

## Exiv2 - Bug #1116

### Issues with namespace 'video'

08 Sep 2015 17:42 - Robin Mills

<b>Status:</b>	Closed	<b>Start date:</b>	08 Sep 2015
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Andreas Huggel	<b>% Done:</b>	100%
<b>Category:</b>	xmp	<b>Estimated time:</b>	4.00 hours
<b>Target version:</b>	0.26		
<b>Description</b>			
This issue relates to the namespace 'video' and probably also applies to 'audio'			
It is listed as available:			
<pre>\$ exiv2 -vV -g video exiv2 0.25 001900 (64 bit build) enable_video=0 xmlns=video:http://www.video \$</pre>			
<b>1) Cannot use it "out of the box"</b>			
<pre>\$ curl --silent -O http://dev.exiv2.org/attachments/download/805/DSC_7154.jpg \$ exiv2 -M"Xmp.video.title movie" DSC_7154.jpg -M option 1: Invalid command `Xmp.video.title' exiv2: Error parsing -M option arguments Usage: exiv2 [ options ] [ action ] file ...</pre>			
Manipulate the Exif metadata of images.			
<pre>\$</pre>			
It can be used if registered:			
<pre>\$ exiv2 -M"reg video http://www.video" -M"set Xmp.video.title movie" DSC_7154.jpg \$ exiv2 -px DSC_7154.jpg Xmp.xmp.Rating                XmpText    1    0 Xmp.xmp.ModifyDate            XmpText    25   2015-07-16T20:25:28+01:00 Xmp.video.title                XmpText    5    movie Xmp.dc.description            LangAlt     1    lang="x-default" Classic View \$</pre>			
<b>2 Throws an exception in exiv2json:</b>			
<pre>\$ bin/exiv2json -x DSC_7154.jpg Caught Exiv2 exception 'XMP Toolkit error 9: Fatal namespace map problem' \$</pre>			
<b>Related issues:</b>			
Related to Exiv2 - Feature #640: method is missing in Exiv2 to get list of kn...		<b>Closed</b>	<b>18 Jul 2009</b>

### Associated revisions

#### Revision 3933 - 08 Sep 2015 22:17 - Robin Mills

#1116. Fix submitted.

#### Revision 3981 - 10 Oct 2015 07:10 - Andreas Huggel

#1116: Reverted changes to Basiclo::Basiclo() introduced in r3933

#### Revision 3982 - 10 Oct 2015 07:19 - Andreas Huggel

#1116: Register missing namespaces (audio, video), changed xapG to xmpG in accordance with the XMP specs.

## History

---

### #1 - 08 Sep 2015 20:07 - Alan Pater

Several random thoughts:

Does the 'video' namespace only apply to video files?

Is video support enabled in the build?

Is video support read only?

If that is the case, then trying to set a Xmp.video property should not work.

### #2 - 08 Sep 2015 22:17 - Robin Mills

- % Done changed from 0 to 70

- Estimated time set to 4.00 h

Fix submitted: [r3933](#).

The command/message in **Fault 1** in the issue report is incorrect. It should be:

```
$ exiv2 -M"set Xmp.video.title movie" DSC_7154.jpg
-M option 1: Invalid key `Xmp.video.title'
exiv2: Error parsing -M option arguments
Usage: exiv2 [ options ] [ action ] file ...
```

```
Manipulate the Exif metadata of images.
662 rmills@rmillsmbp:~/gnu/exiv2/trunk $
```

**Fault 1** (Invalid key Xmp.video.title) is caused by libexiv2 namespaces not being registered with XMPsdk.

Exiv2 has no library initialization function. If we had this, we could register namespaces uniquely known to libexiv2 with XMPsdk. I've hijacked Basiclo::Basiclo to call XmpProperties::registeredNamespaces() to force this. This imposes overhead on every Io object.

**Fault 2** (exiv2json throwing exception XMP Toolkit error 9: Fatal namespace map problem) has two causes.

a) Failure to register libexiv2 namespace (as **Fault 1**).

However even explicitly calling XmpProperties::registeredNamespaces() from exiv2json.cpp/main() was insufficient to fix this.

b) Duplicated namespace.

We were registering the namespace 'video' with two URIs <http://www.video> and <http://www.video/>

### #3 - 09 Sep 2015 03:37 - Andreas Huggel

Exiv2 has no library initialization function. If we had this, we could register namespaces uniquely known to libexiv2 with XMPsdk. I've hijacked Basiclo::Basiclo to call XmpProperties::registeredNamespaces() to force this. This imposes overhead on every Io object.

We do, and it does exactly that; register namespaces in XMP SDK: [Exiv2::XmpParser::initialize](#). There is also a comment in properties.cpp that says new namespaces need to be registered with XMP-SDK there:

```
extern const XmpNsInfo xmpNsInfo[] = {
    // Schemas - NOTE: Schemas which the XMP-SDK doesn't know must be registered in XmpParser::initial
    // ize - Todo: Automate this
    { "http://purl.org/dc/elements/1.1/", "dc", xmpDcInfo, N_("Dublin Core
    schema") },
```

When I last looked (in connection with [#640](#) the other day), this was simply one of the namespaces missing there. I wanted to fix that: those namespaces that Exiv2 knows and XMP-SDK doesn't just need to be registered there (I think), but those that XMP-SDK has and Exiv2 doesn't, will require a bit more thought and work. For the current implementation of XmpProperties::registerNamespaces(), registering the missing namespaces in XMP-SDK in XmpParser::initialize() should suffice.

