M-FILES CORPORATION

INSTALLING, CONFIGURING, AND USING M-FILES GOOGLE DRIVE CONNECTOR

LAST UPDATED 28 JULY 2023

VERSION 3.2

Contents

1. Introduction

M-Files Google Drive Connector is a vault application that connects an M-Files vault to Google Drive for users with Google Workspace account. Gmail accounts are not supported. Vault users can use M-Files Desktop, M-Files Web, or the M-Files mobile apps to locate, view, and edit files in Google Drive. Furthermore, users can add metadata to documents stored in Google Drive via M-Files and thus promote documents to managed objects. Managed objects can then be shown in dynamic M-Files views, even though the original data still resides in Google Drive.

This document provides instructions for installing, configuring, and using the connector. For instructions on setting up the connector, see sections 2 through 6. For user instructions, see section 7.

For details about the connections between the M-Files server and clients, and various Google components, see Appendix A.

For instructions on setting up and using the M-Files for Google Workspace add-on, which allows you to save your Google Drive and Gmail content to your M-Files vault directly from Google Chrome, refer to Setting Up and Using M-Files for Google Workspace Add-On.

2. Prerequisites

Please make sure your environment meets these requirements before moving forward.

2.1 M-Files Software Requirements

PRODUCT	VERSION
M-Files Desktop	20.9.9430.0 or later
M-Files Server	20.9.9430.0 or later
.NET Framework	4.5.2 or later
M-Files Google Drive Connector	19.11.16 or later

Note: You also need an appropriate Intelligent Metadata Layer license to use the connector.

Important: With Google Drive Connector 19.11.16 and later, you must set up a Google service account. In earlier versions, a user with administrative rights is used, for example, for indexing content. In versions 19.11.16 and later, a Google service account (see section 5.2) impersonates a normal Google Drive user for administrative tasks, such as indexing. In other words, it is no longer required for the indexer user to have administrative rights in Google Drive.

2.2 Other Requirements

2.2.1 Google Workspace Accounts

To use M-Files Google Drive Connector, Google Workspace accounts are necessary. Gmail accounts are not supported.

2.2.2 Optional Installations

To get the optimal user experience with M-Files Desktop, install either of the following programs on the client computer.

PRODUCT	VERSION
Google Drive	3.43.2448.9071 or later
Google Drive File Stream	29.1.85.2056 or later

With either of these programs installed, these features are enabled:

- Correct icons are shown for all Google Drive files.
- Double-clicking a Google Drive file opens it up for editing in the default browser.

Without any of the optional programs installed, the connector will still work but the above-mentioned features will be disabled. It is not necessary to log in to either of the programs to make use of the features. It is enough that the programs are installed.

2.2.3 Port Settings

Make sure that your firewall settings on the server computer allow inbound and outbound TCP and UDP connections for port 443. Also make sure that Google endpoints are accessible through port 443. For more information on Google endpoints, see appendix A.

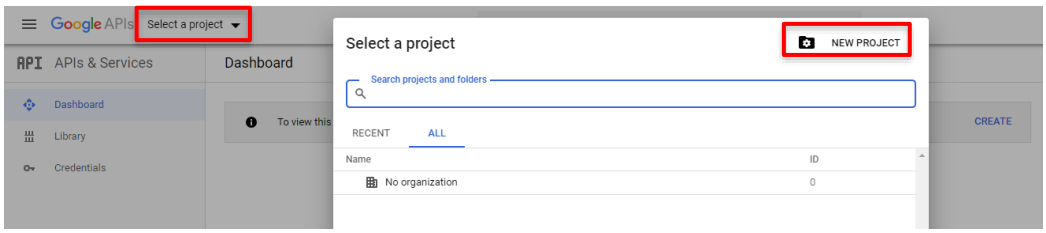
2.2.4 Domain Verification

Your domain must be verified with Google.

3. Registering the Application and Creating a Service Account

Complete these tasks before you install the connector application to your M-Files vault.

