

Exiv2 - Bug #1058

xml:lang should be treated case insensitive

19 Apr 2015 15:48 - Tobias E.

Status:	Closed	Start date:	19 Apr 2015
Priority:	Normal	Due date:	
Assignee:	Robin Mills	% Done:	100%
Category:	xmp	Estimated time:	0.00 hour
Target version:	0.25		
Description			
According to the XMP specs (https://www.adobe.com/content/dam/Adobe/en/devnet/xmp/pdfs/XMPSpecificationPart1.pdf), chapter B.4 (page 42) the xml:lang qualifier is to be compared case insensitive. It would be great if libexiv2 could follow that. It's probably enough to specify a comparator in the definition of value_ in LangAltValue.			
Related issues:			
Related to Exiv2 - Bug # 601: Metadata conversion enhancements		Closed	07 Jan 2009

Associated revisions

Revision 3703 - 19 Apr 2015 18:44 - Robin Mills

#1058. XMP spec chapter B.4 (page 42) the xml:lang qualifier is to be compared case insensitive.

<https://www.adobe.com/content/dam/Adobe/en/devnet/xmp/pdfs/XMPSpecificationPart1.pdf>

Revision 3704 - 19 Apr 2015 20:25 - Robin Mills

#1058. Change comparator to return -1 : 0 : 1

Revision 3706 - 20 Apr 2015 10:44 - Robin Mills

#1058. Fixing the exceptions from test/xmpparser-test.sh. The exceptions from test/conversions.sh require more investigation.

See the issue report for a longer discussion.

Revision 3707 - 20 Apr 2015 18:07 - Robin Mills

#1058. Calming the test suite. LangAltValue comparator causes harmless changes in order of lang reporting.

Revision 3713 - 21 Apr 2015 18:17 - Robin Mills

#1058. xml:lang case insensitive. Working well. Added regression detector.

History

#1 - 19 Apr 2015 18:52 - Robin Mills

- Status changed from New to Assigned
- Assignee set to Robin Mills
- Target version set to 0.25

Fix submitted: r3703

I'm not a metadata expert. I'm a build engineer. I've put in the fix you have suggested. As you have raised this subject, perhaps you could provide test this and provide a suitable test file and an exiv2 command to show "old" and "new" behaviour.

My fix has thrown up exceptions in our test suite. Running xmpparser-test.sh ...

```
Files /Users/rmills/gnu/exiv2/trunk/test/tmp/xmpparser-test.out-stripped and /Users/rmills/gnu/exiv2/trunk/test/data/xmpparser-test.out differ
463c463
```

```
< Xmp.dc.title                      LangAlt  1 lang="en-US" Sonnenuntergang am Strand
---
> Xmp.dc.title                      LangAlt  2 lang="de-DE" Sonnenuntergang am Strand, lang="en-US" Sunset on the beach
result = 1
```

We are also getting exceptions for conversions.sh. I will be having a 1-to-1 with Alan, our conversions engineer, and I'll discuss this with him.

```
Running conversions.sh ...
```

```
Files /Users/rmills/gnu/exiv2/trunk/test/tmp/conversions.out-stripped and /Users/rmills/gnu/exiv2/trunk/test/data/conversions.out differ
24c24,26
```

```
< Xmp.dc.description                LangAlt  1 lang="de-DE" Ciao bella
---
> Warning: Failed to convert Xmp.dc.description to Iptc.Application2.Caption
> Warning: Failed to convert Xmp.dc.description to Exif.Image.ImageDescription
> Xmp.dc.description                LangAlt  2 lang="de-DE" The Exif image description, lang="it-IT" Ciao bella
26c28
< Exif.Image.ImageDescription        Ascii    11 Ciao bella
---
> k.jpg: (No Exif data found in the file)
28,29c30
< Iptc.Envelope.CharacterSet         String   3 $%G
< Iptc.Application2.Caption          String  10 Ciao bella
---
> k.jpg: (No IPTC data found in the file)
33,35c34,38
<   <rdf:li xml:lang="x-default">Ciao bella</rdf:li>
< Xmp.dc.description                LangAlt  1 lang="x-default" Ciao bella
< Exif.Image.ImageDescription        Ascii    11 Ciao bella
---
> Warning: Failed to convert Xmp.dc.description to Iptc.Application2.Caption
> Warning: Failed to convert Xmp.dc.description to Exif.Image.ImageDescription
>   <rdf:li xml:lang="x-default">How to fix this mess</rdf:li>
> Xmp.dc.description                LangAlt  3 lang="x-default" How to fix this mess, lang="de-DE" The Exif image description, lang="it-IT"
Ciao bella
> Exif.Image.ImageDescription        Ascii    21 How to fix this mess
37c40
< Iptc.Application2.Caption          String  10 Ciao bella
---
> Iptc.Application2.Caption          String  20 How to fix this mess
result = 1
553 rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

#2 - 19 Apr 2015 19:10 - Tobias E.

Awesome, that was fast. :)

However, according to <http://www.cplusplus.com/reference/map/map/> the comparator has to return true iff `str1 < str2`. Your function is testing `str1 == str2`.

#3 - 19 Apr 2015 20:29 - Robin Mills

I try to be quick. It's easier for both of us. I've updated the code. r3704. It would be of great help if you could provide a test file and an explanation of the behaviour you expect.

Alan's cancelled our 1-to-1 because of a power cut at his location and I can't discuss the issues in `./conversions.sh` with him until later in the week.

