yes
WADO – WS – Retrieve Rendered Imaging Document Set	No	Yes

446

X.2 TABLE OF CONTENTS

448 A table of contents shall be provided to assist readers in easily finding the needed information.

X.3 INTRODUCTION

X.3.1 REVISION HISTORY

Document Version	Date of Issue	Author	Description
1.1	October 30, 2003	WG 6	Version for Final Text
1.2	August 30, 2007	WG 6	Revised Introduction

X.3.2 AUDIENCE, REMARKS, TERMS AND DEFINITIONS, BASICS OF DICOM COMMUNICATION, ABBREVIATIONS, REFERENCES

See example text in Annex A.3.

X.3.3 ADDITIONAL REMARKS FOR THIS EXAMPLE

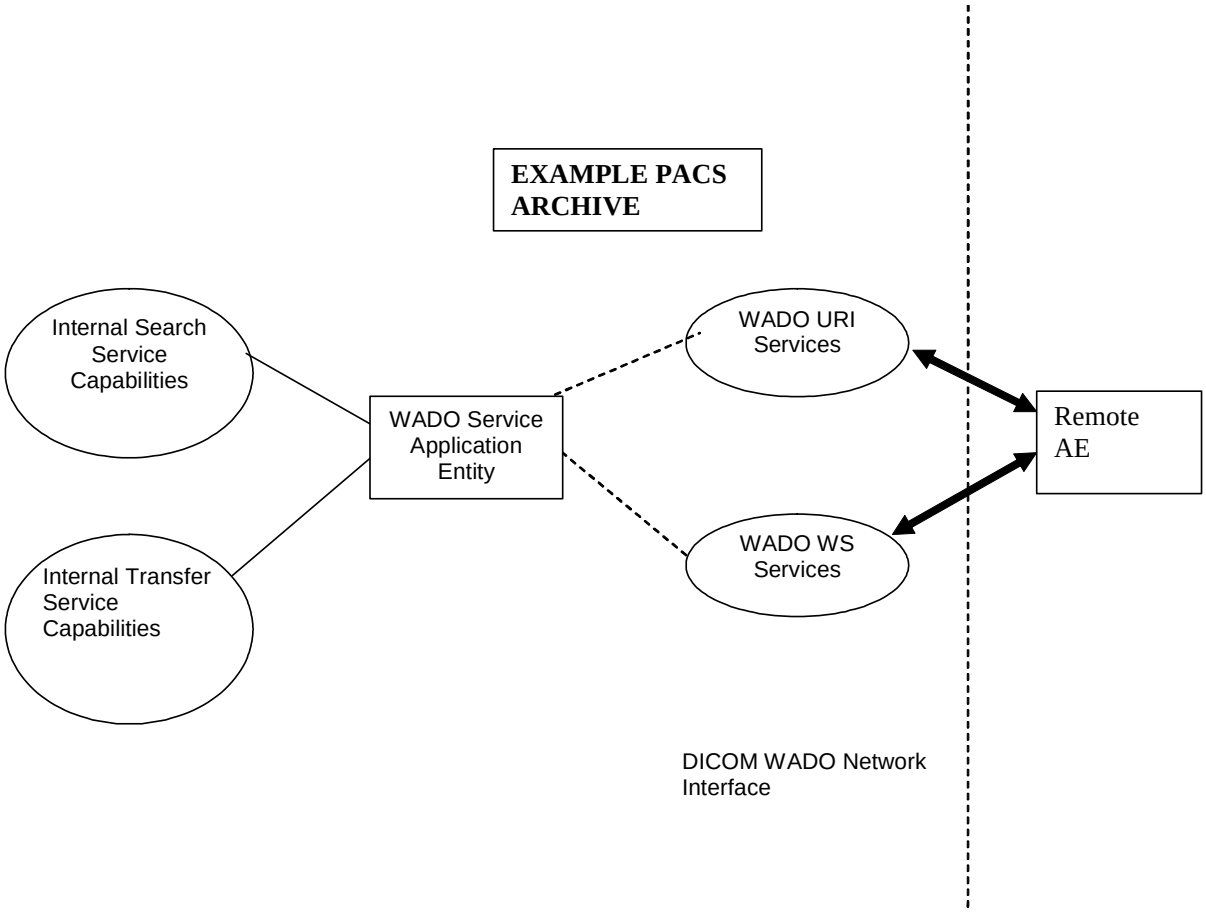
This document is a sample DICOM Conformance Statement created for DICOM PS3.2. It is to be used solely as an example to illustrate how to create a DICOM Conformance Statement for an acquisition modality. The subject of the document, EXAMPLE-WADO-SERVICE, is a fictional product.

460

X.4 NETWORKING

462 ## X.4.1 IMPLEMENTATION MODEL

X.4.1.1 Application Data Flow



464

466 **Figure X.4.1-1**
APPLICATION DATA FLOW DIAGRAM

468 The WADO Service Application receives WADO requests from a remote AE. These requests may be either over the URI or WS interfaces. It is associated with the local real-world activity “Retrieve Images”.
470 It converts these requests into internal lookup functions to find the matching SOP Instances. It then obtains these matching SOP Instances and composes a response back to the requesting remote AE.

472

474 **X.4.1.2 Functional Definition of AEs**

X.4.1.2.1 Functional Definition of WADO Service Application

476 The reception of a WADO request will activate the AE. An internal request is sent to the search capabilities of the EXAMPLE-PACS-ARCHIVE. This request is based upon the request parameters from the WADO
478 request. The response is a list of all SOP instances stored on the EXAMPLE-PACS-ARCHIVE that match the request parameters. If there are no matching instances, the AE will indicate this in the WADO
480 response. For all matching instances, the AE will utilize the internal image transfer request to obtain a copy of each instance. If the request was for retrieval of instances, these instances will be returned. If the
482 request was for retrieval of rendered instances, then the AE will render each instance and return the rendered results.

484

X.4.2 AE SPECIFICATIONS

486 This AE complies with PS 3.18, Annex X, specifications for both WS and URI access.

X.4.2.1 WADO WS Specifications

488 **X.4.2.1.1 WADO WS Retrieve Imaging Document Set**

Table X.4.2-1
490 **WADO WS RETRIEVE IMAGING DOCUMENT SET SPECIFICATION**

Parameter	Restrictions
Transfer Syntaxes Supported	Any transfer syntax supported by the hosting EXAMPLE-PACS-ARCHIVE
SOP Class Restrictions	Any SOP class supported by the hosting EXAMPLE-PACS-ARCHIVE
Size restriction	Any size supported by the hosting EXAMPLE-PACS-ARCHIVE
Anonymization	Supports the DICOM Basic Application Level Confidentiality Profile plus the Retain Patient Characteristics option.

492 **X.4.2.1.2 WADO WS Retrieve Rendered Imaging Document Set**

Table X.4.2-3
494 **WADO WS Retrieve Rendered Imaging Documents Specification**

Parameter	Restrictions
Transfer Syntaxes Supported	Restricted to transfer syntaxes supported by the hosting EXAMPLE-PACS-ARCHIVE
SOP Class Restrictions	Restricted to SOP classes supported by the hosting EXAMPLE-PACS-ARCHIVE
Size restriction	Restricted to sizes supported by the hosting EXAMPLE-PACS-ARCHIVE
Rendered formats available	Supports JPEG and PDF for IMAGE IODS, and PDF for non-IMAGE IODS.
Rows restrictions	Must be in range 16 - 32767
Columns restrictions	Must be in range 16 - 32767
Region restrictions	None
Window Center restrictions	None

Parameter	Restrictions
Window Width restrictions	None
Image Quality restrictions	None
Anonymization	Supports the DICOM Basic Application Level Confidentiality Profile plus the Retain Patient Characteristics option.
Annotation restrictions	None
Compression available	JPEG
Other restrictions	None

X.4.2.1.3 WADO WS Retrieve Imaging Document Set Metadata

Not supported

X.4.2.1.2 Connection Policies

X.4.2.1.2.1 General

All standard WS connection policies apply. There are no extensions for WS options.

X.4.2.1.2.2 Number of Connections

EXAMPLE-WADO-SERVICE limits the number of simultaneous WS requests. Additional requests will be queued after the TCP connection is accepted. When an earlier request completes, a pending request will proceed.

Table X.4.2-4
NUMBER OF WS REQUESTS SUPPORTED

Maximum number of simultaneous WS requests	100 (configurable)

X.4.2.1.2.3 Asynchronous Nature

EXAMPLE-WADO-SERVICE does not support WS asynchronous response.

X.4.2.2 WADO URI Specification

X.4.2.2.1 WADO URI Retrieve Imaging Document Set

Table X.4.2-1
WADO URI RETRIEVE IMAGING DOCUMENTS SPECIFICATION

Parameter	Restrictions
Transfer Syntaxes Supported	Restricted to transfer syntaxes supported by the hosting EXAMPLE-PACS-ARCHIVE
SOP Class restrictions	Restricted to SOP classes supported by the hosting EXAMPLE-PACS-ARCHIVE
Size restriction	Restricted to sizes supported by the hosting EXAMPLE-PACS-ARCHIVE
Anonymization	Supports the DICOM Basic Application Level Confidentiality Profile plus the Retain Patient Characteristics option.

If the URI Retrieve specifies no transfer syntax that is supported by the archive, the SOP Instance will be returned using the Implicit VR Little Endian transfer syntax.

X.4.2.2.2 WADO WS Retrieve Rendered Imaging Document Set

Table X.4.2-3
WADO URI RETRIEVE RENDERED IMAGING DOCUMENTS SPECIFICATION

Parameter	Restrictions
Transfer Syntaxes Supported	Restricted to transfer syntaxes supported by the hosting EXAMPLE-PACS-ARCHIVE
SOP Class restrictions	Restricted to SOP classes supported by the hosting EXAMPLE-PACS-ARCHIVE
Size restriction	Restricted to sizes supported by the hosting EXAMPLE-PACS-ARCHIVE
Rendered formats available	Supports JPEG and PDF for IMAGE IODS, and PDF for non-IMAGE IODS.
Rows restrictions	Must be in range 16 - 32767
Columns restrictions	Must be in range 16 - 32767
Region restrictions	None
Window Center restrictions	Whole window must be in the range of image pixel values.
Window Width restrictions	Must be greater than 4 and whole window must be in the range of image pixel values.
Image Quality restrictions	None
Anonymization	Supports the DICOM Basic Application Level Confidentiality Profile plus the Retain Patient Characteristics option.
Annotation Restrictions	None
Compression available	JPEG
Other restrictions	None

X.4.2.2.3 WADO URI Retrieve Imaging Document Set Metadata

Not supported.

