

DICOM Correction Proposal

STATUS	Final Text
Date of Last Update	2015/11/16
Person Assigned	Jim Philbin (james.philbin@jhmi.edu)
Submitter Name	Jonathan Whitby (jwhitby@vitalimages.com)
Submission Date	2013/11/06

Correction Number	CP1364
Log Summary:	Minor corrections to WADO-RS and STOW-RS
Name of Standard	PS3.6, 3.16, PS3.18, PS3.19 2015c
Rationale for Correction:	A few minor issues with the STOW-RS supplement were identified by Mirada.
Correction Wording:	
Update PS 3.18 Section 6.5.6 as indicated	

6.5.6 WADO_RS – RetrieveMetadata

This action retrieves the DICOM instances presented as the study, series, or instance metadata with the bulk data removed. The response is metadata for the DICOM attributes.

The study, series, or instance metadata includes all attributes; however, a RESTful Service is permitted to replace the Value Field of an attribute with a BulkDataURL for attributes with Value Representations (VR) of **DS**, FL, FD, IS, LT, OB, OD, OF, OW, SL, SS, ST, UL, UN, US, and UT. The client can use the BulkDataURL with the RetrieveBulkData action to retrieve the original Value Field of that attribute.

Note

1. The server is not required to replace any attribute with a BulkDataURL; this is intended to allow the server to provide clients with metadata of a reasonably small size by leaving out large data Value Fields.
2. Attributes with binary Value Fields are encoded as XML Base64 binary values.
3. Some DICOM instances, such as SR documents, may be entirely described in the metadata.

Update 6.6.1.1.2 XML Metadata and Bulk Data Request Message Body as indicated

6.6.1.1.2 XML Metadata and Bulk Data Request Message Body

The XML Metadata and Bulk Data Request Message has a multipart body.

- Content-Type:
 - multipart/related; type=application/dicom+xml; boundary={MessageBoundary}

- The multipart request body contains all the metadata and bulk data to be stored. If the number of bulk data parts does not correspond to the number of unique BulkDataURIs in the metadata then the entire message is invalid and will generate an error status line.
- Each body part is either DICOM PS3.19 XML metadata or a bulk data item from a SOP Instance sent as part of the Store operation. The first part of the multipart message must be XML metadata.
- **Each bulk data item must be preceded by all metadata items that contain a reference to it.**

Note

This requires that all bulk data items for an instance must be preceded by the XML metadata for that instance and if a bulk data item is included in multiple instances it must be preceded by the XML metadata for each instance in which it is included.

- The first part in the multipart request will contain the following HTTP headers:
 - Content-Type: application/dicom+xml; transfer-syntax={TransferSyntaxUID}
- Subsequent items will contain the following HTTP headers (order is not guaranteed):
 - additional metadata with the following headers:
 - Content-Type: application/dicom+xml; transfer-syntax={TransferSyntaxUID}
 - an uncompressed bulk data element encoded in Little Endian binary format with the following headers:
 - Content-Type: application/octet-stream
 - Content-Location: {BulkDataURI}
 - a compressed pixel data object from a SOP Instance in the Study with the following headers:
 - Content-Type: {MediaType}
 - Content-Location: {BulkDataURI}
- Metadata and its associated bulk data shall always be sent in the same POST request.

Note

It is not intended that metadata and bulk data be stored separately in multiple POST requests since the service always requires the metadata for context.

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

Update 6.6.1.1.3 JSON Metadata and Bulk Data Request Message Body as indicated

6.6.1.1.3 JSON Metadata and Bulk Data Request Message Body

The JSON Metadata and Bulk Data Request Message has a multipart body.

- Content-Type:
 - multipart/related; type=application/json; boundary={MessageBoundary}
- The multipart request body contains all the metadata and bulk data to be stored. If the number of bulk data parts does not correspond to the number of unique BulkDataURIs in the metadata then the entire message is invalid and will generate an error status line.