1. Log in to https://console.developers.google.com as a Drive administrator.
2. Create a project for credential storing:
    a. Navigate to https://console.developers.google.com/apis/dashboard.
    b. Click **Select a project** and then **NEW PROJECT**.



    c. In the *Project Name* field, enter the name of the project.
    d. Click **BROWSE** to select the Drive organization (do not use the *No organization* option).
    e. Click **SELECT** to confirm the selection.
    f. Click **CREATE**.
3. Enable the necessary APIs:
    a. In the dashboard for the project, click **ENABLE APIS AND SERVICES**.

b. Find the following APIs and enable them by clicking the **ENABLE** button:
    - Google Drive API (required for interacting with Google Drive itself)
    - Drive Activity API (required for indexing)
    - Admin SDK (required for collecting information about groups and users)

Next, create the OAuth credentials:

4. Navigate to https://console.developers.google.com/apis/credentials/oauthclient and make sure the project you created in step 2 is the active project.
5. Open the **Domain verification** tab to add your domain:
    a. Click **Add domain**.
    b. Enter the domain information.
    c. Click **Add domain**.
6. Open the **OAuth consent screen** tab to configure the consent screen:
    a. **Application type**: Internal
    b. **Application name**: Choose whatever name you want.
    c. **Authorized domains**: Your domain (press Enter after you added the domain to make sure it is added).
    d. Click **Save**.
7. Go back to https://console.developers.google.com/apis/credentials/oauthclient.
8. Select **Web application** as the application type and configure the application:
    a. **Name**: You can call it anything at all. It is the name of the OAuth ID but is not used as a display name in the app.
    b. **Authorized JavaScript origins**: Nothing
    c. **Authorized redirect URIs** (press Enter after adding the URI):
    `https://<M-Files Web DNS>/authentication/MFiles.AuthenticationProviders.OAuth/repositorylogin`
9. Click **Create**.
10. Note down the client ID and the client secret.

> **Note**: The credentials can be used for multiple connectors as long as they all access data residing on the same domain.

Finally, create and configure the service account:

11. Click **Manage Service accounts** > **Create service account**.
12. Enter the required information.
13. Click **Done**.
14. Click the name of the service account.
15. Click **Keys** > **Add key** > **Create new key** > **Create**.

    Result: The JSON-formatted key is downloaded.
16. Click **Details** > **Show advanced settings**.
17. Copy the client ID value to your clipboard.
18. Click **View Google Workspace Admin Console**.
19. Click **Security** > **Access and data control** > **API controls** > **Manage domain wide delegation** > **Add new**.
20. In **Client ID**, paste the client ID from the clipboard.

21. In **OAuth scopes**, enter this string:

```
https://www.googleapis.com/auth/drive,
https://www.googleapis.com/auth/admin.directory.group.readonly,
https://www.googleapis.com/auth/drive.activity.readonly
```

22. Optional: In **OAuth scopes**, add this fourth scope if you use common authentication or specify association methods for usernames or emails:

```
https://www.googleapis.com/auth/drive,
https://www.googleapis.com/auth/admin.directory.group.readonly,
https://www.googleapis.com/auth/drive.activity.readonly,
https://www.googleapis.com/auth/admin.directory.user.readonly
```

23. Click **Authorize**.

4. Installing the Connector Application in M-Files Admin

For instructions on installing the connector to your vault, see Adding a Connector in the M-Files user guide. Remember to install the connector license via the *Applications* dialog.

5. Configuring the Connector in M-Files Admin

After you have installed the connector application to your vault, you need to configure it.

5.1 General Settings

For instructions on accessing the configuration and editing the general connection settings, refer to Configuring a Connector in the M-Files user guide. Skip authenticating the special users (such as common user and indexer user) at this point.

5.1.1 Supported Authentication Types

The connector supports personal and common authentication. For more information about the authentication types, refer to External Repository Authentication in the M-Files user guide.

5.1.2 Mapping Google Drive and M-Files Objects

This table gives a description of the object types that the connector returns from Google Drive. Use these values when you configure the mappings between M-Files object types and the Google Drive object types. The name of the mapping setting for the external object type in M-Files Admin is **External Type**.

EXPORT TYPE	DESCRIPTION
`Document`	Returned for individual files. Should be mapped to an M-Files object type that has the option *Objects of this type can have files* enabled.
`Folder`	Returned for Google Drive folders and that may contain documents and other folders.

If you do not set these mappings, it can be that you do not see content when you browse the external repository with M-Files. This can occur even if all other configuration steps are done correctly.

For instructions on creating M-Files object types of your own, refer to Creating a New Object Type.

5.1.3 Search Indexing

The connector cannot differentiate between file data and metadata when refreshing the search index, and therefore refreshes everything at the same time. Due to this limitation, the search indexing refresh interval is determined by the lowest value of the following two settings:

- **Search Indexing** > **Metadata** > **Folder Indexing Options** > **Refresh Interval in Hours**
- **Search Indexing** > **File Data** > **Folder Indexing Options** > **Refresh Interval in Hours**

5.1.4 Google Drive Properties

See the Google documentation for the list of external properties that are returned with all objects. Only single-value properties can be shown to users when navigating M-Files. A single-value property is any property that is not a list or derived from a list. Everything is returned for the indexer user, so it is possible to index contents for list fields as long as a mapping exists. In the table below, you can find examples of Google properties and information on whether they are returned or not.

