

M-FILES CORPORATION

M-FILES URL PROPERTIES

LAST UPDATED 02 OCTOBER 2025

VERSION 3.1

Contents

1.	Introduction	3
1.1	Basic URL Command Structure	3
2.	Displaying Objects and Files	3
2.1	Mobile	4
2.2	Examples	4
3.	Displaying Views.....	5
3.1	Mobile	5
3.2	Example.....	5
4.	Changing the Appearance of Hyperlinks	6
5.	Creating New Objects.....	7
5.1	Mobile	8
5.2	Examples	8
5.3	Additional Parameters for the Android App.....	9
5.3.1	Examples.....	9
6.	Searching for Objects	10
6.1	Mobile	12
6.2	Examples	12
7.	Adding Vault Connections	14
7.1	Example.....	14
8.	Setting the Pre-Shared Key.....	14
8.1	Example.....	15
9.	URL Resolving Logic.....	15
9.1	URLs Generated in M-Files 9.0 and Later	15
9.2	URLs Generated Before M-Files 9.0	15
9.3	Optimizing URL Handling with Registry Settings	16
9.3.1	Forced Vault Mappings.....	16
9.3.2	Preferred Vault Mappings	16
10.	Change History	16

1. Introduction

Note: This document is for the M-Files hyperlinks that start with m-files:// and are created with the **Get Hyperlink** option. In M-Files January '24 Update and later, these hyperlinks can no longer be created.

For information on the link option that is available for the M-Files version December '22 and later, refer to [Configuring M-Files Links](#).

Different M-Files clients implement URL handlers that allow interacting with the client software by executing certain URLs. This document describes the different URL commands and the parameters that they accept for the classic M-Files Desktop and M-Files Mobile. If you use the classic M-Files Web, refer to [Creating URLs in the Classic M-Files Web](#).

1.1 Basic URL Command Structure

Below, the URL syntax is described in tabular form. The required parameters are wrapped in angle brackets. Optional parameters are surrounded by square brackets. The table given here shows the URL structure that is common for all URLs.

URL	m-files://[command]/<vault>/<parameters>
m-files://	The name of the protocol, followed by a colon and two forward slashes.
command	The URL command that defines what the URL does. If command is omitted, M-Files opens the root of the M-Files drive.
vault	Local name or the GUID of the document vault that the URL targets. If the local name of the vault contains spaces, replace them with %20, which is the URL-encoded counterpart of the space character. For example: Instead of Sample Vault, specify Sample%20Vault in the URL. Note that often the vault GUID is a more reliable identifier for the target vault because different users might have different local names for the same vault. Note also that M-Files mobile apps only support GUID as the vault parameter.
parameters	Parameters for the command. These differ by the command.

2. Displaying Objects and Files

There are five URL commands that can be used for displaying objects or files. These commands differ in what information they display and whether they take any actions upon the object automatically. The syntax for these commands is described in the table given here.

URL	m-files://<command>/<vault>/<type><-<id>[-version]][/fileId]? [object=guid]&[file=guid]
command	Command name: show, open, view, edit, or showmetadata.
vault	Local name or the GUID of the document vault that the URL targets.
type	Numerical ID of the object type, for example, 0 for document.
id	Numerical internal ID of the object, or external ID wrapped in single quotes (').
version	Optional version number. If omitted, the URL refers to the latest version of the object.
fileId	Optional file ID, required when opening files from a multi-file document (MFD).
object=guid	The optional object parameter can be used for specifying the GUID of the target object. If specified, the GUID will be used instead of the other ID information if possible and can enable locating the target object in another vault as well (such as a replicated vault).

file=guid	The optional file parameter can be used for specifying the GUID of the target file. If specified, the GUID will be used instead of the other ID information if possible and can enable locating the target file in another vault as well (such as a replicated vault).
-----------	--

The differences between the commands are described in the table given here.

Command	Action
show	Displays the object in M-Files.
showmetadata	Shows the metadata card for the object.
open	Shows the check-out dialog and opens the document in the default application.
view	Opens the document in the default application in read-only mode.
edit	Checks out the object and opens it for editing in the default application.

2.1 Mobile

In the mobile clients, only the **show** command is implemented. If any other command is used, it will not do anything special and just behave the same way as the **show** command. External IDs are supported in the commands of mobile clients in app version 3.9.0 and later.

2.2 Examples

Notice that actual URLs must not have any line breaks.

