

1	Status	Letter Ballot
2	Date of Last Update	2018/11/11
3	Person Assigned	David Clunie
4		mailto:dclunie@dclunie.com
5	Submitter Name	Rick Busbridge
6		mailto:Rick.Busbridge@philips.com
7	Submission Date	2018/04/09

8	Correction Number CP-1819	
9	Log Summary: Add 64 bit binary VRs	
10	Name of Standard	
11	PS3.3, PS3.5	
12	Rationale for Correction:	
13	Increasingly, medical imaging applications need longer binary integers than 32 bit.	
14	Add appropriate value representations for signed, unsigned and bulk 64 bit integers. The abbreviation "V" for "Very long" is used to	
15	produce new two-letter VRs that are not already used but follow the pattern of two uppercase letters (using numbers in the VR, e.g.,	
16	as in "O8", is known to be a problem for some toolkits, and wouldn't scale in future to more than 9 bytes anyway).	
17	Apply similar changes for waveform data (and add 32 bit).	
18	This CP does not address the consequences of the VL being limited to 32 bit characters, for which new Transfer Syntaxes would	
19	be required, and for which conversion to the default Implicit VR Transfer Syntax would no longer be possible (perhaps requiring a	
20	new Application Context during Association Negotiation).	
21	Correction Wording:	

Amend DICOM PS3.3 as follows (changes to existing text are bold and underlined for additions and ~~struckthrough~~ for removals):

C.10.9.1.5 Waveform Bits Allocated and Waveform Sample Interpretation

Waveform Bits Allocated (5400,1004) specifies the number of bits allocated for each sample, and Waveform Sample Interpretation (5400,1006) specifies the data representation of each waveform sample. Waveform Bits Allocated shall be a multiple of 8. These Attributes are related, and their Defined Terms are specified in Table C.10-10.

Table C.10-10. Waveform Bits Allocated and Waveform Sample Interpretation

Waveform Bits Allocated - Defined Terms	Waveform Sample Interpretation - Defined Terms	Waveform Sample Interpretation Meaning
8	SB	signed 8 bit linear
	UB	unsigned 8 bit linear
	MB	8 bit mu-law (in accordance with ITU-T Recommendation G.711)
	AB	8 bit A-law (in accordance with ITU-T Recommendation G.711)
16	SS	signed 16 bit linear
	US	unsigned 16 bit linear
<u>32</u>	<u>SL</u>	<u>signed 32 bit linear</u>
	<u>UL</u>	<u>unsigned 32 bit linear</u>
<u>64</u>	<u>SV</u>	<u>signed 64 bit linear</u>
	<u>UV</u>	<u>unsigned 64 bit linear</u>

Note

- The set of valid values from within this table may be constrained by definition of the IOD (see ???).
- mu-law and A-law encoding is without the alternate bit inversion used for PCM transmission through the telephone network.

This representation also applies to the Channel Minimum and Maximum Data Values, and Waveform Padding Value.

Amend DICOM PS3.5 as follows (changes to existing text are bold and underlined for additions and ~~struckthrough~~ for removals):

6.2 Value Representation (VR)

...

All new VRs defined in future versions of DICOM shall be of the same Data Element Structure as defined in Section 7.1.2 (i.e., following the format for VRs such as **OV**, OB, OD, OF, OL, OW, SQ and UN).

Note

- Since all new VRs will be defined as specified in Section 7.1.2, an implementation may choose to ignore VRs not recognized by applying the rules stated in Section 7.1.2.
- When converting a Data Set from an Explicit VR Transfer Syntax to a different Transfer Syntax, an implementation may copy Data Elements with unrecognized VRs in the following manner:
 - If the endianness of the Transfer Syntaxes is the same, the Value of the Data Element may be copied unchanged and if the target Transfer Syntax is Explicit VR, the VR bytes copied unchanged. In practice this only applies to Little Endian Transfer Syntaxes, since there was only one Big Endian Transfer Syntax defined.
 - If the source Transfer Syntax is Little Endian and the target Transfer Syntax is the (retired) Big Endian Explicit VR Transfer Syntax, then the Value of the Data Element may be copied unchanged and the VR changed to UN, since

being unrecognized, whether or not byte swapping is required is unknown. If the VR were copied unchanged, the byte order of the value might or might not be incorrect.

