

DICOM Correction Proposal

STATUS	Letter Ballot
Date of Last Update	2016/01/18
Person Assigned	Jim Philbin (james.philbin@jhmi.edu)
Submitter Name	Jim Philbin (james.philbin@jhmi.edu)
Submission Date	2015/09/13

Correction Number	CP1509
Log Summary:	Clarify DICOM media types
Name of Standard	PS3.18
Rationale for Correction:	<p>The definitions of DICOM media types need to be clarified.</p> <p>PS3.18 currently refers to DICOM media types having transfer syntax parameters, but only RS services discusses them.</p> <p>An important design principle of HTTP is that media types are independent of the messages that carry them. This, in turn, means that the transfer syntax parameter should be defined with the DICOM media type specification and should apply to all DICOM media types.</p> <p>This CP defines the syntax of DICOM media types and their parameters. It consolidates the information related to DICOM media types into one section.</p> <p>It also adds a new media type 'image/dicom' which has a default transfer syntax of EVRLE (1.2.840.10008.1.2.1).</p>
Correction Wording:	

5 *Insert the following in PS3.18, Section 6.1.1.7*

6.1.1.7 Selected Media Type

The Selected Media Type is the media type selected by the origin server for the response payload. The media types in the <accept> query parameter and the media ranges in the Accept header field shall each be separately prioritized according to the rules defined in [RFC7231, Section 5.3.1].

10 For multipart payloads the Selected Media Type is determined independently for each message part in the response.

The Selected Media Type is chosen as follows:

...

Insert the following in PS3.18, after Section 6.1.1.7

15 6.1.1.8 DICOM Media Types

This section defines the media types used to represent DICOM Instances, and it also defines:

- The media types that can be used for representations of images and video contained in Instances.
- The syntax of DICOM media types including their transfer syntax and character set parameters.
- 20 • The query parameter for transfer syntax.
- The meaning of Acceptable Transfer Syntaxes and Selected Transfer Syntax.

- The media types supported by each service.

Each DICOM media type defines a representation for DICOM resources. All DICOM resources can have a representation in any of the DICOM media types. The media types may be single part or multipart, and may have optional transfer syntax and/or character set parameters.

Table 6.1.1.8-1 specifies the media types used to encode representations of DICOM Instances. These media types have default encodings for images and video data elements contained in the Instances.

6.1.1.8-1: Media Types for DICOM Instances

Media Type	Descriptions
application/dicom	Encodes Composite SOP Instances in the DICOM File Format defined in PS3.10, Section 7.
application/dicom+xml	Encodes Composite SOP Instances as XML Infosets defined in the Native Dicom Model defined in PS3.19.
application/json	Encodes Composite SOP Instances in the JSON format defined in Annex F.
application/octet-stream	Encodes a Bulkdata object as defined in PS3.19.
image/* video/*	Encodes bulkdata items as a sequence of 8-bit octets (bytes). It is used to encode uncompressed bulkdata values.

If no <transfer-syntax> parameter is specified for the above media types then the default transfer syntax image/dicom (aka EVRLE with uid = 1.2.840.10008.1.2.1)

Table 6.1.1.8-2 specifies, by Resource Category (see Section 6.1.1.2), the various Image and Video Media Types, along with the default and allowed transfer syntax UIDs for each media type. These media types can be used to retrieve image or video bulkdata in a specific transfer syntax.

All services that support SOP Classes that contain images shall support the 'image/dicom' media type. All other media types in Table 6.1.1.8-2 are optional for both the user agent and the origin server.