X.4.2.2.4 Connection Policies

X.4.2.2.4.1 General

All URI connections are limited to HTTP GET requests. The EXAMPLE-WADO-SERVER ignores all unknown HTTP header parameters.

X.4.2.2.4.2 Number of Connections

EXAMPLE-WADO-SERVER limits the number of simultaneous HTTP connections.

Table X.4.2-4
NUMBER OF HTTP REQUESTS SUPPORTED

Maximum number of simultaneous HTTP requests	100 (configurable)

532 **X.4.2.2.4.3 Asynchronous Nature**

EXAMPLE-INTEGRATED-MODALITY supports HTTP pipelined requests and responses.

534 **X.4.3 NETWORK INTERFACES**

X.4.3.1 Physical Network Interface

536 EXAMPLE-WADO-SERVER uses the network interface from the hosting EXAMPLE-PACS-ARCHIVE. See its conformance claim for details.

538 **X.4.3.2 Additional Protocols**

EXAMPLE-WADO-SERVER uses the network services from the hosting EXAMPLE-PACS-ARCHIVE.
540 See its conformance claim for details.

542 **X.4.3.3 IPv4 and IPv6 Support**

This product supports both IPv4 and IPv6 connections.

544 **X.4.4 CONFIGURATION**

X.4.4.1 HTTP URI Interface

546 The EXAMPLE-WADO-SERVER can be configured to respond on two ports, one for unprotected HTTP traffic and one for TLS protected traffic. The TLS port will refuse any connection from a system that is not
548 recognized as authenticated by a known authority.

X.4.4.2 WS Interface

550 The EXAMPLE-WADO-SERVER can be configured to respond on either one or two service endpoints. Each endpoint offers both of the services.

552 The WSDL file to be used by clients is made available at the location http://<servername>/EXAMPLE-WADO-SERVER?WSDL .

554 ## X.5 Media Interchange

Not applicable

X.6 SUPPORT OF CHARACTER SETS

All EXAMPLE-WADO-SERVERs support Unicode UTF-8 for all WS transactions. The EXAMPLE-WADO-SERVER does not convert charactersets when returning SOP Instances using DICOM encoding. The original DICOM encoded charactersets are preserved. When a PDF encoding is returned, characterset conversion is performed and the PDF is returned with a UTF-8 encoding. JPEG renderings, will also utilize UTF-8 encoding for internal labels.

See conformance claim for EXAMPLE-PACS-ARCHIVE for charactersets used within the DICOM instances.

X.7 SECURITY

EXAMPLE-INTEGRATED-MODALITY supports transport level security measures for URI access, and the WS-Security services for WS access.

The transport level security measures are the support for bi-directional authentication using TLS connections. The EXAMPLE-WADO-SERVER can provide it's certificate information, and can be configured with either a direct comparison (self-signed) certificate or a chain of trust certificate.

The EXAMPLE-WADO-SERVER will refuse a connection over TLS from a source that does not have a recognized authentication. For example, a certificate authenticated by "Big Bank Corp." will not be accepted unless the EXAMPLE-WADO-SERVER has been configured to accept authentications from "Big Bank Corp." The list of acceptable certificates for EXAMPLE-WADO-SERVER is not shared with certificates used by other system applications and must be maintained independently.

576 # X.8 ANNEXES

X.8.1 IOD CONTENTS

578 See Conformance claim for the EXAMPLE-PACS-ARCHIVE.

X.8.3 CODED TERMINOLOGY AND TEMPLATES

580 See conformance claim for EXAMPLE-PACS-ARCHIVE

X.8.4 GRAYSCALE IMAGE CONSISTENCY

582 The EXAMPLE-WADO-SERVER assumes that the JPEG images will be displayed with monitors calibrated to the sRGB profile when rendering images.

584 X.8.5 STANDARD EXTENDED / SPECIALIZED / PRIVATE SOP CLASSES

See conformance claim for EXAMPLE-PACS-ARCHIVE

586 X.8.6 PRIVATE TRANSFER SYNTAXES

If you request a DICOM object, it will not be returned in a private transfer syntax.

588

Changes to NEMA Standards Publication PS 3.18-2009

Digital Imaging and Communications in Medicine (DICOM)

Part 18: Web Access to DICOM Persistent Objects (WADO)

Item #1: Modify PS 3.18 Section 1 Scope as indicated.

1 Scope

This standard specifies a web-based service for accessing and presenting DICOM (Digital Imaging and Communications in Medicine) persistent objects (e.g. images, medical imaging reports). This is intended for distribution of results and images to healthcare professionals. It provides a simple mechanism for accessing DICOM persistent object(s) **~~from HTML pages or XML documents~~**, through HTTP/HTTP**S**~~s~~ protocol, using DICOM UIDs (Unique Identifiers). Data may be retrieved either in a presentation-ready form as specified by the requester (e.g. JPEG or GIF) or in a native DICOM format. It does not support facilities for web searching of DICOM images. This standard relates only to DICOM persistent objects (not to other DICOM objects or to non-DICOM objects). Access control beyond the security mechanisms generally available to web applications is outside the scope of this standard.

Item #2: Append PS 3.18 Section 3 Normative Reference as indicated.

IHE ITI TF-2x: Appendix V IHE IT Infrastructure Technical Framework, Volume 2x, Appendix V (Web Services for IHE Transactions)

Move to Informative References

ebRS ebXML Registry Service

Item #3: Append PS 3.18 Section 5 Symbols and abbreviated terms as indicated.

IHE Integrating the Healthcare Enterprise

MTOM Message Transmission Optimization Mechanism

SOAP Simple Object Access Protocol (SOAP12 for SOAP version 1.2)

WS Web Services

WSDL Web Services Description Language

XOP XML-binary Optimized Packaging

Item #4: Modify PS 3.18 Section 6 Data Communication Requirements as indicated.

6.1 INTERACTION

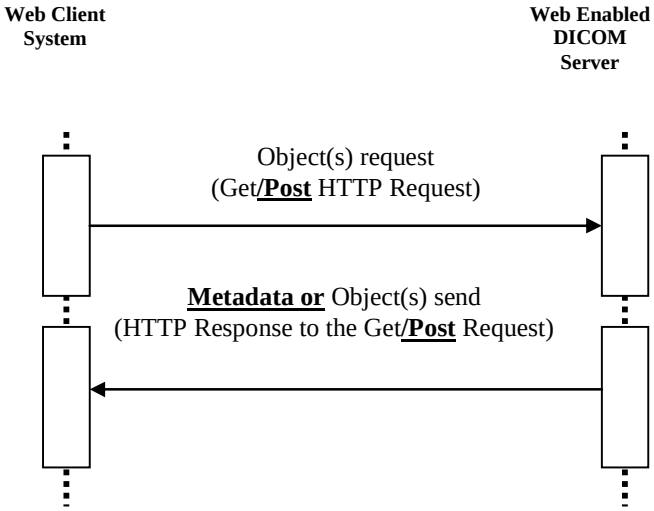


Figure 6-1 — Interaction Diagram

The interaction shall be as shown in Figure 6-1.

Two communications modes are possible:

— **URI based mechanism using HTTP Get: WADO Type request**

— **Web Services (WS) using HTTP Post: WADO WS, either:**

a. DICOM Requester (Retrieve Imaging Document Set)

b. Rendered Requester (Retrieve Rendered Imaging Document Set)

c. Metadata Requester (Retrieve Imaging Document Set Metadata)

6.2 ~~HTTP~~ URI REQUEST

Item #5: Modify PS 3.18 Section 6 Data Communication Requirements as indicated.

6.3 ~~HTTP~~ RESPONSE TO THE URI REQUEST

Item #5: Append PS 3.18 Section 6 Data Communication Requirements as indicated.

6.4 WS REQUEST/RESPONSE

The DICOM Web Service defines several action types. An implementation shall support at least one of these actions. The three action types are:

1. RetrieveImagingDocumentSet

   This action retrieves a set of DICOM instances and other objects. This action corresponds to the IHE XDS-I.b transaction RAD-69. The DICOM instances are formatted in accordance with PS 3.10, and encapsulated in a Web Services response.

2. RetrieveRenderedImagingDocumentSet

   This action retrieves a set of DICOM instances that have been rendered into the requested format. For example, if rendering into JPEG was requested, then these will be the JPEG renderings of the requested set of DICOM objects.

3. RetrieveImagingDocumentSetMetadata

   This action retrieves a set of DICOM instances presented as an Infoset with the bulk data removed. This service can retrieve either the full metadata, or a subset selected by XPath arguments. The XML encoding for the DICOM attributes is defined in PS 3.19.

The Web Services actions shall be fully compliant with the Basic Profile of WS-I as defined in IHE IT Infrastructure Technical Framework Vol 2x Annex V. All <wsa:Action> elements shall have the mustUnderstand attribute set (mustUnderstand="1").

6.4.1 WS - RetrieveImagingDocumentSet

6.4.1.1 Request

The specific Web Services parameters to be used for the Retrieve Imaging Document Set action shall be as follows, in the order that they would appear in the WSDL definition:

- The following types shall be imported (xsd:import) in the /definitions/types section:
  - namespace="urn:ihe:rad:xdsi-b:2009", schema="XDSI.b_ImagingDocumentSource.xsd"

- The baseline XDS.b schema (namespace="urn:ihe:iti:xds-b:2007", schema="XDS.b_DocumentRepository.xsd")

- The /definitions/message/part/@element attribute of the Retrieve Imaging Document Set Request message shall be an "iherad:RetrieveImagingDocumentSetRequest" as defined below.
- The /definitions/message/part/@element attribute of the Retrieve Imaging Document Set Response message shall be an "ihe:RetrieveDocumentSetResponse" as defined below.
- The /definitions/portType/operation/input/@wsaw:Action attribute for the Retrieve Imaging Document Set Request message shall be "urn:ihe:rad:2009:RetrieveImagingDocumentSet".
- The /definitions/portType/operation/output/@wsaw:Action attribute for the Retrieve Imaging Document Set Response message shall be "urn:ihe:iti:2007:RetrieveDocumentSetResponse".
- The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be "urn:ihe:rad:2009:RetrieveImagingDocumentSet".

The <iherad:RetrieveImagingDocumentSetRequest/> element for use with the Retrieve Imaging Document Set Request Message is defined as:

- One or more <iherad:StudyRequest/> elements each of which includes a "studyInstanceUID" attribute identifying the study associated with the DICOM images/ objects being retrieved. Each <iherad:StudyRequest/> element shall contain:
    - One or more <iherad:SeriesRequest/> elements each of which includes a "seriesInstanceUID" attribute identifying the series associated with the DICOM images/ objects being retrieved. Each <iherad:SeriesRequest/> element shall contain:
        - One or more <ihe:DocumentRequest/> elements, each one representing an individual document that the requestor wants to retrieve from the Web Server. Each <ihe:DocumentRequest/> element contains:
            - An optional <ihe:RepositoryUniqueId/> element that identifies the Web Server from which the document is to be retrieved. This value corresponds to XDSDocumentEntry.repositoryUniqueId.

