DICOM Correction Proposal

STATUS	Assigned
Date of Last Update	10 June 2025
Person Assigned	steven.nichols@gehealthcare.com
Submitter Name	Mathieu Malaterre <Mathieu.malaterre@gmail.com>
Submission Date	22 January 2025

Correction Number	CP-2518
Log Summary: Clarify use of Specific Character Set in JSON/XML	
Name of Standard PS3.18, PS3.19	

Rationale for Correction:

The DICOM Standard currently lacks clear guidance on how the attribute Specific Character Set (0008,0005) should be handled in the DICOM JSON Model and the XML Native DICOM Model. This has led to inconsistent or misleading implementations when DICOM datasets are converted into these alternate representations.

For example, a DICOM dataset encoded with ISO_IR 100 may result in the following representation in JSON:

```
"00080005": {
  "vr": "CS",
  "Value": ["ISO_IR 100"]
}
```

or

```
"00080005": {
  "vr": "CS",
  "Value": ["ISO_IR 192"]
}
```

These representations can confuse implementers and users:

- The first example preserves the original DICOM encoding, but in a JSON context, which is always UTF-8 per RFC 8259, it may appear incompatible with standard JSON processing tools.
- The second example (using ISO_IR 192) misrepresents the original dataset's encoding and may imply that the full Unicode character set was supported in the source data, which could be misleading during reconciliation or round-trip conversion back to binary DICOM.

To address this ambiguity, this CP clarifies that:

- The presence of Specific Character Set (0008,0005) in the JSON or XML model is not an indication of the character encoding used by the JSON or XML representation itself.
- Instead, it reflects the character encoding of the original DICOM dataset, preserved for metadata integrity and potential use in reconstructing or analyzing the original encoding context.
- All JSON output shall remain UTF-8 encoded (per RFC 8259), and XML representations shall follow the character set declared in the XML prolog.

WG-27:

Handling of BulkData for text VRs is ambiguous within PS3.18 and PS3.19. Should BulkData containing text VRs:

1. preserve the original DICOM character set encoding (per PS3.19 "might be necessary to determine its encoding"), or
2. convert to match the target format's encoding (per PS3.18 10.4.3.3.2 "replacement for the Value Field")

WG-27 prefers option 2 (target format encoding) to ensure consistency between inline values and BulkData content within each representation format.

Notes:

- “Text” category is in 8.7.3.3.1 Uncompressed Bulkdata Media Types
- CP-1576 proposes a “text/plain” media type in bulk data for string VR’s

Comments from JAHIS (Japanese Association of Healthcare Information Systems Industry)

- Consider a clarification in PS3.3 C.12.1.1.2 that aligns with additions in PS3.18/19.
- No major objections from Japan; they agree with the approach and say it works for real-world Japanese encodings.

- Main concern is long-term clarity and potentially reducing reliance on legacy encodings like ISO_IR 13.

Correction Wording:

Modify PS3.18, 10.4.3.3.2 Metadata Resource Payload as follows:

10.4.3.3.2 Metadata Resource Payload

The payload for a Metadata Resource shall (see Section 10.4.1.1.2) contain all Attributes in the resource. For Data Elements having a Value Representation (VR) of DS, FL, FD, IS, LT, OB, OD, OF, OL, OV, OW, SL, SS, ST, SV, UC, UL, UN, US, UT and UV, the origin server is permitted to replace the Value Field of the Data Element with a Bulkdata URI. The user agent can use the Bulkdata URI to retrieve the Bulkdata **in its original form**.

Modify PS3.18, F.2.2 DICOM JSON Model Object Structure, 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 shall be ordered by their property name in ascending lexicographic (alphabetic) 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, OL, OV, 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, OL, OV, OW, SL, SS, ST, SV, UC, UL, UN, US, UT or UV (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, OL, OV, 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.
3. **The attribute Specific Character Set (0008,0005), if present, reflects the character encoding of the original DICOM dataset (i.e., the application/dicom representation). It does not indicate the character encoding of the JSON representation, which is UTF-8, as specified in RFC 8259. This attribute is preserved as metadata from the original instance and may be used to reconstruct a binary DICOM file or analyze the original character encoding. See PS3.3 Section C.12.1.1.2 for further details.**
4. **When translating BulkData with character set dependent VRs (SH, LO, ST, PN, LT, UC or UT) to JSON representation, the encoding shall be UTF-8, as specified in RFC 8259. For both InlineBinary and BulkData with VR of UN, the original byte encoding is preserved without character set translation. If such data is interpreted as containing character string content, the Specific Character Set (0008,0005) of the original DICOM dataset is used to determine the character encoding.**
5. **The standard does not specify decoding pipeline for converting character strings to Unicode. When generating a DICOM JSON Model, the values of character string attributes shall be represented in JSON as valid UTF-8 strings (per RFC 8259). Implementations may use any decoding method, provided the resulting JSON accurately reflects the intended Unicode content.**

Kommentiert [SN1]: One caveat here is that (0008,0005) might change when IOD's are passed down a chain of DICOM systems.
If during that process the first element of (0008,0005) changes, reconstructing String values from UN attributes will fail.

