



White Paper

Fabasoft Folio Content Addressed Storage

Fabasoft Folio 2020 Update Rollup 3

Copyright © Fabasoft R&D GmbH, Linz, Austria, 2020.

All rights reserved. All hardware and software names used are registered trade names and/or registered trademarks of the respective manufacturers.

No rights to our software or our professional services, or results of our professional services, or other protected rights can be based on the handing over and presentation of these documents.

Contents

1 Introduction	4
2 Software Requirements	4
3 Configuration	4
4 Hash Value and Filing Structure	5
5 Cleaning a CAS Area	5
6 Collisions Handling	5
7 Scaling with Split Points	6
8 Configuring CAS Cache	6
9 Direct access to a CAS area	7
10 Backup Directory	8
10.1 Delayed Backup Directory Cleanup	8
11 Restrictions	9

1 Introduction

Fabasoft software products include the storage system “content addressed storage (CAS)” for Fabasoft Folio MMC Areas.

Content addressed storage ensures that certain content is only saved once. It is possible to reduce disk space usage, especially in larger installations with many content files, because without using a CAS system the same files are generally saved several times.

This document describes the new storage possibility of saved content files in Fabasoft Folio.

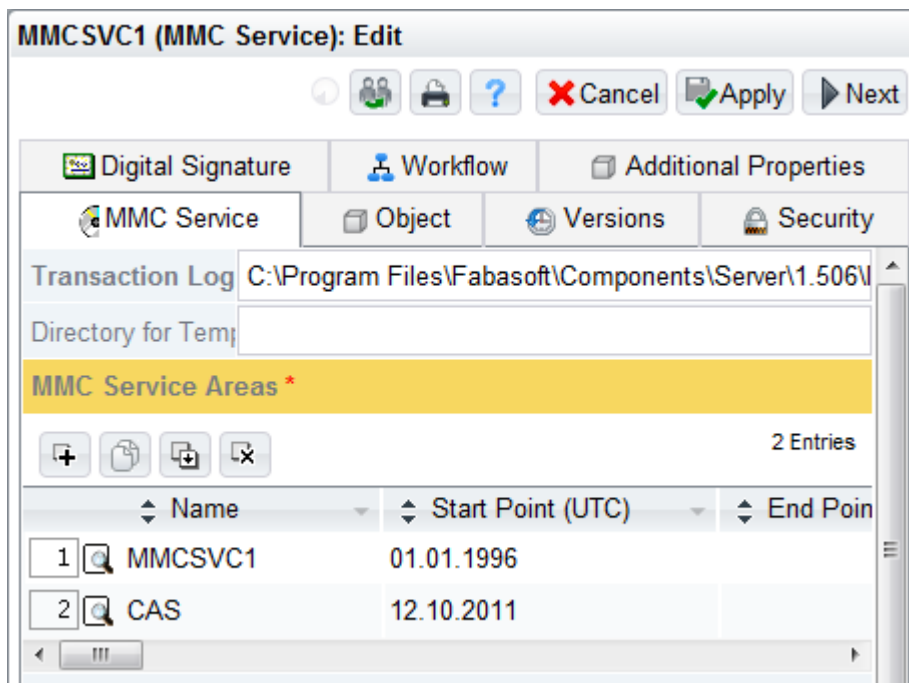
2 Software Requirements

System environment: All information contained in this document implicitly assumes a Microsoft Windows environment or Linux environment.

Supported platforms: For detailed information on supported operating systems and software see the software product information on the Fabasoft distribution media.

3 Configuration

The chosen storage type for content files is set in the Fabasoft Folio MMC Areas in the corresponding Fabasoft Folio MMC Service object. It is not possible to change the storage type later on. To change the storage from “One Directory per Day (Change Date)” to “Content Addressed Storage (CAS)” (or vice versa) the old Fabasoft Folio MMC Area has to be closed and a new one has to be created.

