

2

4

6

Digital Imaging and Communications in Medicine (DICOM)

8

Supplement 174: Rendering for WADO-RS

10

12

14

16

Prepared by:

18 **DICOM Standards Committee, Working Group 27 Web Technology**

1300 N. 17th Street, Suite 1752

20 Rosslyn, Virginia 22209 USA

22

24 Contact: svastagh@medicalimaging.org

VERSION: Draft 04, 10 September 2014

26 Developed in accordance with: DICOM Workitem 2013-12-X

28

OPEN ISSUES

7	Should Accept headers be in preference order? Yes.
8	Should we add a rendered service, or just add optional parameters to existing transactions?
9	Should the Search parameters defined in QIDO section be added to Section 8?
10	Should the Order of Images in Response parameter be included?
11	Should the Rotate parameter be included?
12	Should the Flip parameter be included?
13	What extensions might we want to add in the future?

30

CLOSED ISSUES

1	Should unknown parameter keywords be ignored or should they generate errors? Decision: Ignore unknown parameter keywords. URI and WS should also ignore unknown parameters.
2	Can vendors add other parameters? YES. Any vendor defined parameters should be specified in their Conformance Statement and in their Service Capabilities response.
3	Can the annotation parameter be used with presentation states (in RS)? No. Presentation States should not have any other rendering parameters.
5	What should we do if a resource references a presentation state object? Decision: When a resource contains a reference to a presentation state (series or instance) the images referenced by the presentation state should be rendered using the presentation state and then returned in the response
6	Can the anonymization parameter be used with presentation states? No. Presentation States should not have any other rendering parameters.
7	

32

34

Scope and Field of Application

36 This supplement enables the RESTful Retrieve service to retrieve rendered instances. This is done by
adding query parameters to the request URI. These parameters are similar to those already available in
38 the URI and WS Retrieve services.

A client makes an HTTP request with query parameters specifying how the images ~~should~~ shall be
40 rendered and receives a response containing those images as the result.

42

44

Changes to NEMA Standards Publication PS 3.18-2012

Digital Imaging and Communications in Medicine (DICOM)

46

Part 18: Web Services

Update Section 6.2.1 as follows:

48 6.2.1 Query Parameters of the HTTP Request

The parameters of the *query* component of the Request-Target to be sent to the web Server through the HTTP GET method request shall be represented as defined in [IETF RFC3986](#).

52 ~~Notes: 1. Other components of the Request-URI depend on the configuration, e.g location and script language of the Web Enabled DICOM Server.~~

54 ~~2. The means by which the Web Client System obtains the values of the necessary parameters for web accessing of DICOM objects is out of the scope of the standard.~~

Update Section 6.2.3 as follows:

56 6.2.3 List of character sets supported in the Response

The "Accept-~~e~~Charset" **header** field of the **HTTP** GET method request shall specify the character set of the object(s) to be retrieved. If the "Accept-~~e~~Charset" **header** field of the GET method **request** is not present, or the ~~Web DICOM~~ **Origin-Server** does not support the specified character set, the character set of the response will be at the discretion of the ~~Web Enabled DICOM~~ Server.

62 ~~Note Typically the user of a Web Client does not have control over the "Accept-charset" field. An optional parameter specifies the character set to be used in the returned object.~~

64 Update Section 6.3.1.3 as follows:

6.3.1.3 Transfer Syntax

66 The returned DICOM object shall be encoded using one of the transfer syntaxes specified in the transfer syntax query parameter as defined in Section 8.2.11 below. By default, the transfer syntax shall be "Explicit VR Little Endian"

70 **The returned DICOM object shall be encoded using the transfer syntax specified in the Content-Type header field. If the transfer syntax is not specified, it shall be "Explicit VR Little Endian".**

72 **Notes:** 1. This implies that ~~r~~ Retrieved images are transmitted un-compressed by default.
2. **The use of the transfer syntax query parameter defined in Section 8.6.3 below is discouraged.**

74

Update Section 6.3.2 as follows:

76 **6.3.2 Payload Body of Non-DICOM MIME Media Types** response

77 **6.3.2.1 MIME Content Type**

78 The media MIME type of the response payload shall be one of the MIME media types specified, in preference order, defined in the Accept header field or the contentType parameter of the request, preferably the most desired by the Web Client, and shall be in any case compatible with the 'Accept' header field of the GET method request. The Content-Type header field of the response message shall specify the media type of the response payload.