Modify PS3.19, A.1 Native DICOM Model, as follows:

A.1 Native DICOM Model

A.1.1 Usage

The Native DICOM Model defines a representation of binary-encoded DICOM SOP Instances as XML Infosets that allows a recipient of data to navigate through a binary DICOM data set using XML-based tools instead of relying on tool kits that understand the binary encoding of DICOM.

Note

1. It is not the intention that this form be utilized as the basis for other uses. This form does not take advantage of the self-validation features that could be possible with a pure XML representation of the data.
2. As per the XML standard, XML tags are case sensitive. The case convention for elements is an upper case initial letter, camel case. The case convention for attributes is a lower initial letter, camel case. Keywords referenced in the XML schema are the DICOM title case from the definitions in PS3.6.

With the exception of padding to an even byte length, a data source that is creating a new instance of a Native DICOM Model (e.g., the result from some analysis application) shall follow the DICOM encoding rules (e.g., the handling of character sets) in creating Values for the DicomAttributes within the instance of the Native DICOM Model. Attribute Values encoded in a Native DICOM Model are not required to be padded to an even byte length.

Note

1. Attribute objects within a DICOM JSON Model object are ordered by their property name in ascending order (see Section F.2.2 "DICOM JSON Model Object Structure" in PS3.18). Elements within an XML Infoset following the Native DICOM Model definition are not required to be ordered.
2. The XML is not required to be in a Canonical representation (http://www.w3.org/TR/xml-c14n/).

3. **The attribute Specific Character Set (0008,0005), if present in the Native DICOM Model, reflects the character encoding of the original binary DICOM dataset (i.e., the application/dicom representation). It does not indicate the character encoding of the XML Infoset, which is determined by the Infoset encoding declaration. This attribute is preserved as metadata and may be used to reconstruct a binary DICOM object or analyze the original character encoding. See PS3.3 Section C.12.1.1.2 for further details.**

4. **For BulkData encoded in XML containing character set dependent values (e.g., VRs SH, LO, ST, PN, LT, UC or UT). For other XML encodings, the encoding follows the declared Infoset character encoding. For both InlineBinary and BulkData with VR of UN, the original byte encoding is preserved without character set translation. If such data is interpreted as containing character string content, the Specific Character Set (0008,0005) of the original DICOM dataset is used to determine the character encoding.**

5. **The Standard does not prescribe a specific method for translating character strings from the original DICOM Specific Character Set to the declared Infoset character encoding. When generating a Native DICOM Model, implementations may use any translation approach, provided the resulting XML content accurately reflects the intended character representation of the original DICOM data.**

6. **When the XML Infoset is encoded in UTF-8, the encoding is consistent with the JSON representation defined in PS3.18.**

Kommentiert [SN2]: Suggestion from WG-27. Does WG-06 agree that there should be XML/JSON consistency? Should it be included here or somewhere else?

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

A data recipient that converts data from an instance of the Native DICOM Model back into a binary encoded DICOM object shall adjust the padding to an even byte length as necessary to meet the encoding rules specified in DICOM PS3.5.

Modify PS3.19, Table A.1.5-2. DICOM Data Set Macro, as follows:

Table A.1.5-2. DICOM Data Set Macro

Name	Optionality	Cardinality	Description
…			
>BulkData	C	1	A reference to a blob of data that the recipient may retrieve through use of the GetData() method, a PS3.18 Studies Service Retrieve (WADO-RS) transaction or a PS3.18 Studies Service Store (STOW-RS) transaction. Required if the DICOM Data Element represented is not zero length and an XML Infoset Value, Item, InlineBinary or PersonName element is not present. 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, pixel data or look up tables. 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. Note 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.

Name	Optionality	Cardinality	Description
			All rules (e.g., byte ordering and swapping) in PS3.5 apply. Note**s** Implementers should pay particular attention to the PS3.5 rules regarding the value representations of OD, OF, OL, OV and OW. 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 **used by the recipient**~~**necessary**~~ to determine **the**~~**its**~~ encoding **of the retrieved data**.
…			

Add RFC8259 to 2.2 Internet Engineering Task Force (IETF) and Internet Assigned Names Authority (IANA), as follows:

2.2 Internet Engineering Task Force (IETF) and Internet Assigned Names Authority (IANA)

…

[RFC8259] IETF. December 2017. The JavaScript Object Notation (JSON) Data Interchange Format. http://tools.ietf.org/html/rfc8259 .