

darktable - Bug #12372

Photos of an imported library deleted outside of darktable still appear in darktable

10/20/2018 01:40 PM - Rainer Krienke

Status:	Closed: won't fix	Start date:	10/20/2018
Priority:	Low	Due date:	
Assignee:		% Done:	0%
Category:	General	Estimated time:	0.00 hour
Target version:	2.6.0	bitness:	64-bit
Affected Version:	2.4.3	hardware architecture:	amd64/x86
System:	openSUSE		

Description

Hello,

I noticed a behaviour of darktable that when improved could be very helpful for the user:

I imported a directory into darktable and processed the raw .nef files in there. I exported to jpg and then browsed the jpg photos in a image viewer where I deleted some that didn't appear to be good to me. Next I also (automatically) deleted the corresponding .nef and .xmp files.

Then I started darktable again and it still showed all the icons also those of the physically deleted photos. So I imported the directory with the raw files a second time and now I saw each deleted photo as a bw skull because the file was gone but the database still contained a reference to it.

I know there is a script one can call to remove the dead entries from the database, but I think it would be much better if darktable could handle this common situation in a more comfortable way for the user.

A simple solution would be to add a "resync" button/menu that syncs the currently displayed "film" (actually a directory) with the current contents removing files that no longer exist. It could of course display a warning before.

A still better solution would be to do this completely automatically without a need for the user to click anything.

For me this happens quite often because I do raw developement with darktable but also work with a standard filemanager and image viewer. By the way this is quite exactly what was proposed in a feature request I found from 2 years ago:

<https://redmine.darktable.org/issues/11052>

Would be nice to see an improved behaviour of darktable in respect to this problem.

Thanks a lot
Rainer

History

#1 - 10/21/2018 09:00 PM - Pascal Obry

- Status changed from New to Closed: won't fix

For me this happens quite often because I do raw developement with darktable but also work with a standard filemanager

This has been said multiple time it is not a supported workflow. This is the same ot Lightroom BTW. dt keeps a database and image cache for fast access, so if you want to delete an image do so from darktable interface. There is no way dt will magically knows that the image has been deleted. And again, you should probably never open a directory managed by dt with a filemanager.

#2 - 10/22/2018 09:59 AM - Rainer Krienke

Thanks for the reply,

yes darktable keeps a database, but I do not see any reason why missing images couldn't be removed from the database just like the script `purge_non_existing_images.sh` does but callable from the darktable ui. So there is certainly no technical reason not to do it inside dt.

From a users point of view this would really not harm darktable but make it better usable for all those people like me who simply have a workflow that could be supported this way as well as for people that do not want to call scripts, but use a ui instead which is perfectly ok in my eyes. A photography does not need to be a script fan...

I am certainly not the only one who could benefit from a improvement with this problem. If you google for "dark table missing images" you find many matches. So I think its really worth considering. Tell me a good reason why not?

After all there was already an approach by Tobias Ellinghaus to a very similar problem that would also fix mine (images added into a folder not recognized by dt). This problem is also still today present in dark table:

<https://redmine.darktable.org/issues/9730>

However this is 4 years ago and the issue descibed there has not yet been fixed.

And why should someone not use a filemanager if such a tool is the best do do what ist meant for: managing files? Else you should implement a feature complete filemanager in dark table, but this does not make sense because it would reinvent the wheel.

#3 - 11/29/2018 02:13 PM - Roman Lebedev

- *Target version set to 2.6.0*