- If the source Transfer Syntax is the (retired) Big Endian Explicit VR Transfer Syntax, then the Data Element cannot be copied, because whether or not byte swapping is required is unknown, and there is no equivalent of the UN VR to use when the value is big endian rather than little endian.

The issues of whether or not the element may be copied, and what VR to use if copying, do not arise when converting a Data Set from Implicit VR Little Endian Transfer Syntax, since the VR would not be present to be unrecognized, and if the data element VR is not known from a data dictionary, then UN would be used.

...

Table 6.2-1. DICOM Value Representations

VR Name	Definition	Character Repertoire	Length of Value
OB Other Byte	An octet-stream where the encoding of the contents is specified by the negotiated Transfer Syntax. OB is a VR that is insensitive to byte ordering (see Section 7.3). The octet-stream shall be padded with a single trailing NULL byte value (00H) when necessary to achieve even length.	not applicable	see Transfer Syntax definition
OD Other Double	A stream of 64-bit IEEE 754:1985 floating point words. OD is a VR that requires byte swapping within each 64-bit word when changing byte ordering (see Section 7.3).	not applicable	2^{32} -8 bytes maximum
OF Other Float	A stream of 32-bit IEEE 754:1985 floating point words. OF is a VR that requires byte swapping within each 32-bit word when changing byte ordering (see Section 7.3).	not applicable	2^{32} -4 bytes maximum
OL Other Long	A stream of 32-bit words where the encoding of the contents is specified by the negotiated Transfer Syntax. OL is a VR that requires byte swapping within each word when changing byte ordering (see Section 7.3).	not applicable	see Transfer Syntax definition
<u>OV</u> <u>Other 64-bit Very Long</u>	<u>A stream of 64-bit words where the encoding of the contents is specified by the negotiated Transfer Syntax. OV is a VR that requires byte swapping within each word when changing byte ordering (see Section 7.3).</u>	<u>not applicable</u>	<u>see Transfer Syntax definition</u>
OW Other Word	A stream of 16-bit words where the encoding of the contents is specified by the negotiated Transfer Syntax. OW is a VR that requires byte swapping within each word when changing byte ordering (see Section 7.3).	not applicable	see Transfer Syntax definition
<u>SV</u> <u>Signed 64-bit Very Long</u>	<u>Signed binary integer 64 bits long. Represents an integer n in the range:</u> <u>$-2^{63} \leq n \leq 2^{63}-1$.</u>	<u>not applicable</u>	<u>8 bytes fixed</u>
SL Signed Long	Signed binary integer 32 bits long in 2's complement form. Represents an integer, n, in the range: $-2^{31} \leq n \leq 2^{31}-1$.	not applicable	4 bytes fixed
SS Signed Short	Signed binary integer 16 bits long in 2's complement form. Represents an integer n in the range: $-2^{15} \leq n \leq 2^{15}-1$.	not applicable	2 bytes fixed
<u>UV</u> <u>Unsigned 64-bit Very Long</u>	<u>Unsigned binary integer 64 bits long. Represents an integer n in the range:</u> <u>$0 \leq n < 2^{64}$.</u>	<u>not applicable</u>	<u>8 bytes fixed</u>

VR Name	Definition	Character Repertoire	Length of Value
UL Unsigned Long	Unsigned binary integer 32 bits long. Represents an integer n in the range: $0 \leq n < 2^{32}$.	not applicable	4 bytes fixed
US Unsigned Short	Unsigned binary integer 16 bits long. Represents integer n in the range: $0 \leq n < 2^{16}$.	not applicable	2 bytes fixed

