BaseElements Instruction Manual

1.0 About BaseElements

1.1 What is BaseElements

BaseElements is a FileMaker Developer Tool for analysing FileMaker Pro solutions. By importing the DDR, it gives you a complete cross reference of every element in your solution.

BaseElements is designed to be used by anyone who is using FileMaker Pro to develop systems either for themselves or for other clients. BaseElements gives you the ability to see all of the elements in your solution from the outside, and will tell you what is being used and where and what is broken or not being used at all.

BaseElements can make your development easier by identifying problems before they cause issues by alerting you to broken elements in your solution.

It can speed up your solution by allowing you to remove elements that are no longer being used, saving you space but also allowing you to avoid maintaining sections of the solution that don't need to be maintained. Plus you can used BaseElements to look at where your solution is being slow and why by tracing fields and scripts through their dependencies and optimising them for the best outcomes.

BaseElements is an essential tool for all developers from small to large.

1.2 Requirements

BaseElements is always built using the latest version of FileMaker Pro. So the requirements for using BaseElements are always based on the requirements of the current version of FileMaker Pro.

However, firstly you will need a copy of FileMaker Pro Advanced. You need this to generate the Database Design Report. At a minimum this will need to be FileMaker Pro Advanced 7 or later. BaseElements won't import a DDR from version 6 or earlier. You can use whatever version of the FMPA that you're using to develop the files in, it doesn't need to be the latest version of FMPA, as long as it is late enough to cover all of the features you've used in your solution. (In other words don't develop your solution using FMP 10 with v10 only features but run the DDR with 7, your v10 features won't be included in the report.)

Once you've generated the DDR report you no longer need FMPA to run BaseElements. You can choose between the runtime application which is a standalone application and fmp12 or fmp7 files which require a copy of FMP or FMPA to run.

Runtime Requirements

The BaseElements runtime will always be generated with the current release of FMPA. So the requirements for the runtime are the same as for FMP / FMPA. You can find the current FMP requirements on the FileMaker website [55].

fmp12 Requirements

BaseElements 4 uses features available only in version 12 of FileMaker Pro, so you need to open the fmp12 files in FileMaker Pro 12 or later.

If you are currently using an earlier version of FileMaker Pro, you can use the runtime instead of the fmp12 files as that will allow you to take advantage of the features in version 12 without having to upgrade your own copies just yet.

BaseElements 3 is also still available, and runs in FileMaker Pro 11.

1.3 Updates and Notifications

If you're wanting to keep up to date with new versions of BaseElements, there is a mailing list that you can subscribe to receive notifications. There is a subscription form on the BaseElements website, you just need to fill in your name and email address.

You will only ever receive notifications about new features or updates to BaseElements, and your email address will never be sold or used for spam.

1.4 BaseElements and FileMaker Pro 12

The short version

BaseElements 3.5 adds support for the DDR from FMPA 12 but still runs in FMP 11 itself. It will import the DDR from any version 7 through 12.

BaseElements 4.0 runs in FMP 12 and will also import from any DDR from v7 to v12.

BaseElements 3.5 is a free update from 3.0.

BaseElements 4.0 is a paid upgrade from any previous version. As well as running in FileMaker Pro 12 it includes new features and functionality improvements.

The long version

When deciding which version of BaseElements is best for a project, there are two considerations :

1. Which version of FMPA are you using to generate the DDR?
2. Which version of FMP/A are you going to run BaseElements in?

Which version of FMPA are you using to generate the DDR?

BaseElements 3.0.0 through to 3.0.11 will import successfully any DDR generated with FMPA 7 through 11. If you also want compatibility with FMPA 12 you will need to update to BaseElements 3.5.0. This is a free update for anyone who already has a licence to 3.x.

BaseElements 3.5.0 includes new functionality to import all aspects of the DDR from v7 through to v12. The additional changes in 3.5.0 mean that all references are properly reported, although there is no additional interface for some features such as File level Script Triggers or the new container field functionality.

BaseElements 4.x and later will also import any DDR generated from FMPA 7 through 12 and includes UI to show and reference all of the new items.

Which version of FMP/A are you going to run BaseElements in?

BaseElements 3.x runs in FMP 11 and will not run in earlier versions of FileMaker. You can convert your 3.x copy to FMP 12 but we don't recommend it.