Displays the document with ID 353 in the M-Files user interface:

```
m-files://show/FFE93D16-218E-468C-92A6-02B1149493F2/0-353
```

Displays the document with ID 353 in the M-Files user interface:

```
m-files://show/Sample%20Vault/0-353
```

Displays the file with ID 327 of the document with ID 308 in the M-Files user interface, including the object and file GUID information in the URL:

```
m-files://show/FFE93D16-218E-468C-92A6-02B1149493F2/0-308/327?
object=93757AF5-7129-4D80-B1B2-942E8F5067FD&
file=0AE14C90-3157-4E7B-B5DB-BEBA62AA083B
```

Displays the metadata card for the document with ID 353:

```
m-files://showmetadata/FFE93D16-218E-468C-92A6-02B1149493F2/0-353
```

Opens the document with ID 353 in the default application in read-only mode:

```
m-files://view/FFE93D16-218E-468C-92A6-02B1149493F2/0-353
```

Opens the document with ID 441 in the default application in edit mode:

```
m-files://edit/FFE93D16-218E-468C-92A6-02B1149493F2/0-441
```

Opens the file with ID 476 of the document with ID 438 in the default application, prompting the user to check out the document if not already checked out:

```
m-files://open/FFE93D16-218E-468C-92A6-02B1149493F2/0-438/476
```

3. Displaying Views

In addition to displaying single objects, views can also be opened with M-Files URLs. The basic syntax is described below.

URL	m-files://<command>/<vault>/[path]
command	Command name: viewid or viewidenc .
vault	Local name or the GUID of the document vault that the URL targets.
path	Optional view path. If omitted, M-Files opens the vault root.

When the path is specified, it must start with a view ID, optionally followed by a sequence of folder levels. The view ID and each folder level is separated with a slash (/).

If the command is **viewid**, the folder levels represent values that are specified by using the simple syntax of the **propertyvalue** parameter as specified in the "Searching for objects" section later in this document.

If the command is **viewidenc**, the folder levels are base64-encoded serialized typed value bytes. Additionally, any forward slashes in the base64-encoded data need to be replaced with the character sequence {2} to avoid breaking the overall URL structure by introducing additional forward slashes to it. This allows referring to views that use non-lookup fields as folders, such as views that group items by a date property or another property that cannot be represented with the simple syntax.

3.1 Mobile

For mobile clients, only the **viewid** command is implemented, and using **viewidenc** will not work at all.

- M-Files Mobile App for iOS version 2.6.0 and later.
- M-Files Mobile App for Android version 2.6.3 and later.

External IDs are supported in the commands of mobile clients in app version 3.9.0 and later.

3.2 Example

The URLs must not have any line breaks.

This URL displays the **Picture** folder in the **1. Documents\By Class** view:

```
m-files://viewid/Sample%20Vault/134/86
```

The ID of the **Picture** class is 86, and the ID of the **By Class** view is 134. Do not specify parent view levels.

This URL shows 2004 February Proposals in the **1. Documents\Proposals by Year and Month** view:

```
m-files://viewid/Sample%20Vault/200/T2004-02
```

The text value 2004-02 is used as input in the grouping level function for the **Proposals by Year and Month** view, the ID of which is 200.

4. Changing the Appearance of Hyperlinks

You can use registry settings to change the appearance of M-Files hyperlinks created with the **Get Hyperlink** operation. Add the registry settings to HKEY_LOCAL_MACHINE for client-wide settings or to HKEY_CURRENT_USER for user and vault settings, or to both. HKEY_CURRENT_USER is more important when registry settings are added for both.

In HKEY_LOCAL_MACHINE, the key name is: HKEY_LOCAL_MACHINE\SOFTWARE\Motive\M-Files\<version>\Common\Settings\Hyperlinks.

In HKEY_CURRENT_USER, the key name is: HKEY_CURRENT_USER\SOFTWARE\Motive\M-Files\<version>\Client\MFShell\<vault>\Hyperlinks.

Add these settings on the target computer to change the way hyperlinks are shown:

Value name	FriendlyNameModeForHTML
Value type	REG_DWORD
Description	Defines if the object title is a link in the HTML version of the hyperlink. Values from 1 to 3 also define the type of the link.
Value data	<ul style="list-style-type: none">• 0: No hyperlink• 1: Classic M-Files Desktop hyperlink (default)• 2: Classic M-Files Web hyperlink• 3: M-Files Mobile hyperlink

Tip: M-Files also adds the link to your clipboard in the XML Spreadsheet format. This lets you paste the link to a cell in Microsoft Excel. The type of the link is specified in the FriendlyNameModeForHTML value. If its value is set to 0 or is not specified, the link opens the object in the classic M-Files Desktop.