84 If the Origin-Server cannot provide the requested media type(s) it shall return a Status Code of 406 – Not Acceptable.

86 Note: ~~1. The HTTP behavior is that an error (406 – Not Acceptable) is returned if the required content type cannot be served.~~

The use of the contentType query parameter defined in Section 8.6.1 below is discouraged.

88

6.3.2.2 Content

90 The content payload shall be a single MIME part containing the requested object ~~to be retrieved.~~

92 Note: ~~Multiple objects in a response are not supported by this standard. The parameters select only a single object to retrieve. Most current Web Clients are able to retrieve single objects, within a "non multipart" MIME body, and are not able to support multipart/related or multipart/mixed responses.~~

94 **Update Section 6.5.1.1 as follows:**

6.5.1.1 Request

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

Resource

98 {SERVICE}/studies/{StudyInstanceUID}[? {Query}]*]

where

100 {SERVICE} is the base URL for the service. This may be a combination of protocol (either http or https), host, port, and application.

102 {StudyInstanceUID} is the study instance UID for a single study.

{Query} is one or more query parameters specified in Section 8.

104

~~MIME body, and are not able to support multipart/related or multipart/mixed responses.~~

106 **Insert Section 6.5.1.2.3 as follows:**

6.5.1.2.3 Rendered Response

108 The Content-Type header field of the response contains the Media Type of the multipart rendered response. Each part of the multipart response is a rendered object with its media type specified in the Content-Type header field of each Part.

110

112 **Update Section 6.5.2 as follows:**

6.5.2 WADO-RS - RetrieveSeries

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

6.5.2.1 Request

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

Resource

120 {SERVICE}/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}[? {Query}*

where

122 {SERVICE} is the base URL for the service. This may be a combination of protocol (either http or
https), host, port, and application.

124 {StudyInstanceUID} is the study instance UID for a single study.

{SeriesInstanceUID} is the series instance UID for a single series.

126 **{Query} is one or more query parameters specified in Section 8.**

128 **Insert Section 6.5.1.2.3 as follows:**

6.5.2.2.3 Rendered Response

130 The Content-Type header field of the response contains the Media Type of the multipart rendered response. Each
part of the multipart response is a rendered object with its media type specified in the Content-Type header field of
132 each Part.

134 **Update Section 6.5.3 as follows:**

6.5.3 WADO-RS - RetrieveInstance

136 **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
138 **encapsulated in a multipart MIME response.**

6.5.3.1 Request

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

Resource

142 {SERVICE}/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}[
 ? {Query}*]

144 where

146 {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.

148 {SeriesInstanceUID} is the series instance UID for a single series.

 {SOPInstanceUID} is the SOP Instance UID for a single SOP Instance.

150 **{Query} is one or more query parameters specified in Section 8.**

152 **Insert Section 6.5.3.2.3 as follows:**

6.5.3.2.3 Rendered Response

154 The Content-Type header field of the response contains the Media Type of the multipart rendered response. Each
156 part of the multipart response is a rendered object with its media type specified in the Content-Type header field of
 each Part.

158 **Update Section 6.5.4 as follows:**

6.5.4 WADO-RS - RetrieveFrames

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

162 6.5.4.1 Request

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

164 **Resource**

166 {SERVICE}/studies/{StudyInstanceUID}/series/{SeriesInstanceUID}/instances/{SOPInstanceUID}/frames/
 {FrameList}[? {Query}*]

 where

168 {SERVICE} is the base URL for the service. This may be a combination of protocol (either http or
 https), host, port, and application.

170 {StudyInstanceUID} is the study instance UID for a single study.

 {SeriesInstanceUID} is the series instance UID for a single series.