BaseElements 4.0 runs in FMP 12 and cannot be opened or converted to work in FMP 11.

All versions of BaseElements are available as runtimes, so you can use them without having a copy of that particular version of FileMaker Pro.

Purchasing

Any purchases of BaseElements 3.x made after the 1st of Feb 2012 will receive licences for both BaseElements 3 and 4. Once version 4 is final, purchases will continue to include a licence to version 3 so that you can use BaseElements in both versions of FileMaker Pro.

2.0 Setup

BaseElements fmp12 files

BaseElements for Mac comes in a compressed zip file. You need to copy the 3 BaseElements fmp12 files, or the BaseElements runtime folder to a location on your hard drive.

Multiple Copies

BaseElements will let you import as many different solutions into a single copy as you prefer. However some developers like to keep multiple copies of BaseElements, one for each project or client, and this is perfectly fine within the licence agreement. In this way you can keep a single archive of each project's development.

You can also use FileMaker server, although you can only have one copy on each FileMaker Server at any one time.

2.1 Plugins

BaseElements 4 uses a provided plugin, called "BaseElements.fmplugin" on the Mac and "BaseElements.fmx" on Windows.

Plugins in the Runtime.

There is no extra setup for the plugins in the Runtime on either Mac or Windows. The plugin is already included in the "Extensions" folder, and so will automatically load as required. The only issue may be if you've got an older version of the same plugin in the user's Extensions folder (see below). FileMaker will also try to load plugins from the user folder, even when using a Runtime. If you get warnings about the plugin when running the Runtime, check the user folder (below) for old versions first.

Plugins for the fmp12 versions.

The correct plugin for each platform is included in the BaseElements files and will be loaded automatically when required. If the option to automatically install plugins is switched off from the default, then BaseElements will ask if you want to manually save copies of the plugin. The plugin file needs to be put into the correct folder. In versions of FileMaker after version 9, you should copy the BaseElements plugin from the download folder to :

`C:\Documents and Settings\user_name\Local Settings\Application Data\FileMaker\Extensions\`
Windows Vista / 7 **`C:\users\user_name\AppData\Local\FileMaker\Extensions\`**
Mac OS X **`Macintosh HD/Users/user_name/Library/Application Support/FileMaker/Extensions/`**

You can check if the plugin is installed by launching the BaseElements_4.fmp12 file. You will get a warning or error message if you have not installed the plugin or if you have an older copy of the plugin installed. The latest version of the BaseElements plugin can be found at :

http://www.goya.com.au/baseelements/plugin [56]

The BaseElements plugin is open source and freely available for other developers to use in any project they see fit. Source code is also available. See the website url above for more details.

2.2 Sharing with FileMaker Server

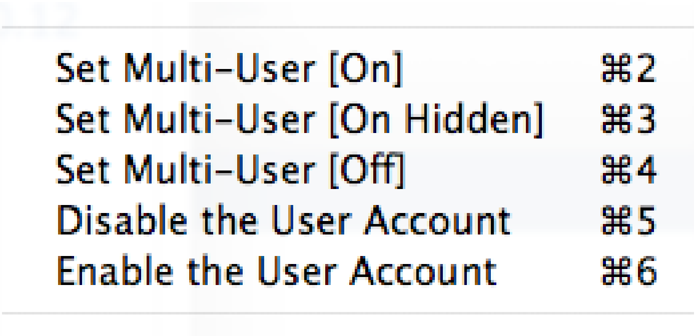
Both the single user and Site Licence versions of BaseElements can be used on FileMaker Server. The single user licence though, will only allow one user at a time to access the database.

When using FileMaker Server you can also use an existing External Authentication (EA) account to limit access to BaseElements. There are two EA accounts in the main BaseElements files :

FMP_BE_Edit
FMP_BE_Read

that have Edit and Read only access enabled respectively. If you need a specific EA group added to the files, we can do that for you. Contact us for details.

However, the regular single user account also has sharing enabled, so without alteration, all users will default to this account and be granted access. In order to remove that access, you need to first open the files on FileMaker Server using one of the EA accounts. Then go to the BaseElements_4 file. In the scripts menu are the following scripts :


Set Multi-User [On] ⌘2
Set Multi-User [On Hidden] ⌘3
Set Multi-User [Off] ⌘4
Disable the User Account ⌘5
Enable the User Account ⌘6

Set Multi-User State

These scripts allow you to turn on or off the multi-user state of BaseElements. The hidden option allows you to have the main BaseElements file not showing in the Open Remote dialog.

