darktable - Feature #12500

Format conversion during import for long term archive

12/27/2018 07:51 AM - Vincent Fregeac

Status:	Incomplete		**Start date:**	12/27/2018
Priority:	Low		**Due date:**	
Assignee:			**% Done:**	20%
Category:	Lighttable		**Estimated time:**	0.00 hour
Target version:				
Affected Version:	git master branch		**bitness:**	64-bit
System:	all		**hardware architecture:**	amd64/x86

Description

RAW files are, currently, the digital equivalent of film negatives but, in a century or so, will be the equivalent of the photographic plate from your great-great uncle you rediscover in the attic of an old family house. For that reason, it's preferable to have all your RAW files in the same format, preferably a openly documented format.

As far as I know, DNG is still the only RAW format with libraries to write this format (including in Darktable for HDR). It is not perfect (it strips the more exotic settings of some camera) but I don't think there is another, better RAW archive format for now. Plus Darktable already write DNGs so it would be easier to implement.

P.S.: I know Adobe DNG Converter can be used with Wine but I don't consider running a Windows software with Wine a long term solution. A native Linux solution is more lasting solution, and probably more reliable. Plus, since Darktable already reads proprietary RAW formats, and already write DNG, it shouldn't require too much to implement.

History

#1 - 12/27/2018 08:50 AM - Roman Lebedev

- Status changed from New to Incomplete

- % Done changed from 0 to 20

Vincent Fregeac wrote:

> RAW files are, currently, the digital equivalent of film negatives but, in a century or so, will be the equivalent of the photographic plate from your great-great uncle you rediscover in the attic of an old family house. For that reason, it's preferable to have all your RAW files in the same format, preferably a openly documented format.

Yes.

> As far as I know, DNG is still the only RAW format with libraries to write this format (including in Darktable for HDR). It is not perfect (it strips the more exotic settings of some camera) but I don't think there is another, better RAW archive format for now. Plus Darktable already write DNGs so it would be easier to implement.

> P.S.: I know Adobe DNG Converter can be used with Wine but I don't consider running a Windows software with Wine a long term solution. A native Linux solution is more lasting solution, and probably more reliable. Plus, since Darktable already reads proprietary RAW formats, and already write DNG, it shouldn't require too much to implement.

DNG conversion is **extremely** lossy.
darktable HDR dng writer is super extremely partial.

In fact, if after such conversion (or after modification of the EXIF in the original raw),
the file no longer loads in darktable, it will not even be considered a bug.
Because then you have intentionally broken the raw file.

There is a **long** road from being able to read raws, to being able to write DNG's,
to being able to losslessly convert to DNG. digicam has(had?) some DNG converter.

But the main question is, what do you think will happen in a century?
Why, all of the sudden, all the knowledge and the code to deal with non-DNG raws will be lost,
but the the DNG knowledge will remain?
Also, do you envision that all the old source code (of e.g. the current darkable version)
will vanish without a trace, completely?

TLDR:
1. the premise is questionable.
2. it **much** more complex than it would appear
3. i think darktable already has a lot of rather-broken-but-kind-of-works things, maybe not add another one to the list :)

#2 - 12/27/2018 11:05 PM - Vincent Fregeac

Hi Roman,

The main reason I favor DNG for my RAW archive is that proprietary RAW formats change (Canon is rolling out CR3, it's third RAW format) and, sadly, companies tend to drop the support of their own old format to force you to buy their latest camera. On the other hand, DNG was initially developed as a long-term archive format, like ISO-9660 for CDs, so it's a format that can be amended but it's core is not supposed to change (same as ISO-9660, amended in 2013, but being able to read ISO-9660:2013 means you can also read ISO-9660:1988). Also, their is no incentive for Adobe to change DNG without backward compatibility because they would lose their credibility as the provider of solutions for professionals (whether they are or not, that's how they want to be perceived). Because of that, their is more chances that software in the future will still be able to read old versions of DNG than old proprietary formats which are not supported anymore by the companies who created them. Of course, it will still possible to analyse the structure of these old formats like it has been done presently, but it's a time-consuming process I'm not sure someone would invest in for 5-10 old photos he found on an old USB key.

