

2

4

6

Digital Imaging and Communications in Medicine (DICOM)

8

Supplement 161: Web Access to DICOM Persistent Objects by RESTful Services (WADO-RS)

10

12

14

16

18

20 *Prepared by:*

DICOM Standards Committee, Working Group 27 Web Technology

22 1300 N. 17th Street, Suite 1752

Rosslyn, Virginia 22209 USA

24

26

VERSION: Final Text, 6 February 2013

28 Developed in accordance with: DICOM Workitem 2008-04-B

Table of Contents

30	Table of Contents	ii
	Scope and Field of Application	1
32	Changes to NEMA Standards Publication PS 3.2-2011	2
4	SYMBOLS AND ABBREVIATIONS	2
34	A.4.2.x.y WADO RS Specifications	2
	I.4 NETWORKING	4
36	I.4.1 IMPLEMENTATION MODEL.....	4
	I.4.1.1 Application Data Flow.....	4
38	I.4.1.2 Functional Definition of AEs.....	5
	I.4.1.2.1 Functional Definition of WADO Service Application	5
40	I.4.2.3 WADO RS Specifications	5
	I.4.2.3.1 WADO RS Retrieve Study.....	5
42	I.4.2.3.2 WADO RS Retrieve Series.....	5
	I.4.2.3.3 WADO RS Retrieve Instance	6
44	I.4.2.3.4 WADO RS Retrieve Frames.....	6
	I.4.2.3.5 WADO RS Retrieve Bulk Data	6
46	I.4.2.3.6 WADO RS Retrieve Metadata	7
	I.4.2.3.7 Connection Policies	7
48	I.4.2.3.7.1 General	7
	I.4.2.3.7.2 Number of Connections.....	7
50	I.4.2.3.7.3 Asynchronous Nature	7
	I.4.4.3 RS Interface	8
52	I.7 SECURITY	8
	Changes to NEMA Standards Publication PS 3.17-2011	9
54	HHH.1 REQUEST AND RESPONSE PARAMETERS.....	9
	HHH.1.1 Request Parameters.....	9
56	HHH.1.2 Response parameters	9
	HHH.1.2.1 URI WADO	9
58	HHH.1.2.2 WADO-WS	9
	HHH.1.2.3 WADO-RS	9
60	HHH.2 WEB <u>AND REST SERVICES</u> IMPLEMENTATION.....	10
	HHH.3 USES FOR WADO WEB <u>AND REST SERVICES</u>	10
62	HHH.3.1 General requirements	10
	HHH.3.2 Analysis of use cases.....	10
64	HHH.3.3.5 DICOM Requestor	11
	HHH.3.3.6 Frame Pixel Data Requestor.....	12
66	HHH.3.3.7 Bulk Data Requestor	12
	HHH.3.3.8 Metadata Requestor	12
68	Changes to NEMA Standards Publication PS 3.18-2011	13
3	Normative references	13
70	5 Symbols and abbreviated terms	13
6	6 Data Communication Requirements	13
72	6.1 INTERACTION.....	13
	6.5 RS REQUEST/RESPONSE	14
74	6.5.1 RS – RetrieveStudy	17
	6.5.1.1 Request.....	17
76	6.5.1.2 Response.....	18

	6.5.1.2.1	DICOM Response	18
78	6.5.1.2.2	Bulk Data Response	18
	6.5.2	RS – RetrieveSeries.....	19
80	6.5.2.1	Request	19
	6.5.2.2	Response	20
82	6.5.2.2.1	DICOM Response	20
	6.5.2.2.2	Bulk Data Response	20
84	6.5.3	RS – RetrieveInstance	21
	6.5.3.1	Request	21
86	6.5.3.2	Response	22
	6.5.3.2.1	DICOM Response	22
88	6.5.3.2.2	Bulk Data Response	23
	6.5.4	RS – RetrieveFrames.....	23
90	6.5.4.1	Request	23
	6.5.4.2	Response	24
92	6.5.4.2.1	Pixel Data Response.....	24
	6.5.5	RS – RetrieveBulkdata.....	25
94	6.5.5.1	Request	25
	6.5.5.2	Response	26
96	6.5.5.2.1	Bulk Data Response	26
	6.5.6	RS – RetrieveMetadata.....	26
98	6.5.6.1	Request	27
	6.5.6.2	Response	27
100	6.5.6.2.1	Metadata Response	27
	6.5.7	Error Codes.....	28
102	7	Persistent Object types.....	28
	7.1	SINGLE FRAME IMAGE OBJECTS.....	28
104	7.1.1	Objects accessed.....	28
	7.1.2	MIME type constraints.....	28
106	7.2	MULTI-FRAME AND VIDEO IMAGE OBJECTS.....	29
	7.2.1	Objects included.....	29
108	7.2.2	MIME type constraints.....	29
	Changes to NEMA Standards Publication PS 3.19-2011.....		31
110	A.1	NATIVE DICOM MODEL.....	31
	A.1.4	Information Model	31
112	A.1.5	Description	33
		Schema.....	34
114			

Scope and Field of Application

116 This Supplement defines Representational State Transfer (REST) Services for providing DICOM images
and other persistent objects to an Electronic Medical Record/Electronic Health Record (EMR/EHR)
118 system.

This supplement deals with retrieval, corresponding to the evolution of the existing WADO to RESTful
120 Services. Native DICOM can be retrieved, as well as separate bulk data, pixel data, or metadata of the
object.

122 Query and notification mechanisms are not defined within this supplement.

Security is beyond the scope of the services defined in this supplement. However there are accepted
124 industry guidelines on the use of generic Web security mechanisms with protected healthcare information
(see DICOM PS 3.15). Several security programming recipes are provided for reference.

126 The healthcare world has need for Web APIs to provide access to image management systems from point
of service (POS) systems. Both Web Services (W3C WS*) and RESTful Web Services are major means
128 for accessing distributed mixed media systems on the World Wide Web and the DICOM community would
benefit by defining standard approaches for both models.

130

Changes to NEMA Standards Publication PS 3.2-2011

132 Digital Imaging and Communications in Medicine (DICOM)

Part 2: Conformance

134

Add to PS 3.2 Section 4 Symbols and Abbreviations

136 4 SYMBOLS AND ABBREVIATIONS

The following symbols and abbreviations are used in this Part.

138 **REST** **Representational State Transfer**

140 **Append to PS 3.12 A.4.x.y "Application Entity <1>"**

A.4.2.x.y WADO RS Specifications

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

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

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

148 **Update PS 3.2 Annex I to replace EXAMPLE-WADO-SERVER and EXAMPLE-INTEGRATED-**
150 **MODALITY with EXAMPLE-WADO-SERVICE in sections : I.4.2.2.4.1, I.4.2.2.4.2, I.4.3.1, I.4.3.2, I.4.4.1,**
I.4.4.2 <twice>, I.6 <twice>, I.8.4., and I.4.2.2.4.3.

152 **Update PS 3.2 Annex I.1 Conformance Statement Overview as indicated.**

I.1 CONFORMANCE STATEMENT OVERVIEW

154 This fictional product EXAMPLE-WADO-SERVICE implements ~~both~~ the WADO URI services, ~~and~~ the
156 WADO **WS** services, ~~and the WADO RS services~~ for access to DICOM SOP Instances that are stored on
an EXAMPLE-PACS-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

158 EXAMPLE-PACS-ARCHIVE. This conformance claim refers to the conformance claim for the EXAMPLE-PACS-ARCHIVE for all such services.