172 {SOPInstanceUID} is the SOP Instance UID for a single SOP Instance.

174 {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).

{Query} is one or more query parameters specified in Section 8.

176

Insert Section 6.5.4.2.3 as follows:

178 **6.5.4.2.3 Rendered Response**

180 The Content-Type header field of the response contains the Media Type of the multipart rendered response. Each part of the multipart response is a rendered object with its media type specified in the Content-Type header field of each Part.

182

184 **Update Section 7.1.2 as follows:**

7.1 SINGLE FRAME IMAGE OBJECTS

186 **7.1.2 ~~MIME~~ Media type constraints**

188 The **Origin-Server** shall **support** be able to send a response in each of the following **single frame media** ~~MIME~~ types:

- application/dicom
- 190 • image/jpeg

192 **The media type of the response payload shall be one of the media types specified, in preference order, in the Accept header field or the *contentType* parameter of the request. If none of the specified media types are supported, the Origin-Server shall return a Status Code of 406.**

194 **If no media type is specified the media type shall default to image/jpeg.**

196 ~~If the 'contentType' parameter is not present in the request, the response shall contain an image/jpeg MIME type, if compatible with the 'Accept' field of the GET method.~~

198 **The Content-Type header field of the response message shall specify the media type of the response.**

200 When an image/jpeg ~~MIME~~ **media** 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.

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

The Server should also support the following ~~MIME~~ **media** types:

- 204 • image/gif
- image/png
- 206 • image/jp2

The Server may also support other ~~MIME~~ **media** types.

208 **Update Section 7.2.2 as follows:**

7.2 MULTI-FRAME IMAGE OBJECTS

7.2.1 Objects included

This category includes ~~are~~ all SOP classes defined in PS 3.3 that are multi-frame image objects.

7.2.2 ~~MIME~~ Media type constraints

The ~~Origin-Server~~ shall support ~~be able to send a response in~~ the following ~~MIME~~ multi-frame media type:

- application/dicom

If the Accept header field or the ~~contentType~~ parameter ~~is~~ are not present in the request, the response shall contain a Content-Type header field of specifying the application/dicom ~~MIME~~ media type and a payload containing an object with that media type.

The Server should also support the following ~~MIME~~ media types:

- video/mpeg
- image/gif

The Server may also support other ~~MIME~~ media types.

Update Section 7.3 as follows:

7.3 TEXT OBJECTS

7.3.1 Objects included

This category contains ~~are~~ all SOP classes defined in PS 3.3 that include the SR Document Content Module.

Note: This includes all SOP Classes that are SR documents, such as narrative text, structured reports, CAD, measurement reports and key object selection documents.

7.3.2 ~~MIME~~ Media type constraints

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

- application/dicom
- text/plain
- text/html

If the Accept header field or the ~~contentType~~ parameter ~~is~~ are not present in the request, or contains only ~~MIME~~ media types that the Server does not support, the response shall contain a text/html ~~MIME~~ media type.

It is recommended that the Server also support the following ~~MIME~~ media types:

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

246 The Server may also support other ~~MIME~~ **media** types.

Update Section 7.4 as follows:

248 **7.4 OTHER OBJECTS**

7.4.1 Objects included

250 The category shall include all objects of all SOP classes defined in PS 3.3 that are not included in the categories described in the sections above, and which are considered in PS 3.3 as classes of persistent objects.

7.4.2 ~~MIME~~ Media type constraints

254 The Server shall be able to send a response in the following ~~MIME~~ **media** type:

- application/dicom

256 The Server may also support other ~~MIME~~ **media** types.

If the **Accept header field** or the *contentType* parameter is **are** not present in the request, the response shall contain an application/dicom ~~MIME~~ **media** type.

260 **Replace Section 8 with the following:**

8 Parameters

262 This section describes parameters used with Web Services.

The URI and RS Retrieve Services use the *query* component of the Target URI to specify parameters. Parameters that are supported by both URI and RS have the same keyword.

Throughout this section information that pertains to a particular service is designated by the phrase “X Service” where X is “URI”, “WS”, or “RS” or some combination thereof.