On the other hand, if the DNG implementation in Darktable is still in a kind-of-work-but-only-partially stage, I agree with you, we should wait until a complete, preferably open-source, DNG conversion library is available. The Adobe DNG Converter with Wine can do in the meantime, even if it's not an ideal solution.

So either close this feature request, or leave it open but on the back-burner.

#3 - 12/28/2018 08:19 AM - Roman Lebedev

Vincent Fregeac wrote:

> Hi Roman,
>
> The main reason I favor DNG for my RAW archive is that proprietary RAW formats change (Canon is rolling out CR3, it's third RAW format) and, sadly, companies tend to drop the support of their own old format to force you to buy their latest camera. On the other hand, DNG was initially developed as a long-term archive format, like ISO-9660 for CDs, so it's a format that can be amended but it's core is not supposed to change (same as ISO-9660, amended in 2013, but being able to read ISO-9660:2013 means you can also read ISO-9660:1988). Also, their is no incentive for Adobe to change DNG without backward compatibility because they would lose their credibility as the provider of solutions for professionals (whether they are or not, that's how they want to be perceived). Because of that, their is more chances that software in the future will still be able to read old versions of DNG than old proprietary formats which are not supported anymore by the companies who created them. Of course, it will still possible to analyse the structure of these old formats like it has been done presently, but it's a time-consuming process I'm not sure someone would invest in for 5-10 old photos he found on an old USB key.

and, sadly, companies tend to drop the support of their own old format to force you to buy their latest camera.
...
Because of that, their is more chances that software in the future will still be able to read old versions of DNG than old proprietary formats which are not supported anymore by the companies who created them.

I'mm not sure what you are saying here. the current code used by dt to load these raws
is certainly **not** maintained/contributed/supported by the camera manufacturer companies.
Therefore it is rather irrelevant what manufacturers do/don't do.
Also, the old versions of code shouldn't magically disappear from everywhere.

On the other hand, if the DNG implementation in Darktable is still in a kind-of-work-but-only-partially stage, I agree with you, we should wait until a complete, preferably open-source, DNG conversion library is available. The Adobe DNG Converter with Wine can do in the meantime, even if it's not an ideal solution.

So either close this feature request, or leave it open but on the back-burner.

#4 - 12/28/2018 09:31 PM - Vincent Fregeac

What I'm saying is, since camera manufacturers do not always support their own old formats, we have to rely on volunteers who develop and maintain open-source support for these formats. But these developers have a life, their priority changes and, because of that, the vast majority of the open-source tools that were available 10 years ago are not maintained anymore. Some, but rarely, still have an active website where you can find the old source code, but many have been closed because no one was ready to keep paying for hosting a software that haven't been maintained for 5 years or more. And I am just talking about the last 10 years. So, yes, open-source code also disappears, not by magic, just because life.

Now, just try to read a betamax tape today. Yes, the format is still known and there are still some people who maintain or recreate betamax tape decks, but you probably won't go through the trouble of finding or building one just for 2 or 3 old tapes you found in the attic, you will probably just ditch them without even knowing what they contain. On the other hand, reading an early ISO-9660 CD is no trouble, even a recent DVD writer still reads the old ISO-9660:1988. That's the difference between a disposable proprietary format and an archive format: how much trouble you have to go through to be able to read it several decades later.

That being said, I was just suggesting it if it was an easy implementation, as DNG is still far from a perfect archive format (I'd rather trust a format backed by ISO than backed only by Adobe), but I still trust it more than a proprietary format that changes every 10 years without backward compatibility. But considering there is no guarantee DNG will become the ISO standard format RAW images need, it is probably not worth investing time in a DNG conversion library for darktable if the existing implementation is not stable or production ready (plus someone may develop one, and it will only require an integration).