```
This change has produced more exceptions from ./xmpparser-test.sh 591 rmills@rmillsmbp:~/gnu/exiv2/trunk/test $ ./xmpparser-test.sh
Files /Users/rmills/gnu/exiv2/trunk/test/tmp/xmpparser-test.out-stripped and /Users/rmills/gnu/exiv2/trunk/test/data/xmpparser-test.out differ
150c150
< Xmp.ns1.ArrayProp2          LangAlt  2 lang="x-two" Item2.2 value, lang="x-one" Item2.1 value
---
> Xmp.ns1.ArrayProp2          LangAlt  2 lang="x-one" Item2.1 value, lang="x-two" Item2.2 value
232d231
< >   <rdf:li xml:lang="x-two">Item2.2 value</rdf:li>
233a233
> >   <rdf:li xml:lang="x-two">Item2.2 value</rdf:li>
282,285d281
< 5c5
< < Xmp.ns1.ArrayProp2          LangAlt  2 lang="x-two" Item2.2 value, lang="x-one" Item2.1 value
< ---
< > Xmp.ns1.ArrayProp2          LangAlt  2 lang="x-one" Item2.1 value, lang="x-two" Item2.2 value
288c284
< Xmp.dc.title                 LangAlt  2 lang="en-US" Sunset on the beach, lang="de-DE" Sonnenuntergang am Strand
---
> Xmp.dc.title                 LangAlt  2 lang="de-DE" Sonnenuntergang am Strand, lang="en-US" Sunset on the beach
345d340
<   <rdf:li xml:lang="en-US">Sunset on the beach</rdf:li>
346a342
>   <rdf:li xml:lang="en-US">Sunset on the beach</rdf:li>
467c463
< Xmp.dc.title                 LangAlt  2 lang="en-US" Sunset on the beach, lang="de-DE" Sonnenuntergang am Strand
---
> Xmp.dc.title                 LangAlt  2 lang="de-DE" Sonnenuntergang am Strand, lang="en-US" Sunset on the beach
592 rmills@rmillsmbp:~/gnu/exiv2/trunk/test $
```

I'm confident about my comparator returning true when `str1 < str2` and tested it on the bench: `#include <iostream>`

```
#include <map>
#include <string>
```

```
/*!
```

```
@brief %LangAltValueComparator
```

```
#1058
```

```
https://www.adobe.com/content/dam/Adobe/en/devnet/xmp/pdfs/XMPSpecificationPart1.pdf
```

```
XMP spec chapter B.4 (page 42) the xml:lang qualifier is to be compared case insensitive.
```

```
*/
```

```
struct LangAltValueComparator {
    bool operator() (const std::string& str1, const std::string& str2) const
    {
```

```

int result = str1.size() < str2.size() ? -1
           : str1.size() > str2.size() ? 1
           : 0
           ;
std::string::const_iterator c1 = str1.begin();
std::string::const_iterator c2 = str2.begin();
if ( result==0 ) for (
    ; result==0 && c1 != str1.end()
    ; ++c1, ++c2
    ) {
    result = tolower(*c1) < tolower(*c2) ? -1
           : tolower(*c1) > tolower(*c2) ? 1
           : 0
           ;
    }
return result < 0 ;
}
};

```

```

typedef std::map<std::string,std::string,LangAltValueComparator> Dict_t;
typedef Dict_t::const_iterator Dict_i;

```

```

int main ()
{
    Dict_t dict;

    dict["robin"]="robin";
    dict["mills"]="mills";
    dict["Robin"]="Robin";
    dict["Mills"]="Mills";
    dict["ROBIN"]="ROBIN";
    dict["MILLS"]="MILLS";

    std::cout << "dict length = " << dict.size() << std::endl;
    for ( Dict_i i = dict.begin() ; i != dict.end() ; i++ ) {
        std::cout << i->first << " : " << i->second << std::endl;
    }

    return 0;
}

```

And the output is: 539 rmills@rmillsmbp:~/gnu/exiv2/trunk \$ make foo; ./foo

make: `foo' is up to date.

dict length = 2

mills : MILLS

robin : ROBIN

540 rmills@rmillsmbp:~/gnu/exiv2/trunk \$

#4 - 20 Apr 2015 01:49 - Alan Pater

Well, I'm back online :-)

If I understand the issue correctly, the following should all be considered the same value:

```
lang="en-US"  
lang="en-us"  
lang="EN-US"  
lang="eN-uS"
```

So if someone wants to change the English title, but makes a typo on the command line, it should not matter, exiv2 should change the correct title anyway.

So

```
exiv2 -M'set Xmp.dc.title lang="en-us" Sunrise on the beach' file.jpg  
exiv2 -M'set Xmp.dc.title lang="EN-US" Sunrise on the beach' file.jpg
```

should both change

```
Xmp.dc.title LangAlt 2 lang="de-DE" Sonnenuntergang am Strand, lang="en-US" Sunset on the beach
```

to

```
Xmp.dc.title LangAlt 2 lang="de-DE" Sonnenuntergang am Strand, lang="en-US" Sunrise on the beach
```

#5 - 20 Apr 2015 01:50 - Alan Pater

Robin, your fix seems to be swapping around the order of the different languages.

```
Xmp.dc.title LangAlt 2 lang="en-US" Sunset on the beach, lang="de-DE" Sonnenuntergang am Strand
```

#6 - 20 Apr 2015 10:44 - Robin Mills

Alan

Thanks for explaining what this is about. I learn something everyday (and forget two things every day)!

I've submitted a fix: r3706 for the exceptions arriving from test/xmpparser.sh

I've puzzled for a couple of hours about why adding the comparator should change the output order. For sure, they now appear in alphabetic order of the lang string. So "de-DE" < "en-US" and so on. "x-default" is always listed first. I don't believe the order of presentation of the language key/value pair "xx-yy" *String* is important. So I've decided to update the test suite reference file: 810 rmills@rmillsmbp:~/gnu/exiv2/trunk/test/tmp \$ cp

```
xmpparser-test.out-stripped ../data/xmpparser-test.out
```

We still have exceptions from test/conversions.sh

```
Files ../tmp/conversions.out-stripped and ../data/conversions.out differ
```

```
26c26
```

```
< Xmp.dc.description LangAlt 2 lang="it-IT" Ciao bella, lang="de-DE" The Exif image description
```

```
---
```

```
> Xmp.dc.description LangAlt 2 lang="de-DE" The Exif image description, lang="it-IT" Ciao bella
```

```
36,40c36,40
```

```
< Warning: Failed to convert Xmp.dc.description to Iptc.Application2.Caption
```

```
< Warning: Failed to convert Xmp.dc.description to Exif.Image.ImageDescription
```

```
< Xmp.dc.description LangAlt 2 lang="de-DE" The Exif image description, lang="it-IT" Ciao bella
```

```

< l.jpg: (No Exif data found in the file)
< l.jpg: (No IPTC data found in the file)
---
> <rdf:li xml:lang="x-default">How to fix this mess</rdf:li>
> Xmp.dc.description           LangAlt   3 lang="x-default" How ..., lang="de-DE" The Exif ..., lang="it-IT" Ciao...
> Exif.Image.ImageDescription   Ascii    21 How to fix this mess
> Iptc.Envelope.CharacterSet    String   3 $%G
> Iptc.Application2.Caption     String   20 How to fix this mess
result = 1
817 rmills@rmillsmbp:~/gnu/exiv2/trunk $

```

You can see that some of those exceptions are the alphabetic report of the `lang="..."`, however not all. The message "Warning: Failed to convert Xmp.dc.description to Iptc.Application2.Caption" is sinister. Having failed to convert, the file does not have IPTC (and Exif) data and this is correctly detected and reported as "l.jpg: (No IPTC data found in the file)". I will intend to investigate test/conversions.sh further.

#7 - 20 Apr 2015 11:54 - Alan Pater

It looks to me that what is failing is line 68 in test/conversions.sh