268 *URI and RS Service:* The parameter names are the same, and the parameters are specified using the *query component* of the Target URI as specified in IETF RFC3986. The syntax is:

`http://{authority}/{path}?{keyword1}={value1}&{keyword2}={value2}&{keyword3}...`

270 For example,

`http://foo.org/bar?requestType=WADO&studyUID=1.2.250.1.59.40211.12345678.678910& ...`

272 *WS Service:* The parameter names for the WS service are similar to those of URI and RS, but the first letter of the parameter is capitalized. The parameters are specified in the request payload (See 6.4.1.1, 6.4.2.1, 6.4.3.1).

276 The tables below specify for each Service the Name, Optionality, and the Value Representation (VR) of the parameter. The Optionality indicates whether the parameter is Required (R), Conditional (C), or Optional (O). All parameter values are strings. Their format is specified using Value Representations (VR), defined

278 in PS 3.5, *except that 1) they shall not be padded to an even length and 2) certain characters may need to*
be percent encoded.

280

8.1 WADO-URI Request Type

282 This parameter specifies that the request is a WADO-URI request. It is only used in URI service. The
value has a VR of Short Text (ST). It is REQUIRED. The value shall be "WADO".

284

Table 8.1-1. Request Type Parameter

Service	Name	Optional	VR
URI	requestType	R	ST

286 8.2 IDENTIFYING DICOM OBJECT(S)

8.2.1 Study Request

288 This parameter identifies a study by its Study Instance UID as specified in PS 3.3.

Table 8.2-1. Study UID Parameter

	Name	Optional	VR
URI	studyUID	R	UI
WS	StudyRequest	R	UI

290

8.2.2 Series Request

292 This parameter specifies the Series Instance UID as specified in PS 3.3.

Table 8.2-2. Series UID Parameter

	Name	Optional	VR
URI	"seriesUID"	R	UI
WS	"SeriesRequest"	R	UI

294

296 *WS Service:* one or more SeriesRequest, each of which has a REQUIRED seriesInstanceUID attribute
as its value. The SeriesRequest(s) are included in the StudyRequest parameter
described above.

298 8.2.3 SOP Instance Request

SOP Instance UID as defined in the PS 3.3.

300

Table 8.2-3. SOP Instance UID Parameter

	Name	Optional	VR
URI	objectUID	R	UI
WS	DocumentRequest	R	UI

302

WS Service: one or more DocumentRequest(s), each of which includes:

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

The DocumentRequest(s) are included in the SeriesRequest parameter described above.

8.2.4 Frame Number

This parameter specifies the frame(s) that shall be returned from a multi-frame image object, as defined in PS 3.3. This parameter shall be ignored for of all objects other than multi-frame objects. The parameter value specifies one frame number.

Table 8.2-4. Frame Number

	Name	Optional	VR
URI	frameNumber	O	IS
WS	FrameNumber	O	IS

8.3 PRESENTATION STATES

The parameters in this section can be used to specify a presentation state that is to be applied to the images specified in the resource. If these parameters are used, then the explicit rendering parameters defined in Section 8.4 shall not be used. If no rendering parameters are specified the server will determine how the images are rendered.

If the resource in the Retrieve request specifies a presentation series or instance and the media type is a rendered media type then the response will contain the set of one or more instances referenced by the presentation state rendered in the media type specified. A resource that specifies a presentation state series or instance cannot be used with the parameters defined in this section.

All presentation state operations are applied in the order specified by the appropriate presentation state pipeline. (See PS 3.4 Annex N)

- Notes:
1. The presentation states can only reference image instances. This restriction also implies that the presentation state parameters defined in this section can only be used with resources that specify images.
 2. The Presentation State must be in the same study as the images it applies to.
 3. If the presentation state parameters are used both parameters shall be present and image rendering parameters shall not be present for URI and WS, for RS either or both presentation state parameters can be present.