#5 - 12/28/2018 09:43 PM - Mica S

Vincent Fregeac wrote:

> Now, just try to read a betamax tape today. Yes, the format is still known and there are still some people who maintain or recreate betamax tape decks, but you probably won't go through the trouble of finding or building one just for 2 or 3 old tapes you found in the attic, you will probably just ditch them without even knowing what they contain. On the other hand, reading an early ISO-9660 CD is no trouble, even a recent DVD writer still reads the old ISO-9660:1988. That's the difference between a disposable proprietary format and an archive format: how much trouble you have to go through to be able to read it several decades later.

The betamax logic here is faulty. RAW files are software, not hardware. You don't need to track down the camera if you already have the RAW files. If you are concerned about archiving things, you should store a copy of the source code in your archive alongside the RAW files. Then when you need to access the RAW files in this dystopian future, you'll have the source to compile and you'll be able to chug away.

> That being said, I was just suggesting it if it was an easy implementation, as DNG is still far from a perfect archive format (I'd rather trust a format backed by ISO than backed only by Adobe), but I still trust it more than a proprietary format that changes every 10 years without backward compatibility. But considering there is no guarantee DNG will become the ISO standard format RAW images need, it is probably not worth investing time in a DNG conversion library for darktable if the existing implementation is not stable or production ready (plus someone may develop one, and it will only require an integration).

I think if you spoke to some people about digital archiving, they'd tell you to archive the source material and the final material. I don't think they'd say "convert to DNG." Look at what adobe has done with its other "open" standards: PDF is a disaster with all sorts of proprietary adobe extensions to its open core. What makes you think DNG is any different?

#6 - 12/28/2018 09:57 PM - Aurélien PIERRE

> What I'm saying is, since camera manufacturers do not always support their own old formats, we have to rely on volunteers who develop and maintain open-source support for these formats.

If old formats have opensource decoders, we don't care about manufacturers anymore. Old formats don't need more work once they have a working decoder in darktable, it's the new ones that require work. And darktable doesn't rely on manufacturers code/libs to do that, it uses its own rawspeed lib (Roman's baby).

I don't get where you go with your betamax tape… Digital files only need a codec, not a device, to be read. dt's decoders are opensource, they won't stop to work overnight, and if they do, the code can be fixed without any third-party action.

#7 - 12/28/2018 10:15 PM - Vincent Fregeac

Source code doesn't read RAW files, executables do. The betamax analogy is only faulty in your eyes because going from source code to an executable is so common for a developer you don't even realize anymore you have to build software the same way you have to build hardware, and, in the same way building hardware is not something everyone can do, building an executable from source code requires an expertise most people don't have.

So, I just archive the executable too, right? But how many OSes today support 8bits executable compiled in the early 80s for the Sinclair Zx81 or the first gen Amiga? And it's already not that easy to find a working Sinclair Zx81, imagine 50 years from now. If you are curator of museum collection, may be.

It seems simple if you only look 5-10 years ahead, not so simple if you start counting decades. The x86 architecture didn't even exist 50 years ago, and seeing how the ARM architecture is spreading, we can't be sure the x86 architecture will still exist 50 years from now. And it's still less than a lifetime. For you, it won't be a problem, you will cross-compile the old source code for another architecture and another OS, fix a few bugs, and voilà. But your kids or grand-children may not have the expertise you have.

#8 - 12/28/2018 10:18 PM - Roman Lebedev

Aurélien PIERRE wrote:

> What I'm saying is, since camera manufacturers do not always support their own old formats, we have to rely on volunteers who develop and maintain open-source support for these formats.

> If old formats have opensource decoders, we don't care about manufacturers anymore. Old formats don't need more work once they have a working decoder in darktable, it's the new ones that require work. And darktable doesn't rely on manufacturers code/libs to do that,

> it uses its own rawspeed lib (Roman's baby).

Factually incorrect. Klaus Post was the original author/maintainer, many developed/contributed. I'm only the current maintainer.