7.1.2 Data Element Structure with Explicit VR

When using the Explicit VR structures, the Data Element shall be constructed of four consecutive fields: Data Element Tag, VR, Value Length, and Value. Depending on the VR of the Data Element, the Data Element will be structured in one of two ways:

- for VRs of OV, OB, OD, OF, OL, OW, SQ and UN the 16 bits following the two byte VR Field are reserved for use by later versions of the DICOM Standard. These reserved bytes shall be set to 0000H and shall not be used or decoded (Table 7.1-1). The Value Length Field is a 32-bit unsigned integer. If the Value Field has an Explicit Length, then the Value Length Field shall contain a value equal to the length (in bytes) of the Value Field. Otherwise, the Value Field has an Undefined Length and a Sequence Delimitation Item marks the end of the Value Field.
- for VRs of UC, UR and UT the 16 bits following the two byte VR Field are reserved for use by later versions of the DICOM Standard. These reserved bytes shall be set to 0000H and shall not be used or decoded. The Value Length Field is a 32-bit unsigned integer. The Value Field is required to have an Explicit Length, that is the Value Length Field shall contain a value equal to the length (in bytes) of the Value Field.

Note

VRs of UC, UR and UT may not have an Undefined Length, i.e., a Value Length of FFFFFFFFH.

- for all other VRs the Value Length Field is the 16-bit unsigned integer following the two byte VR Field (Table 7.1-2). The value of the Value Length Field shall equal the length of the Value Field.

Table 7.1-1. Data Element with Explicit VR of OV, OB, OD, OF, OL, OW, SQ, UC, UR, UT or UN

Tag		VR		Value Length	Value
Group Number (16-bit unsigned integer)	Element Number (16-bit unsigned integer)	VR (2 single byte characters) of " <u>OV</u> ", "OB", "OD", "OF", "OL", "OW", "SQ", "UC", "UR", "UT" or "UN"	Reserved (2 bytes) set to a value of 0000H	32-bit unsigned integer	Even number of bytes containing the Data Element Value(s) encoded according to the VR and negotiated Transfer Syntax. Delimited with Sequence Delimitation Item if of Undefined Length.
2 bytes	2 bytes	2 bytes	2 bytes	4 bytes	'Value Length' bytes if of Explicit Length

Table 7.1-2. Data Element with Explicit VR other than as shown in Table 7.1-1

Tag		VR	Value Length	Value
Group Number (16-bit unsigned integer)	Element Number (16-bit unsigned integer)	VR (2 single byte characters)	(16-bit unsigned integer)	Even number of bytes containing the Data Element Value(s) encoded according to the VR and negotiated Transfer Syntax.
2 bytes	2 bytes	2 bytes	2 bytes	'Value Length' bytes

7.3 Little Endian Byte Ordering

All nonretired Transfer Syntaxes in DICOM require the use of Little Endian Byte Ordering.

Little Endian byte ordering is defined as follows:

- In a binary number consisting of multiple bytes (e.g., a 32-bit unsigned integer value, the Group Number, the Element Number, etc.), the least significant byte shall be encoded first; with the remaining bytes encoded in increasing order of significance.
- In a character string consisting of multiple 8-bit single byte codes, the characters will be encoded in the order of occurrence in the string (left to right).

Big Endian byte ordering was previously described but has been retired, See PS3.5 2016b.

Note

The packing of bits within values of OB or OW Value Representation for Pixel Data and Overlay Data is described in Section 8. The OL and OV Value Representations ~~is are~~ not used for Pixel Data or Overlay Data.

Byte ordering is a component of an agreed upon Transfer Syntax (see Section 10). The default DICOM Transfer Syntax, which shall be supported by all AEs, uses Little Endian encoding and is specified in Section A.1. Alternate Little Endian Transfer Syntaxes are also specified in Annex A.

Note

The Command Set structure as specified in PS3.7 is encoded using the Little Endian Implicit VR Transfer Syntax.