The parameters defined in this section can only be used with the following transactions:

URI Service: Retrieve transaction

WS Service: RetrieveRenderedImagingDocumentSet transaction

RS Service: Retrieve transaction

338 **8.3.1 Presentation State**

340 This parameter specifies the SOP Instance UID of the presentation state object that shall be used to render the images.

Table 8.2-1. Presentation State SOP Instance UID

	Name	Optional	VR
URI	presentationUID	O	UI
WS	PresentationUID	O	UI
RS	N/A	O	UI

342

URI and WS: presentationUID={UID}

344 *RS Service:* presentationUID= {seriesUID}, {instanceUID}
I {InstanceUID}

346

348 For the RS Service, if the presentationUID parameter specifies a series then the resource ~~should~~ **shall** specify the series to which it is to be applied. If the presentationUID parameter specifies a series and an instance, then the presentationSeriesUID parameter shall not be present. If the presentationUID only specifies an instance then the presentationSeriesUID shall be present.

352 The server shall return an error if the Presentation State parameter does not reference any of the SOP Instances specified by the resource.

354 Note: In some older clients it is not possible to determine the dimensions of the viewport of the client display, in which case the intent of the TRUE SIZE mode in the presentation state cannot be satisfied, since the viewport size is not known.

356

8.3.2 Presentation State Series UID

358 This parameter specifies the Series Instance UID of the series containing the presentation state object to be applied to the specified images.

360

Table 8.2-2. Presentation State Series UID

	Name	Optional	VR
URI	presentationSeriesUID	C	UI
WS	PresentationSeriesUID	C	UI
RS	presentationSeriesUID	C	UI

362 *URI and WS:* PresentationSeriesUID={UID}

364 This parameter is REQUIRED and shall be present if and only if "presentationUID" is present.

RS Service: PresentationSeriesUID={UID}

366 This parameter is REQUIRED and shall be present if and only if "presentationUID" is present and specifies a presentation state instance.

368

8.4 PARAMETERS FOR SPECIFYING IMAGE ANNOTATIONS

370 8.4.1 Image Annotation

This parameter specifies that the returned images shall be annotated with information about the patient and/or the performed procedure. If it is not present the returned images will have no annotations applied.

Table 8.4-1. Image Annotation

	Name	Optional	VR
URI	annotation	O	ST
WS	Annotation	O	ST
RS	annotation	O	ST

374

When used in conjunction with a presentation state or a region parameter, the annotations shall be applied after the presentation state has been applied to the images or the region has been selected.

The parameter value is a non-empty list of one or more of the following items, each separated by a "," character:

- *patient*, for displaying patient information on the image (e.g. patient name, birth date,...)
- *technique*, for displaying information of the technique image

Note: The exact nature and presentation of the annotation is determined by the Server. Any additional annotations supported by the server should be documented in the Conformance Statement and should also be documented in the Service Capabilities response. The annotation may be burned into the returned image pixels or it may use overlays.

386

8.5 PARAMETERS FOR SPECIFYING IMAGE RENDERING

The parameters defined in this section specify various rendering transformations to be applied to the DICOM image objects specified in the request. The resulting images, which are returned in the response payload, shall conform to one of the media types specified in the request. Only image media types (see sections 7.1 and 7.2) are supported. These parameters are not supported for non-imaging media types, such as application/dicom or text/xml and they ~~should~~ **shall** not be included in the request.

The parameters defined in this section can only be used with the following transactions:

- *URI Service*: Retrieve transaction
- *WS Service*: RetrieveRenderedImagingDocumentSet transaction
- *RS Service*: Retrieve transaction

The parameters defined in this section, shall not be present if there is a *presentationUID* or *presentationStateUID* parameter present in the request.

The set of transformations specified by the parameters in this section shall be applied to the source images as if they were a presentation state, that-is in the order specified by the applicable image rendering pipeline specified in PS 3.3.

402

8.5.1 Number of Pixel Rows

This parameter specifies height in pixels of the returned image(s).