> I don't get where you go with your betamax tape… Digital files only need a codec, not a device, to be read. dt's decoders are opensource, they won't stop to work overnight, and if they do, the code can be fixed without any third-party action.

#9 - 12/28/2018 10:34 PM - Aurélien PIERRE

Vincent Fregeac wrote:

> It seems simple if you only look 5-10 years ahead, not so simple if you start counting decades. The x86 architecture didn't even exist 50 years ago, and seeing how the ARM architecture is spreading, we can't be sure the x86 architecture will still exist 50 years from now. And it's still less than a lifetime. For you, it won't be a problem, you will cross-compile the old source code for another architecture and another OS, fix a few bugs, and voilà. But your kids or grand-children may not have the expertise you have.

That's why we have packagers, repositories, Linux distributions and releases… And why we try to have portable C/C++ code. I feel like the DNG archiving is only adding one more standard, trying to replace all the other standards.

#10 - 12/28/2018 10:53 PM - Vincent Fregeac

Aurelien, I hope darktable will still be there 50 years from now, but I don't know many software from 50 years ago that are still available today and can run on commonly available modern hardware.

The difference between our point of view is: I am looking decades behind, not years, to plan for decades ahead, probably because I appreciate having negatives and photographic plates that were taken many decades ago by my grandfathers and great-grandfathers. Relying on current software tools only work if you look 10-20 years ahead at most, or if you are a developer who can fix source code and build from it... and you are certain all your kids and grand-children will also be developers and will still code in the same language.

If you don't like the betamax analogy, just try to run some old source code for Sinclair Zx81 on the hardware you have today, but not as a developer, as a regular person who use softwares but do not necessarily have the expertise to fix it or cross-compile them. That's what relying on the permanence of current source code means. One day, darktable will probably be the equivalent of source code for a Sinclair Zx81. And not everyone can, or know someone who can fix and cross-compile old Zx81 code so it can be used on his current computer.

But, as I said, I made that suggestion because I thought the DNG implementation for HDR was stable and it would be therefore relatively easy to implement. If it's not, and this issue has to stay on the back-burner anyway, we can just drop it. I don't want to waste anymore of your time with it. But if you want the next generations to enjoy the photo you take, I would recommend you don't rely on the permanence of any current software or codec, open-source or not. Just my opinion, you have the right to disagree.

#11 - 12/29/2018 01:25 AM - Vincent Fregeac

Sorry, just one last thing, for future consideration: There is in fact an ISO standard for RAW image format and I was wrong when I said DNG, if not perfect is the lesser of two evils. I was looking for another format for archiving my RAW files, since I would rather not keep relying on a closed source software who doesn't even run natively on the OS I use, and found that ISO 12234-2 is in fact the ISO standard for RAW image (although it can also be used for developed images). It's better known as TIFF/EP, not be confused with the more common Adobe TIFF format.

So, converting RAW to TIFF/EP may be a better long-term approach than DNG:
- It is an already accepted long-term standard RAW format supported by ISO;
- TIFF/EP is designed as a RAW format, so it can store all the information to process the sensor data (except, apparently, Sigma's Foveon X3 sensor, but it's really a corner case);
- It supports xmp metadata natively;

- It can be used for RAW image as well as developed images, either with no compression, lossless compression or lossy compression, so it can be the standard format during the entire workflow of an image (except maybe when you publish an image on the web, but the web is not exactly an archiving media).

As a side note, the article where I found the information also mentionned that the [US] Library of Congress Collections identifies raw-file formats as "less desirable file formats". And I think we can agree that they know a thing or two about preserving information with a long term perspective. They accept DNG as an alternative format (it is build on TIFF/EP, as well as NEF for example) but TIFF/EP remains the standard archiving format.

The only "difficulty" is TIFF/EP do not support Exif, it replaces it, so Exif metadata have to be converted to TIFF/EP. I can research if there is a library to write TIFF/EP with all the original RAW metadata, with proven reliability, that is, if you agree to consider implementing a conversion to ISO 12234-2, for those who trust ISO and the US Library of Congress more than Github when it comes to long-term permanence.