160 Table I.1-1 provides an overview of the network services supported by EXAMPLE-~~INTEGRATED-MODALITY~~WADO-SERVICE.

162

**Table I.1-1
 NETWORK SERVICES**

Network Service	User of Service (Client)	Provider of Service (Server)
Web Access to Dicom Objects (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
<u>WADO – RS – Retrieve Study</u>	<u>No</u>	<u>Yes</u>
<u>WADO – RS – Retrieve Series</u>	<u>No</u>	<u>Yes</u>
<u>WADO – RS – Retrieve Instance</u>	<u>No</u>	<u>Yes</u>
<u>WADO – RS – Retrieve Frames</u>	<u>No</u>	<u>Yes</u>
<u>WADO – RS – Retrieve Bulkdata</u>	<u>No</u>	<u>Yes</u>
<u>WADO – RS – Retrieve Metadata</u>	<u>No</u>	<u>Yes</u>

164

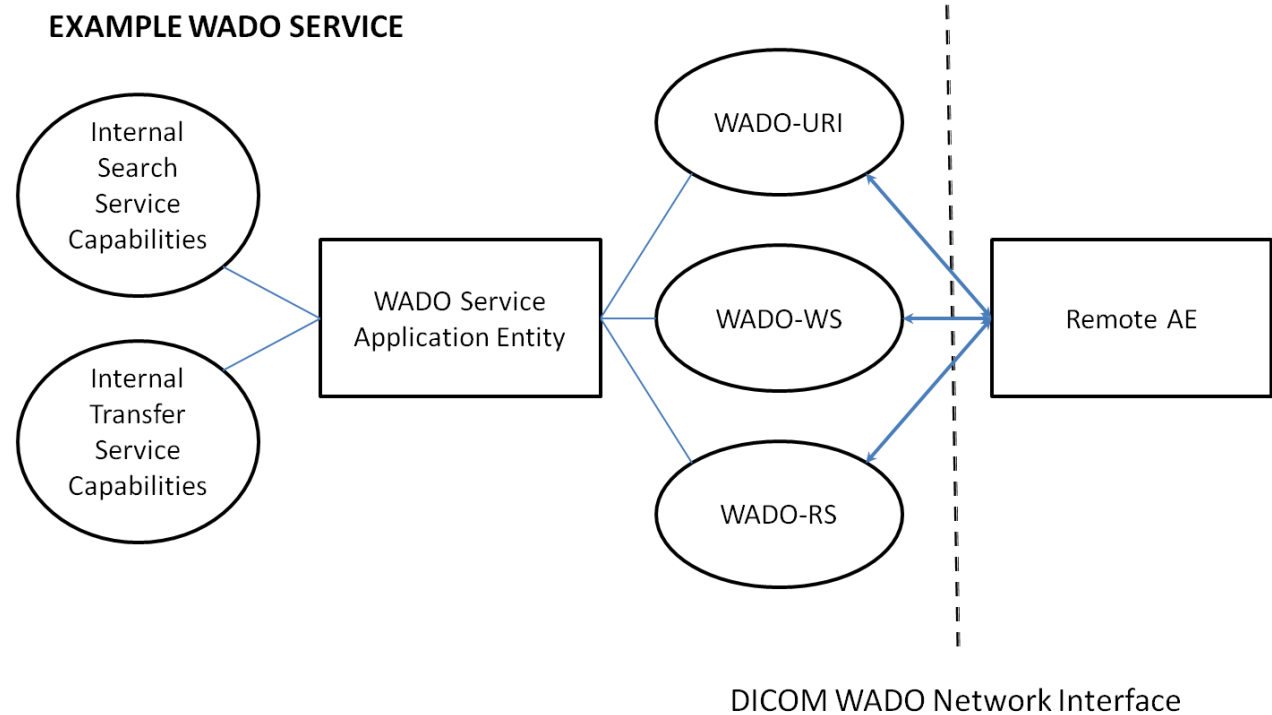
166 **Replace PS 3.2 Section I.4.1 Application Data Flow diagram with the one below**

I.4 NETWORKING

168 I.4.1 IMPLEMENTATION MODEL

I.4.1.1 Application Data Flow

170



172

**Figure I.4.1-1
APPLICATION DATA FLOW DIAGRAM**

174 **Update PS 3.2 Section I.4.1.1 as written below**

176 The WADO Service Application receives WADO requests from a remote AE. These requests may be
177 either over the URI, or WS, or RS interfaces. It is associated with the local real-world activity "Retrieve
178 Images". 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.

180 **Update PS 3.2 Section 1.4.1.2 Functional Definition of AEs as indicated.**

182 **I.4.1.2 Functional Definition of AEs**

I.4.1.2.1 Functional Definition of WADO Service Application

184 The reception of a WADO request will activate the AE. An internal request is sent to the search capabilities
 of the EXAMPLE-~~PACS-ARCHIVE~~**WADO-SERVICE**. This request is based upon the request parameters,
 186 **or the URL resource endpoint** from the WADO 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
 188 instances, the AE will indicate this in the WADO 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
 190 instances, these instances will be returned. If the request was for retrieval of rendered instances, then the
 AE will render each instance and return the rendered results.

192

Update PS 3.2 Annex I.4.2 AE Specifications as indicated.

194 **I.4.2 AE SPECIFICATIONS**

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

196

Append I.4.2.3 to PS 3.2 Annex I.4.2 AE Specifications

198 **I.4.2.3 WADO RS Specifications**

I.4.2.3.1 WADO RS Retrieve Study

200

**Table I.4.2.3-1
WADO RS RETRIEVE STUDY**

Options	Restrictions
Data Types Supported (Accept Type)	Restricted to application/dicom or application/octet-stream
Transfer Syntaxes Supported (transfer-syntax Accept parameter)	Any Transfer Syntax 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 size supported by the hosting EXAMPLE-PACS-ARCHIVE

202

I.4.2.3.2 WADO RS Retrieve Series

204

**Table I.4.2.3-2
WADO RS RETRIEVE SERIES**

Options	Restrictions
Data Types Supported (Accept Type)	Restricted to application/dicom or application/octet-stream
Transfer Syntaxes Supported	Any Transfer Syntax supported by the hosting EXAMPLE-PACS-ARCHIVE

Options	Restrictions
(Transfer-syntax Accept parameter)	
SOP Class Restrictions	Restricted to SOP classes supported by the hosting EXAMPLE-PACS-ARCHIVE
Size Restriction	Restricted to size supported by the hosting EXAMPLE-PACS-ARCHIVE

206

I.4.2.3.3 WADO RS Retrieve Instance

208

**Table I.4.2.3-3
WADO RS RETRIEVE INSTANCE**

Options	Restrictions
Data Types Supported (Accept Type)	Restricted to application/dicom or application/octet-stream
Transfer Syntaxes Supported (Transfer-syntax Accept parameter)	Any Transfer Syntax 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 size supported by the hosting EXAMPLE-PACS-ARCHIVE

210

I.4.2.3.4 WADO RS Retrieve Frames

212

**Table I.4.2.3-4
WADO RS RETRIEVE FRAMES**

Options	Restrictions
Data Types Supported (Accept Type)	Restricted to application/octet-stream
Transfer Syntaxes Supported (Transfer-syntax Accept parameter)	Any Transfer Syntax supported by the hosting EXAMPLE-PACS-ARCHIVE
SOP Class Restrictions	Restricted to Multi-Frame Image Objects as defined in PS 3.3.
Size Restriction	Restricted to size supported by the hosting EXAMPLE-PACS-ARCHIVE

214

I.4.2.3.5 WADO RS Retrieve Bulk Data

216

**Table I.4.2.3-5
WADO RS RETRIEVE BULK DATA**

Options	Restrictions
Data Types Supported	Restricted to application/octet-stream

Options	Restrictions
(Accept Type)	
Transfer Syntaxes Supported (Transfer-syntax Accept parameter)	Any Transfer Syntax 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 size supported by the hosting EXAMPLE-PACS-ARCHIVE

218

I.4.2.3.6 WADO RS Retrieve Metadata

220

**Table I.4.2.3-6
WADO RS RETRIEVE METADATA**

Options	Restrictions
Data Types Supported (Accept Type)	Restricted to application/dicom+xml
Accept-Encoding	Restricted to gzip, deflate, or identity (the use of no transformation whatsoever). See W3C RFC 2616 Protocol Parameters Section 3.5 for more information (http://www.w3.org/Protocols/rfc2616/rfc2616-sec3.html).
SOP Class Restrictions	Restricted to SOP classes supported by the hosting EXAMPLE-PACS-ARCHIVE
Size Restriction	Restricted to size supported by the hosting EXAMPLE-PACS-ARCHIVE

222

I.4.2.3.7 Connection Policies

224 I.4.2.3.7.1 General

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

226 I.4.2.3.7.2 Number of Connections

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

230

**Table I.4.2.3-7
NUMBER OF RS REQUESTS SUPPORTED**

Maximum number of simultaneous RS requests	100 (configurable)
--------------------------------------------	--------------------

232

I.4.2.3.7.3 Asynchronous Nature

234 EXAMPLE-WADO-SERVICE does not support RS asynchronous response.

236 **Append to PS 3.2 Annex I.4.4 Configuration**

I.4.4.3 RS Interface

238 The EXAMPLE-WADO-SERVICE 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
240 recognized as authenticated by a known authority.

242 **Update PS 3.2 Annex I.7 Security as indicated.**

I.7 SECURITY

244 EXAMPLE-~~INTEGRATED-MODALITY~~WADO-SERVICE supports transport level security measures for
URI **and RS** access, and the WS-Security services for WS access.