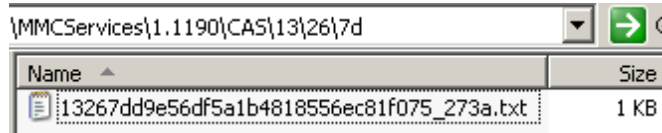


With the content addressed storage technology of the Fabasoft Folio MMC Services it is possible that several Fabasoft Folio MMC Services are using the same data pool. Thus content managed by several Fabasoft Folio MMC Services is only stored once. Otherwise several Fabasoft Folio MMC Services may use different CAS areas (e.g. each Fabasoft Folio Tenant has an own CAS area)

4 Hash Value and Filing Structure

A hash value serves as base for the path of the respective content. Therefore an MD5 hash value is calculated out of the content (MD5: 128 bit = 16 bytes = 32 signs (hexadecimal)). An MD5 hash value is a compromise between uniqueness and performance.

Two characters of the hash value act as directory. Dependent on the hash value, altogether three directories are created. Therefore up to 256 directories on one level may be created.



The hash value, a random number (that is computed at the first writing of the content) and the file extension (in this example “txt”) represent the filename. If the same two content files have different file extensions only one file is stored in the Fabasoft Folio MMC Area with the file extension of the first file.

Fabasoft Folio Kernel stores the hash values, which belong to certain content files in the aggregate `COOSYSTEM@1.1:objlogcontmap`.

5 Cleaning a CAS Area

Changes the content of an existing Fabasoft Folio object, a new hash value is calculated and stored for the modified content. But the previous content is not deleted. To delete content files, which are not in use, call the action “Cleanup Content Areas” via the tool `fscadmin`. All content with hash values that are not referenced by a Fabasoft Folio object are deleted through this operation.

Fabasoft Folio Kernel reads all referenced hash values from the database (from all Fabasoft Folio COO Services) during cleaning the Fabasoft Folio MMC Areas and compares this set of values with existing filenames in the file system. File names which are not included in the set of hash values are not in use anymore and are deleted during cleanup.

Files getting deleted are logged in the corresponding MMC Areas and backup directories. These logs are saved in `logs/casdeletion`.

Attention: Before cleaning a CAS area, a state-of-the-art backup is absolutely necessary.

Note:

- The cleaning process needs, dependent on the extent of a Fabasoft Folio MMC Area, long time and should be planned accordingly. To restrict the cleaning process to specific directories of the MMC Area, the option `casrange` (e.g. `--casrange 00-3f`) can be used.
- To make successful cleaning possible, all Fabasoft Folio COO Services have to be available (important to run the query successfully).
- Content of objects that are located in inactive services are deleted, too.

6 Collisions Handling

Theoretical it is possible that different documents have the same MD5 hash value. If you assume that a hash value is unique, this could lead to data loss.

Therefore Fabasoft Folio compares the content if the hash value is equally. If Fabasoft Folio determines that two different documents have the same hash value, the system adds a new random number to the existing hash value and saves the modified hash value to the content.

Example for a collision:

Name	Size
d4fc63b867d8d120f18ba793aaab4683_7d21.txt	1 KB
d4fc63b867d8d120f18ba793aaab4683_9f9f.txt	1 KB
d4fc63b867d8d120f18ba793aaab4683_439d.htm	1 KB

7 Scaling with Split Points

To enable a distribution of a CAS area, so-called split points can be created. The distribution of data is processed depending on the calculated hash values.

Example for split point configuration:

To distribute a CAS area to three different storage devices, three split points can be created directly in the defined Fabasoft Folio MMC Area. A split point can be a directory or a file. If the split point is a directory then this directory will be used for saving the corresponding data. In case of a split point file the path specified in this file will be used. Such a file can contain a network path or a local directory path. A split point has the following syntax: `splitpoint_<xx>`. `<xx>` represents the first two characters of the hash value.