GOOGLE PROPERTY	TYPE	RETURNED FOR NAVIGATING USERS	RETURNED FOR INDEXER USER
`name`	string	Yes	Yes
`parents`	list	No	Yes
`sharingUser.kind`	string	Yes	Yes

The connector also returns a few artificial properties that can be mapped to M-Files properties. These properties are used for providing extra functionality to the connector. If mapped, the properties are all returned for the indexer user, but only some of them are returned for users navigating M-Files. The properties are described in the table below.

PROPERTY	TYPE	DESCRIPTION	RETURNED FOR NAVIGATING USERS
`nativeContent`	string (multi-line)	Contains a string representation of editable Google Drive files. Use this for indexing the content of those files. Needs to be defined as multi-line text.	No
`googleViewLink`	string (URL)	A direct link for view access to an editable Google Drive file.	Yes
`googleEditLink`	string (URL)	A direct link for edit access to a Google Drive file. See Enabling editing of native Google files for details.	Yes
`googlePresentationLink`	string (URL)	A direct link for the presentation mode for a Google presentation.	Yes

For instructions on how to create new M-Files properties of your own, see New Property Definition.

5.1.5 Enabling Editing of Native Google Files

To enable Google edit links on the metadata card, it is necessary to map the external property *googleEditLink* to an M-Files property and give the M-Files property the alias *googleEditLink.* If no alias is set, the editing capability will not be enabled. For instructions on assigning aliases, refer to Assigning Aliases for Metadata Definitions in the M-Files user guide.

When editing has been enabled, users can click a Google edit link on the metadata card of a native Google file to open and edit the file in their default browser.

5.1.6 Specifying Which Files Are Shown in M-Files

You can use the extension-based filtering settings under **Mapping** > **Object Types** > **Object Type** to specify which external repository objects should be shown in M-Files. The extensions should be entered without a preceding period (for instance: `bmp`), and only a single one should be specified per value field. Excluded external objects are not indexed, either.

5.2 Connector-Specific Settings

Navigate to **Configuration** > **Connector-Specific Settings**, and modify or review at least the settings listed in the tables below. Remember to click the **Save** button once you are done.

SETTING NAME	SETTING DESCRIPTION	VALUE

Root Folder ID	The ID of the folder that should be used as the root.
	
	If you keep the default value (`root`), the root folder will be the root folder of the user's drive.
	
	The ID can be found by navigating to a folder on the Google Drive account and copying everything after `/folders/` in the URL.
		For example:
		
		`0B92U1PJ_r0COUzR2ZHhIV1dFMjo`
		
		`root`

Authentication > **Service Account**		
Email	The email address of the service account. Google generates the address after you create the account.	For example: `test@m-filesconnector.com`
Private Key	The private key of the service account. The value is contained in a JSON file generated when you create the service account. **Important**: Copy the value from the JSON file to the value field on the Configuration tab exactly as it is written between the quotation marks.	`-----BEGIN PRIVATE KEY-----\n<*>\n-----END PRIVATE KEY-----\n` The part **<*>** is replaced with the actual private key content. It is a string that contains hundreds of characters.
Authentication		
Client ID	The client ID noted down earlier if a new project was created.	`45434780512-bcrud71pa0sf6lpq4fli3dsqqmmihsqi.apps.googleusercontent.com`
Client Secret	The client secret noted down earlier if a new project was created.	`TXGWOzC9J1_nYzGU2pfv6w3s`
Redirect URI	The URI to redirect to after authorizing the connector.	`https://<M-Files Web DNS>/authentication/MFiles.AuthenticationProviders.OAuth/repositorylogin`
Enable Logging	Indicates whether authorization events are logged to the local computer.	`Yes` or `No`
Advanced Settings		
Maximum Number of Request Retries	The maximum number of request retries. Each retry doubles the wait time between requests. With the default value of `5`, the expected maximum	Any positive integer or `0`. The default value is `5`.

	execution time is approximately 30 seconds.	
Name of ID Cache Folder	The name to use for the Id Cache folder in the root of the connector.	For example: `__IdCache__`
Depth of ID Cache Folder	The depth of the Id Cache folder structure. Rebuilding of the index must be triggered under the vault maintenance if this value is changed.	Any positive integer. The default value is `3`.
Advanced Settings > User Group Filtering		

User Group Email Filter	Each user group in Google Workspace has an email address. Use this setting to filter the user groups that the connector pulls from Google to M-Files based on the group's email address.
	