246 **The EXAMPLE-WADO-SERVICE supports the following transport level security measures:**

— **HTTP BASIC Authorization over SSL**

248 — **Digest Authorization**

— **SSL Client Certificates**

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

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

258 **The EXAMPLE-WADO-SERVICE can optionally be configured to support the following session
authentication mechanisms:**

260 — **Kerberos Local Domain Sessions**

— **Shibboleth Cross Domain Sessions (using SAML2.0)**

262

Changes to NEMA Standards Publication PS 3.17-2011

264

Digital Imaging and Communications in Medicine (DICOM)

Part 17: Explanatory Information

266 Update PS 3.17 Annex HHH as indicated.

Annex HHH - Evolution of WADO to Web and REST Services (Informative)

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

270 HHH.1 REQUEST AND RESPONSE PARAMETERS

HHH.1.1 Request Parameters

272 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
274 either native DICOM objects or a rendered object (JPEG, PDF etc.). ~~These are summarized as below:~~

276 The WADO-RS request has no parameters because data is requested through well defined URLs and content negotiation through HTTP headers.

The WADO-WS request parameters are summarized as below:

278 Update PS 3.17 HHH.1.2 Response parameters as indicated.

HHH.1.2 Response parameters

280 HHH.1.2.1 URI WADO

282 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.

HHH.1.2.2 WADO-WS

284 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
286 (<http://www.w3.org/TR/soap12-mtom>) mechanism as well as associated metadata.

288 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.

290 HHH.1.2.3 WADO-RS

292 The WADO-RS Service is a transport service that provides resources to enable machine to machine transfers of DICOM, frame pixel data, bulk data, and metadata.

In the REST Services implementation:

- 294 • For the “DICOM Requester”, one or more multipart/related items are returned containing the DICOM instances of a Study or Series, or an individual DICOM SOP Instance.

- 296 • For the “Frame Pixel Data Requester”, one or more multipart/related items are returned containing the pixel data of a multi-frame SOP Instance.
- 298 • For the “Bulk Data Requester”, one or more multipart/related items are returned containing the bulk data of a Study, Series or SOP Instance.
- 300 • For the “Metadata Requester”, an item is returned containing the XML encoded metadata selected from the retrieved objects header as described in the Native DICOM Model defined in PS3.19.

304 **Update PS 3.17 HHH.2 Web Services Implementation as indicated**

306 **HHH.2 WEB AND REST SERVICES IMPLEMENTATION**

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

The Web **and REST** Services technologies have been selected to:

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

314 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).

318 The **WADO-WS** response will be provided as a 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”)

320 The WADO-RS response will be provided as a list of XML and/or binary objects in a multipart/related response. The type of response depends on the Accept type in the HTTP Header.

322

Update PS 3.17 HHH.3 Uses for Web Services as indicated

324

HHH.3 USES FOR WADO WEB AND REST SERVICES

326 **HHH.3.1 General requirements**

328 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.

330 **HHH.3.2 Analysis of use cases**

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

- 332 1. Providing access to images and reports from a point-of-service application e.g., EMR.

- 334 2. Following references to significant images used to create an imaging report and displaying those images.
- 336 3. Following references / links to relevant images and imaging reports in email correspondence or clinical reports e.g., clinical summary.
- 338 4. Providing access to anonymized DICOM images and reports for clinical research and teaching purposes.
- 340 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.
- 342 6. Providing access to summary or selected information from DICOM objects.
- 344 7. **Providing access to complete studies for caching, viewing, or image processing.**

Examples of the use cases described in 1 above are:

- 346 a. The EMR displays in JPEG one image with annotations on it (patient and/or technique related), based upon information provided in a report.
- 348 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).
- 350 c. The EMR displays in JPEG one image per series with information describing every series (e.g. series description).
- 352 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).
- 354 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).
- 356 f. **The EMR displays patient demographics and image slices in a browser by accessing studies through URLs that are cached and rendered in a remote data center.**
- 358
- 360

Append below HHH3.3.4 Metadata (XML without pixel data, waveform data, etc) Requestor

362

HHH.3.3.5 DICOM Requestor

- 364 A. The requesting system is an application capable of making HTTP Service requests and able to process data encoded as a DICOM File, per DICOM PS 3.10 encodings.
- 366 B. Requesting information for DICOM Instances may come from a wide variety of forms. It is expected to include at least the Study UID. This may be encoded as part of an HL7 reference within a CDA document, a DICOM SOP Instance reference, or other formats.
- 368 C. The request specifies
 - 370 1. Requested Dataset
 - a) Study UID
 - 372 2. Optionally, it may also specify subset information
 - a) Series UID
 - 374 b) SOP Instance UID
 - D. The response provides
 - 376 1. SOP Instances, encoded per DICOM PS 3.10.

378 **HHH.3.3.6 Frame Pixel Data Requestor**

- 380 A. The requesting system is an application capable of making HTTP requests and able to process pixel data.
- 382 B. Requesting information for pixel data may come in a wide variety of forms. It is expected to include at least the Study UID, Series UID, Individual SOP Instance, and Frame List information. This may be encoded as part of an HL7 reference within a CDA document, a DICOM SOP Instance reference, or other formats.
- 384 C. The request specifies
 - 386 1. Requested Dataset
 - 388 a) Study UID
 - 390 b) Series UID
 - 392 c) SOP Instance UID
 - 394 d) Frame List comprised of one or more frame numbers
- 396 D. The response provides pixel data

392 **HHH.3.3.7 Bulk Data Requestor**

- 394 A. The requesting system is an application capable of making HTTP requests and able to process bulk data.
- 396 B. Requesting information for bulk data may come in a wide variety of forms. It is expected to include the Bulk Data URL as provided by the RetrieveMetatda resource. This may be encoded as part of an HL7 reference within a CDA document, a DICOM SOP Instance reference, or other formats.
- 398 C. The request specifies
 - 400 1. Requested Dataset
 - 402 a) Bulk Data URL
- 404 D. The response provides bulk data

404 **HHH.3.3.8 Metadata Requestor**

- 406 A. The requesting system is an application capable of making HTTP requests and able to process data encoded as a XML, per DICOM PS 3.19 encodings.
- 408 B. The Study UID may be obtained as part of an HL7 reference within a CDA document, a DICOM SOP Instance reference, or other formats.
- 410 C. Request information
 - 412 1. Requested Dataset
 - a) Study UID
- D. The response provides full study metadata encoded in XML, encoded per DICOM PS 3.19.

414 **Changes to NEMA Standards Publication PS 3.18-2011**

Digital Imaging and Communications in Medicine (DICOM)

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

418 **Append to PS 3.18 Section 3 Normative references**

3 Normative references

420 IETF RFC822 Standard for ARPA Internet Text Messages

422 **Append to PS 3.18 Section 5 Symbols and abbreviated terms**

5 Symbols and abbreviated terms

424 **REST** Representational State Transfer

426 **Update PS 3.18 Section 6.1 INTERACTION as indicated.**

6 Data Communication Requirements

428 **6.1 INTERACTION**

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

430 ~~Two~~**Multiple** communications modes are possible:

— URI based mechanism using HTTP Get: WADO Type request

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

a. DICOM Requester (Retrieve Imaging Document Set)

434 b. Rendered Requester (Retrieve Rendered Imaging Document Set)

c. Metadata Requester (Retrieve Imaging Document Set Metadata)

- 436 — **RESTful Services (RS) using HTTP Get: WADO RS, either:**
- 438 a. **DICOM Requester (Retrieve Study, Series, or Instance DICOM Objects)**
 - 438 b. **Frame Pixel Data Requester (Retrieve Instance Frame Pixel Data)**
 - 440 c. **Bulk Data Requester (Retrieve Study, Series, Instance Bulk Data)**
 - 440 d. **Metadata Requester (Retrieve Study Metadata)**

442 **Append to PS 3.18 Section 6.5 RS REQUEST/RESPONSE**

6.5 RS REQUEST/RESPONSE

444 The DICOM RESTful Service defines several action types. An implementation shall support all the following six action types:

446 1. RetrieveStudy

448 This action retrieves the set of DICOM instances associated with a given study unique identifier (UID). The response can be DICOM or bulk data depending on the “Accept” type, and is encapsulated in a multipart MIME response.

450

452 2. RetrieveSeries

454 This action retrieves the set of DICOM instances associated with a given study and series UID. The response can be DICOM or bulk data depending on the “Accept” type, and is encapsulated in a multipart MIME response.

456 3. RetrieveInstance

458 This action retrieves the DICOM instance associated with the given study, series, and SOP Instance UID. The response can be DICOM or bulk data depending on the “Accept” type, and is encapsulated in a multipart MIME response.

460

462 4. RetrieveFrames

464 This action retrieves the DICOM frames for a given study, series, SOP Instance UID, and frame numbers. The response is pixel data, and encapsulated in a multipart MIME response.

