

Exiv2 - Bug #617

Optimize TIFF writing

01 Mar 2009 00:07 - Andreas Huggel

Status:	Closed	Start date:	28 Feb 2009
Priority:	Normal	Due date:	
Assignee:	Andreas Huggel	% Done:	100%
Category:	tiff parser	Estimated time:	0.00 hour
Target version:	0.19		
Description			
Writing to TIFF uses too much memory.			
Reading			
<pre>ahuggel@mowgli> ls -la P1010003.TIF -rw----- 1 ahuggel ahuggel 9451593 01-Mar-09 P1010003.TIF valgrind exiv2-0.18 P1010003.TIF malloc/free: 2,121 allocs, 2,121 frees, 103,862 bytes allocated.</pre>			
Non-intrusive writing			
<pre>valgrind exiv2-0.18 -M'set Exif.Image.Software Exiv2' P1010003.TIF malloc/free: 6,608 allocs, 6,608 frees, 10,086,737 bytes allocated.</pre>			
Intrusive writing			
<pre>valgrind exiv2-0.18 -M'set Exif.Image.Software The Exiv2 utility' P1010003.TIF malloc/free: 5,940 allocs, 5,940 frees, 19,548,553 bytes allocated.</pre>			
Related issues:			
Related to Exiv2 - Bug #671: Writing to read-only TIFF-like file fails		Closed	06 Jan 2010

Associated revisions

Revision 1755 - 01 Mar 2009 00:12 - Andreas Huggel

#617: For TIFF images, use memory mapping for non-intrusive writing instead of reading image into memory.

Revision 1918 - 07 Nov 2009 06:42 - Andreas Huggel

#617: Changed TiffParser::encode to use an instance of BasicIo instead of a Blob for intrusive writing (API change).

History

#1 - 01 Mar 2009 00:24 - Andreas Huggel

With [r1755](#) the image is no longer loaded into memory for non-intrusive writing. The amount of memory allocated is therefore reduced by about the size of the image (for both write modes, since non-intrusive writing is always attempted first):

Non-intrusive writing

```
valgrind ./exiv2 -M'set Exif.Image.Software Exiv2' P1010003.TIF
malloc/free: 6,746 allocs, 6,746 frees, 650,106 bytes allocated.
```

Intrusive writing

```
valgrind ./exiv2 -M'set Exif.Image.Software The Exiv2 utility' P1010003.TIF
malloc/free: 6,102 allocs, 6,102 frees, 10,114,031 bytes allocated.
```

#2 - 01 Mar 2009 01:18 - Andreas Huggel

The backdraw of the change in [r1755](#) is that in case Exiv2 crashes somewhere in the non-intrusive writing logic, this is now likely to result in modified

and possibly corrupted image.

#3 - 20 May 2009 00:35 - Andreas Huggel

The next step is to replace the use of a Blob in the intrusive writing methods with a Basiclo. This will require an API change in the *Parser::encode methods.

#4 - 20 May 2009 07:55 - Andreas Huggel

- % Done changed from 0 to 30

#5 - 20 May 2009 07:56 - Andreas Huggel

- Status changed from New to Assigned

- Assignee set to Andreas Huggel

#6 - 07 Nov 2009 08:21 - Andreas Huggel

[r1918](#) finally implements the second step: In intrusive writing, the image is no longer written to a Blob but directly to a Basiclo instance, i.e., straight to a file instead of a memory buffer. With this change, the complete image is never loaded into memory anymore.

Non-intrusive writing:

```
valgrind ./exiv2 -M'set Exif.Image.Software Exiv2' P1010003.TIF
malloc/free: 6,322 allocs, 6,322 frees, 644,462 bytes allocated.
```

Intrusive writing:

```
valgrind ./exiv2 -M'set Exif.Image.Software The Exiv2 utility' P1010003.TIF
malloc/free: 5,627 allocs, 5,627 frees, 526,459 bytes allocated.
```

#7 - 07 Nov 2009 08:41 - Andreas Huggel

These changes also have an effect on performance when dealing with large TIFF images. For non-intrusive writing, the performance gain is huge, since the image is now modified in-place, without ever loading it. For intrusive writing, the time is now reasonably close to that of a simple copy operation.

The following tests were done with a 120MB TIFF image, *trunk* is roughly 0.18.2, *unstable* is [r1918](#).

```
big.tif 126,172,596 bytes
time cp big.tif big1.tif
real    0m2.254s
user    0m0.001s
sys     0m0.205s
```

Non-intrusive writing

```
exiv2 -M'set Exif.Photo.DateTimeOriginal Yesterday' big.tif
```

	unstable	trunk
real	0.023s	3.101s
user	0.013s	0.117s
sys	0.011s	0.888s
mem	8,374,303 bytes	134,551,658 bytes

Intrusive writing

```
exiv2 -M'set Exif.Photo.DateTimeOriginal Yesterday noon exactly now' big.tif
```

	unstable	trunk
real	2.654s	3.790s
user	0.044s	0.802s
sys	0.744s	0.995s
mem	6,579,811 bytes	259,330,485 bytes

#8 - 07 Nov 2009 08:42 - Andreas Huggel

- Status changed from Assigned to Resolved

- Target version set to 0.19

- % Done changed from 30 to 100

#9 - 30 Dec 2009 07:50 - Andreas Huggel

- Status changed from Resolved to Closed