Value name	CanShowSpecificLinks
Value type	REG_DWORD
Description	Controls if the HTML version of the hyperlink can include client-specific links after the object title.
Value data	<ul style="list-style-type: none">• 0: No client-specific links are appended.• 1: Client-specific links are appended as specified in the SpecificLinksMode value.

Value name	SpecificLinksMode
Value type	REG_DWORD
Description	If CanShowSpecificLinks is set to 1, controls which of the client-specific links the HTML version of the hyperlink contains. Save the value with the decimal base (Base: Decimal).

	<p>Note: To use links to the new and classic M-Files Web, the URLs to their home pages must be specified in the vault properties. For more information, refer to Document Vault Advanced Properties in the M-Files user guide.</p>
Value data	<ul style="list-style-type: none"> • 0: Automatic (adds all the available client-specific links) • 1: Classic M-Files Desktop • 2: Classic M-Files Web • 3: Classic M-Files Desktop and classic M-Files Web • 4: M-Files Mobile • 5: Classic M-Files Desktop and M-Files Mobile • 6: M-Files Mobile and classic M-Files Web • 7: Classic M-Files Desktop, M-Files Mobile, and classic M-Files Web • 8: New M-Files Web • 9: Classic M-Files Desktop and new M-Files Web • 10: New and classic M-Files Web • 11: Classic M-Files Desktop, new and classic M-Files Web • 12: New M-Files Web and M-Files Mobile • 13: Classic M-Files Desktop, new M-Files Web, and M-Files Mobile • 14: New M-Files Web, M-Files Mobile, and classic M-Files Web • 15: Classic M-Files Desktop, new M-Files Web, M-Files Mobile, and classic M-Files Web

Value name	PlainTextModeForClipboard
Value type	REG_DWORD
Description	Controls the content of the plain-text version of the hyperlink copied to the clipboard.
Value data	<ul style="list-style-type: none"> • 0: All hyperlinks • 1: Classic M-Files Desktop hyperlink only • 2: Classic M-Files Web hyperlink only • 3: M-Files Mobile hyperlink only

5. Creating New Objects

The metadata card can be opened so that it can be interactively used to create a new object. The basic syntax for this URL is below. Some parameter combinations are invalid, and you can get an error when using such a combination.

Command syntax for object creation:

URL	m-files://newobject/<vault>/<type>[?name1=value1 [&name2=value2 [&...]]]
vault	Local name or the GUID of the document vault that the URL targets.
type	Numerical ID of the object type, for example, 0 for document.
[?name1=value1 [&name2=value2 [&...]]]	Query string containing optional parameters, see the table below. Parameter names are case insensitive.

Available query parameters for the **newobject** command:

Parameter	Description
-----------	-------------

class=<value>	Class to be used for the new object. Value is the numerical ID of the class.
template=<value>	Template to be used for the new object. Value is the numerical object ID of the template object (object type is selected implicitly, only the latest version is used). The provided template must match the target class, either directly or indirectly via the "Additional classes" property. The properties of the template will be used instead of the properties in the URL. This includes also the Class property.
titleplain=<value>	Title to be used for the name property of the new object. The value is a plain string. Encode spaces as %20.
title=<encodedvalue>	Title to be used for the name property of the new object. The value is a base64-encoded string. Additionally, any forward slashes in the base64-encoded data need to be replaced with the character sequence {2} to avoid breaking the overall URL structure by introducing additional forward slashes to it.
ext=<value>	Extension to be used for the new object if the SFD mode is selected on the metadata card. The value is a plain string without a leading dot.
view=<path>	View folder to be used for prefilling the object properties. Path follows the convention used with the viewid URLs (see "Displaying views").
property=<propdefID>/<value> property=<propdefID>/-/<encodedvalue>	Property value to be used for prefilling the object properties. PropdefID is the numerical ID of the property definition. The value can take one of the two available forms: <value>: A lookup value can be included in the same way as individual folder references in the view path. See the propertyvalue parameter in the "Searching for objects" section for the syntax of specifying values of different data types. <encodedvalue>: Any typed value can be encoded in the same way as in the viewidenc URLs (see "Displaying views"). Note the additional '-' segment between the property definition ID and the encoded value. You can specify an arbitrary number of property parameters. However, duplicate property values with identical property definition IDs are not merged.