Table 6.1.1.8-2: Media Types and Related Transfer Syntax UIDs for DICOM Image Instances

Resource Category	Media Type	Transfer Syntax UID	Transfer Syntax Name
Single Frame Image	image/dicom	1.2.840.10008.1.2.1 (default)	Explicit VR Little Endian
	image/dicom+jpeg	1.2.840.10008.1.2.4.70 (default)	JPEG Lossless, Non-Hierarchical, First-Order Prediction (Process 14 [Selection Value 1]): Default Transfer Syntax for Lossless JPEG Image Compression
		1.2.840.10008.1.2.4.50	JPEG Baseline (Process 1): Default Transfer Syntax for Lossy JPEG 8 Bit Image Compression
		1.2.840.10008.1.2.4.51	JPEG Extended (Process 2 & 4): Default Transfer Syntax for Lossy JPEG 12 Bit Image Compression (Process 4 only)
		1.2.840.10008.1.2.4.57	JPEG Lossless, Non-Hierarchical (Process 14)
	image/dicom+rle	1.2.840.10008.1.2.5 (default)	RLE Lossless
	image/dicom+jpeg-ls	1.2.840.10008.1.2.4.80 (default)	JPEG-LS Lossless Image Compression
		1.2.840.10008.1.2.4.81	JPEG-LS Lossy (Near-Lossless) Image Compression
	image/dicom+jp2	1.2.840.10008.1.2.4.90 (default)	JPEG 2000 Image Compression (Lossless Only)
		1.2.840.10008.1.2.4.91	JPEG 2000 Image Compression
image/dicom+jpx	1.2.840.10008.1.2.4.92 (default)	JPEG 2000 Part 2 Multi-component Image Compression (Lossless Only)	

		1.2.840.10008.1.2.4.91	JPEG 2000 Image Compression
		1.2.840.10008.1.2.4.93	JPEG 2000 Part 2 Multi-component Image Compression
Multi-Frame Image	image/dicom+jpx	1.2.840.10008.1.2.4.92 (default)	JPEG 2000 Part 2 Multi-component Image Compression (Lossless Only)
		1.2.840.10008.1.2.4.91	JPEG 2000 Image Compression
		1.2.840.10008.1.2.4.93	JPEG 2000 Part 2 Multi-component Image Compression
Video	video/dicom+mpeg	1.2.840.10008.1.2.4.100	MPEG2 Main Profile @ Main Level
		1.2.840.10008.1.2.4.101 (default)	MPEG2 Main Profile @ High Level
	video/dicom+mp4	1.2.840.10008.1.2.4.102 (default)	MPEG-4 AVC/H.264 High Profile / Level 4.1
		1.2.840.10008.1.2.4.103	MPEG-4 AVC/H.264 BD-compatible High Profile / Level 4.1
		1.2.840.10008.1.2.4.104	MPEG-4 AVC/H.264 High Profile / Level 4.2 For 2D Video
		1.2.840.10008.1.2.4.105	MPEG-4 AVC/H.264 High Profile / Level 4.2 For 3D Video
		1.2.840.10008.1.2.4.106	MPEG-4 AVC/H.264 Stereo High Profile / Level 4.2

The Implicit VR Little Endian (1.2.840.10008.1.2), and Explicit VR Big Endian (1.2.840.10008.1.2.2) transfer syntaxes shall not be used with Web Services.

- 40 If a DICOM media type does not specify a transfer syntax, it defaults to Explicit VR Little Endian (1.2.840.10008.1.2.1), which is uncompressed.

The origin server may support additional Transfer Syntaxes.

Note

- 45 1. For the media type 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.
2. The target resource on the origin server may have been encoded in the Deflated Explicit VR Little Endian (1.2.840.10008.1.2.1.99) transfer syntax. If so, the origin server should inflate it, and then convert it into an Acceptable Transfer Syntax.

6.1.1.8.2 DICOM Media Type Syntax

50 The syntax of DICOM media types is:

```
dicom-media-type = (dcm-singlepart / dcm-multipart) [dcm-parameters]
```

Where

```

dcm-singlepart = dcm-mt-name
dcm-multipart           ; see Section 6.1.1.8.2.1.
dcm-parameters = transfer-syntax-parameters ; see Section 6.1.1.8.2.2.
                  / charset-parameters     ; see Section 6.1.1.8.2.3.
dcm-mt-name         = dicom / dicom-xml / json ; DICOM media type name
dicom                = "application/dicom"
dicom-xml            = "application/dicom-xml"
60 json              = "application/json"
```

All DICOM media types may have transfer syntax and character set parameters.

