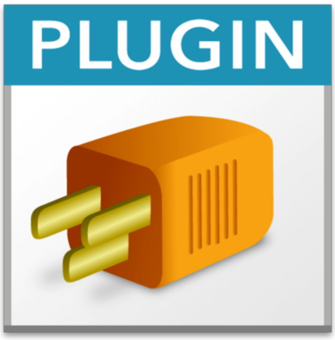
Automatically install MBS FileMaker Plugin



You can automate the installation of MBS FileMaker Plugin via the Install Plugin script step. This is great when you have hundreds of FileMaker Pro installations and you need to use the plugin locally in your scripts. But same techniques apply for server installations for people with a lot of FileMaker servers. Especially if you manage a vertical solution with hundreds of customers running each servers and clients, you like to automate this.

Things to consider
When you plan to add automatic plugin install, you may need to think about the processes and how to define the table and the scripts. For example you can have a table which contains the plugin and metadata. You may want to have the following fields:

- The plugin name, as this can be used for multiple plugins.
- The plugin version
- The platform for the given file: MacOS, Windows, Linux and 32/64bit if needed.
- An active switch, so you can enable/disable this plugin
- Minimum and maximum FileMaker version, so you can have an older plugin for an older FileMaker version and a newer version for a newer FileMaker version.
- The target groups like development, testers, users. Define whether this version should install for a given group of people.
- A force downgrade switch. This may force a downgrade to an older version if needed and you discovered a problem with a newer version of a plugin.

Once you have such a table for your script, you can fill it with plugins. You may have dozens of records there and have last final plugin version there to install for all users, but a newer beta version to install for development or testing only. Once the plugin needs to be pushed to everyone, you can change the target group from development to all users. Then in the scripts below you can make a search to find:

- The newest plugins
- which are set active to install
- for current platform
- for current FileMaker version
- for current mode development/tester/users

And then pick the newest file and check this for installation.
You may need to check platform. We have usually a check for GetAsText(Get (ApplicationArchitecture)) being "x86_64" to verify the FileMaker has 64-bit architecture. If you use Get (SystemPlatform) function in FileMaker, you can check platform. Just be aware that value 4 for Web Direct doesn't tell you whether you are on Windows, MacOS or Linux server, so check that separately, e.g. via IsMacOS, IsLinux or IsWindows.

Check Status

Now you have the record for the plugin you need to install, you check the version of the current plugin. For MBS FileMaker Plugin you can do a query GetAsText(MBS("Version")) to check whether plugin is installed. This is either "?", if the plugin is not installed or the current installed version. Alternatively to GetAsText you could check with Get(LastError) and get the error code for not having the plugin installed. For our plugin you can use GetAsNumber(MBS("VersionAutoUpdate")) to query the version number to compare. This gives an 8 digit number always with version, e.g. 10020009.

Now you compare this version against your database entry. If you have the question mark, you can install plugin. If you have a version number, it may be older than the one you like to install and then you install an update.

Please be aware that the user may have a newer plugin already installed, e.g. a beta version with a critical bug-fix for them. So don't downgrade them unless really required.

Install Plugin

Now you have a plugin in a container and you like to install it. You call the Install Plug-In File script step in a script and pass the right container. It is important to check Get (LastError) and Get (LastExternalErrorDetail) after the call and see whether the installation worked.
Here is a list from Claris with the plugin errors:

Error 1550: Cannot load the plug-in, or the plug-in is not a valid plug-in

- Unrecognized plug-in file
- Plug-in not digitally signed
- Invalid plug-in signature, OSStatus error: Certificate name:
- Invalid plug-in signature, Windows error: Certificate name:
- Plug-in with duplicate id already loaded
- Could not load plug-in code, macOS error:
- Could not load plug-in code, GetLastError:
- Could not load plug-in code, dlerror:
- Plug-in entry point not found
- Invalid plug-in options
- Missing plug-in name

Error 1551: Cannot install the plug-in; cannot delete an existing plug-in or write to the folder or disk

- Compressed contents contain incorrect plug-in extension
- Could not unpack compressed plug-in contents
- Could not create plug-in decompression folder
- Incorrect plug-in extension
- Incorrect plug-in container format

After the install, you can check version again. If the plugin is new, you can now use it right away. If the old and new plugin files have the same file path, the new plugin will not show up before you restart FileMaker process. That is more an OS limitation than a bug in FileMaker to recognize the newer plugin.

More

Please do not hesitate to contact us with questions.

- Example Utility/Install Plugin Update if needed.fmp12
- MBS FileMaker Plugin Update Check Script
- Installation of MBS FileMaker Plugin on FileMaker Server for Windows
- MBS FileMaker Plugin Installation Videos