

Exiv2 - Bug #1178

Olympus E-M5 Mark II raw file not recognized

14 Apr 2016 19:16 - Terence Tay

Status:	Closed	Start date:	14 Apr 2016
Priority:	Normal	Due date:	
Assignee:	Robin Mills	% Done:	100%
Category:	image format	Estimated time:	10.00 hours
Target version:	0.26		

Description

I have a raw file from this camera that is not recognized by Exiv2. Exiv2 reports:

Exiv2 exception in print action for file xxxx.ORF:
xxxx.ORF: The file contains data of an unknown image type

This file can be read by exiftool.

I don't have permission to share this raw file, but I believe the problem is in the first 8 bytes of the ORF header, shown here:

```
49 49 52 4F BC 05 D9 00
```

Exiv2::OrfHeader::read() is expecting:

```
49 49 52 4F 08 00 00 00
```

The difference is in the last 4 bytes where Exiv2 expects it to be an offset with a value of 8. Because of this difference, Exiv2 rejects this file as being invalid. I'm not an expert on ORF formats but the following patch to orfimage.cpp seems to work.

```
257c257
<         setOffset(0x00000008);
---
>         setOffset(getULong(pData + 4, byteOrder()));
```

Please review and if appropriate, please include this fix in Exiv2.

If I get a raw file that demonstrates this issue that I can share, I will attach it.

Associated revisions

Revision 4281 - 19 Apr 2016 20:17 - Robin Mills

#1178 Thanks to Terence for reporting this and providing confidential test data

History

#1 - 15 Apr 2016 11:35 - Robin Mills

- Status changed from New to Assigned
- Target version set to 0.26
- % Done changed from 0 to 10
- Estimated time set to 2.00 h

Thanks for letting me know about this, Terence. Niels, our wonderful makernote engineer, has gone back to college on top of his full-time demanding job, so I've taken on support for this part of the code.

I can probably find a suitable test file on Phil Harvey's Exiftool web site, or here: <http://www.rawsamples.ch/index.php/en/>. I can use 'dd' (or a hex-editor or something) to edit a regular .ORF to the header you have discovered. If you can share a test file privately with me (email, or drop-box or something), that would save me a little time.

You are not being blocked by this issue as you have a workaround. The Libre Graphics Meeting is being held in London this weekend, so I'm off to meet my friends from GIMP and other open-source projects on Saturday. https://en.wikipedia.org/wiki/Libre_Graphics_Meeting Our house remodelling project isn't finished, and we are out of town on Sunday. I will try to find time in the next two or three days to look at this. For sure it sounds quite easy to fix.

One of our outstanding projects is [#992](#) to have a more rigorous investigation of our RAW image support. I added option -pS in v0.25 to print the structure of image files. It's been "beefed up" in v0.26 to support -pR which recursively prints image structures within images. Now, if I could only recruit another engineer to work on this stuff.

Options -pS and -pR aren't useful if we can't even open the file, however I think you've identified the culprit in orfimage.cpp who says "Nope, not my file type" and I'm confident this can be easily remedied.

I believe Exiv2 v0.26 is on track for "code complete" in June (as we have discussed). You can follow progress on "News": <http://dev.exiv2.org/projects/exiv2/news>

#2 - 15 Apr 2016 16:09 - Terence Tay

Thanks for responding to my support issue.

We've asked the photographer to shoot sample images that he is willing to release publicly. I'll forward them along if I get them.

You're right that we're not blocked by this issue. Fortunately, it was a simple fix. We are doing more extensive tests now and I'll let you know if we find something more.

Enjoy your Libre Graphics Meeting and good luck with your house remodelling project.

#3 - 15 Apr 2016 17:30 - Robin Mills

- % Done changed from 10 to 30

- Estimated time changed from 2.00 h to 10.00 h

I've investigated this. I believe an ORF file is a TIFF file with a slight modification. TIFF data-structures begin with II = Intel/little-endian, MM = Motorola/big-endian.

Tiff files start "MM"0x00** or "II"0x0. I guess: ** == 0x2a == 42 (the *meaning of life!*)
Olympus Raw files start "MMOR" or "IIRO". I guess: "OR" = Olympus Raw (I guess), however some start "SR"

I've downloaded all the Olympus files from <http://www.rawsamples.ch/index.php/en/> and dumped the headers (output below). There's exactly one MM file from 35 files. You'll see the others are a mix of IIRO and IISO. If your photographer can send me more MM files, that will be very helpful. I can assure him that his images will only be used for software development/testing purposes and never leave home (they won't be on my backups either).

The next 4 bytes are an unsigned long (in MM/II format) which is the offset to the start of the first TIFF directory. In the test files, it's always 0x0008 (byte 8), however it can be any offset. I suspect the code:

```
setOffset(getULong(pData + 4, byteOrder()));  
if (offset() != 0x00000008) return false;
```

Should simply be:

```
setOffset(getULong(pData + 4, byteOrder()));
```

I'm totally puzzled the following code (remember, I didn't write Exiv2):

```
uint16_t sig = getUShort(pData + 2, byteOrder());  
if (tag() != sig && 0x5352 != sig) return false; // #658: Added 0x5352 for SP-560UZ  
sig_ = sig;
```

I will modify our code and test with the 35 test files.