5.1 Mobile

This command in mobile will not make use of the **ext** or **view** parameters (it does not stop the user from opening the link, they are just ignored). The properties cannot be given with an encoded value. In those cases, they will be ignored, but the link will continue working. For Android there are two additional parameters, go to section 5.3 to see a description of these extra parameters. External IDs are supported in the commands of mobile clients in app version 3.9.0 and later.

5.2 Examples

Notice that actual URLs must not have any line breaks.

Displays the New Document window with the following properties already filled in:

Class = Proposal, Customer = ESTT Corporation; Warwick Systems, Keywords = some keywords, File format = DOCX

```
m-files://newobject/Sample%20Vault/0?
property=100/5&
property=1079/141,142&
property=1004/Tsome%20keywords&
ext=docx
```

Displays the New Customer window with the following properties already filled in:

Title = My First Customer

```
m-files://newobject/Sample%20Vault/136?
titleplain=My%20First%20Customer
```

Displays the New Customer window with the following properties already filled in:
Title = ACME. Uses base64 encoding to ensure that the special characters are properly transmitted.

```
m-files://newobject/Sample%20Vault/136?  
title=QUNNRQ==
```

Displays the New Document window with the class *Sales Invoice* prefilled. Prefilling of the class occurs because the view parameter specifies the *By Class* view and the *Sales Invoice* folder in it:

```
m-files://newobject/Sample%20Vault/0?  
view=134/89
```

5.3 Additional Parameters for the Android App

M-Files Mobile App for Android supports the following additional parameters for URLs that use the **newobject** verb. These parameters are compatible with the Android app version 2.5.5 and later.

Available additional query parameters for the **newobject** command (compatible with the Android app only):

Parameter	Description
close[=<value>]	When no value is added, or if the value is yes or true, this property will cause the app to be closed (pushed into the background) once the new object has been either created or canceled.
propertyfilter=<propdefID>/<value>	<p>Property value to be used for filtering from the titles of the object properties.</p> <p><propdefID>: is the numerical ID of the property definition. If the property defined by the ID is not of type lookup, this parameter will be ignored.</p> <p><value>: A plain string with spaces encoded as %20. This value will be used as the filter to show only those lookup values which title contain it.</p> <p>You can specify an arbitrary number of propertyfilter parameters. However, only the last one of duplicate parameters with identical property definition IDs will be considered.</p> <p>If the same property definition ID is used in a property parameter and a propertyfilter parameter, the propertyfilter parameter takes precedence. Only if the user does not choose one of the filtered values and closes the selection dialog, the property parameter will be used.</p>

5.3.1 Examples

Notice that actual URLs must not have any line breaks.

Displays the New Document window with the following properties already filled in:
Title = Proposal, Class = Proposal

Displays a dialog to choose between all the customers that include in their name “systems” (for example, “Warwick Systems”).

When the object gets saved (or cancelled) the app will close.

```
m-files://newobject/FFE93D16-218E-468C-92A6-02B1149493F2/0?  
titleplain=Proposal&  
property=100/5&
```

```
propertyfilter=1079/systems&
close=yes
```

Displays the New Customer window with the following properties already filled in:

Title = My First Customer

When the object gets saved (or cancelled) the app will close.

```
m-files:///newobject/FFE93D16-218E-468C-92A6-02B1149493F2/136?
titleplain=My%20First%20Customer&
close
```

Displays the New Customer window with the following properties already filled in:

Title = My First Customer

It will display a dialog to choose between all the customers that include in their name “systems” (for example, “Warwick Systems”). If the user does not select any, it will use by default “Warwick Systems” (ID 142).

```
m-files:///newobject/FFE93D16-218E-468C-92A6-02B1149493F2/136?
titleplain=My%20First%20Customer&
propertyfilter=1079/systems&
property=1079/142
```

6. Searching for Objects

M-Files URL can be applied to execute a search and to display search results in a new window or to perform an action on the object that the search returns. The basic syntax for the URL is described in the table given here.

Search command syntax:

URL	<code>m-files:///search/<vault>/[?name1=value1[&name2=value2[&...]]]</code>
vault	Local name or the GUID of the document vault that the URL targets.
[?name1=value1 [&name2=value2 [&...]]]	Query string containing optional parameters, see the table below. Parameter names are case insensitive.