6.1.1.8.2.1 DICOM Multipart Media Types

The syntax of multipart media types is:

```

65 dcm-multipart = "multipart/related"
      OWS ";" OWS "type" "=" dcm-mp-mt-name
      OWS ";" OWS "boundary=" boundary
```

[dcm-parameters]
[related-parameters]

Where

70 dcm-mp-mt-name = dicom / dicom-xml / json / octet-stream

See Section 6.1.1.1 for the definition of <boundary> and <multipart-related>, and see Section 6.1.1.8.2 for the definition of <dcm-parameters>.

Each multipart media type shall include a “type” parameter that defines the media type of the parts, and shall also include a “boundary” parameter that specifies the <boundary> string that is used to separate the parts.

6.1.1.8.2.2 Transfer Syntax Parameters

All DICOM media types may have transfer syntax parameters. Support for <transfer-syntax-parameters> is optional for URI and WS Services. It is required for RS services. Origin servers that support transfer syntax parameters shall specify them in their conformance statement.

80 The syntax is:

```
transfer-syntax-parameters = ts-any-parameter / *ts-uid-parameter
ts-any-parameter           = OWS “;” OWS %s“transfer-syntax” “=” “*”
ts-uid-parameter          = OWS “;” OWS $s“transfer-syntax=” transfer-syntax-uid)
transfer-syntax-uid       ; a UID from PS3.6 Table A-1 with a UID Type of Transfer Syntax
```

85 The value of the transfer syntax parameter may be either a transfer syntax UID from Table 6.1.1.8-2 or the token “*”.

There may be zero or more transfer syntax parameters with a value that is a UID. If there is more than one transfer syntax parameter, they are prioritized from left to right.

For example, to specify that either 1.2.840.10008.1.2.4.50 or 1.2.840.10008.1.2.4.57 are acceptable transfer syntaxes, an Accept header field could be:

90 Accept: application/dicom; transfer-syntax=1.2.840.10008.1.2.4.50; transfer-syntax=1.2.840.10008.1.2.4.57

The token “*” indicates that the user agent will accept any transfer syntax, which allows the origin server to respond without needing to transcode an existing representation to a new transfer syntax. If a media type has a transfer syntax parameter with value “*”, there shall be no other transfer syntax parameters present.

6.1.1.8.2.3 Character Set Parameters

DICOM media type character set parameters are used to specify Acceptable Character Sets for the response. A DICOM media type may have zero or more character set parameters. If there is more than one “charset” parameter, they are prioritized from left to right.

100 The syntax is:

```
charset-parameters = *(OWS “;” OWS %s“charset” “=” charset)
```

All DICOM media types shall have a Default Character Set of UTF-8.

See Section 6.1.2 for character set details.

6.1.1.8.4 Transfer Syntax Query Parameter

105 The <transfer-syntax> query parameter specifies a comma-separated list of one or more Transfer Syntax UIDs, as defined in PS3.6. It is optional.

The syntax is:

```
transfer-syntax = ts-parameter-name “=” (“*” / 1#transfer-syntax-uid)
ts-parameter-name = %s quoted-string
```

110 The URI service defines the <ts-parameter-name> to be “transferSyntax”, which is case-sensitive.

The WS and RS services use the transfer syntax parameters in the media type query parameter (see 6.1.1.8.2.2) and <ts-parameter-name> is not used.

6.1.1.8.5 Acceptable Transfer Syntaxes

Each media type in the Acceptable Media Types has an associated set of Acceptable Transfer Syntaxes.

115 The Acceptable Transfer Syntaxes for a media type can be specified in any of the following ways:

1. The "transfer-syntax" parameters to a media type contained in the <accept> query parameter.
2. The value(s) contained in the <transfer-syntax> query parameter (see Section 6.1.1.8.4)
3. The "transfer-syntax" parameters to a media type contained in the Accept header field.
4. The default transfer syntax for the Selected Media Type.