464

466 5. RetrieveBulkdata

468 This action retrieves the bulk data for a given bulk data URL. The response is a single bulk data item.

468

470 6. RetrieveMetadata

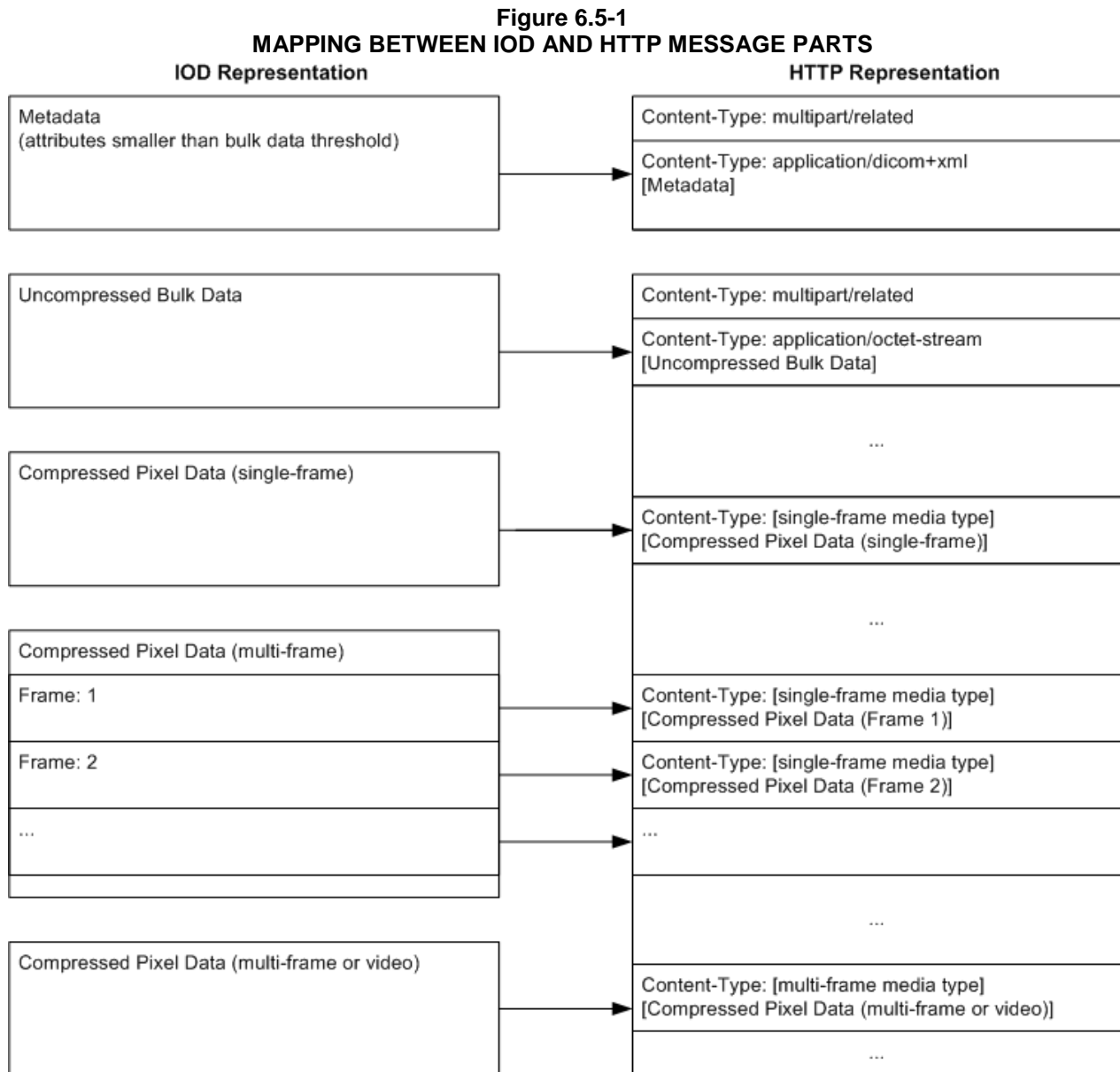
472 This action retrieves the DICOM instances presented as the full study metadata with the bulk data removed. The response is XML encoded metadata for the DICOM attributes as defined in PS 3.19.

474 All responses will be http multipart messages.

476 DICOM objects returned shall be PS3.10 binary objects encoded in a requested Transfer Syntax (Explicit
 VR Little Endian by default) with one message part per DICOM Instance.

478

480



482

Other types of responses will be encoded in the following manner: (see Figure 6.5-1).

484

486

488

490

- All XML responses shall be encoded as described in the Native DICOM Model defined in PS3.19 with one message part per XML object.
- Uncompressed bulk and pixel data shall be encoded in a Little Endian format using the application/octet-stream media type with one message part per bulk data item.
- Compressed pixel data may be encoded in one of three ways:
 - Single-frame pixel data encoded using a single-frame media type (one message part)

- 492 ○ Multi-frame pixel data encoded using a single-frame media type (one frame per message part)
- 494 ○ Multi-frame or video pixel data encoded using a multi-frame media type (multiple frames in one message part)

496 Compressed pixel data shall be encoded using the following Media Types. Media Types corresponding to several DICOM Transfer Syntax UIDs require a transfer-syntax parameter, as shown in Table 6.5-1, to disambiguate the request.

498 Note: If the Transfer Syntax is not specified, then a reversible (lossless) encoding is used.

500

502 **Table 6.5-1
MEDIA TYPE MAPPING TO TRANSFER SYNTAX UID**

DICOM Transfer Syntax UID	Media Type and Parameters
Single-frame media types	
1.2.840.10008.1.2.4.50	image/dicom+jpeg; transfer-syntax=1.2.840.10008.1.2.4.50
1.2.840.10008.1.2.4.51	image/dicom+jpeg; transfer-syntax=1.2.840.10008.1.2.4.51
1.2.840.10008.1.2.4.57	image/dicom+jpeg; transfer-syntax=1.2.840.10008.1.2.4.57
1.2.840.10008.1.2.4.70	image/dicom+jpeg
1.2.840.10008.1.2.4.70	image/dicom+jpeg; transfer-syntax=1.2.840.10008.1.2.4.70
1.2.840.10008.1.2.5	image/dicom+rle
1.2.840.10008.1.2.5	image/dicom+rle; transfer-syntax=1.2.840.10008.1.2.5
1.2.840.10008.1.2.4.80	image/dicom+jpeg-ls
1.2.840.10008.1.2.4.80	image/dicom+jpeg-ls; transfer-syntax=1.2.840.10008.1.2.4.80
1.2.840.10008.1.2.4.81	image/dicom+jpeg-ls; transfer-syntax=1.2.840.10008.1.2.4.81
1.2.840.10008.1.2.4.90	image/dicom+jp2
1.2.840.10008.1.2.4.90	image/dicom+jp2; transfer-syntax=1.2.840.10008.1.2.4.90
1.2.840.10008.1.2.4.91	image/dicom+jp2; transfer-syntax=1.2.840.10008.1.2.4.91
1.2.840.10008.1.2.4.92	image/dicom+jpx
1.2.840.10008.1.2.4.92	image/dicom+jpx; transfer-syntax=1.2.840.10008.1.2.4.92
1.2.840.10008.1.2.4.93	image/dicom+jpx; transfer-syntax=1.2.840.10008.1.2.4.93
Multi-frame media types	
1.2.840.10008.1.2.4.92	image/dicom+jpx
1.2.840.10008.1.2.4.92	image/dicom+jpx; transfer-syntax=1.2.840.10008.1.2.4.92
1.2.840.10008.1.2.4.93	image/dicom+jpx; transfer-syntax=1.2.840.10008.1.2.4.93
1.2.840.10008.1.2.4.100	video/mpeg; transfer-syntax=1.2.840.10008.1.2.4.100
1.2.840.10008.1.2.4.101	video/mpeg; transfer-syntax=1.2.840.10008.1.2.4.101
1.2.840.10008.1.2.4.102	video/mp4; transfer-syntax=1.2.840.10008.1.2.4.102
1.2.840.10008.1.2.4.103	video/mp4; transfer-syntax=1.2.840.10008.1.2.4.103

504 Note: For the media type image/dicom+jp2 Transfer Syntaxes, 1.2.840.10008.1.2.4.90 and 1.2.840.10008.1.2.4.91, the image does not include the jp2 wrapper.

506

508 HTTP Request field Accept is used in the header lines by the client in a HTTP protocol transaction to
indicate the data responses that are acceptable from the server. HTTP Response fields Content-Type and
510 parameters are used in the header lines by the server in a HTTP protocol transaction to indicate the type
and encoding of data returning to the client. All lines are RFC822 format headers. All HTTP header fields
whose use is not defined by WADO-RS are presumed to have the meaning defined by the HTTP standard.
512 The server is required to support uncompressed bulk and pixel data (application/octet-stream) and must
be able to deliver all bulk data in that form unless it is available only in a lossy-compressed format.