	The setting must contain a .NET regular expression string that specifies the format the email addresses must be in for the user groups to be pulled to M-Files and be included in user synchronization and file access evaluation. Leave this value empty to not use a filter.
		For example:
		`sec_fin*@gmail.com`
		or
		`\b.*@domain.com\b`
		
		The first one matches, for example, `sec_fin@gmail.com` and `sec_fi@gmail.com`. The second one matches, for example, `annie.hall@domain.com` and `bertrand_long@domain.com`.

Note: To be able to access a shared drive, it is necessary to set the ID of the shared drive as the root for the connector. It cannot be accessed from other locations such as “root”.

5.3 Authenticating the Indexer User and the Common User

After the configuration, authenticate the indexer user and the common user as needed:

1. If your connection uses `Personal` authentication, or if you want to specify different user accounts for `Common` authentication and for indexing the contents of the Google drive, right-click the connection name in the gray navigation area, and select **Authenticate Indexer User** from the context menu.
    a. In the login dialog, enter the username and password of the account used for indexing the contents of the Google drive.

       This can be any normal Google Drive user. The service account set in the configuration only impersonates this user when the account completes administrative tasks, for example, indexing. If you do not authenticate an indexer user, the service account impersonates the common user.
    b. Click **OK** at the prompt confirming that the authentication was successful.
2. If your connection uses the `Common` authentication type, right-click the connection name in the gray navigation area, and select **Authenticate Common User** from the context menu.
    a. In the login dialog, enter the username and password of the account used for common authentication.
    b. Click **OK** at the prompt confirming that the authentication was successful.

For more information on these user accounts, see the document Intelligent Metadata Layer.

Note: The mapping of users to Google groups starts automatically after the indexer has been authenticated. However, there might be a slight delay before the collection starts.

Note: This connector does not make use of the permissions retriever since normal users can collect the needed permissions.

Note: To be able to operate fully, the connector adds a special app property to all objects residing in Google Drive when accessing them the first time, resulting in the last modified date of the object to be updated as well.

5.4 Giving Access to Content Shared with All Users

If any of the Google groups used in the organization has the special group member *All users in the organization* as a member, it is necessary to create a manual mapping between that user type and an M-Files user group:

1. Wait until the indexer has finished mapping users to Google groups.
2. In the M-Files admin panel for the vault, select **External Repository User Groups.**
3. Right-click the group with the ID *AllGoogleDomainUsers* and click **Properties**.
4. In the **General** tab, click **Add** and select the user group that should be mapped to *AllGoogleDomainUsers*.
5. Click **Add** and then **OK** to confirm the mapping.

Note: The external group called *AllGoogleDomainUsers* will only be visible if any of the Google groups are using the *All users in the organization* member type.

Note: An external group called *anyone* will be visible if any objects have been shared by link with users outside the organization. Manual mapping also must be performed to this group if the shared objects are to be accessible to everyone inside the organization through M-Files.

6. Indexing

Any user account can be used for indexing. However, it is recommended that a dedicated user is set up for indexing. Only objects accessible to the user that is used for indexing will be indexed. This means that the folders and objects that are to be indexed must be shared with this user.

Important: If the root folder ID in the configuration is `root`, make sure that all documents are shared with the user that is used for indexing (indexer user or common user). Otherwise, some functions do not operate correctly. The user must have at least write permissions to the shared objects to collect content and permissions.

To make sure that the indexer has access to all content in a folder and its subfolders, it is recommended to share the folder directly with the indexer and not through a user group. By sharing it directly, the indexer will get access to all content even though there might be special sharing rules for other users. Items shared with a link are neither indexed nor shown to any users apart from the item owner.

To share content with the indexer user:

1. In Google Drive, select the documents that you want to share.

   If your documents are organized into folders, you can easily select the entire folder or multiple folders.
2. Right-click the selection and select **Share** from the context menu.
3. In the *Share with others* dialog, search for the indexer user.

4. Select the indexer user and then click **Send**.

After a while, your Google Drive documents are searchable in M-Files.

> **Note**: During the indexing process, each Google object is tagged with a special cache identifier. This tag is added as an app property and is only visible to the connector application. The tagged objects can be accessed by their tag by navigating to the folder `__IdCache__`, although you normally do not need to do that. This is an artificial folder that does not exist on the drive. The tagging process and the artificial folder are simply needed for the promotion of folders and objects.