Name	Size	Type	Date Modified
splitpoint_00		File Folder	11.05.2007 08:10
splitpoint_55		File Folder	11.05.2007 08:10
splitpoint_aa	1 KB	File	11.05.2007 08:11

In this example the split points are "00", "55" and "aa". By means of this structure all hash values starting between "00" and "54" are stored in the directory `splitpoint_00`, all hash values between "55" and "a9" in the directory `splitpoint_55` and all hash values from "aa" in the directory specified in the `splitpoint_aa` file. According to this illustration it is possible to create up to 256 split points.

If the `splitpoint_00` split point is not created the content is located directly in the defined Fabasoft Folio MMC Area directory.

Split points can be created or deleted as necessary. The right distribution of the content (changes in the file system have to be processed manually) and a restart of Fabasoft Folio MMC Services (the split point structure is only read at first access) are important for successful modification.

Note: The consistence of split points is checked after first access to a CAS area. In cases of inconsistency (files are in the wrong split point directory) an error message is written to the event log.

In a Linux system environment, so-called "symbolic links" can be used to locate split point directories on different storage devices. In a Microsoft Windows system environment, a so-called "mount point" can be created on these directories.

8 Configuring CAS Cache

For performance reasons each Fabasoft Folio Kernel instance has a CAS cache by default. When requesting a content the Fabasoft Folio Kernel first tries to read the content from the CAS cache. If

the content identified by a hash can be found in the CAS cache the content can be used directly from the cache and the Fabasoft Folio MMC Service is not required for reading.

The path of the CAS cache directory can be configured via the environment option `CASCACHEDIR` (e.g. `HKEY_CURRENT_USER\Software\Fabasoft\Environment\CASCACHEDIR` in a Microsoft Windows environment).

By default all CAS areas of all Fabasoft Folio MMC Services are treated as a separate data pool in the cache. In case that several Fabasoft Folio MMC Services are using the same data pool the configuration can be adopted to combine the areas also within the cache. Therefore configure the property *Shared Cache Name (CAS)* (`COOSYSTEM@1.1:dareasharedname`) in the *MMC Service Areas* aggregate of the Fabasoft Folio MMC Services. Areas with the same shared cache name are treated as one CAS data pool.

In order to prevent an unlimited growth of the CAS cache the Fabasoft Folio Kernel calls periodically a cleanup routine for the CAS cache. Information about the cleanup can be found in the event log. The maximum size and the time interval of the recurring cleanup process can be configured via the environment options `CASCACHEMAXMB` and `CASCACHECLEANUPINTERVAL`, respectively. It is also possible to call the cleanup mechanism directly via the action `COOSYSTEM@1.1:CleanupCASCache`. When doing so the environment option `CASCACHECLEANUPINTERVAL` is ignored.

The CAS cache can be disabled via the environment option `ENABLECASCACHEDIR`.

9 Direct access to a CAS area

For reliability and performance reasons it is possible to configure more file shares for the same MMC area. File shares can be configured as UNC paths by the property *Path to Directory on Server* (`COOSYSTEM@1.1:darearemdir`). If file shares are configured the Fabasoft Folio Kernel reads and writes all contents from the shares without the Fabasoft Folio MMC Service.

Path to Directory on Server
\\192.168.100.100\CAS1
\\192.168.100.100\CAS2

When the Fabasoft Folio Kernel reads a content, the file access occurs on a randomly selected file share. (Load distribution). If the content cannot be found on one share another share is selected. (Reliability)

When writing contents the Fabasoft Folio Kernel writes parallel to all configured file shares. The transaction is successful when at least on 50 percent of the configured shares the write operation succeeded.

