



# Troi ClipSave Plug-in 1.1

## USER GUIDE

June 2002



### **Troi Automatisering**

Vuurlaan 18

2408 NB Alphen a/d Rijn

The Netherlands

Telephone: +31-172-426 606

Fax: +31-172-470 539

You can also visit the Troi web site at: <<http://www.troi.com/>> for additional information.

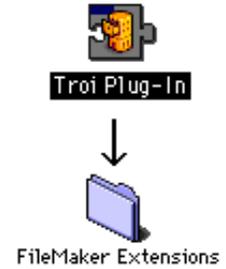
# Table of Contents

Installing plug-ins.....	1
If you have problems.....	1
What can this plug-in do?.....	2
Getting started	2
Using external functions.....	2
Where to add the External Functions?.....	3
Using the ClipSave Plugin	3
Purpose of the ClipSave Plug-in .....	3
Implementing the ClipSave Functions .....	3
Summary of functions.....	4
Function Reference .....	5
Clip-Restore	5
Clip-Save	6
Clip-Version	7

## Installing plug-ins

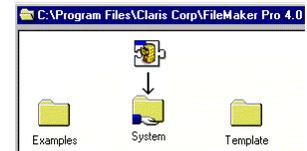
### For Mac OS:

- Quit FileMaker Pro.
- Put the file “ClipSave Plug-In” from the folder “Mac OS Plug-in” into the “FileMaker Extensions” folder in the FileMaker Pro application folder.
- If you have installed previous versions of this plug-in, you are asked: “An older item named “ClipSave Plug-In” already exists in this location. Do you want to replace it with the one you’re moving?”. Press the OK button.
- Start FileMaker Pro. The first time the Troi ClipSave Plug-in is used it will display a File box, indicating that it is loading and showing the registration status.



### For Windows:

- Quit FileMaker Pro.
- Put the file "clipsave.fmx" from the directory "Windows" into the "System" subdirectory in the FileMaker Pro application directory..
- If you have installed previous versions of this plug-in, you are asked: “This folder already contains a file called 'clipsave.fmx'. Would you like to replace the existing file with this one?”. Press the Yes button.
- Start FileMaker Pro. The Troi ClipSave Plug-in will display a dialog box, indicating that it is loading and showing the registration status.



**TIP** You can check which plug-ins you have loaded by going to the plug-in preferences: Choose **Preferences** from the **Edit** menu, and then choose **Plug-ins**.

You can now open the file "All ClipSave Examples.fp5" to see how to use the plug-in's functions. There is also a function overview in the download.

## If you have problems

This user guide tries to give you all the information necessary to use this plug-in. So if you have a problem please read this user guide first. Also you might visit our support web page:

<<http://www.troi.com/support/>>

This page contains FAQ's (Frequently Asked Questions), help on registration and much more. If that doesn't help you can get free support by email. Send your questions to [support@troi.com](mailto:support@troi.com) with a full explanation of the problem. Also give as much relevant information (version of the plug-in, which platform, version of the operating system, version of FileMaker Pro) as possible.

If you find any mistake in this manual or have a suggestion please let us know. We appreciate your feedback!

**TIP** You can get more information on returned error codes from our OSErrrs database on our web site: <<http://www.troi.com/software/oserrrs.html>>. This free FileMaker database lists all error codes for Windows and Mac OS!

# What can this plug-in do?

The Troi ClipSave Plug-in is a very powerful tool. It can save and restore the clipboard. The FileMaker Pro Scripting language has some very useful steps for developers but those steps change the contents of the clipboard. This plug-in makes it possible to use those steps in scripts, without the end-user knowing the clipboard was temporarily changed.

The ClipSave Plug-in can help with the following ScriptMaker steps:

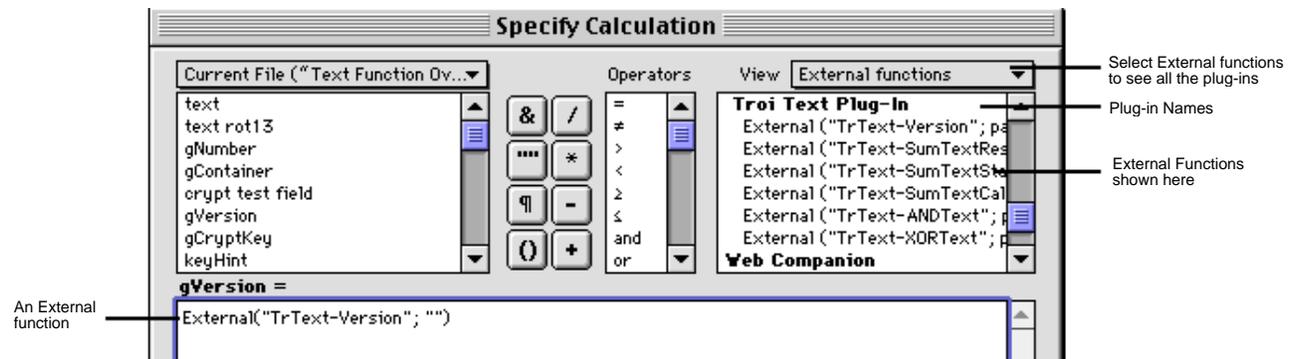
- Copy Record
- Copy All Records
- Copy (of a Page in Preview mode)
- Copy (of a field, with font and style preserved)
- external function steps from plug-ins that use the clipboard (like for example Troi Grabber Plug-in)

The ClipSave function is easily implemented by only 2 script steps.