- The first part in the multipart request will contain a JSON array of DICOM JSON Model Objects (defined in Annex F). Each array element is the metadata from a SOP Instance sent as part of the Store operation. This message part will have the following headers:
 - Content-Type: application/json; transfer-syntax={TransferSyntaxUID}
Where {TransferSyntaxUID} is the UID of the DICOM Transfer Syntax used to encode the inline binary data in the XML metadata.
- Subsequent items will be one of the following:
 - an uncompressed bulk data element encoded in Little Endian binary format with the following headers:
 - Content-Type: application/octet-stream
 - Content-Location: {BulkDataURI}
 - a compressed pixel data object from a SOP Instance in the Study with the following headers:
 - Content-Type: {MediaType}
 - Content-Location: {BulkDataURI}
- JSON Metadata and its associated bulk data shall always be sent in the same POST request.

Note

It is not intended that metadata and bulk data be stored separately in multiple POST requests since the service always requires the metadata for context.

Update PS 3.18 Section 6.6.1.3.2 Response Message Body as indicated.

6.6.1.3.2 Response Message Body

The message body shall provide appropriate status codes for individual SOP Instances indicating success, warning, or failure as defined below.

The message body may also include details about the processing of attributes by the service.

The message body shall also include details of failures that are not associated with a specific SOP Instance.

Table 6.6.1-2 defines the Attributes for referencing SOP Instances that are contained in a Store Instances Response Module in the response message body.

Table 6.6.1-2. Store Instances Response Module Attributes

Attribute Name	Tag	Type	Attribute Description
Retrieve URL	(0008,1190)	2	The URL where the Study is available for retrieval via a WADO-RS Retrieve Study service. Note The VR of this attribute has changed from UT to UR.
Failed SOP Sequence	(0008,1198)	1C	A sequence of Items where each Item references a single SOP Instance for which storage could not be provided.

Attribute Name	Tag	Type	Attribute Description
			Required if one or more SOP Instances failed to store.
<i>>Table 10-11 "SOP Instance Reference Macro Attributes" in PS3.3</i>			
>Failure Reason	(0008,1197)	1	The reason that storage could not be provided for this SOP Instance. See Section 6.6.1.3.2.1.2.
Referenced SOP Sequence	(0008,1199)	1C	A sequence of Items where each Item references a single SOP Instance that was successfully stored. Required if one or more SOP Instances were successfully stored.
<i>>Table 10-11 "SOP Instance Reference Macro Attributes" in PS3.3</i>			
>Retrieve URL	(0008,1190)	2	The URL where the SOP Instance is available for retrieval via a WADO-RS service. Note The VR of this attribute has changed from UT to UR.
>Warning Reason	(0008,1196)	1C	The reason that this SOP Instance was accepted with warnings. Required if there was a warning for this SOP Instance. See Section 6.6.1.3.2.1.1.
>Original Attributes Sequence	(0400,0561)	3	Sequence of Items containing all attributes that were removed or replaced by other values. One or more Items are permitted in this sequence.
>>Attribute Modification DateTime	(0400,0562)	1	Date and time the attributes were removed and/or replaced.
>>Modifying System	(0400,0563)	1	Identification of the system that removed and/or replaced the attributes.
>>Reason for the Attribute Modification	(0400,0565)	1	Reason for the attribute modification. Defined terms are: COERCE = Replace values of attributes such as Patient Name, ID, Accession Number, for example, during import of media from an external institution, or reconciliation against a master patient index. CORRECT = Replace incorrect values, such as Patient Name or ID, for example, when incorrect worklist item was chosen or operator input error.
>>Modified Attributes Sequence	(0400,0550)	1	Sequence that contains all the Attributes, with their previous values, that were modified or removed from the main data set. Only a single Item shall be included in this sequence.

Attribute Name	Tag	Type	Attribute Description
<i>>>Any Attribute from the main data set that was modified or removed; may include Sequence Attributes and their Items.</i>			
Other Failures Sequence	(0008,119A)	1C	Reasons not associated with a specific SOP Instance that storage could not be provided. Each Item references a single storage failure. Required if there are one or more failures not associated with a specific SOP Instance.
>Failure Reason	(0008,1197)	1	The reason that storage could not be provided for this message item. See Section 6.6.1.3.2.1.2.