Available query parameters for the search command:

Parameter	Description
propertyvalue=<propdefID>/<value>	Property value condition. The parameter propdefID is the numerical ID of the property definition. The value can take one of the two available forms:
propertyvalue=<propdefID>/-/<encodedvalue>	<value>: A simple value that requires no encoding, such as a lookup ID or a simple text string with no special characters. For more complex values you should use the encodedvalue syntax. Supported values for the simple syntax are as follows: <i>Lookup value:</i> The value can be an internal object ID or value list ID, or an external object ID, or null. The internal ID is just a single number, e.g., 1234. An external ID is in the form <object type ID>-'<external ID>'. For example: 106-'XYZ2232'
	You can specify multiple lookup values by separating them with commas. For example: 1234,5543,765. The same applies to external IDs, e.g., 106-'XYZ2232',106-'XYZ6655'.

	<p>Text value: A simple text string, preceded by an uppercase T. For example: Tmystring.</p> <p><i>Multi-line text value:</i> A simple text string, preceded by an uppercase M. For example: <i>Mmystring%0D%0Asecond%20line</i>. Encode spaces as %20 and newlines as %0D%0A. For more complex values, you may also use the <code>encodedvalue</code> syntax.</p> <p>Integer value: An integer, preceded by an uppercase I. For example: I987</p> <p>Boolean value: A Boolean value (true or false), preceded by an uppercase B. For example: Btrue or Bfalse</p> <p>In the simple syntax described above, the data type prefix followed by an empty string denotes a null (empty) value. For example, B stands for a null Boolean value, and T stands for a null text value.</p> <p><code><encodedvalue></code>: Any typed value can be encoded in the same way as in the viewidenc URLs (see "Displaying views"). Note the additional '-' segment between the property definition ID and the encoded value.</p> <p>You can specify an arbitrary number of property parameters. However, duplicate property values with identical property definition IDs are not merged.</p>
<p><code>typedvalue=<ObjTypeId>/<value></code></p> <p><code>typedvalue = < ObjTypeId >/-/<encodedvalue></code></p>	<p>Typed value condition. The parameter <code>objTypeId</code> is the numerical ID of the object type definition. It is applied only with lookup values (specify zero for other data types). The value can take one of the two available forms, as specified above for <code>propertyvalue</code>.</p>
<p><code>objecttype=<value></code></p>	<p>The value specifies the object type (numerical ID) to search for.</p>
<p><code>action=<actionname></code></p>	<p>The name of the action to perform on the search result. Can be one of the following:</p> <p><code>showall</code>: Shows the search results in a new window. This is the default action if the action parameter is not specified.</p> <p><code>show</code>: Shows the specified object or file in a new window. Requires that the search returns exactly one matching object. If no objects match the criteria, an error is returned. If more than one object matches the criteria, the <code>showall</code> action is performed instead.</p> <p><code>open</code>: Opens the specified object or file in the appropriate application, prompting the user with the checkout option if the object is not already checked out. Requires that the search returns exactly one matching object. If no objects match the criteria, an error is returned. If more than one object matches the criteria, the <code>showall</code> action is performed instead.</p> <p><code>view</code>: Opens the specified object or file in the appropriate application in read-only mode, i.e., without checking the object out. Requires that the search returns exactly one matching object. If no objects match the criteria, an error is returned. If more than one object matches the criteria, the <code>showall</code> action is performed instead.</p> <p><code>edit</code>: Opens the specified object or file in the appropriate application in edit mode, i.e., automatically checks the object out. Requires that the search returns exactly one matching object. If no objects match the criteria, an error is returned. If more than one object matches the criteria, the <code>showall</code> action is performed instead.</p> <p><code>showmetadata</code>: Displays the metadata card for the specified object. Requires that the search returns exactly one matching object. If no objects match the criteria, an error is returned. If more than one object matches the criteria, the <code>showall</code> action is performed instead.</p>
<p><code>filetitle=<value></code></p> <p><code>filetitle64= <encodedvalue></code></p> <p><code>fileext=<value></code></p> <p><code>fileext64= <encodedvalue></code></p> <p><code>fileindex= <indexvalue></code></p> <p><code>fileid= <idvalue></code></p>	<p>These parameters allow for specifying the target file in a multi-file document. For example, the <code>view</code> action would typically be targeted to an individual file. The first file that matches all specified file criteria is accepted.</p> <p>The parameter names with the 64 suffix use a base64-encoded version of the actual string. In addition, make a note of these remarks:</p> <ul style="list-style-type: none"> Any forward slashes in the base64-encoded data need to be replaced with the character sequence <code>{2}</code>, in order to avoid breaking the overall URL structure by introducing additional forward slashes to it.