              The RepositoryUniqueID is similar to a DICOM AETitle, but is a uniqueID assigned to the WADO-WS Web Server rather than a locally assigned string identifier. There will be a separate RepositoryUniqueID for each web service end point.
            - A required <ihe:DocumentUniqueId/> element that identifies the document within the source. For example, this value could be a SOP Instance UID obtained from a Key Object Selection (KOS) instance.
            - An optional <ihe:HomeCommunityId/> element. See the IHE Profiles for the definition and possible uses of this element.
            - An optional <wado:Anonymize/> element.
            - An optional <wado:FrameNumber/> element.
            - A required <iherad:TransferSyntaxUIDList/> element which contains a list of one or more <ihe:TransferSyntaxUID> elements. Each of the <iherad:TransferSyntaxUID> elements represent one of the transfer syntax encodings that the Imaging Document Consumer is capable of processing.

6.4.1.2 Response

A Web Server shall provide the document(s) indicated in the request. The Web Server shall return the document(s) or an error code when the document could not be returned. The pixel data shall be encoded using one of the DICOM transfer syntaxes included in the Retrieve Imaging Document Set Request

Message. If the Imaging Document Source cannot encode the pixel data using any of the requested transfer syntaxes then an error status shall be returned.

6.4.1.2.1 Form of the Response

The <ihe:RetrieveDocumentResponse/> element for use with the Retrieve Imaging Document Set Response Message is defined as:

- A required /ihe:RetrieveDocumentSetResponse/rs:RegistryResponse element
- An optional sequence of <ihe:DocumentResponse/> elements containing
  - An optional <ihe:HomeCommunityId/> element. The value of this element shall be the same as the value of the /RetrieveImagingDocumentSetRequest/StudyRequest/SeriesRequest/DocumentRequest/HomeCommunityId element in the Retrieve Document Set Request Message. If the <ihe:HomeCommunityId/> element is not present in the Retrieve Document Set Request Message, this value shall not be present.
  - An optional <ihe:RepositoryUniqueId/> that identifies the Imaging Document Source from which the document is to be retrieved. The value of this element shall be the same as the value of the /RetrieveImagingDocumentSetRequest/StudyRequest/SeriesRequest/DocumentRequest/RepositoryUniqueId element in the original Retrieve Imaging Document Set Request Message. This value corresponds to XDSDocumentEntry.repositoryUniqueId.
  - A required <ihe:DocumentUniqueId/> that identifies the document within the Imaging Document Source. The value of this element shall be the same as the value of the /RetrieveImagingDocumentSetRequest/StudyRequest/SeriesRequest/DocumentRequest/DocumentUniqueId element in the original Retrieve Imaging Document Set Request Message. This value corresponds to the SOP Instance UID in the Retrieve Document Request.
  - A conditional <wado:FrameNumber/> that identifies the frame within the source document. It shall be present if and only if <wado:FrameNumber/> was in the request
  - A required <ihe:Document/> element that contains the retrieved document as an XOP infoset.
  - A required <ihe:mimeType/> element that indicates the MIME type of the retrieved document.

The /RetrieveDocumentSetResponse/rs:RegistryResponse/@status attributes provides the overall status of the request: It shall contain one of the following values:

urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Success
urn:ihe:iti:2007:ResponseStatusType:PartialSuccess
urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Failure

See ITI TF-2a: 4.1.13 Error Reporting for the interpretation of these values.

For each document requested in a /RetrieveImagingDocumentSetRequest/StudyRequest/SeriesRequest/DocumentRequest element:

- If the document is successfully retrieved (without warning) then no /RetrieveDocumentSetResponse/rs:RegistryResponse/rs:RegistryErrorList/ rs:RegistryError element shall be present and a /RetrieveDocumentSetResponse/DocumentResponse/Document element shall be returned containing the document as base64binary encoded data.

- If a warning is reported when retrieving the document, then a /RetrieveDocumentSetResponse/rs:RegistryResponse/rs:RegistryErrorList/ rs:RegistryError element shall be returned with:
  - @severity is urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Warning
  - @errorCode is specified
  - @codeContext contains the warning message
  - @location contains the DocumentUniqueId of the document requested
  - The document shall be returned in an instance of /RetrieveDocumentSetResponse/DocumentResponse/Document as base64binary encoded data. The returned document and warning are correlated via the DocumentUniqueId.
- If an error is reported when retrieving a document, then a /RetrieveDocumentSetResponse/rs:RegistryResponse/rs:RegistryErrorList/ rs:RegistryError element shall be returned with:
  - @severity is urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Error
  - @errorCode is specified
  - @codeContext contains the error message
  - @location contains the DocumentUniqueId of the document requested
  - No corresponding RetrieveDocumentSetResponse/DocumentResponse element shall be returned

The error conditions for failures and associated error codes are given below in section 6.4.4. These errors shall be detected and the associated errorCode returned if that error occurs. Additional errors defined in the ebRS standard, in ITI TF-2: 4.1.13 “Error Reporting”, and defined by the implementor may be returned.

6.4.1.2.2 JPIP

If the Web Client specifies a transfer syntax field of 1.2.840.10008.1.2.4.94 (DICOM JPIP Referenced Transfer Syntax) or 1.2.840.10008.1.2.4.95 (DICOM JPIP Referenced Deflate Transfer Syntax), and the Web Server supports the requested transfer syntax the following behavior is expected:

- If the DICOM Image Object(s) already have the same JPIP transfer syntax as the one indicated in the request, the Retrieve Imaging Document Set Response shall include the DICOM Image Objects unchanged.
- If the DICOM Image Object(s) have a transfer syntax that differs from that of the request, the Retrieve Imaging Document Set Response shall include the DICOM image with the transfer syntax changed to the requested transfer syntax. In addition, the pixel data Attribute (7FE0,0010 tag) will have been removed and replaced with a Pixel Data Provider URL (0028,7FE0 tag). The URL represents the JPIP request and will include the specific target information.
- Upon receipt of this Retrieve Imaging Document Set Response, the Imaging Document Consumer may request the pixel data from the pixel data provider using the supplied URL. Additional parameters required by the application may be appended to the URL when accessing the pixel data provider.
- For example, a JPIP request for a 200 by 200 pixel rendition of the entire image can be constructed from the Pixel Data Provider URL as follows:
  - Pixel Data Provider URL (0028,7FE0) = https://server.xxx/jpipserver.cgi?target=imgxyz.jp2,
  - URL Generated by the application = https://server.xxx/jpipserver.cgi?target=imgxyz.jp2&fsiz=200,200

812

814 6.4.2 WS – RetrieveRenderedImagingDocumentSet

6.4.2.1 Request

816 The specific Web Services parameters to be used for the Retrieve Rendered Imaging Document Set action shall be as follows, in the order that they would appear in the WSDL definition:

818 - The following types shall be imported (xsd:import) in the /definitions/types section:
    - namespace="urn:ihe:rad:xdsi-b:2009",
820 schema="XDSI.b_ImagingDocumentSource.xsd"
    - The baseline XDS.b schema (namespace="urn:ihe:iti:xds-b:2007",
822 schema=”XDS.b_DocumentRepository.xsd”)
    - The baseline DICOM WADO-WS schema
824 (namespace=”urn:dicom:wado:ws:yyyymmdd”,
schema=”dicom.wado.ws.yyyymmdd.xsd”)
826 - The /definitions/message/part/@element attribute of the Retrieve Rendered Imaging Document Set Request message shall be a “wado:RetrieveRenderedImagingDocumentSetRequest” as defined
828 below.
    - The /definitions/message/part/@element attribute of the Retrieve Rendered Imaging Document Set
830 Response message shall be a “wado:RetrieveRenderedImagingDocumentSetResponse” as defined below.
832 - The /definitions/portType/operation/input/@wsaw:Action attribute for the Retrieve Rendered Imaging Document Set Request message shall be
834 “urn:dicom:ws:wado:2011:RetrieveRenderedImagingDocumentSet”.
    - The /definitions/portType/operation/output/@wsaw:Action attribute for the Retrieve Imaging
836 Document Set Response message shall be
“urn:dicom:ws:wado:2011:RetrieveRenderedImagingDocumentSetResponse”.
838 - The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be “urn:dicom:ws:wado:2011:RetrieveRenderedImagingDocumentSet”.

840 The <wado:RetrieveRenderedImagingDocumentSetRequest/> element for use with the Retrieve Imaging Document Set Request Message is defined as:

842 - One or more <wado:StudyRequest/> elements each of which includes a “studyInstanceUID” attribute identifying the study associated with the DICOM images/ objects being retrieved. Each
844 <iherad:StudyRequest/> element shall contain:
        - One or more <wado:SeriesRequest/> elements each of which includes a
846 “seriesInstanceUID” attribute identifying the series associated with the DICOM images/ objects being retrieved. Each <iherad:SeriesRequest/> element shall contain:
            - One or more <wado:RenderedDocumentRequest/> elements, each one representing an individual document that the requestor wants to retrieve from the
850 Web Server. Each <wado:DocumentRequest/> element contains:
                - An optional <ihe:RepositoryUniqueId/> element that identifies the Web
852 Server from which the document is to be retrieved. This value corresponds to XDSDocumentEntry.repositoryUniqueId.

854

                  The RepositoryUniqueID is similar to a DICOM AETitle, but is a uniqueID
856 assigned to the WADO-WS Web Server rather than a locally assigned

string identifier. There will be a separate RepositoryUniqueID for each web service end point.

- A required <ihe:DocumentUniqueId/> element that identifies the document within the source. This value corresponds to the SOP Instance UID referenced within the DICOM manifest.
- An optional <ihe:HomeCommunityId/> element that corresponds to the home attribute of the Identifiable class in ebRIM.
- An optional <wado:Annotation/> element.
- An optional <wado:Rows/> element.
- An optional <wado:Columns/> element.
- An optional <wado:Region/> element.
- An optional <wado:WindowCenter/> element.
- An optional <wado:WindowWidth/> element.
- An optional <wado:ImageQuality/> element.
- An optional <wado:PresentationUID/> element.
- An optional <wado:PresentationSeriesUID/> element.
- An optional <wado:Anonymize/> element
- An optional <wado:FrameNumber/> element.
- A required <wado:ContentTypeList/> element which contains a list of one or more <wado:ContentType> elements.
- An optional <wado:CharsetList/> element which contains a list of one or more <wado:Charset> elements.

