**TIFF CHANGE INFORMATION**

**Current Version**: v4.0.5  
**Previous Version**: [v4.0.4](http://docs.google.com/v4.0.4.html)  
**Master FTP Site**:  [download.osgeo.org](ftp://download.osgeo.org/libtiff), directory pub/libtiff  
**Master HTTP Site**: <http://www.simplesystems.org/libtiff/>

This document describes the changes made to the software between the *previous* and *current* versions (see above). If you don't find something listed here, then it was not done in this timeframe, or it was not considered important enough to be mentioned. The following information is located here:

* [Major Changes](#gjdgxs)
* [Changes in the software configuration](#30j0zll)
* [Changes in libtiff](#1fob9te)
* [Changes in the tools](#3znysh7)
* [Changes in the contrib area](#2et92p0)

**MAJOR CHANGES:**

* Support for configure/build using CMake.
* Support for large (> 2GB) files under Microsoft Windows.

**CHANGES IN THE SOFTWARE CONFIGURATION:**

* CMakeLists.txt / CMake
  + Configuration and building using CMake is now supported under Microsoft Windows and on Unix-type systems.
* configure.ac / configure
  + Test for and use fseeko() if it is available. This allows supporting large files on Unix-type systems with a 32-bit 'long' type and a 64-bit 'off\_t' type.

**CHANGES IN LIBTIFF:**

* tiffiop.h:
  + Macros added to use 64-bit equivalents for all standard I/O and POSIX APIs used by libtiff and its tools which are limited to 2GB in Windows builds. Note that these 64-bit equivalents were introduced by the CRT provided with Visual Studio 2005 and if the necessary CRT is not installed on the target computer, the program will not run. The wrapper macros will not be activated unless the definition \_MSC\_VER is at least 1400 or \_\_MSVCRT\_VERSION\_\_ is at least 0x800.
* tif\_unix.c:
  + Updated to support large files under Microsoft Windows. This makes tif\_unix.c a completely viable candidate for use under Windows (in spite of its name) if the CRT is modern enough. Please note that tif\_win32.c already supported large files, but only 'tiffinfo' and 'tiffdump' made any provision to support large files under Windows.
  + \_tiffReadProc() and \_tiffWriteProc() are modified to chunk I/O to a maximum size of 2GB for extremely large I/O requests. This surmounts limitations in the Microsoft Windows read() and write() APIs (which are limited to the range of a 32-bit 'int'), and may avoid poor behavior with extremely large I/O requests on other systems.

**CHANGES IN THE TOOLS:**

* All tools
  + Updated to use I/O wrapper macros from tiffiop.h in order to support large files under Microsoft Windows.

**CHANGES IN THE CONTRIB AREA:**

* None

Last updated $Date: 2016-09-25 20:05:47 $.