514

6.5.1 RS – RetrieveStudy

516 This action retrieves the set of DICOM instances associated with a given study unique identifier (UID). The
response can be DICOM or bulk data depending on the “Accept” type, and is encapsulated in a multipart
518 MIME response.

6.5.1.1 Request

520 The specific Services resource to be used for the RetrieveStudy action shall be as follows:

- Resource
 - 522 ○ {SERVICE}/studies/{StudyInstanceUID}, where
 - 524 ▪ {SERVICE} is the base URL for the service. This may be a combination of
protocol (either http or https), host, port, and application.
 - {StudyInstanceUID} is the study instance UID for a single study.
- 526 • Method
 - GET
- 528 • Headers
 - 530 ○ Accept – A comma-separated list of representation schemes, in preference order,
which will be accepted by the service in the response to this request. The types
allowed for this request header are as follows:
 - 532 ▪ multipart/related; type=application/dicom; [transfer-
534 syntax={TransferSyntaxUID}]
536 Specifies that the response can be DICOM Instances encoded in PS3.10
format. If *transfer-syntax* is not specified the server can freely choose which
Transfer Syntax to use for each Instance.
 - 538 ▪ multipart/related; type=application/octet-stream
540 Specifies that the response can be Little Endian uncompressed bulk data.
 - 542 ▪ multipart/related; type={MediaType}
544 Specifies that the response can be pixel data encoded using a {MediaType}
listed in Table 6.5-1 (including parameters).

Note: An example of a more complicated accept header with multiple transfer syntaxes:

546 User is interested in receiving JPEG2000 pixel data in lossless or compressed format but is willing to accept
JPEG as well. The Accept request would contain the following comma-separated parameters:

548

Accept: multipart/related=image/dicom+jpx; transfer-syntax=1.2.840.10008.1.2.4.92,, multipart/related=image/dicom+jpx;
550 transfer-syntax=1.2.840.10008.1.2.4.93, multipart/related=image/dicom+jpeg

552

or alternatively, multiple Accept headers:

554

Accept: multipart/related=image/dicom+jpx; transfer-syntax=1.2.840.10008.1.2.4.92,

Accept: multipart/related=image/dicom+jpx; transfer-syntax=1.2.840.10008.1.2.4.93

556

Accept: multipart/related=application/dicom+jpeg

558 **6.5.1.2 Response**

The Server shall provide the document(s) indicated in the request. In order to parse the bulk data items it
560 is necessary to also retrieve the XML metadata for the Study.

The Server shall return the document(s), or an error code when the document(s) cannot be returned. If the
562 server cannot convert all of the data to any of the requested media types/Transfer Syntaxes, then an error
code shall be returned, either a "Not Acceptable" response if no data is returned or a "Partial Content"
564 response if only some data is returned.

The client can compare the SOP Instance UIDs or bulk data URLs in the metadata and the message
566 response to determine which bulk data elements have been returned.

All response formats have a content type of multipart/related with a message boundary separator. The
568 response format depends on the Accept header specified in the request.

6.5.1.2.1 DICOM Response

570

- Content-Type:

- multipart/related; type=application/dicom; boundary={MessageBoundary}

572

- The entire multipart response contains every instance for the specified Study that can be converted to one of the requested Transfer Syntaxes.

574

- Each item in the multipart response represents a DICOM SOP Instance with the following http headers:

576

- Content-Type: application/dicom

578 **6.5.1.2.2 Bulk Data Response**

- Content-Type:

580

- multipart/related; type=application/octet-stream; boundary={MessageBoundary}

- multipart/related; type={MediaType}; boundary={MessageBoundary}

582

- The entire multipart response contains all bulk data for the specified Study that can be converted to one of the requested media types.

584

- Each item in the response is one of:

586

- an uncompressed bulk data element encoded in Little Endian binary format with the following headers:

- Content-Type: application/octet-stream
- 588 ▪ Content-Location: {BulkDataURL}
- a compressed bulk data element from a SOP Instance in the Study encoded in a single-
590 frame compression {MediaType} with the following headers:
 - Content-Type: {MediaType}
 - 592 ▪ Content-Location: {BulkDataURL}
- a compressed frame from a multi-frame SOP Instance in the Study encoded in a single-
594 frame media type with the following headers:
 - Content-Type: {MediaType}
 - 596 ▪ Content-Location: {BulkDataURL}/frames/{FrameNumber}
- Note: Each frame will come in a separate part.
- 598
- a set of compressed frames from a SOP Instance in the Study encoded in a multi-frame
600 media type with the following headers:
 - Content-Type: {MediaType}
 - 602 ▪ Content-Location: {BulkDataURL}[/frames/{FrameList}]
 - {FrameList} is a list of frames separated by %2C (comma). It may be
604 omitted if the message part includes all frames for the specified bulk pixel
data object.

606 6.5.2 RS – RetrieveSeries

This action retrieves the set of DICOM instances associated with a given study and series UID. The
608 response can be DICOM or bulk data depending on the “Accept” type, and is encapsulated in a multipart
MIME response.

610 6.5.2.1 Request

The specific resource to be used for the RetrieveSeries action shall be as follows:

- 612 • Resource
 - {SERVICE}/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}, where
 - 614 ▪ {SERVICE} is the base URL for the service. This may be a combination of
protocol (either http or https), host, port, and application.
 - 616 ▪ {StudyInstanceUID} is the study instance UID for a single study.
 - {SeriesInstanceUID} is the series instance UID for a single series.
- 618 • Method
 - GET
- 620 • Headers

- 622 ○ Accept – A comma-separated list of representation schemes, in preference order, which will be accepted by the service in the response to this request. The types allowed for this request header are as follows:
- 624 ▪ multipart/related; type=application/dicom; [transfer-syntax={TransferSyntaxUID}]
- 626 Specifies that the response can be DICOM Instances encoded in PS3.10 format. If *transfer-syntax* is not specified the server can freely choose which Transfer Syntax to use for each Instance.
- 628
- 630 ▪ multipart/related; type=application/octet-stream;
- 632 Specifies that the response can be Little Endian uncompressed bulk data.
- 634 ▪ multipart/related; type={MediaType}
- 636 Specifies that the response can be pixel data encoded using a {MediaType} listed in Table 6.5-1 (including parameters).

638 6.5.2.2 Response

640 The Server shall provide the document(s) indicated in the request. In order to parse the bulk data items it is necessary to also retrieve the XML metadata for the Study.

642 The Server shall return the document(s), or an error code when the document(s) cannot be returned. If the server cannot convert all of the data to any of the requested media types/Transfer Syntaxes, then an error code shall be returned, either a “Not Acceptable” response if no data is returned or a “Partial Content” response if only some data is returned.

644

646 The client can compare the SOP Instance UIDs or bulk data URLs in the metadata and the message response to determine which bulk data elements have been returned.

648 All response formats have a content type of multipart/related with a message boundary separator. The response format depends on the Accept header specified in the request.

6.5.2.2.1 DICOM Response

- 650 ▪ Content-Type:
- 652 ○ multipart/related; type=application/dicom; boundary={MessageBoundary}
- 654 ▪ The entire multipart response contains every instance for the specified Series that can be converted to one of the requested Transfer Syntaxes.
- 656 ▪ Each item in the multipart response represents a DICOM SOP Instance with the following http headers:
- 658 ○ Content-Type: application/dicom

6.5.2.2.2 Bulk Data Response

- 658 ▪ Content-Type:
- 660 ○ multipart/related; type= application/octet-stream; boundary={MessageBoundary}
- 660 ○ multipart/related; type={MediaType}; boundary={MessageBoundary}

- 662 ▪ The entire multipart response contains all bulk data for the specified Series that can be converted to one of the requested media types.
- 664 ▪ Each item in the response is one of:
 - 666 ○ an uncompressed bulk data element encoded in Little Endian binary format with the following headers:
 - 666 ▪ Content-Type: application/octet-stream
 - 668 ▪ Content-Location: {BulkDataURL}
 - 670 ○ a compressed bulk data element from a SOP Instance in the Series encoded in a single-frame media type with the following headers:
 - 672 ▪ Content-Type: {MediaType}
 - 674 ▪ Content-Location: {BulkDataURL}
 - 676 ○ a compressed frame from a multi-frame SOP Instance in the Series encoded in a single-frame media type with the following headers:
 - 678 ▪ Content-Type: {MediaType}
 - 680 ▪ Content-Location: {BulkDataURL}/frames/{FrameNumber}
 - 682 ○ a set of compressed frames from a multi-frame SOP Instance in the Series encoded in a multi-frame media type with the following headers:
 - 684 ▪ Content-Type: {MediaType}
 - 686 ▪ Content-Location: {BulkDataURL}/frames/{FrameList}
 - 688 • {FrameList} is a list of frames separated by %2C (comma). It may be omitted if the message part includes all frames for the specified bulk pixel data object.