6.4.2.2 Response

An Web Server shall render and then provide the document(s) indicated in the request. The Web Server shall return the rendered documents or an error code when the document could not be returned. The rendered forms shall be the subset specified, and in the format requested. If the Imaging Document Source cannot render the pixel data in that manner then an error status shall be returned.

The <wado:RetrieveRenderedImagingDocumentResponse/> element for use with the Retrieve Imaging Document Set Response Message, Retrieve Rendered Imaging Document Set Response Message and Retrieve Imaging Document Set Metadata Response Message is defined as:

- A required /ihe:RetrieveDocumentSetResponse/rs:RegistryResponse element
- An optional sequence of <wado:RenderedDocumentResponse/> elements containing:
    - A <ihe:HomeCommunityId/> element. The value of this element shall be the same as the value of the StudyRequest/SeriesRequest/DocumentRequest/HomeCommunityId element in the Request Message. If the <ihe:HomeCommunityId/> element is not present in the Request Message, this value shall not be present.
    - A required <ihe:RepositoryUniqueId/> that identifies the Imaging Document Source from which the document was retrieved. The value of this element shall be the same as the value of the StudyRequest/SeriesRequest/DocumentRequest/RepositoryUniqueId element in the original Request Message.
    - A required <wado:SourceDocumentUniqueId/> that identifies the source document. The value of this element shall be the same as the value of the StudyRequest/SeriesRequest/DocumentRequest/DocumentUniqueId element in the original Request Message. This value identifies the source, and is not an ID for the resulting rendered document.

902 - A conditional <wado:FrameNumber/> that identifies the frame within the source document. It shall be present if and only if <wado:FrameNumber/> was in the request.
904 - A required <wado:Annotation/> element that contains the actual value used.
- A required <wado:Rows/> element that contains the actual value used.
906 - A required <wado:Columns/> element that contains the actual value used.
- A required <wado:Region/> element that contains the actual value used.
908 - A required <wado:WindowCenter/> element that contains the actual value used.
- A required <wado:WindowWidth/> element that contains the actual value used.
910 - A required <wado:ImageQuality/> element that contains the actual value used.
- A required <wado:PresentationUID/> element that contains the actual value used if a
912 PresentationUID was used.
- A required <wado:PresentationSeriesUID/> element that contains the actual value used
914 if a PresentationSeriesUID was used.
- An optional <wado:Anonymize/> element that shall be present if the rendered instance
916 was anonymized.
- A required <ihe:Document/> element that contains the rendered document encoded as
918 an XOP Infoset.
- A required <ihe:mimeType/> element that indicates the MIME type of the retrieved
920 document.

The /RetrieveDocumentSetResponse/rs:RegistryResponse/@status attributes provides the overall status
922 of the request: It shall contain one of the following values:

urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Success
924 urn:ihe:iti:2007:ResponseStatusType:PartialSuccess
urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Failure
926

For each document requested in a
928 /RetrieveRenderedImagingDocumentSetRequest/StudyRequest/SeriesRequest/DocumentRequest element:

930 - If the document is successfully rendered (without warning) then no
/RetrieveRenderedImagingDocumentSetResponse/rs:RegistryResponse/rs:RegistryErrorList/
932 rs:RegistryError element shall be present and a
/RetrieveRenderedImagingDocumentSetResponse/DocumentResponse/Document element
934 shall be returned containing the rendered document as base64binary encoded data.
- If a warning is reported when retrieving the document, then a
936 /RetrieveRenderedImagingDocumentSetResponse/rs:RegistryResponse/rs:RegistryErrorList/
rs:RegistryError element shall be returned with:
938 - @severity is urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Warning
- @errorCode is specified
940 - @codeContext contains the warning message
- @location contains the DocumentUniqueId of the document requested
942 - The rendered document shall be returned in an instance of
/RetrieveRenderedImagingDocumentSetResponse/DocumentResponse/Document as
944 base64binary encoded data. The returned document and warning are correlated via the DocumentUniqueId.

- If an error is reported when retrieving a document, then a /RetrieveRenderedImagingDocumentSetResponse/rs:RegistryResponse/rs:RegistryErrorList/rs:RegistryError element shall be returned with:
  - @severity is urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Error
  - @errorCode is specified
  - @codeContext contains the error message
  - @location contains the DocumentUniqueId of the document requested
  - No corresponding RetrieveRenderedImagingDocumentSetResponse/DocumentResponse element shall be returned

The error conditions for failures and associated error codes are given below in section 6.4.4. These errors shall be detected and the associated errorCode returned if that error occurs. Additional errors defined in the ebRS standard, in ITI TF-2: 4.1.13 “Error Reporting”, and defined by the implementor may be returned.

6.4.3 WS – RetrieveImagingDocumentSetMetadataRequest

6.4.3.1 Request

The specific Web Services parameters to be used for the Retrieve Imaging Document Set Metadata action shall be as follows, in the order that they would appear in the WSDL definition:

- The following types shall be imported (xsd:import) in the /definitions/types section:
  - namespace="urn:ihe:rad:xdsi-b:2009", schema="XDSI.b_ImagingDocumentSource.xsd"
  - The baseline XDS.b schema (namespace="urn:ihe:iti:xds-b:2007", schema=”XDS.b_DocumentRepository.xsd”)
  - The baseline DICOM WADO-WS schema (namespace=”urn:dicom:wado:ws:yyyymmdd”, schema=”dicom.wado.ws.yyyymmdd.xsd”)
- The /definitions/message/part/@element attribute of the Retrieve Imaging Document Information Set Request message shall be defined an “wado:RetrieveImagingDocumentSetInformationRequest” as defined below.
- The /definitions/message/part/@element attribute of the Retrieve Imaging Document Set Information Response message shall be defined an “wado:RetrieveImagingDocumentSetInformationResponse” as defined below.
- The /definitions/portType/operation/input/@wsaw:Action attribute for the Retrieve Imaging Document Set Information Request message shall be “urn:wado:2011:RetrieveImagingDocumentSetInformation”.
- The /definitions/portType/operation/output/@wsaw:Action attribute for the Retrieve Imaging Document Set Information Response message shall be “urn:wado:2011:RetrieveImagingDocumentSetInformationResponse”.
- The /definitions/binding/operation/soap12:operation/@soapAction attribute shall be “urn:wado:2011:RetrieveImagingDocumentSetInformation”.

The <wado:RetrieveImagingDocumentSetInformationRequest/> element for use with the Retrieve Imaging Document Set Request Message is defined as:

- One or more <wado:StudyRequest/> elements each of which includes a “studyInstanceUID” attribute identifying the study associated with the DICOM images/ objects being retrieved. Each <iherad:StudyRequest/> element shall contain:

- One or more <wado:SeriesRequest/> elements each of which includes a "seriesInstanceUID" attribute identifying the series associated with the DICOM images/ objects being retrieved. Each <iherad:SeriesRequest/> element shall contain:
    - One or more <wado:DocumentInformationRequest/> elements, each one representing an individual document that the requestor wants to retrieve from the Web Server. Each < wado:DocumentInformationRequest /> element contains:
        - An required <ihe:RepositoryUniqueId/> element that identifies the Web Server from which the document is to be retrieved. This value corresponds to XDSDocumentEntry.repositoryUniqueId.

          The RepositoryUniqueID is similar to a DICOM AETitle, but is a uniqueID assigned to the WADO-WS Web Server rather than a locally assigned string identifier. There will be a separate RepositoryUniqueID for each web service end point.
        - A required <ihe:DocumentUniqueId/> element that identifies the document within the source. For example, this value could be a SOP Instance UID obtained from a Key Object Selection (KOS) instance.
        - An optional <ihe:HomeCommunityId/> element. See the IHE Profiles for the definition and possible uses of this element.
        - An optional <wado:Anonymize/> element
        - A required <wado:XPath/> that contains the text corresponding to the XPath "filter" applied to the Native DICOM Model transposition of the object, as defined in PS 3.19.

          Note: If the requested filter is "/", then all of the metadata is requested.

6.4.3.2 Response

An Web Server shall extract information from each document specified in a Document Set Information Request. This shall be done by the logical equivalent of:

1. convert the non-pixel data for each of the requested data into an XML encoded form
2. apply each of the wado:XPath elements to this XML encoded form
3. provide the XPath response as part of the Document Set Information Response.

See PS 3.19 for details on conversion to XML encoded form.

The Web Server shall return the XPath results or an error code when the document could not be processed.

The <wado:RetrieveImagingDocumentSetInformationResponse/> element for use with the Retrieve Imaging Document Set Response Message is additionally defined as:

- A required /wado:RetrieveImagingDocumentSetInformationResponse/rs:RegistryResponse element
- An optional sequence of <wado:DocumentInformationResponse/> elements containing:
    - A <ihe:HomeCommunityId/> element. The value of this element shall be the same as the value of the StudyRequest/SeriesRequest/DocumentRequest/HomeCommunityId element in the Request Message. If the <ihe:HomeCommunityId/> element is not present in the Request Message, this value shall not be present.
    - A required <ihe:DocumentUniqueId/> that identifies the document within the Web Server. The value of this element shall be the same as the value of the

StudyRequest/SeriesRequest/DocumentRequest/DocumentUniqueId element in the original Request Message. This value corresponds to the SOP Instance UID.

- A conditional <wado:FrameNumber/> that identifies the frame within the source document. It shall be present if and only if <wado:FrameNumber/> was in the request.
- One <wado:XPathResponseList/> containing:
  - A required <wado:XPathResponse> that contains the XPath results for each <wado:XPath/> elements, in the same order as in the request encoded as an XOP Infoset. The response element shall be empty if there was no XPath match.

The /RetrieveImagingDocumentSetInformationResponse/rs:RegistryResponse/@status attributes provides the overall status of the request: It shall contain one of the following values:

urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Success

urn:ihe:iti:2007:ResponseStatusType:PartialSuccess

urn:oasis:names:tc:ebxml-regrep:ResponseStatusType:Failure

For each document requested in a /RetrieveImagingDocumentSetInformationRequest/StudyRequest/SeriesRequest/DocumentRequest element:

- If the document is successfully retrieved (without warning) then no /RetrieveImagingDocumentSetInformationResponse/rs:RegistryResponse/rs:RegistryErrorList/rs:RegistryError element shall be present and a /RetrieveImagingDocumentSetInformationResponse/DocumentResponse/Document element shall be returned containing the document as base64binary encoded data.
- If a warning is reported when retrieving the document, then a /RetrieveImagingDocumentSetInformationResponse/rs:RegistryResponse/rs:RegistryErrorList/rs:RegistryError element shall be returned with:
  - @severity is urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Warning
  - @errorCode is specified
  - @codeContext contains the warning message
  - @location contains the DocumentUniqueId of the document requested
  - The document shall be returned in an instance of /RetrieveDocumentSetResponse/DocumentResponse/Document as base64binary encoded data. The returned document and warning are correlated via the DocumentUniqueId.
- If an error is reported when retrieving a document, then a /RetrieveImagingDocumentSetInformationResponse/rs:RegistryResponse/rs:RegistryErrorList/rs:RegistryError element shall be returned with:
  - @severity is urn:oasis:names:tc:ebxml-regrep:ErrorSeverityType:Error
  - @errorCode is specified
  - @codeContext contains the error message
  - @location contains the DocumentUniqueId of the document requested
  - No corresponding RetrieveDocumentSetResponse/DocumentResponse element shall be returned

The error conditions for failures and associated error codes are given below in section 6.4.4. These errors shall be detected and the associated errorCode returned if that error occurs. Additional errors defined in the ebRS standard, in ITI TF-2: 4.1.13 “Error Reporting”, and defined by the implementor may be returned.