The option to turn it off avoids the annoying dialog about hosting not being enabled when not used on FileMaker Server.

Disable / Enable the User Account

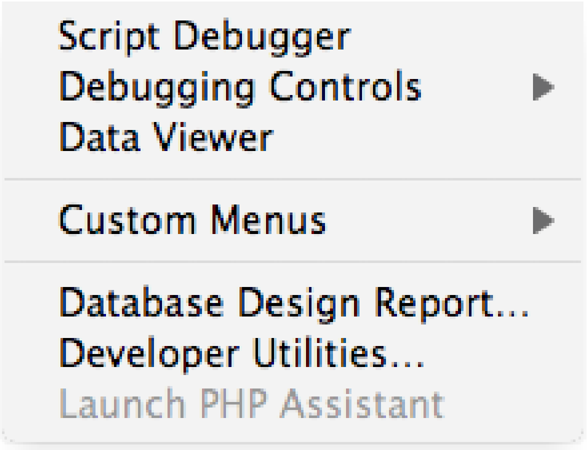
These will allow you to turn off the normal access account and so BaseElements will be restricted to only the EA groups and accounts you've set.

3.0 The Database Design Report

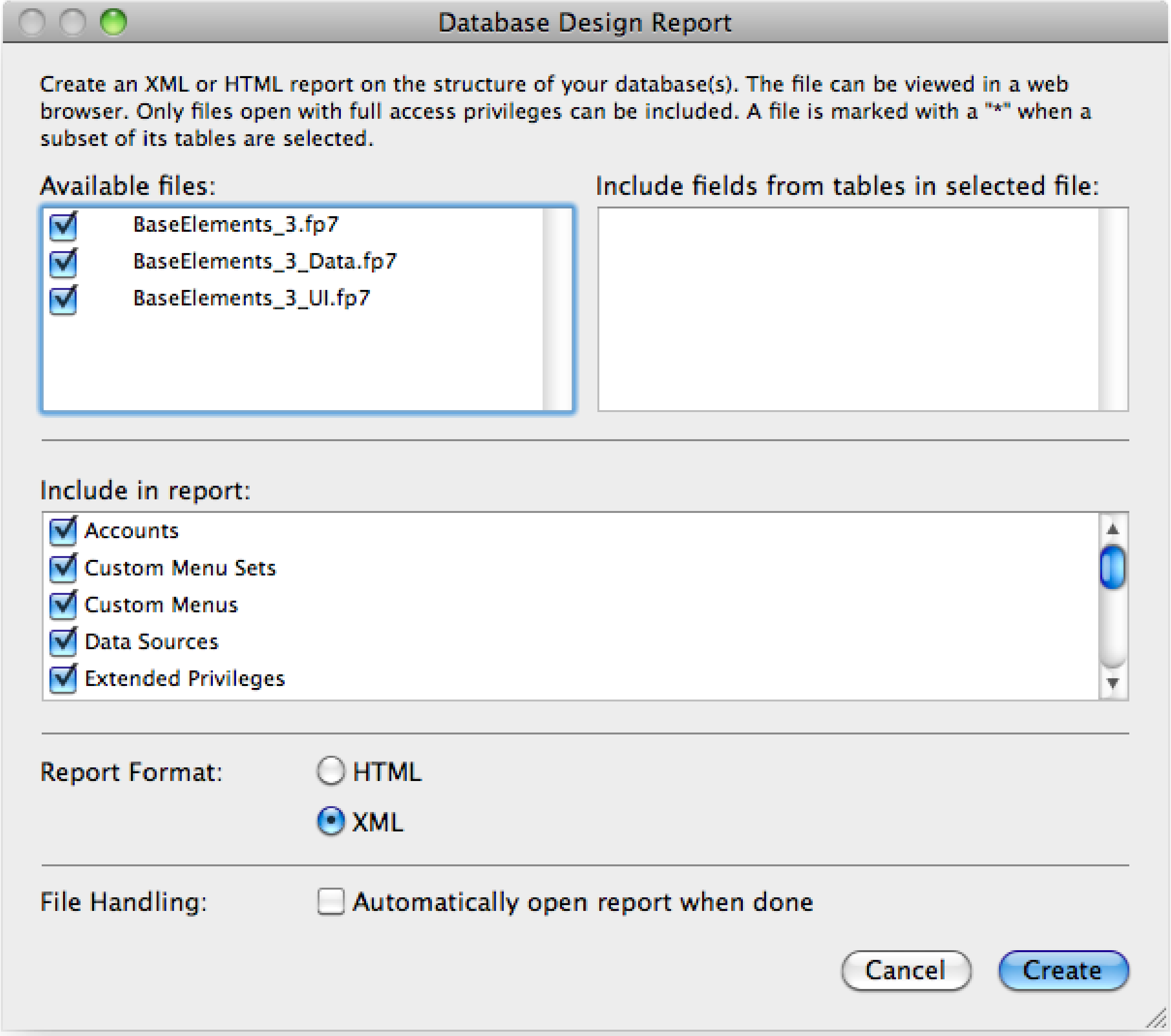
BaseElements requires the Database Design Report (DDR) to function. The DDR is generated by FileMaker Pro Advanced and is a complete documentation of all of the elements in your solution. We use this report and tranform it into useful data which we then import into BaseElements. Without this report, and thus also without a copy of FMPA to generate it, you can't run BaseElements.