	<ul style="list-style-type: none"> The search64 parameter should be encoded with ISO-8859-1. Otherwise, the search string can be corrupted during the search. <p>The filetype parameter can be used for specifying the title part of the file name. For example, <i>filetitle=signed</i> would match both Signed.pdf and Signed.tif.</p> <p>The fileext parameter can be used for specifying the extension part of the file name, without the leading dot. For example, <i>fileext=pdf</i> would match Agreement.pdf but not Agreement.doc.</p> <p>The fileindex parameter can be used for identifying the target file with a one-based index. For example, <i>fileindex=1</i> will select the first file in the multi-file document.</p> <p>The fileid parameter can be used for specifying the ID of the target file. File IDs are not in general visible in the M-Files UI but can be retrieved with M-Files API calls. For example, <i>fileid=6547</i> will match the file with the ID 6547 only.</p> <p>File criteria are applied only for the object that is found when the search is executed. That is, you cannot use file criteria without specifying some object criteria as well.</p>
<pre>search= <value> search64= <encoded value> ftsflags= <flags></pre>	<p>These parameters allow performing full-text searches in the classic M-Files Desktop.</p> <p>Note: Requires M-Files 2018 or later.</p> <p>The search parameter can be used for specifying the search phrase for a full-text search condition. Use this parameter to specify the words to search for, just as they would be typed in the quick search box in the M-Files user interface.</p> <p>The search64 parameter is similar to the search parameter but it uses a base64-encoded version of the actual string. Additionally, any forward slashes in the base64-encoded data need to be replaced with the character sequence {2} to avoid breaking the overall URL structure by introducing additional forward slashes to it.</p> <p>The ftsflags parameter specifies the full-text search flags for the search as an MFFullTextSearchFlags enumeration. By default, the full-text search mode is "All words", "Metadata and file contents", and searching in inflected forms of the words is enabled. This corresponds to the combination of the flag values: MFFullTextSearchFlagsTypeAllWords (131072), MFFullTextSearchFlagsLookInMetaData (268435456), MFFullTextSearchFlagsLookInFileData (536870912), and MFFullTextSearchFlagsStemming (4).</p>

6.1 Mobile

The **search**, **search64** and **ftsflags** parameters are available for the following mobile client versions:

- M-Files Mobile App for iOS version 3.5.0 and later.
- M-Files Mobile App for Android version 3.5.0 and later.

For the **action** parameter, **showall** option behaves as expected and is the default. Then **show**, **open**, **view** and **edit** all behave the same way, if there is only one result it will open it (an object, or a file if it is specified). The option **showmetadata** will always show the metadata of the object even if a file is indicated.

- M-Files Mobile App for iOS version 2.6.0 and later.
- M-Files Mobile App for Android version 2.6.5 and later.

External IDs are supported in the commands of mobile clients in app version 3.9.0 and later.

6.2 Examples

Notice that actual URLs must not have any line breaks.

Search for documents (object type 0) that refer to an assignment object (object type 10) with ID 1234.

```
m-files://search/FFE93D16-218E-468C-92A6-02B1149493F2?
objecttype=0&
typedvalue=10/1234
```

Search for all objects that refer to the object of type 106 having the external ID XYZ2232.

```
m-files://search/VaultName?
typedvalue=106/106-'XYZ2232'
```

Search for all objects that have a NULL value in their property 1140.

```
m-files://search/FFE93D16-218E-468C-92A6-02B1149493F2?
propertyvalue=1140/NULL
```

Search for a single document that matches the criteria Customer = ESTT Corporation AND Proposal number = 7720. Opens the document in edit mode.

```
m-files://search/Sample%20Vault?
objecttype=0&
propertyvalue=1079/141&
propertyvalue=1103/I7720&
action=edit
```

Search for a single document that matches the criteria Project = Web Site Graphics for ESTT AND Class = Confirmation of Order or Agreement. Opens the first PDF file in the document in read-only mode.

```
m-files://search/Sample%20Vault?
objecttype=0&
propertyvalue=1078/22&
propertyvalue=100/25,4&
fileext=pdf&
action=view
```

Perform a full-text search with the search phrase *estt proposal* in the vault with the GUID 2F005834-52BB-42B6-BD1D-7E272F60BA9A.

```
m-files://search/2F005834-52BB-42B6-BD1D-7E272F60BA9A/?
search=estt proposal
```

Note: Supported in M-Files 2018 and later.

Perform a full-text search with the search phrase *estt proposal* from metadata only using the any word condition (ftsflags 268435456 + 262144 = 268435456) in the vault with the GUID 2F005834-52BB-42B6-BD1D-7E272F60BA9A using the 64-bit-encoded search parameter.

```
m-files://search/2F005834-52BB-42B6-BD1D-7E272F60BA9A/?
search64=ZXN0dCBwcm9wb3NhbA==&
ftsflags=268435456
```

Note: Supported in M-Files 2018 and later.