1082 **6.4.4 Error Codes**

The following errorCodes are defined and shall be used to report any of the associated error and warning
1084 situations. Other errorCodes may be present for other error and warning situations.

ErrorCode	Error Situation
urn:dicom:wado:XXX1	Unable to anonymize the requested instance(s).
urn:dicom:wado:XXX2	Web Server does not support anonymization.
urn:dicom:wado:XXX3	The requested instance(s) are not immediately available, but can be made available by manual request.
urn:dicom:wado:XXX4	Instance is no longer available, e.g., document retention rules have caused it to be removed or relocated.
urn:dicom:wado:XXX5	The requested instance(s) cannot be returned because the size or count exceeds resource limits.
urn:dicom:wado:XXX6	Web Server does not support the requested format or transfer syntax.
urn:dicom:wado:XXX7	The requested instance(s) cannot be provided in the requested format or transfer syntax.
urn:dicom:wado:XXX8	Single image format is not available for multi-frame images.
urn:dicom:wado:XXX9	Identifier does not match SOP Class (See PS3.7 C-MOVE)
urn:dicom:wado:XX10	Inconsistent identifiers, e.g., Study and Series are correct but Series is in a different Study (See PS3.7 C-MOVE)
urn:dicom:wado:XX11	SOP Class not supported. (See PS3.7 C-MOVE)
urn:dicom:wado:XX12	Invalid parameter value in request (See PS3.7 C-MOVE)
urn:dicom:wado:XX13	Unsupported parameter in request (See PS3.7 C-MOVE)
urn:dicom:wado:XX14	Processing Failure (See PS3.7 C-MOVE)
urn:dicom:wado:XX15	Study Instance UID not known
urn:dicom:wado:XX16	Series Instance UID not known
urn:dicom:wado:XX17	Document UID not known
urn:dicom:wado:XX18	Out of range Frame number
urn:dicom:wado:XX19	Presentation UID not known
urn:dicom:wado:XX20	Presentation Series UID not known

1086 **Item #6: Modify PS 3.18 Section 7.2 Multi-Frame Image Objects as indicated.**

7.2 MULTI-FRAME AND VIDEO IMAGE OBJECTS

1088 **7.2.1 Objects included**

In this category are all SOP classes defined in PS 3.3 that are multi-frame **or video** image objects.

Item #7: Modify PS 3.18 Section 7.3.2 MIME type constraints as indicated.

— a "CDA" MIME type, in conformance to HL7 CDA **R2**, e.g. ~~application/x-hl7-cda-level-one+xml~~ **text/xml**.

Item #8: Modify PS 3.18 Section 8 Parameters as indicated.

8 Parameters of the request

8.1 PARAMETERS AVAILABLE FOR ALL DICOM PERSISTENT OBJECTS

Parameters specified in this section are applicable to all supported DICOM SOP Classes.

Note: To identify a DICOM Object, only one UID is required, because any UID is globally unique. However, the standard requires that the UID of the higher levels in the DICOM Information Model are specified (i.e., series and study), in order to support the use of DICOM devices that support only the baseline hierarchical (rather than extended relational) Query/Retrieve model, which requires the Study Instance UID and Series Instance UID to be defined when retrieving an SOP Instance, as defined in PS 3.4.

8.1.1 Request type

Type of request performed. This parameter is REQUIRED **for URI based mode.**

The parameter name shall be “requestType”.

The value shall be "WADO".

Note: This parameter allows other types of requests to be introduced in the future, using a similar syntax.

8.1.2 Unique identifier of the study

Study Instance UID as defined in PS 3.3. This parameter is REQUIRED.

The parameter name shall be “studyUID” **for URI based mode, and “StudyRequest” which contains a required “studyInstanceUID” attribute for the WS mode**.

The value shall be encoded as a Unique Identifier (UID) string, as specified in PS 3.5, except that it shall not be padded to an even length with a NULL character.

8.1.3 Unique identifier of the series

Series Instance UID as defined in the PS 3.3. This parameter is REQUIRED.

The parameter name shall be “seriesUID” **for URI based mode, and, for the WS mode, one or multiple “SeriesRequest” which is included into the above described “StudyRequest” and which contains a required “seriesInstanceUID” attribute**.

The value shall be encoded as a Unique Identifier (UID) string, as specified in PS 3.5, except that it shall not be padded to an even length with a NULL character.

8.1.4 Unique identifier of the object

SOP Instance UID as defined in the PS 3.3. This parameter is REQUIRED.

The parameter name shall be “objectUID” **for URI based mode, and for the WS mode one or multiple “DocumentRequest” which is included into the above described “SeriesRequest” and which include each one:**

- **a required “DocumentUniqueId” which contains the Instance UID,**
- **an optional “RepositoryUniqueId” which contains the UID of the DICOM server, and**
- **an optional “HomeCommunityId” which contains the UID of the “clinical affinity domain”**.

The value shall be encoded as a unique identifier (UID) string, as specified in PS 3.5, except that it shall not be padded to an even length with a NULL character.

8.1.5 MIME type of the response

MIME types desired by the Web Client for the response from the Server, as defined in the IETF RFC2616. This parameter is OPTIONAL **for URI based mode, it shall be present for the WS mode “Rendered Requester” and shall not be present in the other WS mode transactions**.

The parameter name shall be “contentType” **for URI based mode, and, for the WS mode, “ContentTypeList” which contains one or multiple “ContentType”**.

In URI based mode, t~~T~~he value shall be a list of MIME types, separated by a "," character, and potentially associated with relative degree of preference, as specified in IETF RFC2616. **In WS mode, it contains one or more “ContentType” elements containing each one MIME type.**

In URI based mode, t~~T~~he Web Client shall provide list of content types it supports in the "Accept" field of the Get method. The value of the ContentType parameter of the request shall be one of the values specified in that field.

Notes: 1. In URI based mode, typically the Accept field will be sent by a Web Client as “*/*”, which is compatible with any MIME types.

2. When this parameter is absent, the default content type of the response is dictated by the "MIME type constraints" sub-sections of Section 7 (i.e. 7.1.2, 7.2.2, 7.3.2, 7.4.2).

8.1.6 Charset of the response

Character set with which the returned object**s** is to be encoded, as defined in the IETF RFC2616. This parameter is OPTIONAL **for URI based mode, and for the WS mode “Rendered Requester” and shall not be present in the other WS mode transactions**.

The parameter name shall be “charset” **for URI based mode, and “CharsetList” containing one or more elements “Charset” for the WS mode**.

For the URI mode, the value shall be a list of character sets, separated by a "," character, and potentially associated with relative degree of preference, as specified in IETF RFC2616.

In URI based mode, t~~T~~he Web Client may provide a list of character sets it supports in the "Accept-charset" field of the Get method. If this field is present, the value of the charset parameter of the request shall be one of the values specified in it.

The **Web** Server may or may not support character set conversion. If character set conversion is supported:

— text based DICOM objects retrieved other than as application/dicom MIME type (e.g., text/plain) may be returned in the requested character set (converted if necessary)

— DICOM objects retrieved as application/dicom MIME type have all contained strings returned in the requested character set (converted if necessary) and the Specific Character Set (0008,0005) updated (if necessary)

Notes: 1. The IANA Character Set registrations specify names and multiple aliases for most character sets. The standard value for use in WADO is the one marked by IANA as "preferred for MIME." If IANA has not marked one of the aliases as "preferred for MIME", the name used in DICOM shall be the value used for WADO.

2. The table in Annex D provides an informative mapping of some IANA values to DICOM Specific Character Set Defined Terms.

8.1.7 Anonymize object**s**

Removal of all patient identification information from within the DICOM object**s**, if not already done, as defined in PS 3.15. This parameter is OPTIONAL. **In the URI based mode,** it shall only be present if contentType is application/dicom.

This parameter is Optional.

The parameter name shall be “anonymize” **for URI based mode, and “Anonymize” for the WS mode**.

The value shall be “yes”.

The Server may return an error if it either cannot or refuses to anonymize ~~**that**~~ **these** object**s**.

In WS mode, the metadata describing the objects or information extracted from them in the response shall be anonymized if requested.

The Server shall return a new SOP Instance UID if the content of the object**s** has not already been anonymized.

Notes: 1. This standard does not introduce any security-related requirements. It is likely that the information contained within DICOM objects identifies the patient. The protocol used (that is HTTP) can be replaced by HTTPs, which is its secure extension, to protect the information in transit. The underlying DICOM implementation decides whether or not to grant access to a particular DICOM object based on whatever security policy or mechanism it has in place. A server is unlikely to **fulfill** a request from an unknown user (e.g., accessed via the HTTP protocol) unless it is certain that the data requested has no patient identifying information within it and has been approved for public viewing.

2. The Anonymize object enables, for example, teaching files systems or clinical trial applications to offer an access to original images stored in a PACS, without disclosing the patients identity, and requiring storage of a (de-identified) copy of the original image**s**. Anonymization is the responsibility of the Server. In order to preserve patient confidentiality, the Server likely will refuse to deliver an anonymized SOP instance to an unknown or unauthorized person unless the Server is certain that the SOP instance holds no patient identifying information. This would include "blanking out" any annotation area(s) containing nominative information burned into the pixels or in the overlays.

8.1.9 Retrieve partial information from objects

Retrieval of additional information from the DICOM objects, using a filtering mechanism based on the XML mapping of DICOM IODs, as described in the Native DICOM Model defined in PS 3.19. This parameter is defined only for the WS mode “Information Requester” transaction.

The parameter name shall be “XPath”.

8.2 PARAMETERS FOR DICOM IMAGE PERSISTENT OBJECTS

These parameters shall only be included when a request is made for a Single Frame Image Object**s** or Multi-Frame Image **or video** Object**s** as defined in Section 7.2.