7. Using M-Files Google Drive Connector

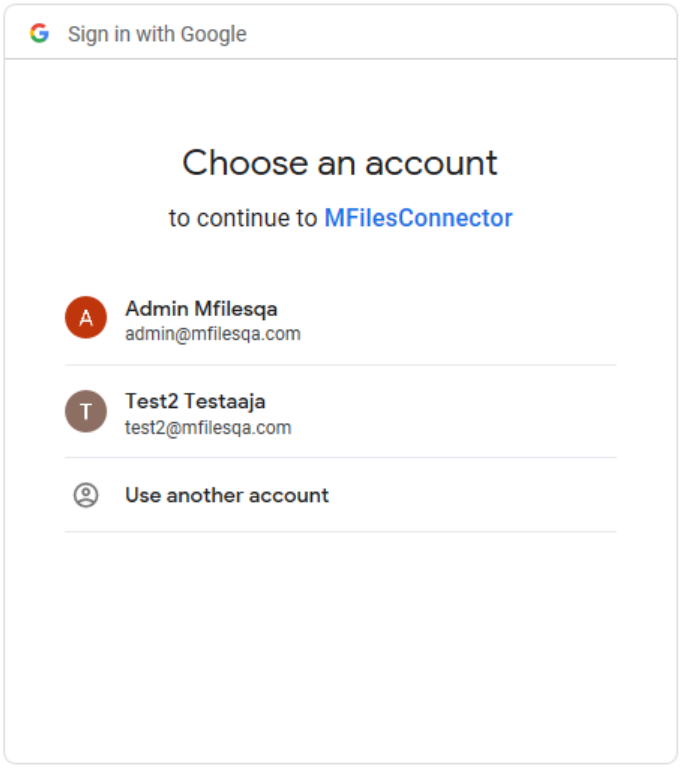
This section contains instructions on using M-Files Google Drive Connector. Generally, when using the connector, a couple of things need to be considered:

- If an item should be searchable through the connector, it must be shared with the indexer either as part of a shared folder or individually.
- If an item is to be promoted, it must be shared with the indexer before the promotion, either as part of a shared folder or individually. For more information about promoting objects, see the document M-Files Intelligent Metadata Layer.
- Items shared by link will neither be indexed nor be shown to any users apart from the item owner. If other people should be able to access content through the connector, it must be shared with them either through groups or directly.
- The permissions used by M-Files are always the least permissive. This means that if an item has been shared with only read permissions to a group the item owner is part of, the item owner will only have read rights to the object through M-Files.
- File type conversions are not allowed, which is why the PDF conversion commands cannot be used.

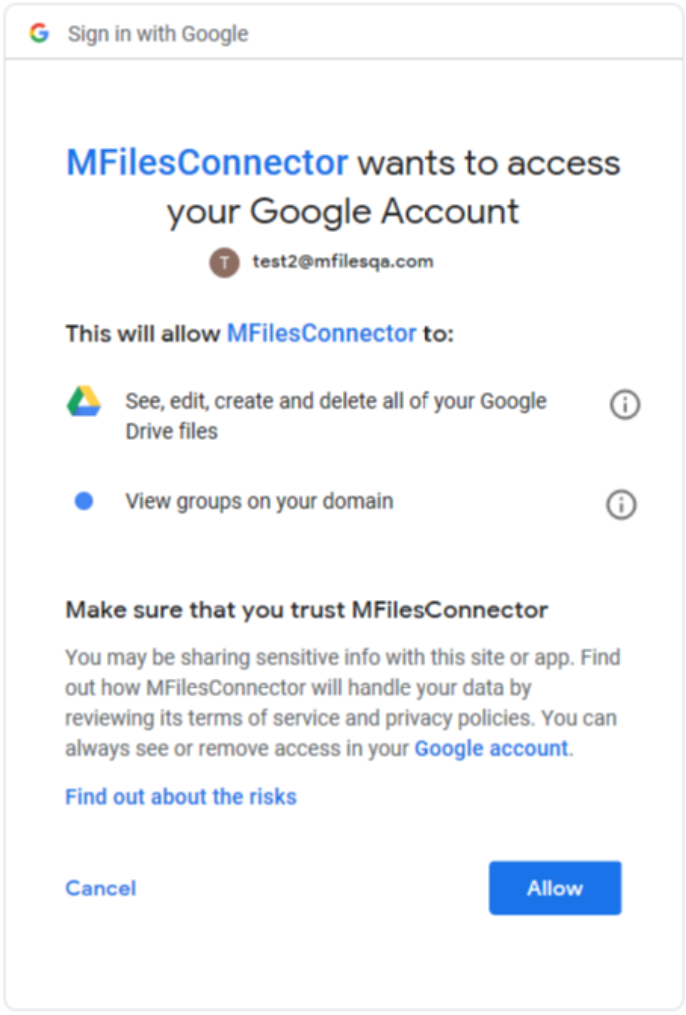
7.1 Getting Started with M-Files Google Drive Connector

