

## DCMTK - Feature #853

### Make IJG JPEG library accessible to DCMTK users by installing the header files

2018-11-19 11:31 - Marco Eichelberg

|  |         |                        |            |
|--|---------|------------------------|------------|
| <b>Status:</b>   | New     | <b>Start date:</b>     | 2018-11-19 |
| <b>Priority:</b>   | Normal  | <b>Due date:</b>       |            |
| <b>Assignee:</b>   |         | <b>% Done:</b>         | 0%         |
| <b>Category:</b>   | Library | <b>Estimated time:</b> | 0:00 hour  |
| <b>Target version:</b>   |         | <b>Compiler:</b>       |            |
| <b>Module:</b>   | dcmjpeg |                        |            |
| <b>Operating System:</b>   |         |                        |            |
| <b>Description</b><br>DCMTK internally uses the Independent JPEG Group library with the lossless patch to perform JPEG encoding and decoding. Apparently some external developers would like to use this library directly, not just the DCMTK wrapper that actively hides the IJG API.<br>This would simply require us to install the IJG header files when running "make install" since the libraries get installed anyway. For this purpose, the header files should perhaps be moved into a separate directory, so that they could be used for example like this: <code>#include "dcmtdcmjpeg/ijg/jpeglib12.h"</code> |         |                        |            |

#### History

##### #1 - 2018-12-06 13:36 - Michael Onken

Here are all options originally proposed by the submitter:

- Convince you to not to bundle the library but refer to some public source.
- Convince the developers to fix the bad applications. Not going well so far.
- Convince you to to hide the symbols, break the bad applications and force them to get fixed. (Or, the worse case, they will stay with that old version.)
- Patch everything myself, maintain the forks, backport to new versions. Doable once but not sustainable long-term.
- Find a different way to replace the library.
- Convince you to publish the headers.

Regarding the wrapping options, he added:

I might be spoiled into laziness to just make thin unique wrapper for each non-unique static library, link wrapper and the library with "-Wl,-exclude-libs,ALL" and use the wrappers freely without fear of clashes.

... which would also be worth a try.

##### #2 - 2020-05-25 13:29 - Michael Onken

- Target version deleted (3.6.6)