Exiv2 - Bug #1129
Different behaviour of exiv2 between remote and local file.

12 Oct 2015 19:50 - Robin Mills

Status:	Closed	**Start date:**	12 Oct 2015
Priority:	Normal	**Due date:**	
Assignee:	Robin Mills	**% Done:**	100%
Category:	tiff parser	**Estimated time:**	2.00 hours
Target version:	0.26		

Description

This issue has surfaced during discussion of #1080.

The file exiv2 -pa http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg behaves differently when local. 690 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pa http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
Exiv2 exception in print action for file http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg:
Failed to read image data
691 rmills@rmillsmbp:~/gnu/exiv2/trunk $ curl http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg | exiv2 -pa -

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  4404    0  4404    0     0  45235      0 --:--:-- --:--:-- --:--:-- 46851
```

Error: Directory Image: Next pointer is out of bounds; ignored.
Warning: Directory Image, entry 0x3030 has unknown Exif (TIFF) type 12336; setting type size 1.
Error: Directory Image, entry 0x3030 has invalid size 808464432*1; skipping entry.
... hundreds of similar lines deleted ...
Error: Directory Image, entry 0x3030 has invalid size 808464432*1; skipping entry.
Warning: JPEG format error, rc = 5
Exif.Image.ExifTag Long 1 217
Floating point exception: 8
692 rmills@rmillsmbp:~/gnu/exiv2/trunk $

Curiously, the file appears to corrupt and is diagnosed consistently with option -pS 694 rmills@rmillsmbp:~/gnu/exiv2/trunk $ curl http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg | exiv2 -pS -

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  4404    0  4404    0     0  60926      0 --:--:-- --:--:-- --:--:-- 63826
STRUCTURE OF JPEG FILE: 1444679207.exiv2_temp
 address | marker     | length  | data
       2 | 0xd8 SOI   |       0
       4 | 0xe1 APP1  |    4400 | Exif..MM.*.....00000000000000000
    4404 | 0xffffffff ?HU?Exiv2 exception in print action for file -:
This does not look like a JPEG image
695 rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

The floating point exception reported by the local file is the subject of #1080.

The scope of this issue is to investigate why FileIo and HttpIo result in different output with option -pa.

Related issues:

Related to Exiv2 - Bug # 1080: Division by zero / crash on malformed input file	**Closed**	**13 May 2015**

Associated revisions

Revision 3991 - 13 Oct 2015 21:19 - Robin Mills

#1129. Fix submitted.

History

#1 - 12 Oct 2015 20:10 - Robin Mills

699 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pS http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
STRUCTURE OF JPEG FILE: http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg
address | marker | length | data
2 | 0xd8 SOI | 0
4 | 0xe1 APP1 | 4400 | Exif..MM.*.....00000000000000000
4404 | 0xffffffff q$V?Exiv2 exception in print action for file http://dev.exiv2.org/attachments/download/786/exiv2-divzero.jpg:
This does not look like a JPEG image
700 rmills@rmillsmbp:~/gnu/exiv2/trunk $

#2 - 13 Oct 2015 21:19 - Robin Mills

- Status changed from Assigned to Resolved

- % Done changed from 0 to 100

- Estimated time set to 2.00

Fix submitted: r3991

This is a very interesting bug. It's caused by different semantics of eof() in the classes RemoteIo and FileIo.

In jpgimage.cpp#365 in function jpegBase::ReadMetadata(), we have: io_->read(rawExif.pData_, rawExif.size_);
if (io_->error() || io_->eof()) throw Error(14);
RemoteIo::eof() returns true when every byte in the file has been read. class FileIo is a wrapper for the standard "C" FILE pointer and FileIo::eof() returns feof(f) != 0. However feof(f) only returns > 0 when there has been an attempt to read past the end-of-file. This is an artefact of the stdio.h implementation. The function feof(f) returns the status of the EOF bit which is only set when an attempt to read beyond the final byte is requested. That every byte in the file has been consumed is not understood/respected by feof(f).

My fix makes RemoteIo and FileIo consistent as follows: bool FileIo::eof() const
{
assert(p_->fp_ != 0);
return feof(p_->fp_) != 0 || tell() >= size() ;
}
The test suite fails on bugfixes-test.sh bug=480 in largeiptc-test.cpp in the following code: Exiv2::DataBuf buf(io.size());
std::cout << "Reading " << buf.size_ << " bytes from " << data << "\n";
io.read(buf.pData_, buf.size_);
if (io.error() || io.eof()) throw Exiv2::Error(14);
This code reads the complete file into memory and throws if the file is at EOF. However, I've modified the semantics of eof() to report true when the whole file has been read. The fix for largeiptc-test.cpp is to require eof() to be true: Exiv2::DataBuf buf(io.size());
std::cout << "Reading " << buf.size_ << " bytes from " << data << "\n";
io.read(buf.pData_, buf.size_);
if (io.error() || !io.eof()) throw Exiv2::Error(14);
And now we pass the test suite.

I am a little nervous of this change as we are changing the semantics of basicio::eof(). The documentation http://www.exiv2.org/doc/classExiv2_1_1BasicIo.html states: Returns true if the IO position has reached the end, otherwise false.
I believe my RemoteIo implementation is totally correct and FileIo should be modified to implement this definition.

The messages from local fileio such as: Error: Directory Image, entry 0x3030 has invalid size 808464432*1; skipping entry.
... hundreds of similar lines deleted ...
Error: Directory Image, entry 0x3030 has invalid size 808464432*1; skipping entry.

are totally bogus and coming from TiffParser. We shouldn't be in TiffParser. The file is corrupt and jpegBase::ReadMetadata() now behaves correctly by throwing for both local and remote files.

There is a English typo in the documentation generated from basicio.hpp and has been fixed.

#3 - 06 Dec 2015 21:01 - Robin Mills

- Status changed from Resolved to Closed