7. Adding Vault Connections

Administrators can set a QR code or create links to easily establish vault connections in end-users' mobile apps. To login, the end user needs to scan the QR code or open the link from the email with their mobile device and input their username and password. After a successful login, the vault connection information is saved to the user's device. For more information on adding vault connection with QR code, see: [Connecting to Vaults with QR Codes in M-Files Mobile.pdf](#).

Note: Adding vault connection with link will not be supported in Mobile client version 23.12.0 and later.

The **addvault** command syntax:

URL	m-files://addvault?url=<server url>&vault=<vault GUID>[&name=<vault name>&key=<pre-shared key>]
url	URL to M-Files Web Access Server
vault	Vault GUID without the { and } brackets.
name	Optional parameter. The display name of the vault connection.
key	Optional parameter. Pre-shared key. Refer to Securing Access to M-Files Vaults with a Pre-Shared Key article in M-Files Knowledge Base for more details about pre-shared keys.

7.1 Example

Notice that actual URLs must not have any line breaks.

Add a vault connection to Sample Vault (vault GUID {1DF46AF7-B2E1-4D12-9901-BB7ABACC79D1}) on <https://mfiles.mydomain.com> if the server requires a pre-shared key 'qiPvhh34OREFsZww1BuPTDFYWFeOxg2kwQfGaXMmn0EQCoAexa0KWaU'.

```
m-files://addvault?url=https://m-files.mydomain.com&
vault=1DF46AF7-B2E1-4D12-9901-BB7ABACC79D1&
name=Sample%20Vault&
key=qiPvhh34OREFsZww1BuPTDFYWFeOxg2kwQfGaXMmn0EQCoAexa0KWaU
```

8. Setting the Pre-Shared Key

The easiest way to set the pre-shared key for mobile clients is by using an M-Files URL that will save it in the device.

Note: Pre-shared keys are not used with gRPC connections.

The **setkey** command syntax:

URL	m-files://setkey?url=https://<servername>&key=<presharedkey>
url	URL to M-Files Web Access Server
key	Pre-shared key. If empty or not added, it will remove a previously set up key. Refer to Securing Access to M-Files Vaults with a Pre-Shared Key article in M-Files knowledge base for more details about pre-shared keys.

8.1 Example

Notice that actual URLs must not have any line breaks.

Save a pre-shared key 'qiPvhh34OREFsZww1BuPTDFYWFeOxg2kwQfGaXMmn0EQCoAexa0KWaU' to be used with the server `https://mfiles.mydomain.com`.

```
m-files://setkey?url=https://m-files.mydomain.com&
key=qiPvhh34OREFsZww1BuPTDFYWFeOxg2kwQfGaXMmn0EQCoAexa0KWaU
```

9. URL Resolving Logic

This section explains the basic logic of how M-Files URL resolver manages M-Files URLs.

9.1 URLs Generated in M-Files 9.0 and Later

The M-Files URLs generated with M-Files 9.0 and later contain the "object=guid" and "file=guid" query string parameters.

The URL "m-files://show/ACF7FC8D-2A41-43FD-BACC-FDD895870357/0-91026?object=487B5435-7AC8-408E-BBB1-0EC206C7021E" is resolved in M-Files 9.0 and later as follows:

1. If the vault connection to the indicated vault is configured, the object identification is interpreted as local IDs within that vault. This is the default resolution step.
2. If the object was not found in step 1, all the available vault connections are then examined. A vault connection is considered available if the vault connection is configured and the user is already logged in using the connection. For each such connection, the value of the "object" query string parameter is used as an Object GUID when searching for the object.

Additionally, it is possible to optimize the vault scanning phase in step 2 by configuring referenced vault connections to be used instead of scanning all available vault connections. See section [9.3.2](#) for details.

9.2 URLs Generated Before M-Files 9.0

The M-Files URLs generated with M-Files 8.0 and earlier do not contain the "object" and "file" query string parameters. They contain only the verb, vault GUID, object type ID, object ID, and optionally version ID.

The URL "m-files://show/ACF7FC8D-2A41-43FD-BACC-FDD895870357/0-91026" is resolved in M-Files 9.0 and later as follows:

1. If the vault connection to the indicated vault is available, the object identification is interpreted as local IDs within that vault. This is the default resolution step.
2. If the object was not found in step 1, all the available vault connections are then examined. A vault connection is considered available if the user is already logged in using the connection. For each such connection, the vault GUID and the object identification are interpreted as the original identity of the object (original vault GUID + original object type ID + original object ID) when searching.