3.1 Generating the DDR

You need a copy of FileMaker Pro Advanced to generate the Database Design Report. Open a copy of all of your solution files in FMPA and then go to the Tools menu, and to "Database Design Report...".



In the dialog, make sure all of your solution files are checked (default) and all of the options are checked (default). Change the Report Format to "XML", and you can also uncheck the "File Handling" checkbox. Click Create and choose a location to save your XML files to.



Create an XML or HTML report on the structure of your database(s). The file can be viewed in a web browser. Only files open with full access privileges can be included. A file is marked with a "*" when a subset of its tables are selected.

Available files:
- ☑ BaseElements_3.fp7
- ☑ BaseElements_3_Data.fp7
- ☑ BaseElements_3_UI.fp7

Include fields from tables in selected file:

Include in report:
- ☑ Accounts
- ☑ Custom Menu Sets
- ☑ Custom Menus
- ☑ Data Sources
- ☑ Extended Privileges

Report Format: ○ HTML ● XML

File Handling: ☐ Automatically open report when done

The name of the report isn't critical, although it's simpler to just leave it as the default (Summary.xml on English platforms).

The location of the XML files also isn't critical, although BaseElements does write to the folder containing the XML files, so you do need read and write access to that folder.

3.2 Comparison of BE with the HTML DDR

FileMaker Pro Advanced includes an HTML DDR which outputs a useful list of lots of detail about a solution. You do need to be aware though that this report is not complete and you probably shouldn't use it for serious analysis of a solution, especially in a multiple file solution. Most of the detail about items is included, but links between multiple files are not. So scripts that call scripts in other files are not linked in both ways, only as an outgoing link. So you can be looking at a script and and not know that it is called by other scripts.

This limitation is in all areas though, not just scripts. So be careful if you use the HTML DDR in a multiple file solution.

Obviously there are some big differences between a HTML report and the FileMaker file that BaseElements uses. Firstly there isn't the multiple file limitation that the HTML DDR has, you can see links between every element in your solution across as many files as there are.

The big benefit of BaseElements is in it's search-ability and link-ability. You can search within every calculation for a certain text, then link from the found set to every object that uses that text, and from there to a list of layouts. Everything in BaseElements has those links available. And you can open as many windows with list, detail or table views as you want. So you have a lot of flexility in your use of BaseElements so you can get the best out of your solution.

Features

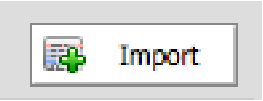
BaseElements includes a lot of additional features that the HTML DDR doesn't analyse and won't tell you about, plus extra options for reporting. Some examples :

Plugins.
Variables.
Errors.
Unreferenced elements.
Warnings.
Comparison Report.
Consolidation Report.
Error, Unreferenced and Warning Report.
FileMaker Go Compatibility Report.

plus lots more. BaseElements is being added to and enhanced with each version, so there will always be lots of reasons to use it that help you develop your best solutions.