In the case of Little Endian encoding, Big Endian Machines interpreting Data Sets shall do 'byte swapping' before interpreting or operating on certain Data Elements. The Data Elements affected are all those having VRs that are multiple byte Values and that are not a character string of 8-bit single byte codes. VRs constructed of a string of characters of 8-bit single byte codes are really constructed of a string of individual bytes, and are therefore not affected by byte ordering. The VRs that are not a string of characters and consist of multiple bytes are:

- 2-byte US, SS, OW and each component of AT
- 4-byte OF, OL, UL, SL, and FL
- 8 byte OV, OD, FD, UV and SV

Note

For the above VRs, the multiple bytes are presented in increasing order of significance when in Little Endian format. For example, an 8-byte Data Element with VR of FD, might be written in hexadecimal as 68AF4B2CH, but encoded in Little Endian would be 2C4BAF68H.

8.3 Waveform Data and Related Data Elements

The DICOM protocol provides for the exchange of encoded time-based signals, or waveforms, encoded in the Waveform Data (5400,1010).

Note

Per ???, an IOD supporting multiple sets of Waveform Data will encapsulate Waveform Data (5400,1010) within a Sequence.

Encoded Waveform Data of various bit depths is accommodated through the Waveform Bits Allocated (5400,1004) Data Element. This element defines the size of each waveform data sample within the Waveform Data (5400,1010). Allowed values are ~~8 and~~ 16, 32 and 64 bits.

The Value Representation of the Waveform Data (5400,1010) shall be OW; OB shall be used in cases where Waveform Bits Allocated has a value of 8, but only with Transfer Syntaxes where the Value Representation is explicitly conveyed.

Note

1. Under the Default Transfer Syntax, OB and OW VRs have the identical byte transfer order.
2. Conversion of a SOP Instance from the Default Transfer Syntax to an Explicit VR Transfer Syntax (uncompressed) requires the interpretation of the Waveform Bits Allocated (5400,1004) Data Element, to determine the proper VR of the Waveform Data.

The following data elements related to Waveform Data shall be encoded with the same VR as Waveform Data: Channel Minimum Value (5400,0110), Channel Maximum Value (5400,0112) and Waveform Padding Value (5400,100A).

A Transfer Syntax Specifications (Normative)

A.1 DICOM Implicit VR Little Endian Transfer Syntax

This Transfer Syntax applies to the encoding of the entire DICOM Data Set. This implies that when a DICOM Data Set is being encoded with the DICOM Implicit VR Little Endian Transfer Syntax the following requirements shall be met:

- a. The Data Elements contained in the Data Set structure shall be encoded with Implicit VR (without a VR Field) as specified in Section 7.1.3.
- b. The encoding of the overall Data Set structure (Data Element Tags, Value Length, and Value) shall be in Little Endian as specified in Section 7.3.
- c. The encoding of the Data Elements of the Data Set shall be as follows according to their Value Representations:
 - For all Value Representations defined in this part, except for the Value Representations OB and OW, the encoding shall be in Little Endian as specified in Section 7.3.
 - For the Value Representations **OV**, OB, OL and OW, the encoding shall meet the following specification depending on the Data Element Tag:
 - Pixel Data (7FE0,0010) has the Value Representation OW and shall be encoded in Little Endian.

Note

The **OV and** OL Value Representations ~~is are~~ not used for Pixel Data, even if it has a Bits Allocated (0028,0100) of **64 or** 32, since **OV and** OL ~~was were~~ added to the standard after the encoding of Pixel Data had been established

- ...
- Waveform Data (5400,1010) shall have Value Representation OW and shall be encoded in Little Endian.
-
- ...