8.2.1 Annotation on the object

Annotation of ~~an~~ object**s** retrieved and displayed as an image. This parameter is OPTIONAL **for the URI based mode and the WS mode "Rendered Requester" transaction.** It shall not be present if contentType is application/dicom, or is a non-image MIME type (e.g., text/*). When it is not present for ~~an~~ image object**s**, no **additional** annotation may be burnt in.

When used in conjunction with a presentation state object, it shall be applied after the presentation on the image**s**. When used in conjunction with the region parameter, it shall be applied after the selection of the region.

The parameter name shall be "annotation" **for URI based mode, and "Annotation" for the WS mode**. Its value is a non-empty list of one or more of the following items, separated by a "," character:

- "patient", for displaying patient information on the image**s** (e.g. patient name, birth date,…)
- "technique", for displaying technique information of the image**s** (e.g. image number, study date, image position,…).

Note: The exact nature and presentation of the annotation is determined by the Server. The annotation is burned into the returned image pixels.

8.2.2 Number of pixel rows

The parameter name shall be "rows" **for URI based mode, and "Rows" for the WS mode**.

The value shall be expressed as an integer, representing the image height to be returned. It is OPTIONAL **for the URI based mode and the WS mode "Rendered Requester" transaction. It shall not be present for other WS mode transactions**. It shall not be present if contentType is application/dicom.

If both "rows" and "columns" are specified, then each shall be interpreted as a maximum, and a size will be chosen for the image**s** within these constraints, maintaining the correct aspect ratio. If the number of rows is absent and the number of columns is present, the number of rows shall be chosen in order to maintain the correct aspect ratio. If both are absent, the image**s** (or selected region) ~~is~~ **are** sent in ~~its~~ **their** original size (or the size of the presentation state applied on the image**s**), resulting as one pixel of screen image for each value in the image**s** data matrix.

The value shall be encoded as an integer string (IS), as specified in PS 3.5.

8.2.3 Number of pixel columns

The parameter name shall be "columns" **for URI based mode, and "Columns" for the WS mode**.

The value shall be expressed as an integer, representing the image width to be returned. It is OPTIONAL **for the URI based mode and the WS mode "Rendered Requester" transaction.** It shall not be present if contentType is application/dicom.

If both "rows" and "columns" are specified, then each shall be interpreted as a maximum, and a size will be chosen for the image**s** within these constraints, maintaining the correct aspect ratio. If the number of columns is absent and the number of rows is present, the number of columns shall be chosen in order to maintain the correct aspect ratio. If both are absent, the image**s** (or selected region) ~~is~~ **are** sent in ~~its~~ **their** original size (or the size of the presentation state applied on the image**s**), resulting as one pixel of screen for one pixel of the image**s**.

The value shall be encoded as an integer string (IS), as specified in PS 3.5.

8.2.4 Region of the image

This parameter allows selection of a rectangular region of an image matrix to be retrieved. The purpose of this parameter is to allow a user to view a selected area of the image matrix, for example at higher magnification.

The parameter is OPTIONAL **for the URI based mode and the WS mode "Rendered Requester" transaction. It shall not be present for other WS mode transactions**.

The parameter name shall be "region" **for URI based mode, and "Region" for the WS mode**.

It shall not be present if contentType is application/dicom.

The value shall be expressed as a list of four positive decimal strings, separated by the ',' character, representing the region of the source image**s** to be returned. These decimal values shall be values in a normalized coordinate system relative to the size of the original image matrix measured in rows and columns, with values ranging from 0.0 to 1.0, and representing in the following order:

- the x position of the top left hand corner of the region to be retrieved, 0.0 corresponding to the first column of the image matrix. **In the WS mode, this value is encoded into an XML element "XMin".**

- the y position of the top left hand corner of the region to be retrieved, 0.0 corresponding to the top row of the image matrix. **In the WS mode, this value is encoded into an XML element "YMin".**

- the x position of the bottom right hand extent of the region, 1.0 corresponding to the last column of the image matrix, 0.0 being forbidden. **In the WS mode, this value is encoded into an XML element "XMax".**

- the y position of the bottom right hand extent of the region, 1.0 corresponding to the last row of the image matrix, 0.0 being forbidden. **In the WS mode, this value is encoded into an XML element "YMax".**

Note: The Server may or may not support this parameter.

If this parameter is supported, an image matrix corresponding to the specified region shall be returned with size corresponding to the specified normalized coordinate values otherwise the complete image matrix shall be returned. If the presentationUID parameter is present, the region shall be selected after the corresponding presentation state has been applied on the image**s**.

8.2.5 Window center of the image

The parameter name shall be "windowCenter" **for URI based mode, and "WindowCenter" for the WS mode**.

Controls the luminosity of the image**s** as defined in PS 3.3. This parameter is OPTIONAL for **the URI based mode and the WS mode "Rendered Requester" transaction**. It shall not be present for other WS mode transactions. It is REQUIRED if "windowWidth" is present. This parameter shall not be present if there is a presentationUID parameter. It shall not be present if contentType is application/dicom.

The value shall be encoded as a decimal string (DS), as specified in PS 3.5.

8.2.6 Window width of the image

The parameter name shall be “windowWidth” **for URI based mode, and “WindowWidth” for the WS mode**.

Controls the contrast of the image**s** as defined in PS 3.3. This parameter is OPTIONAL **for the URI based mode and the WS mode “Rendered Requester” transaction. It shall not be present for other WS mode transactions**. It is REQUIRED if "windowCenter" is present. This parameter shall not be present if there is a presentationUID parameter. It shall not be present if contentType is application/dicom.

The value shall be encoded as a decimal string (DS), as specified in PS 3.5.

8.2.7 Frame Number

The parameter name shall be “frameNumber" **for URI based mode, and “FrameNumber” for the WS mode**.

Specifies that the single frame with that number within a multi-frame image object, as defined in PS 3.3, shall be returned. It is OPTIONAL and shall be ignored in the case of all objects other than multi-frame objects.

The value shall be encoded as an integer string (IS), as specified in PS 3.5.

8.2.8 Image Quality

The parameter name shall be “imageQuality" **for URI based mode, and “ImageQuality” for the WS mode**. It is OPTIONAL **for the URI based mode and the WS mode “DICOM requester” and “Rendered Requester” transactions.** It shall not be present if contentType is application/dicom, except if the transferSyntax parameter is present and corresponds to a lossy compression.

If the requested MIME type is for a lossy compressed image (e.g. image/jpeg), this parameter indicates the required quality of the image to be returned within the range 1 to 100, 100 being the best quality.

Note: Decompression and recompression may degrade the image quality if the original image was already irreversibly compressed. In case the image has been already lossy compressed using the same format as required (e.g. jpeg), it may be sent as it is without decompressing and recompressing it.

The value shall be encoded as an integer string (IS), as specified in PS 3.5.

Note: The specific interpretation of the meaning of this parameter is left to the interpretation of the implementers of the standard.

8.2.9 Unique identifier of the presentation object

The parameter name shall be “presentationUID" **for URI based mode, and “PresentationUID” for the WS mode**.

SOP Instance UID of the presentation state storage object to be applied to the image**s**. This parameter is OPTIONAL **for the URI based mode and the WS mode “Rendered Requester” transaction.** It shall not be present if contentType is application/dicom.

The value shall be encoded as a unique identifier (UID) string, as specified in PS 3.5, except that it shall not be padded to an even length with a NULL character.

If this parameter is combined with region and/or annotation parameters, the presentation state shall be applied to the image**s** prior to selecting a region and burning in annotations.

If the Presentation Size Mode in the presentation state is SCALE TO FIT or TRUE SIZE, then the displayed area specified in the presentation shall be scaled to fit the size specified by the rows and columns parameters if present, otherwise the displayed area selected in the presentation state will be returned without scaling.

Notes: 1. The intent of the TRUE SIZE mode in the presentation state cannot be satisfied, since the physical size of the pixels displayed by the web browser is unlikely to be known. If the Presentation Size Mode in the presentation state is MAGNIFY, then the displayed area specified in the presentation shall be magnified (scaled) as specified in the presentation state. It will then be cropped to fit the size specified by the rows and columns parameters, if present.

2. Any Displayed Area relative annotations specified in the presentation state are rendered relative to the Specified Displayed Area within the presentation state, not the size of the returned image**s**.

Though the output of the presentation state is defined in DICOM to be in P-Values (grayscale values intended for display on a device calibrated to the DICOM Grayscale Standard Display Function PS 3.14), the grayscale or color space for the image**s** returned by the request is not defined by this standard.

8.2.10 Unique identifier of the series containing the presentation object

The parameter name shall be “presentationSeriesUID" **for URI based mode, and “PresentationSeriesUID” for the WS mode**.

Series Instance UID of the series containing the presentation state storage object to be applied on the image**s**. This parameter is REQUIRED and shall only be present if "presentationUID" is present.

The value shall be encoded as a unique identifier (UID) string, as specified in PS 3.5, except that it shall not be padded to an even length with a NULL character.

Note: As specified in DICOM, the Presentation State will be in the same study as the image**s** it applies to.

8.2.11 Transfer Syntax UID

The parameter name shall be “transferSyntax" **for URI based mode, and “TransferSyntaxUIDList” containing one or more “TransferSyntaxUID” elements for the WS mode**.

The Transfer Syntax(es) to be used within the DICOM image object**s**, as specified in PS 3.6. This parameter is OPTIONAL **for the URI based mode and the WS mode “DICOM Requester” transaction.** It shall not be present if contentType is other than application/dicom.

By default the DICOM objects returned shall be encoded in Explicit VR Little Endian. Neither Implicit VR, nor Big Endian shall be used. The response shall be the Transfer Syntax requested if possible. If it is not possible for the response to be sent using the requested transfer syntax then the Explicit VR Little Endian Uncompressed Transfer Syntax shall be used.

Note: The transfer syntax can be chosen as one of the values of TransferSyntaxUID corresponding to JPIP, in case of which the returned objects will contain the URL of the JPIP session to launch for retrieving the corresponding image.

The value shall be encoded as a unique identifier (UID) string, as specified in PS 3.5, except that it shall not be padded to an even length with a NULL character.

Item #10: Append PS 3.18 by the following annex.