Update PS 3.18 Section 6.6.1.3.2.2 Response Message Body Example as indicated.

6.6.1.3.2.2 Response Message Body Example

The following is an example of a PS3.18 XML Store Instances Response Module in the response message body containing 2 failed SOP Instances, 1 successful SOP Instance, and 1 accepted SOP Instance with a warning:

```
<?xml version="1.0" encoding="utf-8" xml:space="preserve" ?>
<NativeDicomModel xmlns="http://dicom.nema.org/PS3.19/models/NativeDICOM"
...

```

Update F.2.2 DICOM Data JSON Model Object as follows

F.2.2 DICOM JSON Model Object Structure

The DICOM JSON Model object is a representation of a DICOM Data Set.

The internal structure of the DICOM JSON Model object is a sequence of objects representing attributes within the DICOM Data Set.

Attribute objects within a DICOM JSON Model object must be ordered by their property name in ascending order.

Group Length (gggg,0000) attributes shall not be included in a DICOM JSON Model object.

The name of each attribute object is:

- The eight character uppercase hexadecimal representation of a DICOM Tag

Each attribute object contains the following named child objects:

- vr: A string encoding the DICOM Value Representation. The mapping between DICOM Value Representations and JSON Value Representations is described in Section F.2.3.
- At most one of:
 - Value: An array containing one of:
 - The Value Field elements of a DICOM attribute with a VR other than PN, SQ, OB, OD, OF, OW, or UN (described in Section F.2.4)

The encoding of empty Value Field elements is described in Section F.2.5

- The Value Field elements of a DICOM attribute with a VR of PN. The non-empty name components of each element are encoded as a JSON strings with the following names:
 - Alphabetic
 - Ideographic
 - Phonetic
- JSON DICOM Model objects corresponding to the sequence items of an attribute with a VR of SQ

Empty sequence items are represented by empty objects

- BulkDataURI: A string encoding the WADO-RS URL of a bulk data item describing the Value Field of an enclosing Attribute with a VR of **DS**, FL, FD, IS, LT, OB, OD, OF, OW, SL, SS, ST, UL, UN, US, or UT (described in Section F.2.6)
- InlineBinary: A base64 string encoding the Value Field of an enclosing Attribute with a VR of OB, OD, OF, OW, or UN (described in Section F.2.7)

Note

1. For Private Data Elements, the group and element numbers will follow the rules specified in Section 7.8.1 in PS3.5
2. The person name representation is more closely aligned with the DICOM Data Element representation than the DICOM PS3.19 XML representation.

Update InlineBinary in PS 3.19 Table A.1.5-2 DICOM Data Set Macro as follows

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	The keyword as defined in PS 3.6. Required unless the DICOM Data Element is unknown to the host.
...			

>InlineBinary	C	1	<p>The Value Field of the enclosing Attribute encoded as base64.</p> <p>Required if the DICOM Data Element represented is:</p> <ul style="list-style-type: none"> • not zero length • the VR if the enclosing Attribute is either OB, OD, OF, OW, or UN • an XML Infoset Value or BulkData XML element is not present <p>Shall not be present otherwise.</p> <p>Note that There is a single InlineBinary 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 InlineBinary element.</p> <p>All rules (e.g. byte ordering and swapping) in DICOM PS 3.5 apply.</p> <p>Note: Implementers should in particular pay attention the PS 3.5 rules regarding the value representations of OD, OF and OW.</p>
---------------	---	---	--

Update PS 3.6 with new data element:

Table 6-1. Registry of DICOM Data Elements

Tag	Name	Keyword	VR	VM	
<u>(0008,119A)</u>	<u>Other Failures Sequence</u>	<u>OtherFailuresSequence</u>	<u>SQ</u>	<u>1</u>	