3.3 Importing into BaseElements

Once you've generated your DDR, you then need to import it into BaseElements. Open BaseElements and go to the Home screen. On the top left side of the Home layout is the import button :



Click this button, and then select the "Summary.xml" file you generated when you created the DDR. This file will be called something else if you're using a different language version of FileMaker.

BaseElements will close the UI files and bring the Data file to the front and show you the progress dialog for the import. Large DDRs (could be anything from 500MB and up) will take time to import. Because the XSLT transform happens in the plugin if you have a large DDR it can at times appear that FileMaker is locked up. Leave it running, and any issues that happen will be shown via a dialog.

Once the import is complete, it will return to the UI file, showing the details of your just imported solution.

3.4 XSL Import Types

In BaseElements 3.0.5 and later there is an extra radio button choice in the Setup for "XSL Type". This is an option to select from two different available XSL import engines. The difference between the options are :

Xalan

Xalan is the name for the internal FileMaker XSL engine that is built into the Import Script Step. It is generally going to be a slower import than LibXSL but will also import very large solutions that will cause LibXSL to run out of memory. However, we have also seen situations where Xalan would cause a FileMaker crash where LibXSL will manage the same import just fine.

LibXSL

LibXSL is the import engine included in the BaseElements plugin. It will generally be faster than Xalan because it's processing the XML into CSV files which are quicker to import than XML. In some smaller solutions you may not notice the difference at all though. We have found that due to limitations in the plugin environment that we can run out of memory when trying to import some very large solutions using LibXSL. The plugin environment inside FileMaker is limited to either 2 or 4 GB or Ram (depending on OS), regardless of how much memory is inside the machine. If you are getting out of memory errors, switch back to Xalan for that solution.

Auto

The Auto setting is the default, and this option will try to use the faster LibXSL engine inside the plugin. If it encounters a failure at any point, it will retry with Xalan and after 2 failures will do all subsequent import steps with Xalan.

What to choose

The default option is Auto option will be fine for most people, as that will work in most situations and is the faster option. LibXSL though doesn't have access to as much memory as Xalan, so if you get out of memory errors or other failures when trying to import, please re-try the import with the setting set to Xalan before reporting an issue with the import.

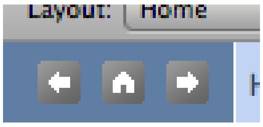
4.0 BaseElements Basics

BaseElements is designed to be familiar to anyone who is a FileMaker Developer. We've stuck with well known FileMaker conventions where possible and tried to use as many native user interface controls and methods.

A couple of basic tips : First it's just FileMaker, so you can create new windows and switch between form, list and table views at will. A sample use case might be to go to a list of related items, which automatically switches to list view. You can then open a new window, and keep that found set for later. Switch back your old window and use the back and forward buttons to go back to old layouts.

The find interface isn't modified at all, you can do regular finds as well as constrain or extend, and because you have field access, you can right click for "Find Matching Records" as well as context dependent sorting. It's all just standard FileMaker.

Back and forward.



The buttons are the Back, Home, and Forward buttons respectively. The back and forward is to track your history in BaseElements and to be able to back track to a previous layout and record. The back and forward will remember the layout, tabs and record that you were on previously. It doesn't remember your found set or sort order though, so be aware if you go back to a record that isn't in the found set you will alter the found set to include it. If you've gone backwards, you can then also track back forwards. Once you link somewhere though, the forward is cleared out, in much the same way that a web browser would.

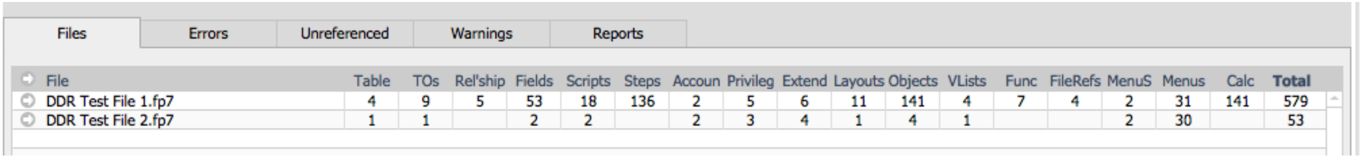
This is a scripted back process, so it records only when you navigate via a button inside the UI. If you manually choose a layout from the layouts menu it isn't included in the back history.

