**TIFF CHANGE INFORMATION**

**Current Version**: v3.9.0  
**Previous Version**: [v3.9.0beta](http://docs.google.com/v3.9.0beta.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](#tyjcwt)
* [Changes in the software configuration](#30j0zll)
* [Changes in libtiff](#1fob9te)
* [Changes in the tools](#3znysh7)
* [Changes in the contrib area](#2et92p0)

**MAJOR CHANGES:**

* All of the major changes since 3.8.0 were incorporated in the v3.9.0beta release. This is a stable release from a stable maintenance branch which incorporates many bug and security fixes needed by existing users of 3.X.X series libraries. Any future major changes will be in the 4.0.X series of releases.

**CHANGES IN THE SOFTWARE CONFIGURATION:**

* Updated autotools: Autoconf 2.64, Automake 1.11, libtool 2.2.6.
* Enabled support for Automake silent build rules (--enable-silent-rules or 'make V=0')
* Enabled support for Automake colorized tests.
* Added detection of a 64-bit integer type so that bundled utilities (like tiffcrop) can use it when necessary. Note that this version of libtiff does not require a 64-bit integer type but libtiff 4.0 does.

**CHANGES IN LIBTIFF:**

* libtiff/{Makefile.am, Makefile.v}: Do not distribute tiffconf.h, remove tif\_config.h/tiffconf.h during cleaning. As per bug <http://bugzilla.remotesensing.org/show_bug.cgi?id=1573>
* tif\_dirinfo.c: \_TIFFMergeFields() now only merges in field definitions that are missing. Existing definitions are silently ignored. (Bug #1585)
* tif\_write.c: Rip out the fancy logic in TIFFAppendToStrip() for establishing if an existing tile can be rewritten to the same location by comparing the current size to all the other blocks in the same directory. This is dangerous in many situations and can easily corrupt a file. (observed in esoteric GDAL situation that's hard to document). This change involves leaving the stripbytecount[] values unaltered till TIFFAppendToStrip(). Now we only write a block back to the same location it used to be at if the new data is the same size or smaller - otherwise we move it to the end of file.
* tif\_dirwrite.c: Try to avoid writing out a full readbuffer of tile data when writing the directory just because we have BEENWRITING at some point in the past. This was causing odd junk to be written out in a tile of data when a single tile had an interleaving of reading and writing with reading last. (highlighted by gdal autotest/gcore/tif\_write.py test 7.
* tif\_predict.c: use working buffer in PredictorEncodeTile to avoid modifying callers buffer. <http://trac.osgeo.org/gdal/ticket/1965>
* tif\_predict.c/h, tif\_lzw.c, tif\_zip.c: Improvements so that predictor based encoding and decoding works in read-write update mode properly. <http://trac.osgeo.org/gdal/ticket/1948>
* configure.com, libtiff/tif\_vms.c: Better OpenVMS support. Patches from Alexey Chupahin.
* tif\_fax3.c: fix leak of FAXCS state (per bug 1603).
* tif\_fax3.c: Make find0span() and find1span() non-inline to make MSVC 6.0 compiler happy.
* tif\_codec.c: Avoid NULL pointer dereferencing for exotic compression codec codes.
* tif\_dirread.c: zero tif->tif\_dir after freeing the directory in TIFFReadCustomDirectory(). I don't exactly remember why this was important.
* tif\_dirwrite.c: Fix potential memory leak writing large double tags.
* tif\_dirread.c: Fix unchecked malloc result.
* libtiff/tif\_lzw.c: Properly zero out the codetable. As per bug <http://bugzilla.maptools.org/show_bug.cgi?id=1929>
* libtiff/tif\_lzw.c: Properly zero out the string table. Fixes CVE-2008-2327 security issue.
* libtiff/tif\_jbig.c: Support the JBIG-KIT 2.0 (compatibility with the older versions retained).
* libtiff/tif\_getimage.c,tiffio.h: removed all use of UaToAa and Bitmap16to8 arrays in TIFFRGBAImage structure to restore ABI compatability. These were just an attempt to speed up processing with precalculated tables. <http://bugzilla.maptools.org/show_bug.cgi?id=1979>
* libtiff/tif\_codec.c: Avoid printing c->name if it does not exist.
* libtiff/tif\_getimage.c, tiffio.h: More ABI corrections. Removed SubsamplingHor/Ver from TIFFRGBAImage structure. <http://bugzilla.maptools.org/show_bug.cgi?id=1980>
* libtiff/tif\_jpeg.c: Avoid errors if the application writes a full strip for the last partial strip in a jpeg compressed file. <http://bugzilla.maptools.org/show_bug.cgi?id=1981>
* libtiff/tiffio.h: GCC will now validate format specifications for TIFFError(), TIFFErrorExt(), TIFFWarning(), and TIFFWarningExt() in order to reveal bugs. Cleaned up resulting warnings throughout for 32 bit build only.
* libtiff/tiffiop.h: Add private type declarations for int64, and uint64 so that bundled utilities (like tiffcrop) can use it when necessary.
* libtiff/tif\_predict.c: Add support for 32bit integer horz. predictors. <http://bugzilla.maptools.org/show_bug.cgi?id=1911>
* libtiff/tif\_luv.c: Fix handling of tiled logluv images. <http://bugzilla.maptools.org/show_bug.cgi?id=2005>
* libtiff/tif\_write.c: do not override the planar configuration to be contig for one sample files if planar configuration is already set. <http://bugzilla.maptools.org/show_bug.cgi?id=2057>
* libtiff/tif\_lzw.c: Fix buffer underflow bug. <http://bugzilla.maptools.org/show_bug.cgi?id=2065>
* libtiff/tif\_luv.c: correct return codes from encoderow to be 1 on success instead of zero. <http://bugzilla.maptools.org/show_bug.cgi?id=2069>
* libtiff/tif\_print.c (TIFFPrintDirectory): Applied patch for "tag error may cause segfault in tif\_print.c." <http://bugzilla.maptools.org/show_bug.cgi?id=1896>

**CHANGES IN THE TOOLS:**

* tools/tiff2pdf.c: Fixed setting of alpha value per report on list.
* tools/tiffsplit.c: Get rid of unsafe strcpy()/strcat() calls when doing the filename/path construction.
* tools/tiff2pdf.c: More appropriate format string in t2p\_write\_pdf\_string(); avoid signed/unsigned mismatch.
* tools/tiffsplit.c: Use dynamically allocated array instead of static when constructing output file names.
* tools/tiffcrop.c, man/tiffcrop.1: Incorporated significant functionality update from Richard Nolde.
* tools/tiff2ps.c: Incorporated significant functionality update from Richard Nolde. In particular, support for rotating the image by 90, 180, 270, and 'auto' has been added.
* tools/tiff2ps.c: Remove spurious message printed to stderr.
* tools/tiffsplit.c: fix sampleformat to be shortv instead of longv.
* tools/{rgb2ycbcr.c, tiff2rgba.c}: Applied patch for CVE-2009-2347 libtiff: integer overflows in various inter-color space conversion tools. <http://bugzilla.maptools.org/show_bug.cgi?id=2079>

**CHANGES IN THE CONTRIB AREA:**

* None

Last updated $Date: 2016-09-25 20:05:46 $.