Annex E – WADO WS Schemas and Examples

E.1 WADO WS XSD SCHEMA (INFORMATIVE)

The following XSD schema can be used for the WADO WS implementation:

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns="urn:ihe:rad:xdsi-b:2009" xmlns:xs="http://www.w3.org/2001/XMLSchema"
       targetNamespace="urn:ihe:rad:xdsi-b:2009" elementFormDefault="qualified"
       attributeFormDefault="unqualified" xmlns:tns="urn:oasis:names:tc:ebxml-
regrep:xsd:rs:3.0">
       <xs:import namespace="urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0"/>
       <xs:import namespace="urn:ihe:iti:xds-b:2007"/>
       <xs:simpleType name="LongName">
         <xs:restriction base="xs:string">
           <xs:maxLength value="256"/>
         </xs:restriction>
       </xs:simpleType>
       <xs:complexType name="RetrieveDocumentSetRequestType">
         <xs:sequence>
           <xs:element name="DocumentRequest" maxOccurs="unbounded">
             <xs:complexType>
               <xs:sequence>
                 <xs:element name="HomeCommunityId" type="LongName" minOccurs="0">
                   <xs:annotation>
                     <xs:documentation>This corresponds to the home attribute of the
Identifiable class in regrep RIM (regrep-rim-3.0-os.pdf, page 20)</xs:documentation>
                   </xs:annotation>
                 </xs:element>
                 <xs:element name="RepositoryUniqueId" type="LongName" minOccurs="0">
                   <xs:annotation>
                     <xs:documentation>This is the XDSDocumentEntry.repositoryUniqueId
attribute in the XDS metadata</xs:documentation>
                   </xs:annotation>
                 </xs:element>
                 <xs:element name="DocumentUniqueId" type="LongName">
                   <xs:annotation>
                     <xs:documentation>This is the XDSDocumentEntry.uniqueId attribute
in the XDS metadata</xs:documentation>
                   </xs:annotation>
                 </xs:element>
               </xs:sequence>
             </xs:complexType>
           </xs:element>
         </xs:sequence>
       </xs:complexType>
       <xs:complexType name="RegistryErrorType">
         <xs:simpleContent>
           <xs:extension base="xs:string">
             <xs:attribute name="codeContext" type="xs:string" use="required"/>
             <xs:attribute name="errorCode" type="xs:string" use="required"/>
             <xs:attribute default="urn:oasis:names:tc:ebxml-
regrep:ErrorSeverityType:Error"
               name="severity" type="xs:anyURI"/>
             <xs:attribute name="location" type="xs:string" use="optional"/>
           </xs:extension>
         </xs:simpleContent>
       </xs:complexType>
       <xs:complexType name="RegistryErrorListType">
         <xs:annotation>
           <xs:documentation xml:lang="en">The RegistryErrorList is derived from the
ErrorList element from the ebXML Message Service Specification</xs:documentation>
         </xs:annotation>
         <xs:sequence>
           <xs:element maxOccurs="unbounded" name="RegistryError"
type="RegistryErrorType"/>
         </xs:sequence>
```

```
1432          <xs:attribute name="highestSeverity" type="xs:anyURI" use="optional"/>
        </xs:complexType>
1434    <xs:complexType name="RegistryResponseType">
          <xs:annotation>
1436        <xs:documentation xml:lang="en">Base type for all ebXML Registry
responses</xs:documentation>
1438      </xs:annotation>
          <xs:sequence>
1440        <!-- every response may be extended using Slots. -->
            <xs:element minOccurs="0" type="RegistryErrorListType"
1442 name="RegistryErrorList"/>
          </xs:sequence>
1444      <xs:attribute name="status" type="xs:anyURI" use="required"/>
          <!-- id is the request if for the request for which this is a response -->
1446      <xs:attribute name="requestId" type="xs:anyURI" use="optional"/>
        </xs:complexType>
1448    <xs:element name="RetrieveDocumentSetRequest"
type="RetrieveDocumentSetRequestType"/>
1450    <xs:complexType name="RetrieveDocumentSetResponseType">
          <xs:sequence>
1452        <xs:element name="RegistryResponse" type="RegistryResponseType"/>
            <xs:sequence minOccurs="0">
1454          <xs:element maxOccurs="unbounded" name="DocumentResponse">
                <xs:complexType>
1456              <xs:sequence>
                    <xs:element minOccurs="0" name="HomeCommunityId" type="LongName">
1458                  <xs:annotation>
                        <xs:documentation>This corresponds to the home attribute of the
1460 Identifiable class in regrep RIM (regrep-rim-3.0-os.pdf, page 20)</xs:documentation>
                      </xs:annotation>
1462                </xs:element>
                    <xs:element name="RepositoryUniqueId" type="LongName">
1464                  <xs:annotation>
                        <xs:documentation>This is the
1466 XDSDocumentEntry.repositoryUniqueId attribute in the XDS metadata</xs:documentation>
                      </xs:annotation>
1468                </xs:element>
                    <xs:element name="DocumentUniqueId" type="LongName">
1470                  <xs:annotation>
                        <xs:documentation>This is the XDSDocumentEntry.uniqueId
1472 attribute in the XDS metadata</xs:documentation>
                      </xs:annotation>
1474                </xs:element>
                    <xs:element minOccurs="0" name="Document" type="xs:base64Binary"/>
1476              </xs:sequence>
                </xs:complexType>
1478          </xs:element>
            </xs:sequence>
1480      </xs:sequence>
        </xs:complexType>
1482    <xs:element name="RetrieveDocumentSetResponse"
type="RetrieveDocumentSetResponseType"/>
1484    <xs:complexType name="RetrieveRenderedImagingDocumentSetRequestType">
          <xs:sequence>
1486        <xs:element maxOccurs="unbounded" name="StudyRequest">
              <xs:complexType>
1488            <xs:sequence>
                  <xs:element maxOccurs="unbounded" name="SeriesRequest">
1490                <xs:complexType>
                      <xs:sequence>
1492                    <xs:element maxOccurs="unbounded"
name="RenderedDocumentRequest">
1494                      <xs:complexType>
                            <xs:sequence>
1496                          <xs:element name="HomeCommunityId" type="LongName"
                              minOccurs="0">
1498                          <xs:annotation>
```

```
                                <xs:documentation>This corresponds to the home attribute
of the Identifiable class in regrep RIM (regrep-rim-3.0-os.pdf, page
20)</xs:documentation>
                                </xs:annotation>
                                </xs:element>
                                <xs:element name="RepositoryUniqueId"
                                type="LongName" minOccurs="0">
                                <xs:annotation>
                                <xs:documentation>This is the
XDSDocumentEntry.repositoryUniqueId attribute in the XDS metadata</xs:documentation>
                                </xs:annotation>
                                </xs:element>
                                <xs:element name="DocumentUniqueId" type="LongName">
                                <xs:annotation>
                                <xs:documentation>This is the XDSDocumentEntry.uniqueId
attribute in the XDS metadata</xs:documentation>
                                </xs:annotation>
                                </xs:element>
                                <xs:element minOccurs="0" name="Annotation"
                                type="xs:string"/>
                                <xs:element minOccurs="0" name="Rows"
                                type="xs:string"/>
                                <xs:element minOccurs="0" name="Columns"
                                type="xs:string"/>
                                <xs:element minOccurs="0" name="Region"
                                type="xs:string"/>
                                <xs:element minOccurs="0" name="WindowWidth"
                                type="xs:string"/>
                                <xs:element minOccurs="0" name="WindowCenter"
                                type="xs:string"/>
                                <xs:element minOccurs="0" name="ImageQuality"
                                type="xs:string"/>
                                <xs:element minOccurs="0"
                                name="PresentationSeriesUID" type="xs:string"/>
                                <xs:element minOccurs="0" name="Anonymize"
                                type="xs:string"/>
                                <xs:element minOccurs="0" name="FrameNumber"
                                type="xs:string"/>
                                <xs:element minOccurs="1" name="ContentTypeList"
                                type="xs:string" maxOccurs="unbounded"/>
                                <xs:element minOccurs="1" name="CharsetList"
                                type="xs:string" maxOccurs="unbounded"/>
                            </xs:sequence>
                          </xs:complexType>
                        </xs:element>
                      </xs:sequence>
                    </xs:complexType>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
        <xs:element name="RetrieveRenderedImagingDocumentSetRequest"
          type="RetrieveRenderedImagingDocumentSetRequestType"/>
        <xs:complexType name="RetrieveRenderedImagingDocumentSetResponseType">
          <xs:sequence>
            <xs:element name="RegistryResponse" type="RegistryResponseType"/>
            <xs:element maxOccurs="unbounded" name="RenderedDocumentSetResponse"
minOccurs="0">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="HomeCommunityId" type="LongName" minOccurs="0">
                    <xs:annotation>
                      <xs:documentation>This corresponds to the home attribute of the
Identifiable class in regrep RIM (regrep-rim-3.0-os.pdf, page 20)</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element name="RepositoryUniqueId" type="LongName" minOccurs="1">
```

```
                    <xs:annotation>
                      <xs:documentation>This is the XDSDocumentEntry.repositoryUniqueId
attribute in the XDS metadata</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element name="SourceDocumentUniqueId" type="LongName">
                    <xs:annotation>
                      <xs:documentation>This is the XDSDocumentEntry.uniqueId attribute
in the XDS metadata</xs:documentation>
                    </xs:annotation>
                  </xs:element>
                  <xs:element minOccurs="1" name="Annotation" type="xs:string"/>
                  <xs:element minOccurs="1" name="Rows" type="xs:string"/>
                  <xs:element minOccurs="1" name="Columns" type="xs:string"/>
                  <xs:element minOccurs="1" name="Region" type="xs:string"/>
                  <xs:element name="WindowWidth" type="xs:string"/>
                  <xs:element minOccurs="1" name="WindowCenter" type="xs:string"/>
                  <xs:element minOccurs="1" name="ImageQuality" type="xs:string"/>
                  <xs:element minOccurs="1" name="PresentationSeriesUID"
type="xs:string"/>
                  <xs:element minOccurs="0" name="Anonymize" type="xs:string"/>
                  <xs:element minOccurs="0" name="FrameNumber" type="xs:string"/>
                  <xs:element minOccurs="1" name="mimeType" type="xs:string"
maxOccurs="1"/>
                  <xs:element minOccurs="1" name="Document" type="xs:base64Binary"/>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
        <xs:element name="RetrieveRenderedImagingDocumentSetResponse"
          type="RetrieveRenderedImagingDocumentSetResponseType"/>
        <xs:complexType name="RetrieveImagingDocumentSetInformationRequestType">
          <xs:sequence>
            <xs:element maxOccurs="unbounded" name="StudyRequest">
              <xs:complexType>
                <xs:sequence>
                  <xs:element maxOccurs="unbounded" name="SeriesRequest">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element maxOccurs="unbounded"
                          name="DocumentInformationRequest">
                          <xs:complexType>
                            <xs:sequence>
                              <xs:element name="HomeCommunityId" type="LongName"
                              minOccurs="0">
                              <xs:annotation>
                              <xs:documentation>This corresponds to the home attribute
of the Identifiable class in regrep RIM (regrep-rim-3.0-os.pdf, page
20)</xs:documentation>
                              </xs:annotation>
                              </xs:element>
                              <xs:element name="RepositoryUniqueId"
                              type="LongName" minOccurs="1">
                              <xs:annotation>
                              <xs:documentation>This is the
XDSDocumentEntry.repositoryUniqueId attribute in the XDS metadata</xs:documentation>
                              </xs:annotation>
                              </xs:element>
                              <xs:element name="DocumentUniqueId" type="LongName">
                              <xs:annotation>
                              <xs:documentation>This is the XDSDocumentEntry.uniqueId
attribute in the XDS metadata</xs:documentation>
                              </xs:annotation>
                              </xs:element>
                              <xs:element minOccurs="0" name="Anonymize"
                              type="xs:string"/>
                              <xs:element minOccurs="1" name="XPath"
                              type="xs:string" maxOccurs="1"/>
```