120 6.1.1.8.6 Selected Transfer Syntax

The Selected Transfer Syntax is the transfer syntax selected by the origin server to encode a single message part in the response. The Selected Transfer Syntax has the following characteristics:

- The Selected Transfer Syntax shall be one of transfer syntaxes supported by the Selected Media Type.
- 125 • Both transfer syntax media type parameters and transfer syntax query parameters are prioritized from left to right.
- The origin server shall first determine the Selected Media Type as defined in Section 6.1.1.X, and then determine the Selected Transfer Syntax:

130 If the Select Media Type was contained in the <accept> query parameter, then the Selected Transfer Syntax is determined as follows:

1. Select the first supported transfer syntax parameter of the Selected Media Type, if any;
2. Otherwise, select the first supported transfer syntax in the <transfer-syntax> query parameter value for the Selected Media Type, if any;
3. Otherwise select the default transfer syntax for the Selected Media Type

135 If the Selected Media Type was contained in the Accept header field, then the Selected Transfer Syntax is determined as follows:

1. Select the first supported transfer syntax parameter for the Selected Media Type;
2. Otherwise, select the default transfer syntax for the Selected Media Type.

Note

- 140 1. The Selected Transfer Syntax may be different for each message part contained in a response.
2. Implementers may use a different selection algorithm as long as the result is the same.

6.1.1.8.7 Support for DICOM Media Types by Service

The URI, WS, and RS APIs support the following media types:

145 uri-media-type = dicom [dcm-parameters]
 ws-media-type = dicom-xml [dcm-parameters]
 rs-media-types = (multipart-dicom / multipart-dicom-xml / json) [dcm-parameters]

Support for the "transfer-syntax" and "charset" parameters is optional for URI and WS Services. It is required for RS Services.

150

Insert PS3.18 Section 6.2.2.1 as follows:

6.2.2.1 DICOM Media Type

The WADO-URI service supports the following media types:

 uri-media-type = dicom [dcm-parameters] ; See Section 6.1.1.8.2

155 Support for the "transfer-syntax" and "charset" parameters is optional for the WADO-URI Service.

Update PS3.18 Section 6.3.1.3 as follows:

6.3.1.3 Transfer Syntax

160 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".

Note

This implies that retrieved images are sent uncompressed by default.

165 If the Selected Media Type is a DICOM media type, the representations in the response shall be encoded using the Selected Transfer Syntax. See Section 6.1.1.8.6.

Insert PS3.18 Section 6.4.5 as follows:

6.4.5 DICOM Media Type

The WADO-WS service supports the following media types:

170 ws-media-type = dicom+xml [dcm-parameters] ; See Section 6.1.1.8.2

Support for the "transfer-syntax" and "charset" parameters is optional for the WADO-WS Service.

Update PS3.18 Section 6.5 as follows:

...

175 Compressed pixel data shall be encoded using the following Media Type media types and transfer syntaxes specified in Table 6.1.1.8-2. 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.

Note

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

180 **Table 6.5-1. Media Type Mapping to Transfer Syntax**

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+rlc

DICOM Transfer Syntax UID	Media Type and Parameters
1.2.840.10008.1.2.5	image/dicom+rlc; 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
1.2.840.10008.1.2.4.84	image/dicom+jpeg-ls; transfer-syntax=1.2.840.10008.1.2.4.84
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

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.

185 ...

Append the following at the end of PS3.18 Section 6.5:

The RS Services support the following media types:

rs-media-types = (multipart-dicom / multipart-dicom+xml / json) [dcm-parameters]

Support for the "transfer-syntax" and "charset" parameters is required.