Complete the following steps when using M-Files Google Drive Connector for the first time:

1. In the M-Files Desktop home screen, under *External Views*, double-click your Google Drive connection.
2. Select the Google Workspace account that you want to use for signing in with Google.



3. Click **Allow** to give M-Files Google Drive Connector permissions to access your Google Workspace account.



You can now access your Google Drive documents via the M-Files Google Drive Connector connection.

7.2 Searching for Documents

Content from the Google Drive connection can be located and accessed via the search like any other M-Files object.

> **Note**: In M-Files Desktop, when the user has opened the Google Drive external view, the *View* option on the *Filters* tab does not work. The *View* option can be used for other external repositories than the Google Drive connection.

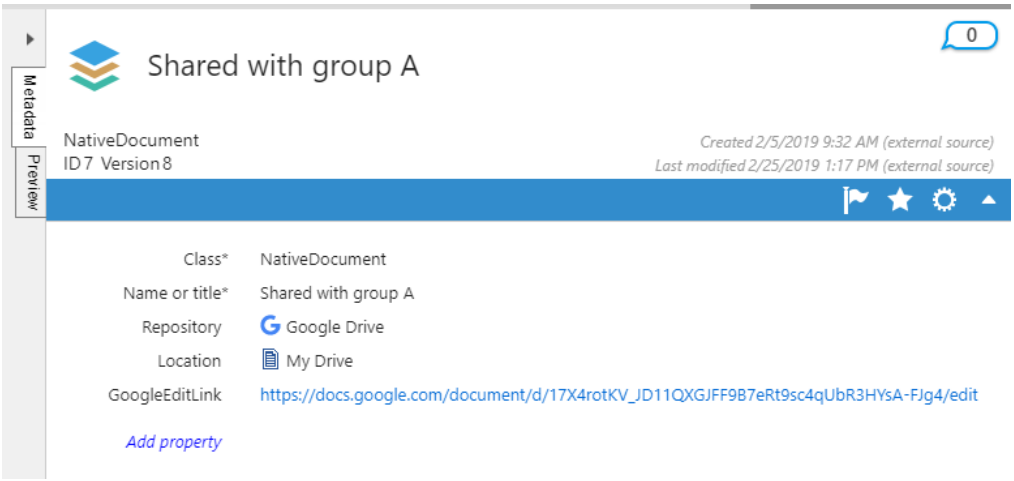
> **Note**: If a Google Drive document has a name that ends with a period, M-Files adds an asterisk (*) to the end.

We recommend that you do not use names that end with a period for Google Drive documents that are shown in M-Files.

7.3 Viewing and Editing Native Google Drive Documents in M-Files

Complete the following steps to view and edit native Google Drive documents in M-Files:

1. In M-Files, locate and select a native Google Drive document obtained via the M-Files Google Drive Connector connection.
2. On the metadata card of the document, click the Google edit link.



For M-Files Mobile, if the Google Workspace account of your mobile device does not have access rights to the document, a *Request access* screen is shown. In the screen, you can request access to the document.