```
                              </xs:sequence>
                            </xs:complexType>
                          </xs:element>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
          <xs:element name="RetrieveImagingDocumentSetInformationRequest"
            type="RetrieveImagingDocumentSetInformationRequestType"/>
          <xs:complexType name="RetrieveImagingDocumentSetInformationResponseType">
            <xs:sequence>
              <xs:element name="RegistryResponse" type="RegistryResponseType"/>
              <xs:element maxOccurs="unbounded" name="DocumentInformationResponse"
minOccurs="0">
                <xs:complexType>
                  <xs:sequence>
                    <xs:element name="HomeCommunityId" type="LongName" minOccurs="0">
                      <xs:annotation>
                        <xs:documentation>This corresponds to the home attribute of the
Identifiable class in regrep RIM (regrep-rim-3.0-os.pdf, page 20)</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element name="RepositoryUniqueId" type="LongName" minOccurs="1">
                      <xs:annotation>
                        <xs:documentation>This is the XDSDocumentEntry.repositoryUniqueId
attribute in the XDS metadata</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element name="DocumentUniqueId" type="LongName">
                      <xs:annotation>
                        <xs:documentation>This is the XDSDocumentEntry.uniqueId attribute
in the XDS metadata</xs:documentation>
                      </xs:annotation>
                    </xs:element>
                    <xs:element minOccurs="0" name="FrameNumber" type="xs:string"/>
                    <xs:element minOccurs="1" name="XPathResponseList">
                      <xs:complexType>
                        <xs:sequence>
                          <xs:element maxOccurs="unbounded" name="XPathResponse"
                            type="xs:string"/>
                        </xs:sequence>
                      </xs:complexType>
                    </xs:element>
                  </xs:sequence>
                </xs:complexType>
              </xs:element>
            </xs:sequence>
          </xs:complexType>
          <xs:element name="RetrieveImagingDocumentSetInformationResponse"
            type="RetrieveImagingDocumentSetInformationResponseType"/>
</xs:schema>
```

E.4 WADO WS REQUEST EXAMPLE (informative)

Example of requesting the retrieval of images from a series in JPEG resized to 300 pixels max with associated information on modality and instance number. Note that the multipart related MTOM/XOP encoding is used in the request as well as response as required by the Web Services standards.

```
POST /tf6/services/xdsrepositoryb HTTP/1.1
Content-Type: multipart/related;
boundary=MIMEBoundaryurn_uuid_DCD262C64C22DB973512563039513 23;
type="application/xop+xml";
start="<0.urn:uuid:DCD262C64C22DB97351256303951324@apache.org>"; start-
info="application/soap+xml";
action="urn:dicom:ws:wado:2011:RetrieveRenderedImagingDocumentSet"
```

```
User-Agent: Axis2
Host: localhost:5000

--MIMEBoundaryurn_uuid_DCD262C64C22DB97351256303951323
Content-Type: application/xop+xml; charset=UTF-8; type="application/soap+xml"
Content-Transfer-Encoding: binary
Content-ID: <0.urn:uuid:DCD262C64C22DB97351256303951324@apache.org>
<s:Envelope
         xmlns:s="http://www.w3.org/2003/05/soap-envelope"
         xmlns:a="http://www.w3.org/2005/08/addressing">
       <s:Header>
         <a:Action
s:mustUnderstand="1">urn:dicom:ws:wado:2011:RetrieveRenderedImagingDocumentSet
</a:Action>
         <a:MessageID>urn:uuid:0fbfdced-6c01-4d09-a110-2201afedaa02</a:MessageID>
         <a:ReplyTo s:mustUnderstand="1">
           <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
         </a:ReplyTo>
         <a:To >http://localhost:2647/XdsService/DocSource.svc</a:To>
       </s:Header>
       <s:Body>
         <RetrieveImagingDocumentSetRequest xmlns:iherad="urn:ihe:rad:xdsi-b:2009"
xmlns:ihe="urn:ihe:iti:xds-b:2007">
           <StudyRequest studyInstanceUID="1.3.6.1.4...101">
               <SeriesRequest seriesInstanceUID="1.3.6.1.4...201">
                   <ihe:DocumentRequest>

       <ihe:RepositoryUniqueId>1.3.6.1.4...1000</ihe:RepositoryUniqueId>
                       <ihe:DocumentUniqueId>1.3.6.1.4...2300</ihe:DocumentUniqueId>
                       <Rows>300</Rows>
                       <Columns>300</Columns>
                       <ContentTypeList>
                         <ContentType>image/jpeg</ContentType>
                       </ContentTypeList>
                   </ihe:DocumentRequest>
                   <ihe:DocumentRequest>
       <ihe:RepositoryUniqueId>1.3.6.1.4...1000</ihe:RepositoryUniqueId>
                       <ihe:DocumentUniqueId>1.3.6.1.4...2301</ihe:DocumentUniqueId>
                       <Rows>300</Rows>
                       <Columns>300</Columns>
                       <ContentTypeList>
                         <ContentType>image/jpeg</ContentType>
                       </ContentTypeList>
                   </ihe:DocumentRequest>
               </SeriesRequest>
           </StudyRequest>
         </RetrieveRenderedImagingDocumentSetRequest>
       </s:Body>
</s:Envelope>

--MIMEBoundaryurn_uuid_DCD262C64C22DB97351256303951323—
```

E.5 WADO WS RESPONSE EXAMPLE

Example of the response corresponding to the above request:

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: multipart/related;
boundary=MIMEBoundaryurn_uuid_F862C3E04D9E35266C1256303956115;
type="application/xop+xml";
start="0.urn:uuid:F862C3E04D9E35266C1256303956116@apache.org"; start-
info="application/soap+xml"; action="urn:ihe:iti:2007:RetrieveDocumentSetResponse"
Date: Fri, 23 Oct 2009 13:19:11 GMT

```

```
--MIMEBoundaryurn_uuid_F862C3E04D9E35266C1256303956115
Content-Type: application/xop+xml; charset=UTF-8; type="application/soap+xml"
Content-Transfer-Encoding: binary
Content-ID: <0.urn:uuid:F862C3E04D9E35266C1256303956116@apache.org>

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing">
      <s:Header>
        <a:Action
s:mustUnderstand="1">urn:ihe:iti:2007:RetrieveRenderedImagingDocumentSetResponse</a:Act
ion>
        <a:RelatesTo>urn:uuid:0fbfdced-6c01-4d09-a110-2201afedaa02</a:RelatesTo>
      </s:Header>
      <s:Body>
        <RetrieveDocumentSetResponse
            xmlns="urn:ihe:iti:xds-b:2007"
            xmlns:lcm="urn:oasis:names:tc:ebxml-regrep:xsd:lcm:3.0"
            xmlns:query="urn:oasis:names:tc:ebxml-regrep:xsd:query:3.0"
            xmlns:rim="urn:oasis:names:tc:ebxml-regrep:xsd:rim:3.0"
            xmlns:rs="urn:oasis:names:tc:ebxml-regrep:xsd:rs:3.0">
          <rs:RegistryResponse status="urn:oasis:names:tc:ebxml-
regrep:ResponseStatusType:Success"/>
          <DocumentResponse>
            <ihe:RepositoryUniqueId>1.3.6.1.4...1000</ihe:RepositoryUniqueId>
            <SourceDocumentUniqueId>1.3.6.1.4...2300</SourceDocumentUniqueId>
            <Annotation>patient</Annotation>
            <Rows>300</Rows>
            <Columns>300</Columns>
            <Region>
              <Xmin>0.0</Xmin>
              <Ymin>0.0</Ymin>
              <Xmax>1.0</Xmax>
              <Ymax>1.0<Ymax>
            </Region>
            <WindowCenter>2000</WindowCenter>
            <WindowWidth>4096</WindowWidth>
            <ImageQuality>30</ImageQuality>
            <mimeType>image/jpeg</mimeType>
            <Document>
<xop:Include href="cid:1.urn:uuid:F862C3E04D9E35266C1256303956117@apache.org"
xmlns:xop="http://www.w3.org/2004/08/xop/include"/>
            </Document>
          </DocumentResponse>
          <DocumentResponse>
            <RepositoryUniqueId>1.3.6.1.4...1000</RepositoryUniqueId>
            <DocumentUniqueId>1.3.6.1.4...2301</DocumentUniqueId>
            <Annotation>patient</Annotation>
            <Rows>300</Rows>
            <Columns>250</Columns>
            <Region>
              <Xmin>0.0</Xmin>
              <Ymin>0.0</Ymin>
              <Xmax>1.0</Xmax>
              <Ymax>1.0<Ymax>
            </Region>
            <WindowCenter>2000</WindowCenter>
            <WindowWidth>4096</WindowWidth>
            <ImageQuality>30</ImageQuality>
            <mimeType>image/jpeg</mimeType>
            <Document>
<xop:Include href="cid:2.urn:uuid:F862C3E04D9E35266C1256303956117@apache.org"
xmlns:xop="http://www.w3.org/2004/08/xop/include"/>
            </Document>
          </DocumentResponse>
        </RetrieveDocumentSetResponse>
      </s:Body>
</s:Envelope>

--MIMEBoundaryurn_uuid_F862C3E04D9E35266C1256303956115
```

```
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary
Content-ID: <1.urn:uuid:F862C3E04D9E35266C1256303956117@apache.org>

This is the binary JPEG payload for the first image.

--MIMEBoundaryurn_uuid_F862C3E04D9E35266C1256303956115
Content-Type: application/octet-stream
Content-Transfer-Encoding: binary
Content-ID: <2.urn:uuid:F862C3E04D9E35266C1256303956117@apache.org>

This is the binary JPEG payload for the second image.

--MIMEBoundaryurn_uuid_F862C3E04D9E35266C1256303956115—

```