I'd like you to stick with your work-around for the moment. Next week I will consider some refactoring of our code base. The existing exiv2 code has separate sources for each image type, for example: tiffimage.cpp orfimage.cpp.

```
623 rmills@rmillsmbp:~/gnu/exiv2/ttt $ wc src/tiffimage.cpp src/orfimage.cpp  
2574 11331 117064 src/tiffimage.cpp  
284 813 8855 src/orfimage.cpp  
2858 12144 125919 total  
624 rmills@rmillsmbp:~/gnu/exiv2/ttt $
```

As you can see, orfimage.cpp is modest, however I suspect he can be refactored to be even smaller. And I'd like to add support for -pS and -pR for .orf files. I'm not an expert in image formats (although learning fast from working on -pS and -pR) and I suspect there are other RAW formats which are essentially tiffs with different a signature in byte[2:3]. Refactoring now will pay dividends when servicing [#992](#) (raw image support).

It's always nice to hear from you and Mike. I hope you're both well. The house project will finish this summer. Because of the house project, I only have Saturday on which to attend the conference, however I'm hoping to meet some Exiv2 users. Maybe they'll buy me lunch!

```
630 rmills@rmillsmbp:~/gnu/exiv2/ttt $ for i in ~/Downloads/RAW_OL*.ORF ; do echo $(od -a -N 8 $i | head -1)  
$i ; done  
0000000 I I R S bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_C5050Z.ORF
```

```

0000000 I I R S bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_C5060WZ.ORF
0000000 I I R S bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_C8080.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E-330.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E-450.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E-M10.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E-M10MARKII.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E-M5MARKII.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E-PL6.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E-PL7.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E1.ORF
0000000 M M O R nul nul nul bs /Users/rmills/Downloads/RAW_OLYMPUS_E20.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E3.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E30.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E300.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E410.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E420.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E5.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E500.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E510.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E520.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_E600.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EM1.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EM5.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EP1.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EP2.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EP3.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EPL1.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EPL3.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_EPL5.ORF
0000000 I I R S bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_SP350-2.ORF
0000000 I I R S bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_SP350.ORF
0000000 I I R S bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_SP500UZ.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_TG-4.ORF
0000000 I I R O bs nul nul nul /Users/rmills/Downloads/RAW_OLYMPUS_XZ-2.ORF
631 rmills@rmillsmbp:~/gnu/exiv2/ttt $

```

#4 - 15 Apr 2016 22:10 - Robin Mills

- % Done changed from 30 to 50

Terence has sent me a private test file. I believe the file is OK. My workaround is to comment off one line of code:

```

bool OrfHeader::read(const byte* pData, uint32_t size)
{
    if (size < 8) return false;

    if (pData[0] == 0x49 && pData[1] == 0x49) {
        setByteOrder(littleEndian);
    }
    else if (pData[0] == 0x4d && pData[1] == 0x4d) {
        setByteOrder(bigEndian);
    }
    else {
        return false;
    }
    uint16_t sig = getUShort(pData + 2, byteOrder());
    if (tag() != sig && 0x5352 != sig) return false; // #658: Added 0x5352 for SP-560UZ
    sig_ = sig;
    setOffset(getULong(pData + 4, byteOrder()));
    // if (offset() != 0x00000008) return false;

    return true;
} // OrfHeader::read

```

This fix is fine for readMetadata() (*which use Andreas/Brad file parsers*).
This fix will not unlock printStructure() which has its own file parsers (*which I wrote*).

The 0x5352 is attempting to forgive the header IISR or something. That code might be wrong. I'll study this further next week.

Dumping a .ORF file

Create a new file with the header from a TIFF and the body of a .ORF:

```

$ dd count=4 bs=1 if=msvc2005/testimages/test.tiff > orf.tiff
$ dd bs=1 skip=4 count=16000000 if=Any.ORF >> orf.tiff

```

```
$ exiv2 -pR orf.tiff
```

I'll update orfimage.cpp next week to support -pS/-pR

#5 - 19 Apr 2016 20:28 - Robin Mills

- % Done changed from 50 to 80

[r4281](#) Thank you to Terence for reporting this and providing test data. The test data is private and cannot be submitted.

For Terence's workaround purpose, the correct (one-line) fix is listed above: <http://dev.exiv2.org/issues/1178#note-4> In src/orfimage.cpp comment off:

```
// if (offset() != 0x00000008) return false;
```

My fix is rather more complex than the simple work-around. I have derived OrfImage from TiffImage, because TIFF and Orf are almost identical. This enables OrfImage::printStructure() which implements exiv2(.exe) options -pS/-pR/-pC/-pX. I've looked at the other raw formats to see if they can be derived from TiffImage. The only likely candidate is cr2image (Canon Raw). I decided to leave cr2image.cpp as is because it already implements printStructure(). We might revisit this when [#992](#) is serviced in a future release. [#992](#) is not scheduled for v0.26.

#6 - 24 Apr 2016 05:17 - Robin Mills

- Status changed from Assigned to Closed

- % Done changed from 80 to 100