9.3 Optimizing URL Handling with Registry Settings

M-Files URLs work across multiple vaults and can be used for replicated objects. You can optimize the URL resolving with the registry values discussed in this section.

9.3.1 Forced Vault Mappings

You can specify mappings between replicated vaults. If, for instance, a vault with {B7F9864E-ABB5-48BA-8FAA-C50DFFC27F6F} GUID is a replica of the vault {3278463C-AC3E-4077-8AE0-9D563D66AD75}, you can force M-Files Client to always search for objects with {B7F9864E-ABB5-48BA-8FAA-C50DFFC27F6F} vault GUID in {3278463C-AC3E-4077-8AE0-9D563D66AD75} vault.

Key	HKEY_CURRENT_USER\SOFTWARE\Motive\M-Files\<version>\Client\MFStatus\VaultPreferencesInURLMapping\Forced
Value name	<vault GUID with curly brackets>
Data Type	REG_SZ
Value Data	Vault GUID that is used instead of the GUID in the URL. Can be entered either with or without curly brackets.
Description	Only the vault indicated by the value of this setting is accessed when trying to resolve the URL target. If the user is not logged in the vault, the login prompt is shown normally. Example in a REG file format: <pre>[HKEY_CURRENT_USER\Software\Motive\M-Files\<version>\Client\MFStatus\VaultPreferencesInURLMapping\Forced] "{B7F9864E-ABB5-48BA-8FAA-C50DFFC27F6F}"="{3278463C-AC3E-4077-8AE0-9D563D66AD75}"</pre>

9.3.2 Preferred Vault Mappings

With this setting, the vault GUID in the URL is used when trying to resolve the URL target. If the URL cannot be resolved, the vault indicated by the value of this setting is accessed next. If the user is not logged in to that vault, the login prompt is shown normally. If the URL cannot still be resolved, all other vaults the user is logged in are accessed.

Key	HKEY_CURRENT_USER\SOFTWARE\Motive\M-Files\<version>\Client\MFStatus\VaultPreferencesInURLMapping
Value name	<vault GUID with curly brackets>
Data Type	REG_SZ
Value Data	Vault GUID that is used instead of the GUID in the URL. Can be entered either with or without curly brackets.
Description	Example in a REG file format: <pre>[HKEY_CURRENT_USER\Software\Motive\M-Files\<version>\Client\MFStatus\VaultPreferencesInURLMapping] "{B7F9864E-ABB5-48BA-8FAA-C50DFFC27F6F}"="{3278463C-AC3E-4077-8AE0-9D563D66AD75}"</pre>

10. Change History

Version	Essential changes
1.0	Initial published version.
1.1	Instructions to add vault connections added.
1.2	New parameters for the Android App added (section 5.3).
1.3	New full-text search parameters added (section 6).
1.4	Instructions to use the setkey command added. Information about mobile clients for each command added.

1.5	Updated information regarding Android support for object searches (section 6.1).
1.6	Added section 4: URL resolving logic.
1.7	Added information regarding the missing support of external IDs in mobile clients (all <i>Mobile</i> sections). Moved the <i>URL resolving logic</i> as the last section.
1.8	Changed information regarding M-Files mobile clients support for external IDs (all <i>Mobile</i> sections). Both mobile clients (iOS and Android) support external IDs from version 3.9.0 forward.
1.9	Specified what an available vault connection means.
2.0	Added a link to the M-Files Web URL specification.
2.1	Added a time-based example to section 3.2.
2.2	Small changes throughout the document.
2.3	Added a note about the new links in the M-Files December '22 version and later to the introduction.
2.4	Added a note about pre-shared keys and gRPC. Moved the content to the newer template.
2.5	Deleted some obsolete content.
2.6	Added a note to section 6. about ending the add vault link support in M-Files mobile client 23.12.0 and later. Added information about creating a vault connection with QR code.
2.7	Added the information that with M-Files January '24 Update and later, hyperlinks can no longer be created. Also moved the section 4 Changing the Appearance of Hyperlinks from M-Files user guide to this document.
2.8	Added information to section 6 that the search64 parameter should be encoded with ISO-8859-1.
2.9	Added more information about the parameter <i>template=<value></i> to section 5.
3.0	Terminology changes throughout the document.
3.1	Fixed a URL.