6.5.3 RS – RetrievalInstance

684 This action retrieves the DICOM instance associated with the given study, series, and SOP Instance UID. The response can be DICOM or bulk data depending on the “Accept” type, and is encapsulated in a
686 multipart MIME response.

6.5.3.1 Request

688 The specific resource to be used for the RetrievalInstance action shall be as follows:

- 690 • Resource
 - 692 ○ {SERVICE}/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}, where
 - 694 ▪ {SERVICE} is the base URL for the service. This may be a combination of protocol (either http or https), host, port, and application.
 - 696 ▪ {StudyInstanceUID} is the study instance UID for a single study.
 - 698 ▪ {SeriesInstanceUID} is the series instance UID for a single series.

- 696 ▪ {SOPInstanceUID} is the SOP Instance UID for a single SOP Instance.
- Method
- 698 ○ GET
- Headers
- 700 ○ Accept – A comma-separated list of representation schemes, in preference order,
702 which will be accepted by the service in the response to this request. The types
702 allowed for this request header are as follows:
- multipart/related; type=application/dicom; [transfer-
704 syntax={TransferSyntaxUID}]
- 706 Specifies that the response can be DICOM Instances encoded in PS3.10
708 format. If *transfer-syntax* is not specified the server can freely choose which
708 Transfer Syntax to use for each Instance.
- multipart/related; type=application/octet-stream;
- 710 Specifies that the response can be Little Endian uncompressed bulk data.
- multipart/related; type={MediaType}
- 712 Specifies that the response can be pixel data encoded using a {MediaType}
714 listed in Table 6.5-1 (including parameters).

716

6.5.3.2 Response

718 The Server shall provide either a single DICOM PS3.10 object for the SOP Instance or one or more bulk
720 data items. In order to parse the bulk data items it is necessary to also retrieve the XML metadata for the
720 Study.

722 The Server shall return the document(s), or an error code when the document(s) cannot be returned. If the
724 server cannot convert all of the bulk data to any of the requested media types, then an error code shall be
724 returned, either a “Not Acceptable” response if no data is returned or a “Partial Content” response if only
724 some data is returned.

726 The client can compare the bulk data URLs in the metadata and the message response to determine
726 which bulk data elements have been returned.

728 All response formats have a content type of multipart/related with a message boundary separator. The
728 response format depends on the Accept header specified in the request.

6.5.3.2.1 DICOM Response

- 730 ▪ Content-Type:
- multipart/related; type=application/dicom; boundary={MessageBoundary}
- 732 ▪ The multipart response contains a single item representing the specified DICOM SOP Instance
with the following http headers:
- Content-Type: application/dicom
- 734

6.5.3.2.2 Bulk Data Response

- 736 ▪ Content-Type:
- 738 ○ multipart/related; type=application/octet-stream; boundary={MessageBoundary}
 - 738 ○ multipart/related; type={MediaType}; boundary={MessageBoundary}
- 740 ▪ The entire multipart response contains all bulk data for the specified Instance that can be converted to one of the requested media types.
- 742 ▪ Each item in the response is one of:
- 742 ○ an uncompressed bulk data element encoded in Little Endian binary format with the following headers:
 - 744 ▪ Content-Type: application/octet-stream
 - 744 ▪ Content-Location: {BulkDataURL}
 - 746 ○ a compressed bulk data element from a SOP Instance encoded in a single-frame media type with the following headers:
 - 748 ▪ Content-Type: {MediaType}
 - 748 ▪ Content-Location: {BulkDataURL}
 - 750 ○ a compressed frame from a multi-frame SOP Instance encoded in a single-frame media type with the following headers:
 - 752 ▪ Content-Type: {MediaType}
 - 752 ▪ Content-Location: {BulkDataURL}/frames/{FrameNumber}
 - 754 ○ a set of compressed frames from a multi-frame SOP Instance encoded in a multi-frame media type with the following headers:
 - 756 ▪ Content-Type: {MediaType}
 - 756 ▪ Content-Location: {BulkDataURL}/[frames/{FrameList}]
 - 758 • {FrameList} is a list of frames separated by %2C (comma). It may be omitted if the message part includes all frames for the specified bulk pixel data object.
 - 760

762 6.5.4 RS – RetrieveFrames

This action retrieves the DICOM frames for a given study, series, SOP Instance UID, and frame numbers.

764 The response is pixel data, and is encapsulated in a multipart MIME response.

6.5.4.1 Request

766 The specific Services resources to be used for the RetrieveFrames action shall be as follows:

- Resource

- 768 ○ {SERVICE}/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames/{FrameList}, where
- 770 ▪ {SERVICE} is the base URL for the service. This may be a combination of protocol (either http or https), host, port, and application.
- 772 ▪ {StudyInstanceUID} is the study instance UID for a single study.
- 774 ▪ {SeriesInstanceUID} is the series instance UID for a single series.
- 776 ▪ {SOPInstanceUID} is the SOP Instance UID for a single SOP Instance.
- 776 ▪ {FrameList} is a comma or %2C separated list of one or more non duplicate frame numbers. These may be in any order (e.g. ../frames/1,2,4,3).
- Method
 - 778 ○ GET
 - Headers
 - 780 ○ Accept
 - 782 ▪ multipart/related; type=application/octet-stream
 - 784 ▪ multipart/related; type={MediaType}
- 786 Specifies that the response can be Little Endian uncompressed pixel data
- 786 Specifies that the response can be pixel data encoded using a {MediaType} and {TransferSyntaxUID} listed in Table 6.5-1 (including parameters).

788 6.5.4.2 Response

790 The Server shall provide the document(s) indicated in the request. In order to parse the bulk data items it is necessary to also retrieve the XML metadata for the Study.

792 The Server shall return the document(s) or an error code when the document(s) cannot be returned. If the server cannot encode the pixel data using any of the requested media types, then an error status shall be returned.

794 All response formats has a content type of multipart/related with a message boundary separator.

6.5.4.2.1 Pixel Data Response

- 796 ▪ Content-Type:
- 798 ○ multipart/related; type=application/octet-stream; boundary={MessageBoundary}
 - 798 ○ multipart/related; type={MediaType}; boundary={MessageBoundary}
- 800 ▪ The entire multipart response contains all requested Frames for the specified Instance.
 - 800 ▪ Each item in the response is one of:
 - 802 ○ an uncompressed frame encoded in Little Endian binary format with the following headers:
 - 802 ▪ Content-Type: application/octet-stream

- 804
 - a compressed frame encoded in a single-frame media type with the following headers:
 - Content-Location: {BulkDataURL}/frames/{FrameNumber}
- 806
 - Content-Type: {MediaType}
 - Content-Location: {BulkDataURL}/frames/{FrameNumber}
- 808
 - a set of compressed frames encoded in a multi-frame media type with the following headers:
 - Content-Type: {MediaType}
 - Content-Location: {BulkDataURL}/frames/{FrameList}
 - 812
 - {FrameList} is a list of frames separated by %2C (comma). It may be omitted if the message part includes all frames for the specified bulk pixel data object.
- 814
 - The frames will be returned in the order specified by the Frame List.

6.5.5 RS – RetrieveBulkdata

816 This action retrieves the bulk data for a given bulk data URL. The response is a single bulk data item.

6.5.5.1 Request

818 The specific Services resource to be used for the RetrieveBulkdata action shall be as follows:

- Resource
 - 820
 - {BulkDataURL}, where
 - 822
 - {BulkDataURL} is the URL of a bulk data element. This may be the URL attribute of a BulkData element from a DICOM PS3.19 XML file received in response to a WADO-RS RetrieveMetadata request.
 - 824
 - The server shall always return the same bulk data for a specified BulkData URL if the data is available.
 - 826
 - If the resource specified by the BulkData URL is not available, the server shall return:
 - 828
 - 404 – Not Found, if the server expects to be able to return the resource again in the future
 - 830
 - 410 – Gone, if the server does not expect the resource to be valid in the future
 - 832
 - The server determines the period of time a BulkData URL resource is available.
 - 834
 - Method
 - GET
 - 836
 - Headers

- Accept

838 ▪ multipart/related; type=application/octet-stream

840 Specifies that the response can be Little Endian uncompressed bulk data.

- multipart/related; type={MediaType}

842 Specifies that the response can be pixel data encoded using a {MediaType}
844 listed in Table 6.5-1 (including parameters).

