DCMTK - Feature #1014

Silently remove illegal zero pad bytes when dcmEnableAutomaticInputDataCorrection is activated–

2022-01-12 17:47 - Marco Eichelberg

Status:	Closed	**Start date:**	2022-01-12
Priority:	Normal	**Due date:**	
Assignee:	Marco Eichelberg	**% Done:**	100%
Category:		**Estimated time:**	1:00 hour
Target version:			
Module:	dcmdata	**Compiler:**	
Operating System:			

Description

The DICOM parser in dcmdata has a global setting named "dcmEnableAutomaticInputDataCorrection" that enables a few silent data corrections when reading DICOM files, such as replacing incorrect space paddings at the end of a UID, or removing padding characters from other text fields. The parser does not, however, silently fix illegal zero bytes used as pad bytes, as can be seen in the attached (anonymized) example file.

It should be trivial for all classes derived from DcmByteString to silently remove such trailing zero bytes, just as if they were space characters.
Things would be more complicated for DcmCharString, where trailing zero bytes might actually be part of a multi-byte character when a multi-byte character set other than UTF-8 is active.

In any case, it would be desirable to provide an automatic data correction of illegal zero bytes when dcmEnableAutomaticInputDataCorrection is on.

History

#1 - 2022-05-04 12:17 - Marco Eichelberg

Update: None of the multi-byte character sets supported by the DICOM standard use a zero byte in a multi-byte character sequence. It is, therefore, safe to remove trailing zero bytes.

#2 - 2025-02-04 15:23 - Marco Eichelberg

- Status changed from New to Closed

- Assignee set to Marco Eichelberg

- % Done changed from 0 to 100

- Estimated time set to 1:00 h

Closed by commit #31924835b.

Files

trailing_zero_bytes.dcm	74.3 KB	2022-01-12	Marco Eichelberg