A.2 DICOM Little Endian Transfer Syntax (Explicit VR)

This Transfer Syntax applies to the encoding of the entire DICOM Data Set. This implies that when a DICOM Data Set is being encoded with the DICOM Little Endian Transfer Syntax the following requirements shall be met:

- a. The Data Elements contained in the Data Set structure shall be encoded with Explicit VR (with a VR Field) as specified in Section 7.1.2.
- b. The encoding of the overall Data Set structure (Data Element Tags, Value Length, and Value) shall be in Little Endian as specified in Section 7.3.
- c. The encoding of the Data Elements of the Data Set shall be as follows according to their Value Representations:

- For all Value Representations defined in this part, except for the Value Representations OB and OW, the encoding shall be in Little Endian as specified in Section 7.3.
- For the Value Representations **OV**, OB, OL and OW, the encoding shall meet the following specification depending on the Data Element Tag:
 - Pixel Data (7FE0,0010)
 - where Bits Allocated (0028,0100) has a value greater than 8 shall have Value Representation OW and shall be encoded in Little Endian;
 - where Bits Allocated (0028,0100) has a value less than or equal to 8 shall have the Value Representation OB or OW and shall be encoded in Little Endian.

Note

The **OV and OL** Value Representations ~~is are~~ not used for Pixel Data, even if it has a Bits Allocated (0028,0100) of **64 or 32**, since **OV and OL** ~~was were~~ added to the standard after the encoding of Pixel Data had been established

- ...
- Waveform Data (5400,1010) has the Value Representation specified in its Explicit VR Field. The component points shall be encoded in Little Endian.
- ...

A.3 DICOM Big Endian Transfer Syntax (Explicit VR)

This Transfer Syntax was retired in 2006. For the most recent description of it, see PS3.5 2016b.

A.4 Transfer Syntaxes For Encapsulation of Encoded Pixel Data

These Transfer Syntaxes apply to the encoding of the entire DICOM Data Set, even though the image Pixel Data (7FE0,0010) portion of the DICOM Data Set is the only portion that is encoded by an encapsulated format. These Transfer Syntaxes shall only be used when Pixel Data (7FE0,0010) is present in the top level Data Set, and hence shall not be used when Float Pixel Data (7FE0,0008) or Double Float Pixel Data (7FE0,0009) are present. This implies that when a DICOM Message is being encoded according to an encapsulation Transfer Syntax the following requirements shall be met:

1. The Data Elements contained in the Data Set structure shall be encoded with Explicit VR (with a VR Field) as specified in Section 7.1.2.
2. The encoding of the overall Data Set structure (Data Element Tags, Value Length, etc.) shall be in Little Endian as specified in Section 7.3.
3. The encoding of the Data Elements of the Data Set shall be as follows according to their Value Representations:
 - For all Value Representations defined in this part of the DICOM Standard, except for the Value Representations OB and OW, the encoding shall be in Little Endian as specified in Section 7.3.
 - For the Value Representations **OV**, OB, OL and OW, the encoding shall meet the following specification depending on the Data Element Tag:
 - Pixel Data (7FE0,0010) may be encapsulated or native.

It shall be encapsulated if present in the top-level Data Set (i.e., not nested within a Sequence Data Element).

Note

The distinction between fixed value length (native) and undefined value length (encapsulated) is present so that the top level Data Set Pixel Data can be compressed (and hence encapsulated), but the Pixel Data within an Icon Image Sequence may or may not be compressed.

1 If native, it shall have a defined Value Length, and be encoded as follows:

- 2 • where Bits Allocated (0028,0100) has a value greater than 8 shall have Value Representation OW and shall be encoded
3 in Little Endian;
- 4 • where Bits Allocated (0028,0100) has a value less than or equal to 8 shall have the Value Representation OB or OW and
5 shall be encoded in Little Endian.

6 **Note**

7 a. The OV and OL Value Representations is are not used for Pixel Data, even if it has a Bits Allocated (0028,0100)
8 of 64 or 32, since OV and OL was were added to the standard after the encoding of Pixel Data had been es-
9 tablished

10 b. That is, as if the Transfer Syntax were Explicit VR Little Endian.

11 If encapsulated,

- 12 • ...
- 13 • Waveform Data (5400,1010) has the Value Representation specified in its Explicit VR Field. The component points shall be
14 encoded in Little Endian.
- 15 • ...