Now, I understand that not everyone is really concerned about what their images will become 100 years from now. Most people, and I supposed you too, have thought about it, maybe not as far as 100 years, but didn't necessarily had the time, or the will, to really research it and consider the implications of creating images (or documents in general) that cannot be read natively by humans (in opposition to negative films or printed documents). Because some old images from my ancestors, as well as other old family documents, were apparently very valuable to regional or national archive organisation, I did spent quite some time researching how to ensure, or at least do my best, that future generations will still be able to still read these documents or images. And I'm not the only one, so I didn't have to come up with the ideas I expressed in my feature requests, they are in general recommendations from people much more knowledgeable than me when it comes to long-term archives. And, yes, sometime I miss something, like the ISO 12234-2 standard I didn't think was applicable to RAW images, but I try to only request features that have become the general consensus. Now, you can consider long-term permanence of images a corner case and ignore it, but darktable is the only real FOSS alternative who try to offer a reasonably complete image workflow, and is not only used by wedding or travel photographers (nothing wrong with that, some are really talented photographers, they just don't need to store their images for the next century).

So, may be not right now, but implementing a conversion to a standardized long-term format (TIFF/EP, or DNG since the Library of Congress consider DNG an acceptable alternative) should be in the roadmap of darktable. If you are willing to consider it, I will do the first part, find a library with proven reliability, and/or test it extensively on my own images, before asking you to integrate the library in darktable (That's already done for exiftool and DNG, but not yet for TIFF/EP if you prefer to go with an already accepted ISO standard).

#12 - 12/29/2018 06:56 AM - dar ix

I find it amusing that you really think over the time frames that you specified ... that one format parser might be preserved better than another :)

#13 - 12/29/2018 08:07 AM - Vincent Fregeac

Sorry, my bad, I obviously misrepresented the concept I was trying to convey. I do not favor any format, parser, or codec. I am just relaying the concept that relying on several, short-time, proprietary formats, is not the preferred approach for long term archives, even though these formats are still supported by open-source solutions years after the company who created these format stopped supporting them.

Although digital information has a rather short history, compared to "analog formats" or formats that can be "read" natively with human senses, consensus about digital archives have already emerged and they all share the same basic concepts: an open format, preferably designed for long-term archiving, ideally supported by an international standard organisation, and storing all the relevant information in a single container, is the current consensus.

And, although the idea that RAW images are assets worth archiving is a recent concept, there is already a consensus about RAW image permanence, the same consensus as any other digital information: open format, designed for long-term archive, ideally supported by an international organisation or at least by a major international orgranisation, and able to store any kind of present or future information (as far as people specialized

in that field can imagine future requirements for metadata).

Today, for RAW images, it seems there is only two contenders: the TIFF/EP format meets most of the criteria but the Foveon X3 sensor proved that it may not be the ideal flexible format able to store any current and future RAW images. DNG, based itself on TIFF/EP, is more flexible and able to store the information for every current sensor technology, including the unusual approach of the Foveon X3, but, although it's backed by strong internation organisation, it is not yet an ISO standard.

But there is one aspect where the consensus is clear and definitive: proprietary RAW format is the less prefered format, despite the support of the open-source community. And the reason is, open-source, even when it is backed by large organisation like Canonical or Red Hat, has not yet proved to be a guarantee for permanence in a scale of decades. It doesn't mean the FOSS concept is not vastly superior to the previous concept of closed source and secrecy, it just means that, as good as it can be, when it comes to long-term permanence, FOSS has not proved yet to be a reliable solution, quite the contrary in most cases, despite all the other advantages of the FOSS concept.

And you may think you know better than ISO, the Library of Congress and the Archives Nationales de France, but I know enough to know I don't know as much as they do. And, as usual, you have the right to disagree.