```
runTest exiv2 -M'set Xmp.dc.description lang="x-default" How to fix this mess' l.xmp
```

'**lang="x-default" How to fix this mess**' is not being added to the description property. Without a x-default conversions to Exif and IPTC do not happen.

#8 - 20 Apr 2015 12:11 - Robin Mills

I'm debugging this at the moment.

I haven't understood your message, Alan. If it's not to be converted, why have we not seen this message before: Warning: Failed to convert Xmp.dc.description to Iptc.Application2.Caption

There's something in the code about looking for language index = 0. That looks suspicious. Now that we're storing in alphabetic order, 'x-default' is unlikely to be index 0 when there's more than one language.

#9 - 20 Apr 2015 12:50 - Alan Pater

I don't know why the conversion does not happen without a "x-default", but that is the baseline. Test 4 of test/conversions.sh is the same. We see the message as a result of Test 4.

Test 5 tries to add an "x-default", so those messages should go away. Previous to the fix for this issue, they did go away. With the fix, "x-default" is no longer being added, so the messages appear for Test 5 also.

#10 - 20 Apr 2015 12:56 - Robin Mills

It's something like that. I think test4 has always, and and is still producing the message. No change at that point.

I think something's going wrong when we try to add the x-default which is intended to remove the message. However after failing to update, the message is reported and a confusing cascade of trouble follows.

Let me continue to debug this. I'll update this report when I understand it better.

#11 - 20 Apr 2015 18:27 - Robin Mills

- Status changed from Assigned to Resolved

Fix submitted r3707. I'm going change the status of this issue to Resolved. Perhaps we should add a regression detector to the test suite. Something like:

```
$ exiv2 -M'set Xmp.dc.title lang="en-GB" Sunrise on the beach' file.jpg
$ exiv2 exiv2 -M'set Xmp.dc.title lang="de-DE" zee toweels ease on zee sun loungers' file.jpg
$ exiv2 exiv2 -M'set Xmp.dc.title lang="DE-de" Sonnenuntergang am Strand file.jpg
```

If the Germans still have their towels on the loungers, we're in trouble. For that matter, if the Germans outnumber the Brits, we'll also be very unhappy!

I'm not having a go at the Germans. Some of the best guys I've worked with are Germans: Daniel B, Michael D, Tobias H, Uwe L ... and many more.

#12 - 21 Apr 2015 10:17 - Robin Mills

- Status changed from Resolved to Assigned

The comparator seems to be working correctly. However there is something wrong because we can achieve duplicate lang="de-DE" values as follows:

```
1173 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.title lang="en-GB" BRITS' ~/R.jpg
1174 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.title lang="de-DE" GERMANS' ~/R.jpg
1175 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pX ~/R.jpg | xmllint -pretty 1 -
<?xml version="1.0"?>
<?xpacket begin="ï" id="W5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 4.4.0-Exiv2">
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
    <rdf:Description xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="" xmp:Rating="0"
xmp:ModifyDate="2015-02-13T20:46:51-06:00">
      <dc:title>
        <rdf:Alt>
          <rdf:li xml:lang="de-DE">GERMANS</rdf:li>
          <rdf:li xml:lang="en-GB">BRITS</rdf:li>
        </rdf:Alt>
      </dc:title>
    </rdf:Description>
  </rdf:RDF>
</x:xmpmeta>
<?xpacket end="w"?>
1176 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.title lang="de-DE" germans' ~/R.jpg
1177 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pX ~/R.jpg | xmllint -pretty 1 -
<?xml version="1.0"?>
<?xpacket begin="ï" id="W5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 4.4.0-Exiv2">
  <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
    <rdf:Description xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="" xmp:Rating="0"
xmp:ModifyDate="2015-02-13T20:46:51-06:00">
      <dc:title>
        <rdf:Alt>
          <rdf:li xml:lang="de-DE">germans</rdf:li>
```

```

<rdf:li xml:lang="en-GB">BRITS</rdf:li>
<rdf:li xml:lang="de-DE">GERMANS</rdf:li>
</rdf:Alt>
</dc:title>
</rdf:Description>
</rdf:RDF>
</x:xmpmeta>
<?xpacket end="w"?>

```

```
1178 rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

However the comparator is OK: 1178 rmills@rmillsmbp:~/gnu/exiv2/trunk \$ exiv2 -M'set Xmp.dc.title lang="DE-de" the germans are upside down' ~/R.jpg

```
1179 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pX ~/R.jpg | xmllint -pretty 1 -
```

```

<?xml version="1.0"?>
<?xpacket begin="ï" id="W5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmp:tk="XMP Core 4.4.0-Exiv2">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="" xmp:Rating="0"
xmp:ModifyDate="2015-02-13T20:46:51-06:00">

```

```

<dc:title>
<rdf:Alt>
<rdf:li xml:lang="de-DE">the germans are upside down</rdf:li>
<rdf:li xml:lang="en-GB">BRITS</rdf:li>
<rdf:li xml:lang="de-DE">GERMANS</rdf:li>
</rdf:Alt>
</dc:title>

```

```

</rdf:Description>
</rdf:RDF>
</x:xmpmeta>
<?xpacket end="w"?>
1180 rmills@rmillsmbp:~/gnu/exiv2/trunk $

```

There is also something very destructive about lang="x-default": 1180 rmills@rmillsmbp:~/gnu/exiv2/trunk \$ exiv2 -M'set Xmp.dc.title

```
lang="x-default" boring title' ~/R.jpg
```

```
1181 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -pX ~/R.jpg | xmllint -pretty 1 -
```

```

<?xml version="1.0"?>
<?xpacket begin="ï" id="W5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmp:tk="XMP Core 4.4.0-Exiv2">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="" xmp:Rating="0"
xmp:ModifyDate="2015-02-13T20:46:51-06:00">

