Core - Bug # 47211

Status:	Resolved	**Priority:**	Should have
Author:	Oliver Hader	**Category:**	File Abstraction Layer (FAL)
Created:	2013-04-14	**Assigned To:**	Oliver Hader
Updated:	2013-05-24	**Due date:**	
TYPO3 Version:	6.0		
PHP Version:	5.3		
Complexity:	hard		
Is Regression:			
Sprint Focus:			
Subject:	ProcessedFile is persisted in sys_file as well		

Description

New ProcessedFile objects are stored in sys_file as well instead of just being available in sys_file_processedfile.
The solutions for the time being is to check whether "/_processed_/" is used in the beginning of the file identifier to be created.
Based on this result, either a regular File or ProcessedFile shall be generated.

Related issues:

related to Core - Bug # 47140: ProcessedFile/Thumbnail is always regenerated	**Resolved**	**2013-04-11**
related to Media Management - Bug # 47782: Regenerated and indexed thumbnails...	**Resolved**	**2013-05-01**
related to Core - Bug # 48998: FAL files get reindexed in storage null when u...	**Resolved**	**2012-09-06**

Associated revisions

Revision b31881aa - 2013-05-17 16:55 - Fabien Udriot

[BUGFIX] ProcessedFile is persisted in sys_file as well

The problem appears more generally as soon as getProperties is
called from a file object. FAL will try, by all means, indexing the file
(storing the file into sys_file, in other word)

The change set adds a method telling a File not be indexed by the API.
This is particularly useful if you want to create objects but don't want
them to be persisted.

Change-Id: I8cf0c0f6389dc13e8c3855452f1d2519544b69b9
Fixes: #47211
Fixes: #47782
Releases: 6.0, 6.1, 6.2
Reviewed-on: https://review.typo3.org/20437
Reviewed-by: Alexander Opitz
Tested-by: Alexander Opitz
Reviewed-by: Wouter Wolters
Tested-by: Wouter Wolters
Reviewed-by: Stefan Neufeind
Reviewed-by: Frans Saris
Reviewed-by: Philipp Gampe
Tested-by: Philipp Gampe

Revision 0d725add - 2013-05-17 16:58 - Fabien Udriot

[BUGFIX] ProcessedFile is persisted in sys_file as well

The problem appears more generally as soon as getProperties is
called from a file object. FAL will try, by all means, indexing the file
(storing the file into sys_file, in other word)

The change set adds a method telling a File not be indexed by the API.
This is particularly useful if you want to create objects but don't want
them to be persisted.

Change-Id: I8cf0c0f6389dc13e8c3855452f1d2519544b69b9
Fixes: #47211
Fixes: #47782
Releases: 6.0, 6.1, 6.2
Reviewed-on: https://review.typo3.org/20872
Reviewed-by: Philipp Gampe
Tested-by: Philipp Gampe

Revision f94f4eb1 - 2013-05-17 17:00 - Fabien Udriot

[BUGFIX] ProcessedFile is persisted in sys_file as well

The problem appears more generally as soon as getProperties is
called from a file object. FAL will try, by all means, indexing the file
(storing the file into sys_file, in other word)

The change set adds a method telling a File not be indexed by the API.
This is particularly useful if you want to create objects but don't want
them to be persisted.

Change-Id: I8cf0c0f6389dc13e8c3855452f1d2519544b69b9
Fixes: #47211
Fixes: #47782
Releases: 6.0, 6.1, 6.2
Reviewed-on: https://review.typo3.org/20873
Reviewed-by: Philipp Gampe
Tested-by: Philipp Gampe

History

#1 - 2013-04-25 15:47 - Klaus Hinum

With 6.0.5rc1 still unreferenced sys_file entries. For a used image in a tt_content element I get entries in sys_file for
- the original file from the storage /Notebooks/Apple/macbookpro17_front.jpg - which is fine
- /_processed_/preview_macbookpro17_front_ba969b9235... for the processed preview file in the backend (why that?)
- /typo3temp/_processed_/csm_macbookpro17_front_d819... for the processed frontend file (why is it in typo3temp and why an entry in sys_file).
- /fileadmin/Notebooks/Apple/macbookpro17_front.jpg ... another entry with storage 0 (altough fileadmin is storage 1), this time for the click enlarge file (as the original is taken without shrinking)

So I get 4 entries instead one, and therefore FAL also breaks replication in our case.
Can anyone explain why there are 4 entries? In my point of view there should be only one in sys_file and three in sys_file_processedfile (for the thumbnail, the scaled and click enlarge / original images).

Update: more information what happens

#2 - 2013-05-02 19:49 - Fabien Udriot

The problem appears more generally as soon as getProperties is called from a file object. FAL will try, by all means, indexing the file (storing the file into sys_file, in other word)

I suggest adding a method telling a File not be indexed by the API. This is particularly useful if you want to create objects but don't want them to be persisted.

As info, for the need of Media (which extends File), I had to add a convenient method which is not the nicest. From my scope, it was for solving some recursive Exception appearing when unit testing. Besides, I remember some other cases where such infinite recursion Exception happened, if a file could not be indexed for some reasons.

#3 - 2013-05-02 19:50 - Gerrit Code Review

- Status changed from New to Under Review

Patch set 1 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20437

#4 - 2013-05-02 20:02 - Gerrit Code Review

Patch set 2 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20437

#5 - 2013-05-06 08:10 - Fabien Udriot

As info, here is the code I am using generating a thumbnail. Normally, this would create a processed file and store the info in sys_file_processedfile. But following a regression in FAL, it is now put into sys_file (with all the consequences).

```
$configuration['width'] = $this->file->getProperty('width');
$configuration['height'] = $this->file->getProperty('height');
$taskType = \TYPO3\CMS\Core\Resource\ProcessedFile::CONTEXT_IMAGEPREVIEW;
/** @var $processedFile \TYPO3\CMS\Core\Resource\ProcessedFile */
$processedFile = $this->file->process($taskType, $configuration);
```

#6 - 2013-05-14 12:41 - Gerrit Code Review

Patch set 3 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20437

#7 - 2013-05-14 16:29 - Gerrit Code Review

Patch set 4 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20437

#8 - 2013-05-14 16:58 - Gerrit Code Review

Patch set 5 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20437

#9 - 2013-05-14 17:01 - Gerrit Code Review

Patch set 6 for branch **master** has been pushed to the review server.
It is available at https://review.typo3.org/20437

#10 - 2013-05-17 16:57 - Gerrit Code Review

Patch set 1 for branch **TYPO3_6-1** has been pushed to the review server.
It is available at https://review.typo3.org/20872

#11 - 2013-05-17 16:57 - Gerrit Code Review

Patch set 2 for branch **TYPO3_6-1** has been pushed to the review server.
It is available at https://review.typo3.org/20872

#12 - 2013-05-17 16:59 - Gerrit Code Review

Patch set 1 for branch **TYPO3_6-0** has been pushed to the review server.
It is available at https://review.typo3.org/20873

#13 - 2013-05-17 17:30 - Fabien Udriot

- Status changed from Under Review to Resolved
- % Done changed from 0 to 100

Applied in changeset commit:f94f4eb1dc800a320902a31d897aa69904d7c6b9.

#14 - 2013-05-18 09:34 - Klaus Hinum

guess target version should be 6.1.1 and 6.0.6

#15 - 2013-05-24 12:16 - Klaus Hinum

what about all the sys_file entries from the processed files. Can we delete them savely? E.g. delete all files that are in the _processed directory in the sys_file database.