- Range

846 ▪ See RFC 2616 Section 14.35. If omitted in the request the server shall return
the entire bulk data object.

848 **6.5.5.2 Response**

The Server shall provide the document(s) indicated in the request. In order to parse the bulk data items it
850 is necessary to also retrieve the XML metadata for the Study.

The Server shall return the document(s) or an error code when the document(s) cannot be returned. If the
852 server cannot encode the pixel data using any of the requested media types, then an error status shall be
returned.

854 All response formats have a content type of multipart/related with a message boundary separator. The
response format depends on the Accept header specified in the request.

856 **6.5.5.2.1 Bulk Data Response**

- Content-Type:

- multipart/related; type=application/octet-stream; boundary={MessageBoundary}

- The single item in the response is one of:

- an uncompressed bulk data element encoded in Little Endian binary format with the
following headers:

- Content-Type: application/octet-stream

- Content-Location: {BulkDataURL}

- a compressed bulk data element from a SOP Instance encoded in a single-frame media
type with the following headers:

- Content-Type: {MediaType}

- Content-Location: {BulkDataURL}

- If the Range header is specified in the request, the server shall return only the specified bytes of
the bulk data object. See RFC 2616 Section 14.35.

870 **6.5.6 RS – RetrieveMetadata**

This action retrieves the DICOM instances presented as the full study metadata with the bulk data
872 removed. The response is XML encoded metadata for the DICOM attributes as defined in PS 3.19.

874 The full study metadata includes all DICOM attributes under a certain size threshold which is determined by the server. Some DICOM instances, such as SR documents, may be entirely described in the metadata.

876 6.5.6.1 Request

The specific Services resource to be used for the RetrieveMetadata action shall be as follows:

- 878 • Resource
 - {SERVICE}/studies/{StudyInstanceUID}/metadata, where
- 880 ▪ {SERVICE} is the base URL for the service. This may be a combination of protocol (either http or https), host, port, and application.
- 882 ▪ {StudyInstanceUID} is the study instance UID for a single study.
- Method
- 884 ○ GET
- Headers
- 886 ○ Accept
 - multipart/related; type=application/dicom+xml
- 888 Specifies that the response should be WADO XML.

6.5.6.2 Response

890 The Server shall provide the document(s) indicated in the request. The Server shall return the document(s) or an error code when the document(s) could not be returned.

892 The response format has a content type of application/dicom+xml as described in the Native DICOM Model defined in PS3.19 and must include the URL attribute for each BulkData element.

894 6.5.6.2.1 Metadata Response

- Content-Type:
 - 896 ○ multipart/related; type=application/dicom+xml
- The entire multipart response contains all XML metadata for the specified Study.
- 898 ▪ Each item in the response is the XML encoded metadata for an Instance with the following http headers:
 - 900 ○ Content-Type: application/dicom+xml; transfer-syntax={TransferSyntaxUID}

902 Where {TransferSyntaxUID} is the UID of the DICOM Transfer Syntax used to encode the inline binary data in the XML metadata.

904 Note: The metadata is consistent with the characteristics of the bulk data on the server. If bulk data is requested using specified Transfer Syntaxes or media types, it is possible that the bulk data retrieved may be inconsistent with the metadata. For example, for a Study whose DICOM Tag (0028,2110) "LossyImageCompression" is set to "00", indicating no lossy compression, calling RetrieveStudy and requesting a lossy compression media type will provide pixel data that is inconsistent with the metadata.

906 It is the responsibility of the client to deal with these inconsistencies appropriately.

908

6.5.7 Error Codes

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

| Client Error Code | Client Error Name | Error Situation |
|-------------------|-------------------|--------------------------------------------------------------------------------------------------------|
| 206 | Partial Content | Accept type, Transfer Syntax or decompression method supported for some but not all requested content. |
| 400 | Bad Request | Malformed resource |
| 404 | Not Found | Specified resource does not exist |
| 406 | Not Acceptable | Accept type, Transfer Syntax or decompression method not supported |
| 410 | Gone | Specified resource was deleted |
| 503 | Busy | Service is unavailable |

912

914 **Update PS 3.18 Section 7 PERSISTENT OBJECT TYPES as indicated.**

7 Persistent Object types

916 The provisions for some specific object types shall be as defined in this section.

918 Note: In all cases the categorization depends on the SOP Class of the objects, enabling a client, or application building an HTML page for the client, to determine in advance of the request what the requirements will be.

920

7.1 SINGLE FRAME IMAGE OBJECTS

922 7.1.1 Objects accessed

924 In this category are all object instances of SOP classes defined in PS 3.3 that consist of a single image frame, instances of multi-frame SOP Classes that contain only one frame, or object instances that consist of single frame accessed from instances of multi-frame SOP Classes using the "frameNumber" parameter.

926 7.1.2 MIME type constraints

The Server shall be able to send a response **in for** each of the following MIME types:

928 — WADO-WS

- application/dicom

930

- image/jpeg

— WADO-RS

- 932 ○ application/dicom
- application/octet-stream
- 934 ○ application/dicom+xml

If the contentType parameter is not present in the **request WADO-WS response**, the response shall contain an image/jpeg MIME type, if compatible with the 'Accept' field of the GET method. **If the contentType parameter is not present in the WADO-RS response, the response is dependent on the 'Accept' field and the requested resource.**

When an image/jpeg MIME type is returned, the image shall be encoded using the JPEG baseline lossy 8 bit Huffman encoded non-hierarchical non-sequential process ISO/IEC 10918.

Note: The choice of image/jpeg as the default for continuous tone images is a consequence of the universal support by Web Clients.

944 The Server should also support the following MIME types **for WADO-WS:**

- image/gif
- 946 — image/png
- image/jp2

948 **The Server should also support the following MIME types for WADO-RS:**

- image/dicom
- 950 — image/dicom+jpeg
- image/dicom+rle
- 952 — image/dicom+jpeg-ls
- image/dicom+jp2image/dicom+jpx

954 The Server may also support other MIME types.

7.2 MULTI-FRAME AND VIDEO IMAGE OBJECTS

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.

7.2.2 MIME type constraints

The Server shall be able to send a response **in for** the following MIME types:

- 960 — **WADO-WS**
 - application/dicom
- 962 — **WADO-RS**
 - application/dicom

- 964 ○ **application/octet-stream**
- **application/dicom+xml**

966 If the contentType parameter is not present in the **WADO-WS** request, the response shall contain a application/dicom MIME type.

968 The Server ~~should also~~ **can optionally** support the following MIME types **for WADO-WS:**

— video/mpeg

970 — image/gif

The Server can optionally support the following MIME types for WADO-RS:

972 — **image/dicom+jpx**

— **video/mpeg**

974 — **video/mp4**

The Server may also support other MIME types.

976 **Changes to NEMA Standards Publication PS 3.19-2011**

Digital Imaging and Communications in Medicine (DICOM)

978 **Part 19: Application Hosting**

980 **Add Bulk Data URI to PS 3.19 Annex A.1 Symbols Native DICOM Model**

A.1 NATIVE DICOM MODEL

982 ...