```

```

<dc:title>
<rdf:Alt>
<rdf:li xml:lang="en-GB">BRITS</rdf:li>
<rdf:li xml:lang="de-DE">GERMANS</rdf:li>
</rdf:Alt>
</dc:title>
</rdf:Description>

```

```

</rdf:RDF>
</x:xmpmeta>
<?xpacket end="w"?>
1182 rmills@rmillsmbp:~/gnu/exiv2/trunk $

```

There seems to be a contradiction in the code concerning element index = 0 being x-default. The languages are stored in a map and the concept of index = 0 is suspect.

When I fix this, I will add a regression detector to the test suite. If I don't get this fixed in the next day or so, I'll add to the test suite to detect and report this incorrect behaviour.

I'm not convinced that the -1:0:1 result from the comparator is correct. I believe it's expected to return bool for match.

#13 - 21 Apr 2015 14:00 - Tobias E.

Robin Mills wrote:

```
| I'm not convinced that the -1:0:1 result from the comparator is correct. I believe it's expected to return bool for match.
```

It is. It should return true when `str1 < str2` and false otherwise.

#14 - 21 Apr 2015 14:29 - Robin Mills

Tobias

Thanks. I've changed the comparator in my local copy this morning. That will be submitted soon.

You'll see that I've discovered, and documented a number of issues with this feature. I've already updated the man page `src/exiv2.1` (r3709) to explain this feature more clearly with examples. The current trunk (from r3707 forward) is much improved, however I'll update the code base and the test suite with additional fixes in the next few days.

Robin

#15 - 21 Apr 2015 19:11 - Robin Mills

- Status changed from *Assigned* to *Resolved*

#16 - 21 Apr 2015 22:45 - Alan Pater

- Status changed from *Resolved* to *Assigned*

Ok, I fixed my build.

```
$ exiv2 -vVg svn
exiv2 0.24 001800 (64 bit build)
svn=3715
```

This didn't work:

```
$ exiv2 -M'set Xmp.dc.description lang="IT-IT" ciao bella' 21.jpg
$ exiv2 -g description 21.jpg
Xmp.dc.description          LangAlt  3 lang="x-default" How to fix this mess, lang="de-DE" The Exif image description, lang="it-IT"
ciao bella!!!!
```

But this did:

```
$ exiv2 -M'set Xmp.dc.description lang="it-it" ;;;clao beLLa' 21.jpg
$ exiv2 -g description 21.jpg
Xmp.dc.description          LangAlt  3 lang="x-default" How to fix this mess, lang="de-DE" The Exif image description, lang="it-IT"
;;;clao beLLa
```

By the way, the XMP specification prefers but does not define or require normalization to xx-XX.

B.4 Case-neutral xml:lang values

The values of xml:lang qualifiers, and some standard XMP properties, obey the rules for language identifiers given in IETF RFC 3066. Implementers are encouraged to pay particular attention to these aspects of IETF RFC 3066:

- Both two-letter and three-letter primary subtags as defined by ISO 639-1 and ISO 639-2 are supported.
- When a language has both an ISO 639-1 two-character code and an ISO 639-2 three-character code, the tag derived from the ISO 639-1 two-character code is used.
- All tags are treated as case-insensitive; there exist conventions for capitalization of some of them, but case is not allowed to carry meaning. For instance, ISO 3166 recommends that country codes be capitalized (MN Mongolia), while ISO 639 recommends that language codes be written in lower case (mn Mongolian).

Since the values are required to be treated as case-insensitive, XMP processors are allowed to normalize them on input and to output the normalized values. The recommendations of ISO 639 and ISO 3166 are preferred.

This document does not define or require a normalization policy. Since comparisons are case-insensitive, differences in policy can have no substantive effect.

#17 - 21 Apr 2015 23:00 - Robin Mills

Thanks for reporting this, Alan. I'll look at this tomorrow.

#18 - 22 Apr 2015 06:30 - Robin Mills

Alan

The spec doesn't require normalisation, however it require insensitivity.

| *Since the values are required to be treated as **case-insensitive**, XMP processors are allowed to normalize them*

They are treated as case-insensitive (that's the point of the comparator). I hadn't noticed that our implementation does normalize. Now that I've realised that the implementation does normalize, I think it's possible that all this effort was pointless. However I'm not going to drag the changes back out. We've strengthened our test harness and improved the man pages and I've learn about a part of Exiv2 that I've never explored. I should have insisted on a bug report with a properly defined fault before working on this.

