DICOM Correction Proposal Form

Correction Number	CP-246
Log Summary: Undefined length UN VR value is always implicit VR	

Type of Modification	Name of Standard
Clarification	PS 3.5 2000

Rationale for Correction
The UN VR only needs to ever be used when receiving a dataset in implicit VR little endian (IVRLE) transfer syntax. For all other transfer syntaxes the VR is explicit and therefore UN is not needed. If one were to receive, in IVRLE, a sequence with a defined length, then one would have no way of knowing it was a sequence, and would copy the bytes verbatim to the output and designate the VR as UN. The contents of this value would therefore be "a sequence in IVRLE". There is no way to avoid this. If on the other hand, the value length was undefined, one has to parse it as a sequence, recursively, down to either the data element level or to the level at which a defined length item is found (e.g. as would be the case if it were a encapsulated pixel data in an OB rather than an SQ). In either case one has to make a decision as to whether to regurgitate it with a) the lengths unchanged or not, and b) in explicit or implicit VR. Since: - there is no way of signaling in the output UN value whether it is encoded explicit or implicit VR - in the defined length sequence case it will always be implicit VR - the UN VR only exists to convert IVRLE streams to explicit the only reliable choice is to always encode the value of undefined length VR values received as IVRLE, with implicit VR, not explicit VR. This interpretation is consistent with the existing statement in the standard that "the Unknown (UN) VR shall only be used for … Data Elements previously encoded … using the DICOM Default Transfer Syntax (Implicit VR Little Endian),"

Sections of documents affected
PS 3.5 6.2.2

| Correction Wording: |

6.2.2 Unknown (UN) Value Representation

The Unknown (UN) VR shall only be used for Private Attribute Data Elements and Standard Data Elements previously encoded as some DICOM VR other than UN using the DICOM Default Transfer Syntax (Implicit VR Little Endian), and whose Value Representation is currently unknown. As long as the VR is unknown the Value Field is insensitive to Little/Big Endian byte ordering and shall not be 'byte-swapped' (see section 7.3). **In the case of undefined length sequences, the value shall remain in implicit VR form**. See section 7.8 for a description of Private Data Attribute Elements and section 10 and Annex A for a discussion of Transfer Syntaxes.

The UN VR shall not be used for Private Creator Data Elements (i.e. the VR is equal to LO, see section 7.8.1).

Notes: 1. All other (non-default) DICOM Transfer Syntaxes employ explicit VR in their encoding, and therefore any Private and/or Standard Data Element Value Field Attribute value encoded and decoded using any Transfer Syntax other than the default, and not having been translated to the DICOM Default Transfer Syntax default in the interim, will have a known VR.

2. If at some point an application knows the actual VR for an Attribute of VR UN (e.g. has its own applicable data dictionary), it can assume that the Value Field of the Attribute is encoded in Little Endian byte ordering **with implicit VR encoding**, irrespective of the current Transfer Syntax.

3. This VR of UN is needed when an explicit VR must be given to a Data Element whose Value Representation is unknown (e.g. store and forward). UN is a means to explicitly indicate that the Value Representation of a Data Element is unknown.

4. The length field of the Value Representation of UN may contain the value of "unknown length"**, in which case the contents can be assumed to be encoded with implicit VR**. See section 7.5.1 to determine how to parse Data Elements with an unknown length.

5. An example of a Standard Data Element using a UN VR is a Type 3 or Type U Standard Attribute added to an SOP Class definition. An existing application which does not support that new Attribute (and encounters it) could convert the VR to UN.