See also [http://dev.exiv2.org/projects/exiv2/wiki/Thread\\_safety](http://dev.exiv2.org/projects/exiv2/wiki/Thread_safety)

### #4 - 09 Sep 2015 08:57 - Robin Mills

- Status changed from Assigned to Feedback

- Assignee changed from Robin Mills to Andreas Huggel

- % Done changed from 70 to 80

Thanks, Andreas. I'm pushing this to you for comment (status 'Feedback').

I'm not happy with calling `XmpParser::initialize()` from `main()`. Do we have to also in-line the `CriticalSection` stuff? This is overly complicated and obscure.

Can we add a new API `Exiv2::libraryInitialize()` function and I'll call this from `exiv2json.cpp/main()`?

I will restore `Basiclo::Basiclo()` to the empty in-line function in `include/exiv2/basiclo.hpp`: `Basiclo::Basiclo{}`

`Exiv2::libraryInitialize()` should call a new API `Exiv2::XmpProperties::initialize()`

`XmpProperties::initialize()` can call `XmpParser::initialize()`, because he's a friend class. And he should include the following code, which can then be removed from `XmpProperty::registeredNamespaces()`

```
for (unsigned int i = 0; i < EXV_COUNTOF(xmpNsInfo); ++i) {
    Exiv2::XmpParser::registerNs(xmpNsInfo[i].ns_, xmpNsInfo[i].prefix_);
}
```

#### #5 - 10 Sep 2015 02:36 - Andreas Huggel

The idea of looping through `xmpNsInfo[]` to register the namespaces in the XMP parser automatically is good, I like that. But let's take this step by step. The real issue is that `Exiv2` and `XMPSDK` are not in sync, let me fix that first: `Exiv2` and `XMPSDK` must be in agreement of what are the known namespaces. That will make sure we don't hit namespace-related obscure errors and will make `XmpProperties::registeredNamespaces()` trivial.

As for the API:

You do not need to call `XmpParser::initialize()` from your `exiv2json.cpp main()`. `Exiv2` will take care of that. Only multithreaded apps have to be careful. They need to follow the simple guidelines in the `Thread-safety Wiki` article, then they are fine. For now, we need `XmpParser::initialize()` in the new function to list the registered namespaces, because it uses the XMP parser.

We currently only have `XmpParser::initialize()` because that is all we need: the XMP parser (`XMPSDK`) needs to be initialized. `Exif` and `IPTC` do not need any initialization code, phew! Generally, an initialization function in a C++ library is bad. It should be avoided at all costs. That's why there is no function to initialize the library. I think we should not add one, unless we really really have to, and I hope we never will.

Let's try hard to keep the API as stable and as small as possible. (API: Consider any code change in the `include/exiv2/*` files an API change.) Every time we modify or delete an API, we create an incompatible release. Every API we do **not** add, we will not have to modify or delete later. We have hardly made changes since the last release, but the upcoming release is already binary incompatible.

`Exiv2` is a library to deal with image metadata. We should provide only a very specific and consistent API related to this task and resist the temptation to add kitchen-drawer all-purpose interfaces, as convenient they may seem. Every application has its own utilities and every developer has her own preferences.

The API is already quite overwhelming as it is. Imagine you're new to `Exiv2` and look at our [class list](#). Where do you start? I've had comments "Exiv2 only deals with image metadata, why do you need 100 classes for that?" (97 at the moment, and only counting those in the public API.) The question is valid and difficult to answer, especially if you consider how many `Exiv2` classes you actually need in an average application.

#### #6 - 10 Sep 2015 19:41 - Alan Pater

When adding previously unknown XMP namespaces, I've been listing them in `xmp.cpp`

<http://dev.exiv2.org/projects/exiv2/repository/entry/trunk/src/xmp.cpp#L396>

Do these changes mean that those entries are not longer needed?

#### #7 - 10 Sep 2015 21:43 - Robin Mills

- Status changed from *Feedback* to *Assigned*

Thank Alan and Andreas for your comments.

Alan:  
The calls you make in `XmpParser::initialize()` register those namespace directly with `XMPSdk` using `SXMPMeta::RegisterNamespace()`.

The for loop above is currently in `properties.cpp/XmpProperties::registeredNamespaces()` which calls `xmp.cpp/Exiv2::XmpParser::registerNs()` which calls `SXMPMeta::RegisterNamespace()`.

Andreas:  
I'm sorry to have misunderstood the initialization code. I was under the impression that we did not have a library initialization function. And while I understand your point of view that you don't wish to see a function such as `Exiv2::libraryInitialize()`, we none-the-less have to sync those disparate sources of namespaces. And if we don't push this responsibility on the user, we should assume it in `Basiclo::Basiclo()` - or some such gateway to `libexiv2`. I'll leave you to resolve this. I set this to 70% complete with the intention of revisiting `Basiclo::Basiclo()` to install a lower overhead synchronisation function (possibly `XmpParser::initialize()`).

I'm going to leave you to fix this as you think best. You may begin by rolling back the changes to `Basiclo.{cpp|hpp}` in [r3933](#). The changes in [r3933](#) to `properties.cpp` concerning the declaration of 'video' and 'audio' are essential to fix **fault 2** above.

Gentlemen:  
To help a new user get quickly started with `libexiv2`, I believe our sample applications are of great importance. I've discussed giving the man pages a makeover for this very purpose. [#1061](#). You are both aware that I endeavour to help users on the Forum. We're providing a rich mix of solid

code-base and build, api documentation, good man-pages, simple to understand samples and on-line support.

**#8 - 24 Apr 2017 08:17 - Robin Mills**

- *Status changed from Assigned to Closed*

- *% Done changed from 80 to 100*