Table 8.5-1. Number of Pixel Rows

	Name	Usage	VR
URI	rows	O/O	IS
WS	Rows	O/O	IS
RS	rows	O/R	IS

406

If both *rows* and *columns* are specified, then each shall be interpreted as a maximum, and a size will be chosen for the images within these constraints, maintaining the aspect ratio of the original image(s), and the returned image(s) will be scaled to that size. 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 aspect ratio if the original images. If both are absent, the images (or selected region) are sent in their original size (or the size of the presentation state applied to the images), resulting in one pixel of screen image for each value in the original image's data matrix.

414 **8.5.2 Number of Pixel Columns**

This parameter specifies width in pixels of the returned image(s).

416

Table 8.5-2. Number of Pixel Columns

	Name	Usage	VR
URI	columns	O/O	IS
WS	Columns	O/O	IS
RS	columns	O/R	IS

418 If both "rows" and "columns" are specified, then each shall be interpreted as a maximum, and a size will be chosen for the images within these constraints, maintaining the aspect ratio of the original images. 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 aspect ratio if the original images. If both are absent, the images (or selected region) are sent in their original size (or the size of the presentation state applied to the images), resulting in one pixel of screen image for each value in the original image's data matrix.

424 See Section 8.5.1 for constraints when both *rows* and *columns* are specified.

8.5.3 Image Region

426 This parameter specifies a rectangular region of the original image(s) that will be returned in the response. The purpose of this parameter is to allow a user to view a selected area of an image matrix, for example at higher magnification.

Table 8.5-3. Image Region

	Name	Usage	VR
URI	region	O/O	DS
WS	Region	O/O	DS
RS	region	O/R	DS

430

The value shall be expressed as a list of four positive Decimal Strings (DS), that are separated by the ',' character. These decimal strings specify a region of the source image(s) to be returned. The values shall specify a region in a normalized coordinate system relative to the size of the original image matrix

434 measured in rows and columns, with values ranging from 0.0 to 1.0, and representing in the following
order:

- 436 • 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.
- 438 • the y position of the top left hand corner of the region to be retrieved, 0.0 corresponding to the
first row of the image matrix.
- 440 • 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.
- 442 • 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.

444

If this parameter is specified, an image matrix corresponding to the specified region shall be returned with
446 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
448 corresponding presentation state has been applied to the images.

8.5.4 WindowLevel

450 This parameter controls the luminosity and contrast of the images as defined in PS 3.3. The value will
contain two Decimal Strings separated by a comma (“,”). The first value shall specify the *Window Center*
452 and the second value shall specify the *Window Width*.

Table 8.5-4. Window Level

	Name	Usage	VR
RS	window	O/R	DS, DS

454

For example, “window= 350, 50” specifies that the Window Center is 350 and the Window Width is 50.

456

8.5.5 Window Center

458 This parameter controls the luminosity of the images as defined in PS 3.3. It is CONDITIONAL, and
becomes REQUIRED if the Window Width parameter is present. This parameter shall not be present if
460 there is a *presentationUID* parameter.

Table 8.5-5. Window Center

	Name	Usage	VR
URI	windowCenter	C	DS
WS	WindowCenter	C	DS

462

Note: The *windowCenter* and *windowWidth* parameters must both be present or both be absent.

464

8.5.6 Window Width

466 This parameter controls the contrast of the images as defined in PS 3.3. It is CONDITIONAL, and;
becomes REQUIRED if the Window Center parameter is present. This parameter shall not be used if there
468 is a *presentationUID* parameter present.

Table 8.5-6. Window Width

	Name	Usage	VR
URI	windowWidth	C	DS
WS	WindowWidth	C	DS

470

Note: The *windowCenter* and *windowWidth* parameters must both be present or both be absent.

472

8.5.7 Window Preset

474 This parameter is used to specify values for *Window Width* and *Window Center* that are optimized for
displaying different types of tissue. It is OPTIONAL. This parameter shall not be used if there is a
476 *presentationUID* parameter present.