#14 - 12/29/2018 09:43 AM - Aurélien PIERRE

I have had corrupted raws in the past, my hard-drive hosts pictures showing people who are dead now, so I get the purpose of long-term picture conservation. Where I don't agree with you is the choice of the format. A true archival format would have some sort of integrity hash (MD5 at least) that can be checked to ensure the data remain uncompromised, and some sort of backup mechanism for when bits are lost. As of now, DNG look to me like PDF and TIFF… Adobe's shit that became a standard just because they are the leader, not because the format is safe or good in itself.

So my opinion on that matter is the best we have for now is the original untouched proprietary format. It will change the day we have a true, opensource, exchange format backed by the ISO, with integrity checks built-in.

#15 - 12/29/2018 11:55 AM - Aurélien PIERRE

If I had to design an archival file format, I would store a wavelet decomposition of the image instead of the pure matrix. Wavelets are a way to extract frequencies levels, similarly to the Fourier spectrum. It's used to perform lossless compression in JPEG2000 and in retouch and equalizer modules, in dt, to perform multi-scales editings. To reconstruct the image, you just sum the wavelets decomposition for each pixel and that's it.

So, imagine you save 5 wavelets levels of your picture in your archival format. The probability of having a corrupted bit is high, but the probability for the corruption to affect all the wavelets scales of the same pixel is close to zero. If the corruption affects low-frequencies, you just need to interpolate the closest neighbours to reconstruct the signal almost losslessly. If it affects the high frequency, well, I'm sure clever minds will find a way to do a probabilistic reconstruction of the pixel using the low-frequencies layers and the high-frequency neighbours (joint total-variation inpainting with hyper-laplacian prior, for example).

Now, to get the coordiate of the corrupted pixel, you compute a MD5 hash for each line and each column of the full output image (wavelets scales stacked), and another MD5 hash for each wavelet level. You store these values in the file header. Everytime you open the file, the codec recomputes the actual hashes and does a check with the saved hashes from the header. In case of some corruption, the faulty lines/columns/layers hashes will give the coordinates of the affected pixels, and the multi-scale storage will allow a clever reconstruction based on what remains, or, worse case scenario, lose only one frequency of each affected pixel.

That's what I call an archival format : you get redundancy of data and integrity self-checking. So DNG and TIFF are just other dumb formats to me.

#16 - 12/29/2018 12:19 PM - Roman Lebedev

Aurélien PIERRE wrote:

> If I had to design an archival file format, I would store a wavelet decomposition of the image instead of the pure matrix. Wavelets are a way to extract frequencies levels, similarly to the Fourier spectrum. It's used to perform lossless compression in JPEG2000 and in retouch and equalizer modules, in dt, to perform multi-scales editings. To reconstruct the image, you just sum the wavelets decomposition for each pixel and that's it.
>
> So, imagine you save 5 wavelets levels of your picture in your archival format.

(that is what gopro's vc5 does, and *likely* canon's cr3)

#17 - 12/29/2018 10:13 PM - Vincent Fregeac

Redundancy and data integrity is not the role of the archive format, it's the role of the vault, through 3-2-1 redundancy or more, hash/checksums of files or directly of the media, and periodic validation of the archived data even if you don't touch it for years. Also, when you loose archived digital information, it's rarely one byte or one pixel, it's the entire media. So if you can't read the files anymore, the fact that the file format has inherent data integrity won't help. The role of the archive format is to remain as close as possible to the original and still be easily readable in the future. That's why archives keep parchment and dusty old books which are even dumber formats than TIFF/EP or DNG.

A wavelet representation with inherent data intergrity would be great for the darkroom though, because you have a higher risk of corrupting single bytes or pixels when you process an image (except if you use EEC memory, redundant processors and so on, but that would make a fairly expensive digital darkroom). Being able to restore corrupted pixels directly in the darkroom, without going back to the originals stored in your digital vault, would be really convenient. But not storing the original matrix means you are stuck for ever with the artifacts of the current demosaic algorithms, which are better than they were, but still not perfect, so I still prefer to archive the dumb original matrix than a processed matrix.

