

darktable - Bug #12516

darktable 2.6 can not delete duplicates of local copies "cannot remove local copy when the original is not accessible"

12/31/2018 06:23 PM - David Schaefer

Status:	New	Start date:	12/31/2018
Priority:	Low	Due date:	
Assignee:		% Done:	0%
Category:	General	Estimated time:	0.00 hour
Target version:	2.6.0	bitness:	64-bit
Affected Version:	git master branch	hardware architecture:	amd64/x86
System:	Ubuntu		

Description

Hi,

darktable 2.6 can not delete duplicates of local copies. A toast with the message "cannot remove local copy when the original is not accessible" is shown.

History

#1 - 01/04/2019 03:30 AM - Aurélien PIERRE

And are the originals accessible ?

When the local copies are deleted, darktable resynchronizes the development history with the XMP of the originals. So the operation is aborted when this can't be achieved.

Although, it could be nice to have a dialog allowing to force the deleting anyway.

#2 - 01/04/2019 08:45 AM - David Schaefer

Hi,

And are the originals accessible ?

No but that doesnt matter. Please read below.

When the local copies are deleted, darktable resynchronizes the development history with the XMP of the originals. So the operation is aborted when this can't be achieved.

Although, it could be nice to have a dialog allowing to force the deleting anyway.

I completely agree, but this is different from my concern.

When you have a local copy it doesnt matter if the original is accessible or not. You work on the local copy. Only if you replay the local copy you need to have access because the synchronization and update process obviously needs it. Now lets consider a different use-case: A duplicate of a local copy:

If we have a duplicate of an image, may it be a local copy or not (image O), we get a new Image (image D). Since we are able to create the duplicate for both of them (also if O from the local copy is not available), we should also be able to remove the duplicate.

For me the requirement to be available makes only sense for the local copy O, not any derived images D from O.

#3 - 01/04/2019 10:25 PM - Pascal Obry

If we have a duplicate of an image, may it be a local copy or not (image O), we get a new Image (image D). Since we are able to create the duplicate for both of them (also if O from the local copy is not available), we should also be able to remove the duplicate.

But there is no way to tell if the duplicate come from the local copy or the original I think. So from dt point of view, the syncho is required and it refuses to delete something that may be need a sync later. Maybe we can improve that, but not sure it is trivial...