

# SuperNav 1.0

*A Navigation Solution for FileMaker*



1999- 2004

Copyright Datamatic S.C. and Model Masters  
All rights reserved 1999-2004.  
SuperNav 1.0 plug-in is  
Copyright 1999-2004 Datamatic S.C.  
SuperNav 1.0 application is  
Copyright 1999-2004 Model Masters

## Table of Contents

I. INTRODUCTION.....	3
II. INSTALLING THE PLUG-IN .....	4
III. SUPERNAV 1.0 FUNCTIONS .....	5
Snav-Version.....	6
Snav-OpenFile .....	7
Snav-CloseFile .....	8
Snav-TriggerScript .....	9
IV. SUPERNAV PLUG-IN PARAMETERS.....	10
V. PARAMETER SYNTAX .....	11
VI. SUPPORT & FINAL CONSIDERATIONS .....	12

## I. Introduction.

SuperNav 1.0 is a navigation solution for FileMaker which includes a FileMaker Solution and a FileMaker plug-in. Together they provide you with an easy and fast way to implement browser type navigation capabilities in your new or legacy FileMaker applications.

The SuperNav Demo and SuperNav Developer plug-ins are compatible with FileMaker Pro 5.0/5.5/6.0 for Macintosh and Windows. This plug-in adds four functions to the FileMaker Pro calculation engine:

**Version:** Returns the plug-in version, this provides verification of the plug-in availability and is used for plug-in validation purposes.

**OpenFile:** Opens any FileMaker Database in the local network.

**CloseFile:** Closes any opened FileMaker Database in the local network.

**ScriptTrigger:** Triggers any script in any opened Database.

**The SuperNav plug-in is a commercial shareware product, and is part of the SuperNav Plug-In Demo and the SuperNav Solution. It is not sold separately.**

Just as with any other shareware software, the SuperNav plug-in can be fully tested before you buy it. If you find it useful and decide to obtain the Developer version, you must pay for a registered Developer version.

The SuperNav Developer version is only sold as an Unlimited Distribution license for use in bound (Mac only) and unbound (Mac/Win) FileMaker solutions for an unlimited number of users and can be fully distributed with your own solution(s).

The SuperNav Solution and SuperNav plug-in is a joint venture of Datamatic Software ([datamaticsoftware.com](http://datamaticsoftware.com)) and Model Masters, ([fmplsolutions.com](http://fmplsolutions.com)).

### Registering SuperNav 1.0.

Paying the registration is securely and easily done through the secure payment processing service, Kagi Online at: <http://order.kagi.com/?36P> or through our sites <http://www.datamaticsoftware.com/> or <http://www.fmplsolutions.com/>

## II. INSTALLING THE PLUG-IN.

Quit FileMaker Pro.

**Macintosh:** Once the file "SuperNav Mac Demo or SuperNav Developer plug-in" is on your hard disk move it into the "FileMaker Extensions" folder, located inside the FileMaker Pro 5.0/5.5/6.0 folder.

**Windows:** Once the file "SuperNav Win 1.0.fmx" is on your hard disk put it into the "System" folder, located inside the FileMaker Pro 5.0/5.5/6.0 folder.

If a previous version of this file is already installed, a dialog asking if you want to replace it with the one you are moving in will appear. Press the **OK** button.

**IMPORTANT: If a SuperNav Plug-In DEMO version is already installed, please make sure to remove it from the folder because files can have different names but use the same signature within FileMaker which can create a conflict.**

Start FileMaker Pro. The SuperNav Demo or Developer plug-in is now ready to use.

If you are testing the SuperNav Demo plug-in evaluation version, a splash screen will appear at the start and the SuperNav plug-in will remain operational for 30 days.

Open the file "SuperNav Plug-In Demo.fp5" to test the plug-in functions.

Before testing the plug-in, make sure that the plug-in used in FileMaker is enabled in the preferences dialog box with a mark in the boxes in front of their names. You can prompt this dialog in the Edit/ Preferences/ Applications/ Plug-ins in FileMaker Pro 5.0/5.5/6.0

The SuperNav SNav 1.0 Demo (Mac/Win) and the SNav 1.0 Developer (Mac/Win) plug-ins run under FileMaker 5.0/5.5/6.0 for Windows and Macintosh (Classic & OSX) platforms.