190

Update PS3.18 Section 6.5.1.1 as follows:

6.5.1.1 Request

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

- Resource
 - 195 • {SERVICE}/studies/{StudyInstanceUID}, where
 - {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.
 - Method
 - 200 • GET
 - Headers
 - 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:
 - multipart/related; type=application/dicom; [transfer-syntax={TransferSyntaxUID}]
 - 205 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.
 - multipart/related; type=application/octet-stream
 - Specifies that the response can be Little Endian uncompressed bulk data.
 - multipart/related; type={Image-media-type}
 - 210 Specifies that the response can be pixel data encoded using ~~a {MediaType} listed in Table 6.5-1 (including parameters).~~ the media types and transfer syntaxes specified in Table 6.1.1.8-2.
- ...

Update PS3.18 Section 6.5.2.1 as follows:

6.5.2.1 Request

- 215 ...
 - multipart/related; type={MediaType}
 - ~~Specifies that the response can be pixel data encoded using a {MediaType} listed in Table 6.5-1 (including parameters).~~
 - 220 Specifies that the response can be pixel data encoded using the media types and transfer syntaxes specified in Table 6.1.1.8-2.

6.5.2.2 Response

Update PS3.18 Section 6.5.3.1 as follows:

6.5.3.1 Request

- 225 ...
 - multipart/related; type={MediaType}
 - ~~Specifies that the response can be pixel data encoded using a {MediaType} listed in Table 6.5-1 (including parameters).~~

230 Specifies that the response can be pixel data encoded using the media types and transfer syntaxes specified in Table 6.1.1.8-2.

6.5.3.2 Response

Update PS3.18 Section 6.5.4.1 as follows:

6.5.4.1 Request

235 ...

- multipart/related; type={MediaType}

~~Specifies that the response can be pixel data encoded using a {MediaType} listed in Table 6.5-4 (including parameters).~~

240 Specifies that the response can be pixel data encoded using the media types and transfer syntaxes specified in Table 6.1.1.8-2.

6.5.4.2 Response

Update PS3.18 Section 6.5.5.1 as follows:

6.5.5.1 Request

245 ...

- multipart/related; type={MediaType}

~~Specifies that the response can be pixel data encoded using a {Image media type} listed in Table 6.5-4 (including parameters).~~

Specifies that the response can be pixel data encoded using the media types and transfer syntaxes specified in Table 6.1.1.8-2.

6.5.5.2 Response

Update PS3.18 Section 6.6 as follows:

6.6 STOW-RS Request/Response

The STOW-RS Service defines one action type. An implementation shall support the following action type:

1. Store Instances

255 This action creates new resources for the given SOP Instances on the Server or appends to existing resources on the Server.

All request messages are HTTP/1.1 multipart messages. The organization of SOP Instances into message parts depends on whether the SOP Instances are structured as PS3.10 binary instances, or metadata and bulk data.

PS3.10 binary instances shall be encoded with one message part per DICOM Instance.

260 Metadata and bulk data requests will be encoded in the following manner: (see Figure 6.5-1 Mapping between IOD and HTTP message parts):

- All XML request messages shall be encoded as described in the Native DICOM Model defined in PS3.19 with one message part per XML object.
- All JSON requests shall be encoded as an array of DICOM JSON Model Objects defined in Annex F.
- 265 • 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 shall be encoded in one of two ways:

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

270 Compressed pixel data shall be encoded ~~using the Media Types as described in Table 6.5-1 WADO-RS Media Type Mapping to Transfer Syntax UID~~ **using the media types and transfer syntaxes specified in Table 6.1.1.8-2.**

. Media Types corresponding to several DICOM Transfer Syntax UIDs may require a transfer-syntax parameter to disambiguate the request.

275 HTTP Request field Content-Type is used in the header lines by the client in an HTTP/1.1 transaction to indicate the type of data being sent to the Service. All lines are RFC822 or RFC7230 format headers. All HTTP header fields whose use is not defined by STOW-RS shall have the meaning defined by the HTTP standard.

The Service is required to support uncompressed bulk and pixel data (multipart/related; type= application/octet-stream).

280