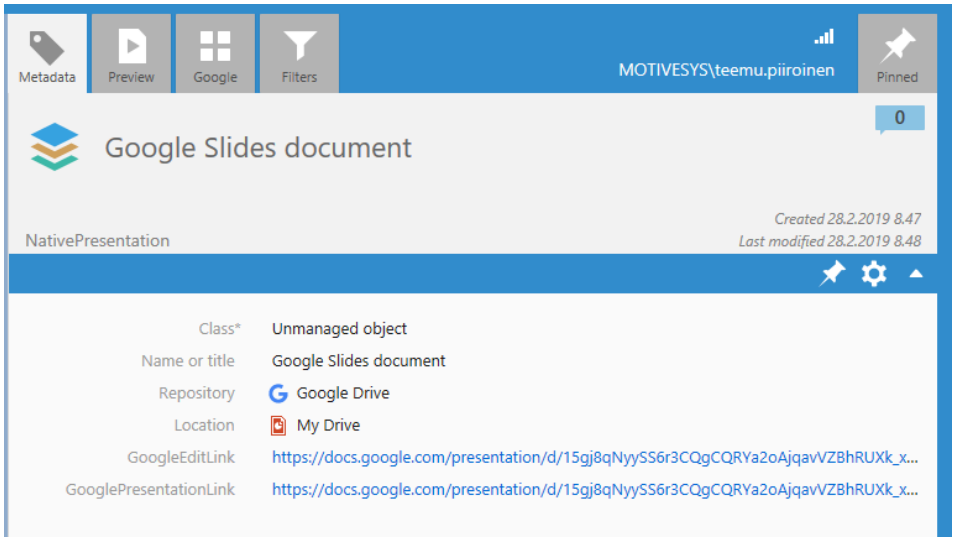
Remarks about document editing

- The **Check Out** and **Check In** commands are not used for native Google Drive documents.
- The Google user used for editing is not automatically logged out if the M-Files user logs out and another user logs in. A manual logout must be done if multiple M-Files users are using the same Windows account to access Google Drive through M-Files.

Whether a document can be edited via Google Drive is based on the MIME-type of the file and not the extension. Changing the extension of a document to one of the supported extensions does not make it editable. For information about editing Microsoft Office files via Google Drive, see Work with Office files using Office editing.

7.4 Opening a Google Slides Document in Presentation Mode

You can open a Google Slides document in the presentation mode by accessing the Google presentation link on the metadata card. Clicking the Google presentation link opens a new browser window for viewing the presentation.



7.5 Editing Native Google Files Outside of M-Files and Showing Correct Icons

If you want to be able to open native Google files for editing in your browser instead of through M-Files, you need to install either Google Drive or Google File Stream. With either of these programs installed, it is possible to double-click any native Google file to open it in your default browser. Installing any of the programs will also make the correct icons for the native files show in the file listings.

It is not necessary to log in to the programs to get this functionality. It is enough to have the programs installed.

8. Change History

The table below describes the essential changes by document version.

VERSION	DATE	ESSENTIAL CHANGES
1.0	2019/01/17	Initial published version.
1.1	2019/02/21	Added section 7.
1.2	2019/04/30	Updated section 2.1. Added notes to sections 7.2 and 5.4. Updated the second table in section 5.1.4.
1.3	2019/07/18	Added section 0. Note about special app properties added to sections 5.3 and 6. Remarks about file conversion limitations and object promotion added to section 7. Section 7.2 updated.
1.4	2019/09/06	Minor updates throughout the document.
1.5	2019/09/20	Updated the M-Files Google Drive Connector version requirement in section 2.1
1.6	2019/12/04	Added information about the View filter.
1.7	2020/02/21	Added Appendix A. Added information about port settings to the prerequisites section.
1.8	2020/05/05	Added information about the M-Files for Google Workspace add-on and a link to its configuration instruction to section 1.

1.9	2020/06/02	Added section 5.1.6.
2.0	2020/09/22	Updated the connector-specific settings in section 5.2.
2.1	2020/12/11	Added information about the filtering settings to section 5.2.
2.2	2021/01/11	Added information about the service account settings to section 5.2.
2.3	2021/02/23	Moved the contents of abstract to section 1. Removed from several sections instructions related to using the Native Editing UI extension because it is no longer supported. Minor other changes throughout the document.
2.4	2021/03/03	In section 6, highlighted the importance of sharing all documents with the indexer user.
2.5	2021/03/19	Removed the description of the *User to impersonate* setting.
2.6	2021/06/15	Note about Google endpoints added to section 2.2.2. Added notes to sections 4.1 and 4.2.
2.7	2021/10/26	Implementation now requires a Google service account. The indexer user can now be a normal user account with no administrative rights in Google Drive. Multiple sections edited.
2.8	2021/12/03	Section 3 updated. Many other, smaller updates.
2.9	2021/12/17	Information about site verification moved to the prerequisites section.
3.0	2022/02/04	Added section 2.2.1 and added to section 1 that free Gmail accounts are not supported.
3.1	2022/05/03	Added information about file names that end with period to section 7.2.
3.2	2023/07/28	Improvements to section 5.1.2.