Table 8.5-7. Window Presets

	Name	Required	VR
URI	windowPreset	O	ST
WS	WindowPreset	O	ST
RS	windowPreset	O	ST

478

This standard does not define the names or window values (center, width); however, the client can use the
480 Server Capabilities Service to retrieve the names and associated values of the presets support by that
server. This parameter shall not be used if the *window*, *windowCenter*, or *windowWidth* parameters are
482 present.

8.5.8 Image Quality

484 This parameter specifies the quality of the returned images. It is only supported for image media types
that allow lossy compression. The value shall be an integer with a range of 1 to 100 inclusive.

486

Table 8.5-8. Image Quality

	Name	Usage	VR
URI	imageQuality	O	IS
WS	ImageQuality	O	IS
RS	imageQuality	O	IS

488 Note: Decompression and recompression may degrade the image quality if the original image was already
irreversibly compressed. If the image(s) has already been lossy compressed in the same format as the
490 requested media type (e.g. jpeg), then the image(s) may be sent "as is" without decompressing and
recompressing.

492

8.5.9 Order of Images in Response

494 This parameter controls the order of the rendered images or objects in the response.

Table 8.5-9. Image Order

	Name	Usage	VR
RS	order	O	SS

496

Its value is one of the keywords defined in Table 8.3-11. Other keywords

498

Table 8.5-10. Keywords for Ordering Images

Keyword	Description	Attribute
datetime	The objects in the response will be in ascending order by acquisition data and time.	{0008,0012} InstanceCreationDate
number	The objects in the response will be in ascending order by series number, instance number and frame number.	(0020,0011) SeriesNumber, (0020,0013) InstanceNumber, (0020,9156) FrameAcquisitionNumber
position	The objects in the response will be in ascending order by series, instance and frame position.	(0020,0032) ImagePositionPatient
phase	The objects in the response will be in ascending order by series, instance and frame position.	TemporalPositionIndex

500

8.5-10 Rotate

502 This parameter controls the rotation of rendered images. Its value is a positive or negative integer
503 between -360 and 360, where a positive value indicates clockwise rotation and a negative value indicates
504 counterclockwise rotation.

Table 8.5-10. Rotate

	Name	Usage	VR
RS	rotate	R	IS

506

8.5.11 Flip

508 This parameter controls flipping of images in the response. Its value is either “horizontal” or “vertical,”
509 which specifies the axis around which the image is flipped.

510

Table 8.5-11. Flip

	Name	Usage	VR
RS	flip	R	SS

512 **8.5-12.**

514

8.6 MISCELLANEOUS PARAMETERS

516 **8.6.1 De-Identification or Anonymization**

This parameter specifies that the objects returned in the response shall have all *Individually Identifiable Health Information* removed from them, as defined in PS 3.15. This process is called *de-identification* and is sometimes referred to as *anonymization*; although, this term is discouraged. De-identified objects shall have a new SOP Instance UID that is different from that of the original identified objects. The Server may return an error if either it cannot or refuses to anonymize the requested objects.

522

Table 8.6-1. De-Identification

	Name	Usage	VR	Value
--	------	-------	----	-------

URI	anonymize	O	SS	yes
WS	Anonymize	O	SS	yes
RS	de-identify or anonymize	R	SS	yes

524 Note: The de-identified object(s) can be used for teaching or clinical trial applications to provide access to the
 526 original images, without disclosing any person's identity, or requiring storage for a (de-identified) copy of
 528 the original image. De-identification is the responsibility of the Server. In order to preserve patient
 530 confidentiality, the Server will likely refuse to deliver a de-identified SOP instance to an unknown or
 532 unauthorized person unless the Server is certain that the SOP instance contains no *Individually
 Identifiable Health Information*. This includes removing or "blanking out" any annotation area(s)
 containing *Individually Identifiable Health Information* burned into the pixels or in overlays. (See PS3.15
 Section E. Attribute Confidentiality Profiles)

8.6.2 Retrieve Partial Information

534 This parameter specifies the retrieval of 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.
 536 The parameter name shall be "XPath" and shall only be used in the WS
RetrieveImagingDocumentSetMetadata transaction.