I can't reproduce your findings. I had similar trouble yesterday and spent about about 2 hours in the debugger. Then I took a nice clean file and everything worked as I would expect. I think it's possible that files which have been processed by the old code may be damaged in some way. Try a nice new clean file. Here's what I see with a nice clean file: 932 rmills@rmillsmbp:~/gnu/exiv2/trunk \$ cp ~/DSC_5900.jpg 21.jpg

```
933 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -px 21.jpg
Xmp.xmp.Rating          XmpText  1 0
```

```

Xmp.xmp.ModifyDate      XmpText  25 2015-02-13T20:46:51-06:00
934 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.description lang="IT-IT" ciao bella' 21.jpg
935 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -px 21.jpg
Xmp.xmp.Rating          XmpText   1 0
Xmp.xmp.ModifyDate      XmpText  25 2015-02-13T20:46:51-06:00
Xmp.dc.description      LangAlt   1 lang="it-IT" ciao bella
936 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.description lang="it-it" ITALIAN' 21.jpg
937 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -px 21.jpg
Xmp.xmp.Rating          XmpText   1 0
Xmp.xmp.ModifyDate      XmpText  25 2015-02-13T20:46:51-06:00
Xmp.dc.description      LangAlt   1 lang="it-IT" ITALIAN
938 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.description everybody else' 21.jpg
939 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -px 21.jpg
Xmp.xmp.Rating          XmpText   1 0
Xmp.xmp.ModifyDate      XmpText  25 2015-02-13T20:46:51-06:00
Xmp.dc.description      LangAlt   2 lang="x-default" everybody else, lang="it-IT" ITALIAN
940 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.description lang="it-it" 21.jpg
941 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -px 21.jpg
Xmp.xmp.Rating          XmpText   1 0
Xmp.xmp.ModifyDate      XmpText  25 2015-02-13T20:46:51-06:00
Xmp.dc.description      LangAlt   1 lang="x-default" everybody else
942 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -M'set Xmp.dc.description lang="it-it" Anybody for icecream' 21.jpg
943 rmills@rmillsmbp:~/gnu/exiv2/trunk $ exiv2 -px 21.jpg
Xmp.xmp.Rating          XmpText   1 0
Xmp.xmp.ModifyDate      XmpText  25 2015-02-13T20:46:51-06:00
Xmp.dc.description      LangAlt   2 lang="x-default" everybody else, lang="it-IT" Anybody for icecream
944 rmills@rmillsmbp:~/gnu/exiv2/trunk $

```

You'll see that I added similar code to test/bugfixes-test.sh yesterday (r3713)

If you have your original, possibly damaged file, you can dump the XMP with the command: `$ exiv2 -pX file | xmllint -pretty 1 -`
 Perhaps you could have a look at that to see if there is something odd about it.

#19 - 22 Apr 2015 10:16 - Tobias E.

I don't think that it was pointless. With the following example you see that the old behaviour did indeed normalize what got put into the map, however when looking up a value the requested string was taken verbatim.

Code:

```

// g++ -W -Wall -g `pkg-config --cflags --libs exiv2` -o lang_test lang_test.cpp

#include <exiv2/easyaccess.hpp>
#include <exiv2/image.hpp>
#include <exiv2/exif.hpp>

#include <iostream>

int main(int argc, char **argv)
{
  if(argc < 2)
  {
    std::cerr << "usage: " << argv[0] << " <image filename>" << std::endl;
  }
}

```

```

return 0;
}

Exiv2::Image::AutoPtr image;
Exiv2::XmpData::const_iterator xmp_pos;

image = Exiv2::ImageFactory::open(argv[1]);
if(image.get() == 0) exit(0);
image->readMetadata();

Exiv2::XmpData &xmpData = image->xmpData();

if((xmp_pos = xmpData.findKey(Exiv2::XmpKey("Xmp.dc.title"))) != xmpData.end())
{
    std::string str = xmp_pos->toString();
    std::cout << "Xmp.dc.title: " << str << std::endl << std::endl;

    const Exiv2::LangAltValue &value = static_cast<const Exiv2::LangAltValue &>(xmp_pos->value());

    str = value.toString("de-de");
    std::cout << " de-de: " << str << std::endl << std::endl;

    for(Exiv2::LangAltValue::ValueType::const_iterator it = value.value_.begin(); it != value.value_.end(); ++it)
    {
        std::string lang = it->first;
        std::string text = it->second;
        std::cout << " " << lang << ": " << text << std::endl;
    }
}
else
    std::cout << "Xmp.dc.title missing" << std::endl;

return 1;
}