### III. SUPERNAV PLUG-IN FUNCTIONS.

New functions Are Added To the FileMaker calculator engine Using FileMaker Plug-Ins. Therefore, to use them, you should use Set Field script steps via the "External functions" list to use a plug-in.

SuperNav Demo plug-in and the SuperNav Developer plug-in add four functions via the FileMaker calculation engine:

- **Version** – Returns the plug-in version number.
- **OpenFile** – Opens any FileMaker database in the local network.
- **CloseFile** – Closes any opened FileMaker database.
- **TriggerScript** – Triggers any script on any opened database.

***IMPORTANT: List separators are different in each country. Commas are used in the US and semicolons are used in some other countries. Please use the appropriate list separator according to your FileMaker version.***

#### **Example:**

External("Snav-Version", *parameter*) (US)

External("Snav-Version"; *parameter*) (Some other countries)

## 1. Snav-Version

The plug-in version number is returned.

### IMPORTANT:

***Always use this function in your scripts to verify that the plug-in is loaded and properly installed before trying to use the plug-in.***

***It will provide a safe way to avoid any problem in your application if someone removes the plug-in from the Win (System) or Mac (Extensions) folder.***

FileMaker script example:

```
If[ External("Snav-Version", 0) = "SuperNav 1.0"]
  SetField[myField; External("Snav-Version", 0)]
Else
  ShowMessage["SuperNav plug-in is not installed"]
EndIf
```

This means:

```
If[SuperNav exists and is version 1.0]
  Use other SuperNav functions
Else
  Show a message or halt your program
```

### FILE Pathname Note:

The Main Menu file (part of the SuperNav application) has the following field calculations installed:

- a) **FilePathAuto**
- b) **Path**
- c) **Path2Root**
- d) **PathWinMac**
- e) **DB\_ID**
- f) **HostName**
- g) **DB\_PW**
- h) **CurrentFMPVersion**
- i) **CurrentPlatform**

The purpose of these calculation fields is to automatically provide the required file paths when using the SuperNav plug-in.

## j) 2. Snav-OpenFile

This function opens any FileMaker file in the current workstation.

One purpose of the SuperNav plug-in is to trigger scripts, but to do the target file must be open.

To open a FileMaker file, the full pathname of the file and the password (if one is defined) must be provided as one of the parameters required in order to open the file. The password used can be any defined password. Note the file will open with the permissions attached to that password.

### Example:

```
External ("Snav-OpenFile", parameter)
```

The OpenFile function requires several sub-parameters following the function name, these sub-parameters are represented in the example above under the variable "parameter", and **must** be configured as follows:

```
"|File=Macintosh HD:TestFile.fp5|Password=5215cd"
```

Please, follow the syntax rules defined in the "Parameters" section below to avoid any mistake in a sub-parameter definition.

To use the OpenFile function, provide these parameters:

```
|File=<full pathname of the file you want to open>  
|Password=<the password of the file> (optional)
```

### Examples:

Macintosh:

```
External("Snav-OpenFile",  
"| File= MacintoshHD: MyFolder: TestFile.fp5| Password= 5215cd")
```

Windows:

```
External("Snav-OpenFile",  
"| File= C:\MyFolder\ TestFile.fp5|Password= 5215cd")
```

### 3. Snav-CloseFile

This function closes any open FileMaker file.

In this case, as the file is already open, only the database name is needed as a parameter (not the full path).

**Example:**

```
External("Snav-CloseFile, "| Database= MyOpenDatabase.fp5")
```

Note that the parameters for any of the four functions **must** to be passed as text, enclosed within double quotes.

#### 4. Snav-ScriptTrigger

This function triggers any script defined within any open FileMaker file.

In this case, as the file is already open, only the database name (not the full path) and the script name are required as sub-parameters.

##### Example:

```
External("Snav-ScriptTrigger, parameter)
```

Within the FileMaker calculation engine, this functionality is used to trigger FileMaker scripts according to the results obtained from any valid ScriptMaker calculation.

This means that scripts can be used for validation or to detect any change in the user interface, like typing a new value in a field, changing the selection of a menu or creating a new record, etc.

This function is also very useful to trigger external scripts defined in other files.

This function requires two sub-parameters, the name of the file including the file extension and the name of the script to be called.