The home button will also take you back towards the home screen. If you're on any of the data layouts, you go back to the Analysis first, and then another click takes you to the Home screen.

New Windows.

The New Window function is such a useful way of keeping a found set for later, it's been mapped to the Command-N keyboard shortcut. There is no need to create new records in BaseElements as they're all created via import, so this is a convenient shortcut.

Analysis Layout.



File	Table	TOs	Rel'ship	Fields	Scripts	Steps	Accoun	Privileg	Extend	Layouts	Objects	VLists	Func	FileRefs	MenuS	Menus	Calc	**Total**
DDR Test File 1.fp7	4	9	5	53	18	136	2	5	6	11	141	4	7	4	2	31	141	579
DDR Test File 2.fp7	1	1		2	2		2	3	4	1	4	1			2	30		53

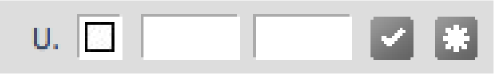
This is the main starting point for any solution. This portal shows each of the files in your solution and counts for all of the items in each file. Each of those numbers is also a link to the respective items. So, in the screenshot above you can click on the number 18 below scripts to go directly to a list of the 18 scripts in that file. The same applies to each of the other counts, as well as the same details on the Errors, Unreferenced and Warnings tab.

This is the best starting point for looking at any solution, you can jump to a list of the items you want to look at with one click.

Another feature of this portal, is the totals at the bottom of the portal. These are totals for all of the files in these solutions. You can also click on any one of these totals and view a list of all of the items in this solution.

4.01 Conventions and Abbreviations

There are very few abbreviations used in BaseElements, except for some important ones to remember. In the top right of any Form view layout are the following three fields and two buttons :



These three fields are the **Unreferenced**, **Warning Count** and **Error Count** fields respectively. The two buttons are the Mark as Fixed [43] and Highlight [44] buttons which have their own sections in the manual.

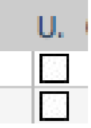
Unreferenced

The Unreferenced field is a checkbox field, and when the box is checked, the element being referred to is considered Unreferenced. This is abbreviated to "U:" to save space and be consistent with the "U:" header used at the top of portals or list views.

There are three possible states for the Unreferenced checkbox :

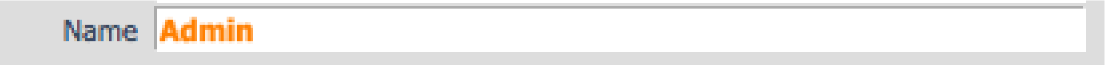
- Unchecked - The element is in use by other elements in the solution.
- Checked - The element is completely unreferenced.
- Dash - The item is unreferenced, but is either showing in a menu (for scripts or layouts) or your solution is using Indirection [28] and may be referring to this element by name.

Unreferenced Header at the top of a Portal :



Warning Count

The Warning Count field operates in much the same way as Errors, but refers to the count of warnings on this object. The Warning Count field is orange, and when the element has a warning associated with it, the name of the element will also highlight in orange.