```

XMP file:

```

<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmpptk="XMP Core 4.4.0-Exiv2">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description rdf:about="" xmlns:dc="http://purl.org/dc/elements/1.1/">
<dc:title>
<rdf:Alt>
<rdf:li xml:lang="x-default">standard</rdf:li>
<rdf:li xml:lang="de-De">german</rdf:li>
<rdf:li xml:lang="en-US">us english</rdf:li>
<rdf:li xml:lang="en-GB">proper english</rdf:li>
</rdf:Alt>
</dc:title>
</rdf:Description>
</rdf:RDF>
</x:xmpmeta>

```

Output without your change:

```
Xmp.dc.title: lang="x-default" standard, lang="de-DE" german, lang="en-GB" proper english, lang="en-US" us english
```

```
de-de:
```

```
de-DE: german
```

```
en-GB: proper english
```

```
en-US: us english
```

```
x-default: standard
```

So looking up de-de isn't returning anything which is wrong.

PS: I know that the XMP file is malformed, the x-default value has to be present with a real language identifier, too. But it doesn't matter here.

#20 - 22 Apr 2015 11:20 - Robin Mills

Gosh, Tobias. Your words are very helpful indeed. I was thinking "Now that I know what this lang feature is about, I ought to go back and see how things used to be". You've made that journey and provided the evidence that this has been worth the effort.

Oh and there's another useful by-product of this effort. I updated sample/exiv2json.cpp to respect AltLang and create a JSON object.

<http://dev.exiv2.org/issues/1054#note-3>

I haven't understood your PS:

```
| PS: I know that the XMP file is malformed, the x-default value has to be present with a real language identifier, too. But it doesn't matter here.
```

Is there something that requires attention here? xmllint doesn't report the XML as malformed: 529 rmills@rmillsmbp:~/gnu/exiv2/team/meetings

```
$ exiv2 -pX ~/R.jpg | xmllint -pretty 1 -
```

```
<?xml version="1.0"?>
```

```
<?xpacket begin="□" id="W5M0MpCehiHzreSzNTczkc9d"?>
```

```
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmpptk="XMP Core 4.4.0-Exiv2">
```

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
```

```
<rdf:Description xmlns:xmp="http://ns.adobe.com/xap/1.0/" xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="" xmp:Rating="0"
```

```
xmp:ModifyDate="2015-02-13T20:46:51-06:00">
```

```
<dc:title>
```

```
<rdf:Alt>
```

```
<rdf:li xml:lang="x-default">all other languages</rdf:li>
```

```
<rdf:li xml:lang="en-GB">the Brits are in the bar</rdf:li>
```

```
<rdf:li xml:lang="de-DE">wee german</rdf:li>
```

```
</rdf:Alt>
```

```
</dc:title>
```

```
</rdf:Description>
```

```
</rdf:RDF>
```

```
</x:xmpmeta>
```

```
<?xpacket end="w"?>
```

```
530 rmills@rmillsmbp:~/gnu/exiv2/team/meetings $
```

By the way, I have dual citizenship. I am British/European and a US Citizen. Being Scottish, I don't speak proper English in England or the States!

If you and Alan are happy, I'd like to mark this resolved.

#21 - 22 Apr 2015 13:52 - Tobias E.

The XML itself is well formed, but the XMP specs say in part 1, chapter 8.2.2.4 "Language Alternative"

An `xml:lang` value of "x-default" may be used to explicitly denote a default item. If used, the "x-default" item shall be first in the array and its simple text value should be repeated in another item in which `xml:lang` specifies its actual language. However, an "x-default" item may be the only item, in which case there is only a default value in no defined language.

That's nothing `exiv2` itself could handle I guess. Even if a user specified a default value and at least one language it would be unknown what language the default would be in.

#22 - 22 Apr 2015 14:24 - Alan Pater

I get the same result as Robin with his example commands:

```
asp@exiv2:~$ exiv2 -g description clean.jpg
Xmp.dc.description          LangAlt  2 lang="x-default" everybody else, lang="it-IT" Anybody for icecream
```

It's when I try to take it a bit further that it stops working for me:

```
asp@exiv2:~$ exiv2 -M'set Xmp.dc.description lang="de-DE" beer' clean.jpg
asp@exiv2:~$ exiv2 -g description clean.jpg
Xmp.dc.description          LangAlt  3 lang="x-default" everybody else, lang="de-DE" beer, lang="it-IT" Anybody for icecream
```

```
asp@exiv2:~$ exiv2 -M'set Xmp.dc.description lang="DE-de" oktoberfest' clean.jpg
asp@exiv2:~$ exiv2 -g description clean.jpg
Xmp.dc.description          LangAlt  3 lang="x-default" everybody else, lang="de-DE" beer, lang="it-IT" Anybody for icecream
```

```
asp@exiv2:~$ exiv2 -M'set Xmp.dc.description lang="IT-it" icecream for everybody' clean.jpg
asp@exiv2:~$ exiv2 -g description clean.jpg
Xmp.dc.description          LangAlt  3 lang="x-default" everybody else, lang="de-DE" beer, lang="it-IT" Anybody for icecream
```