##### Example:

```
External("Snav-ScriptTrigger,  
"| Database= MyDatabase.fp5| Script= MyScript")
```

#### IV. SUPERNVAV PLUG-IN PARAMETERS

The SuperNav plug-in provides four different sub-parameters that can be used as follows:

- **File.**

This is the full pathname of the file and is required to open the file.

On Macintosh systems it takes the form of:

```
"|File=Macintosh HD:MyApplicationFolder:TestFile.fp5"
```

On Windows, it is in the form of a Windows pathname:

```
"|File=C:\MyApplicationFolder\TestFile.fp5"
```

- **Database.**

It is the name of an open FileMaker database file. It does not include the full path.

```
"|Database=TestFile.fp5"
```

- **Script.**

It is the name of the script to trigger. It should be exactly **to** the name of the script as defined in ScriptMaker.

```
"|Script=MyScript"
```

- **Password.**

This parameter is optional and it is only used if the file you are trying to open has a password defined.

```
"|Password=5215cd"
```

**NOTE:** The entire password sentence (KEYWORD **and** password) must be enclosed within double quotes, independent of the number of sub-parameters passed to the SuperNav plug-in.

**Example - Number Validation**

```
External("Snav- ScriptTrigger,
  \"|Database= MyDatabase.fp5| Script= SCRIPT: Number Validation")
```

**Script: Number Validation**

```
# The gnNumberfield has been created as a number field
# and is set to validate when the field content changes.
# Then it is changed to a global number field
```

```
If [ gnNumber = 1000 ]
Show Message [ Buttons: "OK", "", ""; Data: "The number is 1000" ]
Else
  If [ gnNumber < 1000 ]
    Show Message [ Buttons: "OK", "", ""; Data: "The number is smaller than 1000" ]
    Else
      If [ gnNumber > 1000 ]
        Show Message [ Buttons: "OK", "", ""; Data: "The number is larger than
          1000" ]
      End If
    End If
  End If
End If
```

**Script: BTN: Set Text Field**

```
# This Script Causes The Value List Attached To gtField
# To Be Displayed. The User Then Selects An Item From This Value List
# This action Enables the gtText Field Validation By Inserting A Script Name
# In The Field: gtDispatchTextValidation". This Script Name is
# Used When The A Field Validates (After The Content of the Field: gtText
# Changes.)
```

```
Set Field [ gtText, "" ]
Clear [ gtText ]
Set Field [ gtDispatchTextValidation, " SCRIPT: List Selection" ]
Exit Script
```

**Script: BTN: List Selection**

```
# " This Script Disables The gtField Validation By Removing"
# " The Script Name From The Field: gtDispatchTextValidation"
# " "
```

```
Set Field [ gtDispatchTextValidation, "" ]
Go to Field [  ]
```

```
Pause/Resume Script [ "0:00:00" ]
Show Custom Dialog [ Title: "You Selected . . ."; Message: gtText;
  Buttons: "OK", "", "" ]
```

## V. PARAMETER SYNTAX.

The following syntax rules are used to form the **Parameter** required by the SuperNav plug-in.

- All parameters always start with a pipe character (|) (As the first part of a **sub-parameter**.)
- The sub-parameter name should immediately follow the pipe with no space between the equals sign (=) and the **sub-parameter** information.
- **Sub-parameter** names always start with upper case letters.
- An equals sign (=) follows the name of each sub-parameter with no space between the equals sign (=) and the **sub-parameter** information.
- **Sub-parameters** can be in any order.
- The information for a particular sub-parameter follows the equal sign "=" with no space between the equal sign (=) and the **sub-parameter** information.

## VI. SUPPORT AND FINAL CONSIDERATIONS

There is a FAQ section, on our sites with several scripts, to help solve some common issues for our customer. Whatsoever, we will be pleased to assist you in getting started we may be contacted at:

[support@datamaticsoftware.com](mailto:support@datamaticsoftware.com) or [support@fmpsolutions.com](mailto:support@fmpsolutions.com).

Please read the manual and the user license agreement carefully, before using the plug-in and the SuperNav Plug-In Demo and/or the SuperNav Studio application. By using it you fully agree with the terms and conditions of such agreement.

This plug-in and the application have been thoroughly tested by Datamatic Software and Model Masters respectively. Datamatic Software nor Model Masters are not responsible for any damage resulting from the use or testing of this software. It should be tested on a test database as some conditions in your system (unknown to us) could interfere in the proper functioning of the plug-in. Use it at your own risk.