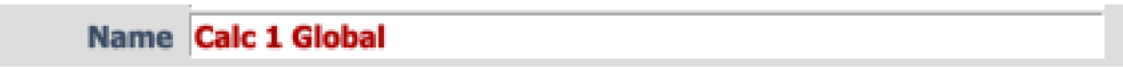


Name Admin

Error Count

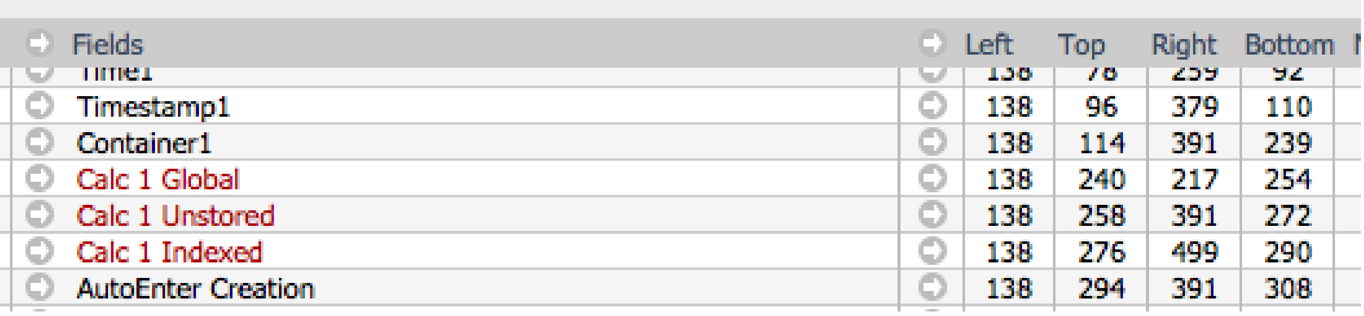
The Error Count field is a number, and when zero the field is empty. You can always tell when you're looking at the error field as it's highlighted in red and bold. The only thing that is ever highlighted red is an error, so you know exactly where to look for errors.

A red error name :



Name Calc 1 Global

A red error name when shown in a list of fields on a layout :



Fields	Left	Top	Right	Bottom
Time1	138	78	259	92
Timestamp1	138	96	379	110
Container1	138	114	391	239
Calc 1 Global	138	240	217	254
Calc 1 Unstored	138	258	391	272
Calc 1 Indexed	138	276	499	290
AutoEnter Creation	138	294	391	308

When an element has both errors and warnings, then the error highlight will override the warning highlight and the name will appear in red throughout the entire solution.

4.02 Navigating the UI

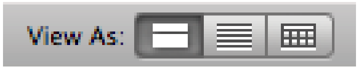
There are five features in BaseElements that make up the central core of the BaseElements user interface.

- Form, List and Table Views

- Tabs
- Portals
- Windows

Form, List and Table Views

BaseElements 3 uses the built in view changing UI that FileMaker provides. So the "View As" UI item in the toolbar :



which allows you to switch view for the current layout, is actually being overridden by a Custom Menu item, and within BaseElements it will switch layouts instead. You can then easily switch between our Form, List or Table views with one click. You can also use the Menu Items in the View menu to achieve the same effect.

When in List view, there is also a "View" button on the left hand side of the List that will take you from the list view to the Form view of any one specific element.

Tabs

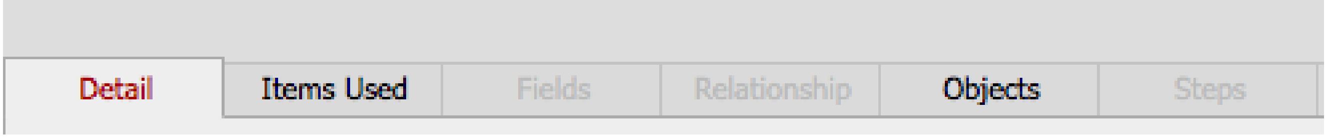
Whenever there is more information about an element than will fit on a single layout, we used tabs to break the item into sections. The tabs for every type of element are different, as each one has different attributes and features, but there are some core consistencies in how they are presented.

Initially, there will be some "Details" tabs that show information about the element itself. There may only be one of these, or in the case of elements like "Privilege Sets", there are many to detail all of the settings for each Privilege Set.

For any element that uses other elements, like Fields used in a Calculation or Value Lists used by an Object, there will be an "Items Used" tab. These contain up to five other portals : Fields and TOs, ValueLists, CustomFunctions, Variables, and Plugins. Which ones are displayed depend on which ones are used by the element.

Tabs that follow the "Items Used" tab are places where the current element is being used. Again there may be one or more of these, and the "Steps" tab will have sub tabs, as there are four different types of Steps used in other places (in Scripts, as Layout Object buttons, Menu Item steps and Script Triggers).