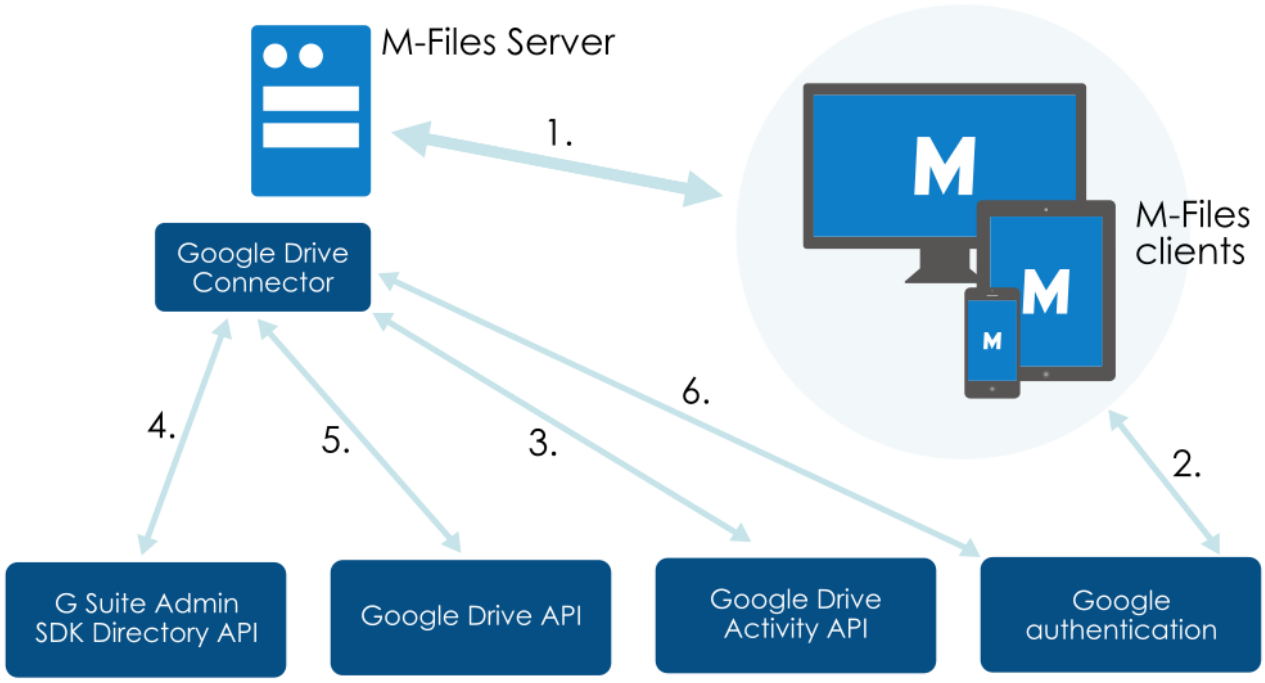
9. Reference Documents

You may want to see these articles for additional information:

- Setting Up and Using M-Files for Google Workspace Add-On
- M-Files Intelligent Metadata Layer

Appendix A: Connections Between M-Files and Google

The following diagram provides an overview of the connections between the M-Files server, M-Files clients, and the various, required Google components, such as Google Drive API and Google Authentication.



1) The connection between the M-Files server and M-Files clients can use the RPC, HTTPS, or gRPC protocol.

- See Adding a Vault Connection for an overview about vault–server connections, and Enabling RPC over HTTPS connections to M-Files Server for a more detailed description.

2) The authentication between M-Files clients and the Google authentication service is carried out by using HTTPS and OAuth 2.0.

- Google Drive Connector does not perform the authentication, but instead delegates it to the M-Files client.
- Authorization endpoint: `https://accounts.google.com/o/oauth2/v2/auth`
- Token endpoint: `https://www.googleapis.com/oauth2/v4/token`

3) Communication between Google Drive Connector and Google Drive Activity API is performed over HTTPS and is required for indexing new and modified documents.

- Use Google API Client Library for working with Drive Activity API v2.
- Endpoint: `https://driveactivity.googleapis.com/v2`

4) Communication between Google Drive Connector and Google Workspace Admin SDK Directory API performed over HTTPS and is required for collecting information about users and user groups.

- Use Google API Client Library for working with Admin Directory API v1.
- Endpoint: `https://www.googleapis.com/admin/directory/v1`

5) Communication between Google Drive Connector and Google Drive API is performed over HTTPS and is required for interacting the Google Drive service.

- Use Google API Client Library for working with Drive API v3.
- Endpoint: `https://www.googleapis.com/drive/v3`

6) Communication between Google Drive Connector and the Google authentication service is performed over HTTPS and is required for refreshing the access token.

- Endpoint: `https://oauth2.googleapis.com/token`