For me, the ideal RAW archive format would store the original matrix with its original analog values, but we don't have the technology to store analog values yet and the closest equivalent, 32 bits ADCs, are still more expensive than the most expensive DSLR (plus they are fairly slow, they would take 2-3 minutes to digitize a simple 24MP matrix). So we have to do with a fairly lossy 12 or 14 bits representation of the original for now. But it's still worth keeping this coarse representation of the real original because better algorithms to process it are created regularly, so you don't want to be stuck with a representation of the defects and artifacts of the current algorithms.

And I'm sure there will be better, more precise, more flexible, archive formats for RAW images in the future, but for now, the best solution to keep an image as close as possible to the original, in a format that is accepted industry-wide (I mean the archiving industry), is still something like TIFF/EP or, in a lesser measure, DNG. They are not perfect, but the alternatives are worse.

By the way, TIFF/EP is not TIFF. TIFF is an Adobe format but TIFF/EP is exactly what you said you wanted in an archive format: an open-source, exchange format backed by ISO. It doesn't have built-in redundacy or data intergrity check built-in because it's the vault, not the format, which is supposed to provide redundancy.

#18 - 12/30/2018 06:31 AM - Aurélien PIERRE

Roman Lebedev wrote:

> (that is what gopro's vc5 does, and *likely* canon's cr3)

And I *thought* I was being original… The most difficult part in innovation seems to be born early enough.

#19 - 12/30/2018 04:43 PM - Vincent Fregeac

Roman, you are still original, at least for RAW formats: CR3 is just using ISO/IEC 14496-12 (Base Media Format for MPEG4) instead of TIFF for the image, plus custom tags, and it introduces an optional lossy compression for RAW images. Otherwise, it is still a true RAW, just as "dumb" as CR2 but also just as close to the original as CR2. Details here: https://github.com/lclevy/canon_cr3

While it's still a proprietary format, it's a step in the right direction as it uses a real standard instead of a defacto standard, and it stores the original without any processing, storing even the dual pixel track, although I don't see how it can be used once the picture is taken. But someone may later find a clever way to use it for better demosaic, or noise reduction, or something else, so it's a good thing they stored everything.

And Harvey is already working on CR3 support in exiftool so supporting CR3 in darktable should not be too much trouble (that is, if you move to exiftool, which exiv2 main developer kind of recommend except for corner case applications where you need as much speed as possible).

#20 - 12/30/2018 04:54 PM - Roman Lebedev

I still don't get it, are you posting your comments in full seriousness, or are you trolling.

There is something off about them, as-if they are posted only after a quick look at the top 3 results in the google search
about the subject, thus they look as-if they are on-topic, and have some details, but also a incredibly superficial,
with no deep insight, and with some details being wrong, and lot's of water.

Vincent Fregeac wrote:

> ...

No comment on that.

> And Harvey is already working on CR3 support in exiftool so supporting CR3 in darktable should not be too much trouble (that is, if you move to
> exiftool, which exiv2 main developer kind of recommend except for corner case applications where you need as much speed as possible).

Uh huh, how do you figure?

#21 - 01/01/2019 01:02 AM - Vincent Fregeac

May be the opinion I try to express will be easier for you to consider seriously if it comes from a reputable institution

Sustainability of Digital Formats: Planning for Library of Congress Collections: https://www.loc.gov/preservation/digital/formats/sustain/sustain.shtml

It's not too long to read, about 3 pages, and it is reasonably complete. Although it is from an archiving institution, all the points they make also apply to images (or at least mine), just replace "scholars" and "custodians" by whoever you are keeping your images for. I don't link these reference documents usually because they are quite verbose (you probably have better things to do) and most of my references are in French (english is my second language). But if I'm not summarizing them properly, maybe it's better if you go to the source.

P.S.: And, yes, I need google search to find these references because I don't have an encyclopedic memory. I have converted my RAW files to an open format since 2008, as well as storing metadata directly in the RAW files, I know why I'm doing that but I don't necessarily remember, 10 years later, where I read the reasons why I'm doing that (or I need to find an English equivalent to the document I read in French at that time).