## Getting started

### Using external functions

The Troi ClipSave Plug-in adds new functions to the standard functions that are available in FileMaker Pro. The functions added by a plug-in are called external functions. You can see those extra functions for all plug-ins at the top right of the Specify Calculation Box:



You use special syntax with external functions: External("function name",parameter) where function name is the name of an external function. The parameter is required, even if it's only "". Plug-ins don't work directly after installation. To access a plug-in function, you need to add the calls to the function in a calculation for example in a text calculation in Define Fields or in a ScriptMaker Script.

**IMPORTANT** In the United States, commas act as list separators in functions. In other countries semicolons might be used as list separators. The separator being used depends on the operating system your computer uses, as well as the separator used when the file was created. All examples show the functions with commas. For example: External("Clip-Version", "") will become External("Clip-Version"; "") in such a file.

## Where to add the External Functions?

External functions for this plug-in are intended to be used in a script step using a calculation. For most functions of this plug-in it makes no sense to add them to a define field calculation, as the functions will have side effects.

## Using the ClipSave Plug-in

### Purpose of the ClipSave Plug-in

This plug-in is mainly intended to be used by developers of FileMaker databases. The functions of this plug-in should be added to ScriptMaker scripts only. This plug-in will make it possible to use the clipboard in the scripts, without the end user knowing the clipboard was temporary changed. Some useful script steps that change the clipboard are:

- Copy Record
- Copy All Records
- Copy (of a Page in Preview mode)
- Copy (of a field, with font and style preserved)
- External function steps from Plug-ins that use the clipboard (see for example our Grabber plug-in)

More uses might be possible.

### Implementing the ClipSave functions

You can implement the ClipSave functionality with 2 functions. These are the main steps to preserve the clipboard:

- 1- Save the clipboard
- 2- Use the clipboard as you need it
- 3- Restore the clipboard

In ScriptMaker this will look as follows:

```
Set Field [gErrorCode, External("Clip-Save", "") ]  
... add here your step(s) that mess with the clipboard  
Set Field [gErrorCode, External("Clip-Restore", "") ]
```

## Example Usage

This script uses the the Copy All records step to get a list of all the IDs of the records. First the Clipboard is saved, the Copy All Records step is done, and finally the clipboard is restored to the original value.

```
Set Field [gErrorCode, External("Clip-Save", "") ]
Go to Layout ["Copy All Records"]
Copy All Records
Go to Layout [original layout]
Paste [Select, "gAllRecordIDs"]
Go to Field []
Set Field [gErrorCode, External("Clip-Restore", "") ]
```

## Plug-in Limitations

The plug-in may fail if there is not enough memory to save the clipboard. The clipboard is then not saved, but FileMaker will continue to work.

Please take a close look at the included example files, as they provide a great starting point. From there you can move on, using the 48 functions of the plug-in as building blocks. Together they are a powerful tool set for working with files on the disk.

# Summary of functions

The Troi ClipSave Plug-in adds the following functions:

<u>function name</u>	<u>short description</u>
Clip-Version	check for correct version of the plug-in
Clip-Save	save the clipboard
Clip-Restore	restore the previously saved clipboard

# Function Reference

## Clip-Restore

**Syntax**      Set Field [ result, External("Clip-Restore", "" )]

This function will restore the previous saved clipboard as the current clipboard.

### Parameters

*none, leave blank for future use.*

### Returned result

If successful it returns 0.

If unsuccessful it returns an error code starting with \$\$ and the error code. Possible codes are:

\$\$-50 = parameter error, the clipboard was not saved first.

\$\$-41 = not enough memory

Other error codes might be returned.

### Special considerations

You need to have called the function Clip-Save first to save the clipboard.

### Example usage

```
External("Clip-Restore", "")
```

# Clip-Save

**Syntax**      Set Field [ result, External("Clip-Save", "" )]

This function will save the current contents of the clipboard.

## Parameters

*none, leave blank for future use.*

## Returned result

If successful this function returns 0.

If unsuccessful it returns an error code starting with \$\$ and the error code. Possible codes are:  
\$\$-41 = not enough memory

Other error codes might be returned.

## Special considerations

Use this function to save the current contents of the clipboard. The contents of the clipboard is saved into memory of FileMaker. After you have called this function you can use the clipboard for your own purposes. Call the function Clip-Restore to restore the clipboard to its previous state.

## Example usage

Set Field [gErrorCode , External("Clip-Save", "" )]

This will save the contents of the clipboard to memory.

# Clip-Version

**Syntax**      Set Field [ result, External("Clip-Version", "") ]

Use this function to see which version of the plug-in is loaded.  
Note: This function is also used to register the plug-in.

## Parameters

*switches*      *determine the behaviour of the function*

*switches can be one of this:*

- GetVersionString      *the version string is returned (default)*
- GetVersionNumber      *Returns the version number of the plug-in*
- ShowFlashDialog      *Shows the Flash Dialog of the plug-in (returns 0)*

*If you leave the parameter empty the version string is returned.*

## Returned result

The function returns "" if this plug-in is not loaded. If the plug-in is loaded the result depends on the input parameter. It is either a:

-GetVersionString:

If you asked for the version string it will return for example "ClipSave Plug-in 1.0"

-GetVersionNumber:

If you asked for the version number it returns the version number of the plug-in x1000. For example version 1.1 will return number 1100.

-ShowFlashDialog:

This will show the flash dialog and then return the error code 0.

## Special considerations

Important: always use this function to determine if the plug-in is loaded. If the plug-in is not loaded use of external functions may result in data loss, as FileMaker will return an empty field to any external function that is not loaded.

## Example usage

External(Clip-Version, "") will for example return "ClipSave Plug-in 1.0"

## Example 2

External("Clip-Version", "-GetVersionNumber") will return 1100 for version 1.1.

External("Clip-Version", "-GetVersionNumber") will return 1001 for version 1.0b1

So for example to use a feature introduced with version 1.1 test if the result is equal or greater than 1100.