8.7 PARAMETERS CORRESPONDING TO REQUEST HEADER FIELDS

538 The parameters in this section have counterparts in the HTTP/1.1 header fields. These parameters were
 540 defined in the early days of the Web, when some Web servers did not support the corresponding header
 fields. The use of these parameters is discouraged.

8.7.1 Content Type

544 The Accept header field should be used rather than the Content Type parameter.

This parameter specifies a list of Media Type(s) for the response that are acceptable to the User-Agent.

546 **Table 8.7-1. Acceptable Content Type**

	Name	Usage	VR
URI	contentType	O	ST
WS	ContentTypeList	R	ST

548 *URI Service*: the value shall be a list of Media Types, in preference order, separated by a "," character.

550 *WS Service*: the value is one or more Content Type elements each of which has one media type
 attribute. This parameter shall be present for the *RetrieveRenderedImagingDocumentSet*
 transaction, but shall not be present in the other WS transactions.

552 Note: The recommended method of specifying the media types is to use the HTTP Accept header field
 554 (See section X.Y above). When this parameter is absent, the media type returned will be the default
 media type for the object(s) being returned.

8.7.2 Character Set

556 The Accept-Charset header field should be used rather than the Character Set parameter.

558 This parameter specifies a list of character sets for the response that are acceptable to the Client
application. The character set names are defined in the IANA Character Set Registry, which can be found
560 at <http://www.iana.org/assignments/character-sets/character-sets.xhtml>.

Table 8.7-2. Character Set

	Name	Usage	VR
URI	charset	O	ST
WS	CharsetList	R	ST

562

The Web Server may or may not support character set conversion; however, if character set conversion is
564 supported:

- 566 • text based DICOM objects retrieved with a non-DICOM media type (e.g., text/plain) may be
returned in the requested character set (converted if necessary)
- 568 • DICOM objects retrieved with a DICOM media type have all contained strings returned in the
requested character set (converted if necessary) and with the Specific Character Set (0008,0005)
attribute updated (if necessary)

570

Notes: 1. The IANA Character Set registrations specify names and multiple aliases for most character sets. The
572 standard value for use in URI 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
574 2. The table in Annex D provides an informative mapping of some IANA values to DICOM Specific
Character Set Defined Terms.
576 3. In HTTP/1.1, the acceptable character sets are specified in the Accept-Charset header field of the
request message. If this field is present, the value of the *charset* query parameter, if any, of the request
578 shall be one of the values specified in the Accept-Charset header field. It is recommended that this
parameter no longer be used.

580 *URI Service:* the value shall be a list of character sets, in preference order, separated by a ","
character, as specified in IETF RFC 3986. This parameter is OPTIONAL for URI Service.

582 *WS Service:* the value is a *CharsetList* containing one or more Charset elements. This parameter
shall be present for the *RetrieveRenderedImagingDocumentSet* transaction, but shall not be
584 present in the other WS transactions.

586 **8.7.3 Transfer Syntax UID**

The Transfer-Syntax parameter for DICOM media types should be used rather than the Transfer Syntax
588 parameter.

This parameter specifies the Transfer Syntax of the returned DICOM objects, as specified in PS 3.5. It
590 shall only be present for DICOM media types (see X.Y). The value shall have a VR of Unique Identifier
(UID). It is OPTIONAL.

592

Table 8.7-3. Transfer Syntax UID

	Name	Usage	VR
URI	transferSyntax	O	UI
WS	TransferSyntaxUIDList	O	UI

594 The objects returned in the response shall be in the requested Transfer Syntax, if possible. If it is not
possible, then the default transfer syntax for Web Services, Explicit VR Little Endian, shall be used.
596 *Transfer syntaxes using Implicit VR, or Big Endian shall not be used for Web Services.*

Notes: 1. The transfer syntax can be chosen as one of the values of *TransferSyntaxUID* corresponding to JPIP,
598 in which case the returned objects will contain the URI of the JPIP session to launch for retrieving the
corresponding image.
600 2. The preferred method of specifying Transfer Syntax is to use the media type parameter "transfer-
syntax". The Transfer-Syntax UID parameter should no longer be used.
602