Note:

- File servers must process write requests synchronously or honor filesystem synchronization requests to prevent data loss. In case of Samba (CIFS), the server option `strict sync = yes` is required. In case of Linux/nfsd (NFS), the export option `sync` is required.
- The user context in which the Fabasoft Folio Kernel is running (e.g. Web Service User) must have read and write privileges on the configured shares.
- On Linux systems, the configured UNC paths will be resolved automatically based on currently mounted filesystems.
- Changing the MMC area configuration requires the execution of "Synchronize Registry Entries".
- Changing the MMC area configuration requires a restart of the kernel.

- The number of writer threads can be configured via the environment option `MMCWRITETHREADCOUNT` and is 12 by default.
- For each Fabasoft Folio Kernel a preferred read share can be configured via the environment option `MMCREADDIR_<areaname>` (e.g. `MMCREADDIR_MMCSVC1 = "\\192.168.100.100\CAS2"`).

10 Backup Directory

A backup directory can be configured for online and offline backups. Both are based on a daily directory structure. An online backup directory configured via the properties *Backup Directory on Server (Online)* (`COOSYSTEM@1.1:dareabackupdir`) or *Path to Backup Directory on Server (Online)* (`COOSYSTEM@1.1:dareabackupremdir`) means that contents will be backed up in the directory during normal operations. Additionally, this kind of backup directory can also be synchronized with the configured area paths. On the other hand, an offline backup directory configured via the property *Path to Backup Directory on Server (Offline)* (`COOSYSTEM@1.1:dareaofflinebackupremdir`) will not be used during normal operations. This backup directory will only be used for manual backup tasks.

When using the options `synctobackup` or `syncfrombackup` of the action *Cleanup Content Areas* in the tool `fscadmin`, the configured area paths can be synchronized with the backup directory. Using the option `cleanupbackup`, a cleanup of the backup directory can be started. The cleanup is based on file system operations meaning that each file in the backup directory which is no longer available in one of the configured area paths will be deleted. `cleanupbackup` only makes sense when the operation is executed after removing unreferenced files on the configured area paths using the action `cleanup`.

Using the option `offlinebackup` with `synctobackup`, `syncfrombackup`, or `cleanupbackup` means that the configured offline backup directory is used for the specified operation instead of the online backup directory.

Note:

- The synchronization using `synctobackup` or `syncfrombackup` is based on the log files written in the directory `logs` of the configured directory.
- A content will not be written to the daily backup directory when the content already exists in the single instance areas.
- Changing the MMC area configuration requires the execution of "Synchronize Registry Entries".

10.1 Delayed Backup Directory Cleanup

The tool `cleanbackup` removes files from the backup directory by using the deletion log generated during cleaning up an MMC Area. This tool takes the parameters `root`, which specifies the path of the backup directory to be cleaned up and `days` to define the minimum number of passed days since the deletion of a file in the MMC Area for deleting it in the backup directory, too.

Note:

- If a cleanup with `fscadmin` has already been run once and no CAS deletion logs exist, a verbose `cleanupbackup` dry-run needs to be performed with `fscadmin`. The date range should be until yesterday. This output of `fscadmin` needs to be converted to deletion logs using the tool `logtohashlog`. The generated deletion logs need to be placed in `logs/casdeletion` in the associated backup directory.

- The backup can be spread across multiple paths. In this case, *Path to Backup Directory on Server (Online)* (`COOSYSTEM@1.1:dareabackupremdir`) represents the active backup path containing most recent backup data. Old backup data can be moved to other paths that are not configured in Fabasoft Folio. In this scenario, `cleanbackup` must be run for all paths. As CAS deletion logs are only copied to the active backup path, logs in `logs/casdeletion` must be copied to all other paths before executing `cleanbackup` for the active backup path.
- On Linux systems `cleanbackup` also deletes immutable files if run as root.

11 Restrictions

Full-text search in Fabasoft Folio MMC Areas of type “content addressed storage (CAS)” is only supported with Fabasoft Mindbreeze Enterprise. Full-text search based on Microsoft Index Service is not supported by these Fabasoft Folio MMC Areas.