```
asp@exiv2:~$ exiv2 -pX clean.jpg | xmllint -pretty 1 -
```

```
<?xml version="1.0"?>
<?xpacket begin="□" id="W5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 4.4.0-Exiv2">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="">
<dc:description>
<rdf:Alt>
```

```

<rdf:li xml:lang="x-default">everybody else</rdf:li>
<rdf:li xml:lang="it-IT">icecream for everybody</rdf:li>
<rdf:li xml:lang="de-DE">beer</rdf:li>
<rdf:li xml:lang="it-IT">Anybody for icecream</rdf:li>
</rdf:Alt>
</dc:description>
</rdf:Description>
</rdf:RDF>
</x:xmpmeta>
<?xpacket end="w"?>

```

Is it just me, or can this be reproduced?

#23 - 22 Apr 2015 15:06 - Robin Mills

Alan:

The file "clean.jpg" is damaged. It has two lang="it-IT" elements. Do you have a work-flow to create such a file starting from test/data/exiv2-empty.jpg

I had a file like that yesterday and spent 2 hours in the debugger with it. You might be able to fix it with a new feature I added to remove a language definition: \$ exiv2 -M'set Xmp.dc.description lang="it-IT"

Setting a language (including x-default) to the empty string simply removes the language. If you really want a language with an empty string, you'll need to set it to a string with a blank or a null byte or something. I think it's desirable to be able to remove languages without removing the whole key and re-inserting the languages you want to keep using the *dentist's method*. You know the *dentist's method*: you pull them out, one at a time!

Tobias:

Thanks for the clarification about "x-default". Exiv2 isn't in the guessing business. We're engineers and want things to be deterministic. The idea that "x-default" must be first makes total sense. If you're reading the XMP with an xml-player (such as expat events), you store the value when you see "x-default" and modify it when/if your language rolls past. Perfect. No backtracking necessary.

#24 - 22 Apr 2015 15:19 - Alan Pater

The work flow is from comment #18. Let me start from scratch and look at the xml ...

```

asp@exiv2:~$ cp ~/src/exiv2/trunk/test/data/exiv2-empty.jpg clean.jpg

asp@exiv2:~$ exiv2 -M'set Xmp.dc.description lang="IT-IT" ciao bella' clean.jpg

asp@exiv2:~$ exiv2 -pX clean.jpg | xmllint -pretty 1 -

<?xml version="1.0"?>
<?xpacket begin="ï" id="W5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 4.4.0-Exiv2">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="">
<dc:description>
<rdf:Alt>
<rdf:li xml:lang="it-IT">ciao bella</rdf:li>
</rdf:Alt>

```

```
</dc:description>
</rdf:Description>
</rdf:RDF>
</x:xmpmeta>
<?xpacket end="w"?>
```

```
asp@exiv2:~$ exiv2 -M'set Xmp.dc.description lang="it-it" ITALIAN' clean.jpg
```

```
asp@exiv2:~$ exiv2 -pX clean.jpg | xmllint -pretty 1 -
```

```
<?xml version="1.0"?>
<?xpacket begin="□" id="W5M0MpCehiHzreSzNTczkc9d"?>
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="XMP Core 4.4.0-Exiv2">
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
<rdf:Description xmlns:dc="http://purl.org/dc/elements/1.1/" rdf:about="">
<dc:description>
<rdf:Alt>
<rdf:li xml:lang="it-IT">ciao bella</rdf:li>
<rdf:li xml:lang="it-IT">ITALIAN</rdf:li>
</rdf:Alt>
</dc:description>
</rdf:Description>
</rdf:RDF>
</x:xmpmeta>
<?xpacket end="w"?>
```

If that is not what you get, then my system is broken.

#25 - 22 Apr 2015 15:39 - Alan Pater

OK, sorry, it was my build again. A clean and fresh checkout and build of trunk and everything is working as it should be. I even tested with Spanish and Aymara. Everything works good.

#26 - 22 Apr 2015 16:25 - Robin Mills

Right. I think we've beaten this issue (and each other) to destruction here. Very good team-work, Gentlemen. I'm going to set the status to "Resolved".

"Resolved" means we intend no further work on this. However should something arise, it'll be assigned for further activity. During review prior to shipping it will be set to "Closed" and never opened again. If something raises after being "closed", we'd open a new issue and reference this.

#27 - 22 Apr 2015 20:00 - Robin Mills

- Status changed from Assigned to Resolved

#28 - 08 May 2015 15:59 - Robin Mills

- % Done changed from 0 to 100

#29 - 21 Jun 2015 16:39 - Andreas Huggel

- Status changed from Resolved to Closed