A.1.4 Information Model

984 A diagram of the Native DICOM Model appears in Figure A.1.4-1.

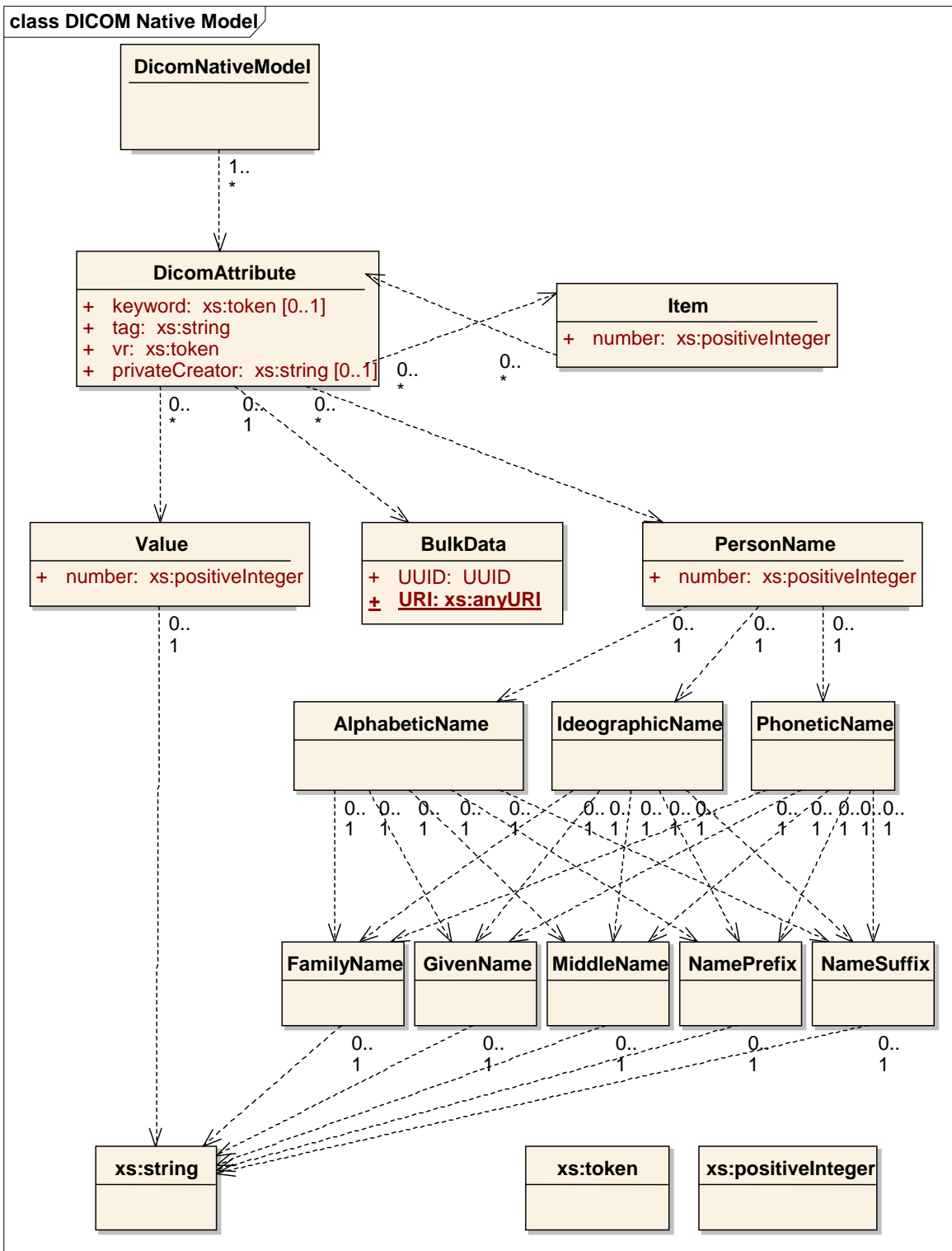


Figure A.1.4-1 Native DICOM Model

A.1.5 Description

988

Table A.1.5-1 Native DICOM Model

| Name | Optionality | Cardinality | Description |
|-----------------------------------------------------|-------------|-------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| NativeDicomModel | R | 1 | <p>An Infoset (as defined in W3C Recommendation XML Information Set “http://www.w3.org/TR/xml-infoset/”) representing the content of a DICOM Data Set (as defined in PS3.5), which may be either:</p> <ul style="list-style-type: none"> - the contents of an entire DICOM Composite Instance (as defined in PS3.3) in response to a native model request, or - the contents of part of a DICOM Composite Instance in response to a query on a native model, or - the contents of a Sequence Item (as defined in PS3.5), recursively included within an Infoset Value element. <p>The directive <code>xml:space="preserve"</code> shall be included.</p> |
| <i>Include ‘DICOM DataSet Macro’ Table A. 1.5-2</i> | | | |

990

Table A.1.5-2 DICOM Data Set Macro

| Name | Optionality | Cardinality | Description |
|----------------|-------------|-------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| DicomAttribute | O | 0-n | An Infoset element corresponding to each DICOM Attribute. |
| >keyword | C | A | <p>The keyword as defined in PS3.6.</p> <p>Required unless the DICOM Data Element is unknown to the host.</p> |
| ... | | | |
| >BulkData | C | 1 | <p>A reference to a blob of data that the recipient may retrieve through use of the GetData() method or a WADO-RS call.</p> <p>Required if the DICOM Data Element represented is not zero length and an XML Infoset Value, Item, or PersonName element is not present.</p> <p>The provider of the data may use a BulkData reference at its discretion to avoid encoding a large DICOM Value Field as text by value in the Infoset. For example, a provider may include large</p> |

| | | | |
|--------|----------|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| | | | <p>binary values such as pixel data or look up tables, which typically would be located in a file, as BulkData references.</p> <p>Note that there is a single BulkData Infoset element representing the entire Value Field, and not one per Value in the case where the Value Multiplicity is greater than one. E.g., a LUT with 4096 16 bit entries that may be encoded in DICOM with a Value Representation of OW, with a VL of 8192 and a VM of 1, or a US VR with a VL of 8192 and a VM of 4096 would both be represented as a single BulkData element.</p> <p>All rules (e.g. byte ordering and swapping) in DICOM PS3.5 apply.</p> <p>Note: Implementers should in particular pay attention the PS3.5 rules regarding the value representations of OW and OF.</p> <p>If the BulkData has a string or text Value Representation, the value(s) of the DICOM Specific Character Set Data Element, if present, might be necessary to determine its encoding.</p> |
| >>UUID | <u>C</u> | A | <p>An identifier of this bulk data reference formatted as a UUID using the hexadecimal representation defined in ITU-T Recommendation X.667.</p> <p><u>Required if BulkData URI is not present. Shall not be present otherwise.</u></p> |
| >>URI | <u>C</u> | <u>A</u> | <p><u>The HTTP(S) URI for this bulk data reference.</u></p> <p><u>Required if the NativeDicomModel was:</u></p> <p><u>- returned in response to a WADO-RS Retrieve Metadata request</u></p> <p><u>Shall not be present otherwise.</u></p> |

992 **Schema**

The Normative version of the XML Schema for the Native DICOM Model follows:

994 default namespace="http://dicom.nema.org/PS3.19/models/NativeDICOM"

996 # This schema was created as an intermediary, a means of describing

Supplement 161 WADO by means of RESTful Services

Page 35

```
998 # native binary encoded DICOM objects as XML Infosets, thus allowing
# one to manipulate binary DICOM objects using familiar XML tools.
# As such, the schema is designed to facilitate a simple, mechanical,
1000 # bi-directional translation between binary encoded DICOM and XML-like
# constructs without constraints, and to simplify identifying portions
1002 # of a DICOM object using XPath statements.
#
1004 # Since this schema has minimal type checking, it is neither intended
# to be used for any operation that involves hand coding, nor to
1006 # describe a definitive, fully validating encoding of DICOM concepts
# into XML, as what one might use, for example, in a robust XML
1008 # database system or in XML-based forms, though it may be used
# as a means for translating binary DICOM Objects into such a form
1010 # (e.g. through an XSLT script).

1012 start = element NativeDicomModel { DicomDataSet }

1014 # A DICOM Data Set is as defined in PS3.5. It does not appear
# as an XML Element, since it does not appear in the binary encoded
1016 # DICOM objects. It exists here merely as a documentation aid.
DicomDataSet = DicomAttribute*
1018
DicomAttribute = element DicomAttribute {
1020   Tag, VR, Keyword?, PrivateCreator?,
   ( BulkData | Value+ | Item+ | PersonName+ )?
1022 }
BulkData = element BulkData { UUID | URI }
1024 Value = element Value { Number, xsd:string }
Item = element Item { Number, DicomDataSet }
1026 PersonName = element PersonName {
   Number,
1028   element SingleByte { NameComponents }?,
   element Ideographic { NameComponents }?,
1030   element Phonetic { NameComponents }?
}
1032
NameComponents =
1034   element FamilyName {xsd:string}?,
   element GivenName {xsd:string}?,
1036   element MiddleName {xsd:string}?,
   element NamePrefix {xsd:string}?,
1038   element NameSuffix {xsd:string}?

1040 # keyword is the attribute tag from PS3.6
# (derived from the DICOM Attribute's name)
1042 Keyword = attribute keyword { xsd:token }
# canonical XML definition of Hex, with lowercase letters disallowed
1044 Tag = attribute tag { xsd:string{ minLength="8" maxLength="8" pattern="[0-9A-F]{8}" } }
VR = attribute vr { "AE" | "AS" | "AT" | "CS" | "DA" | "DS" | "DT" | "FL" | "FD"
1046   | "IS" | "LO" | "LT" | "OB" | "OF" | "OW" | "PN" | "SH" | "SL"
   | "SQ" | "SS" | "ST" | "TM" | "UI" | "UL" | "UN" | "US" | "UT" }
1048 PrivateCreator = attribute privateCreator{ xsd:string }
UUID = attribute uuid { xsd:string }
1050 URI = attribute uri { xsd:anyURI }
Number = attribute number { xsd:positiveInteger }
```