There is also conditional formatting for the tabs to make navigation faster and easier. For example in the following screenshot from the Fields layout :

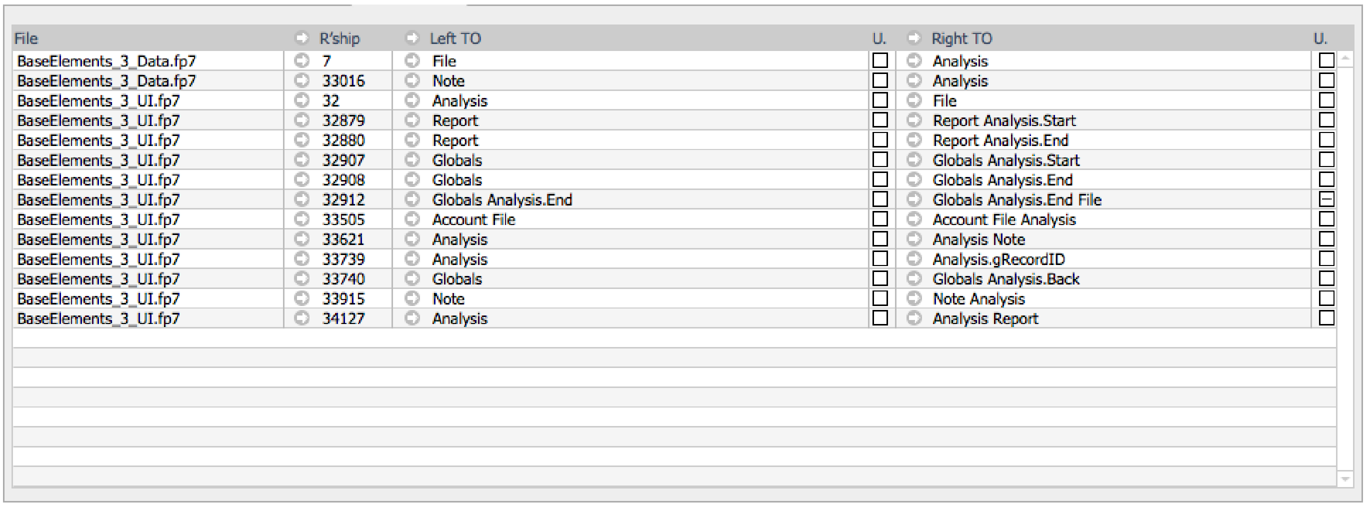


The red conditional formatting for the first "Detail" tab tells you that there is an error on this item. The specific error would be highlighted further down on the Detail tab. The "Items Used" tab has black text, so it's an active tab. In this case it means that this field uses other elements. On that tab there will be data in the portals for this field.

The remaining tabs are the "Items that use this Field" tabs. In this case you can see at a glance from the conditional formatting that the only place that this Field is used is in Objects. So this field is either on a layout, or used by a layout object somewhere. It's not used by other fields, in Relationships or in Steps.

Portals

Portals are the core user interface element in BaseElements. Portals are used to show the complete lists of every element related to any other element. So, for example, when you're looking at a Field (in this case one from BaseElements itself), and looking at the Relationships tab it shows you every Relationship that this field is used in :



File	R'ship	Left TO	U.	Right TO	U.
BaseElements_3_Data.fp7	7	File	☐	Analysis	☐
BaseElements_3_Data.fp7	33016	Note	☐	Analysis	☐
BaseElements_3_UI.fp7	32	Analysis	☐	File	☐
BaseElements_3_UI.fp7	32879	Report	☐	Report Analysis.Start	☐
BaseElements_3_UI.fp7	32880	Report	☐	Report Analysis.End	☐
BaseElements_3_UI.fp7	32907	Globals	☐	Globals Analysis.Start	☐
BaseElements_3_UI.fp7	32908	Globals	☐	Globals Analysis.End	☐
BaseElements_3_UI.fp7	32912	Globals Analysis.End	☐	Globals Analysis.End File	⊟
BaseElements_3_UI.fp7	33505	Account File	☐	Account File Analysis	☐
BaseElements_3_UI.fp7	33621	Analysis	☐	Analysis Note	☐
BaseElements_3_UI.fp7	33739	Analysis	☐	Analysis.gRecordID	☐
BaseElements_3_UI.fp7	33740	Globals	☐	Globals Analysis.Back	☐
BaseElements_3_UI.fp7	33915	Note	☐	Note Analysis	☐
BaseElements_3_UI.fp7	34127	Analysis	☐	Analysis Report	☐

All of the fields are accessible, so you can click into them to copy data or perform finds. In this example, the portal shows the File, Relationship, and Left and Right TO that the Relationship uses. You can link to any one of the relationships by clicking the arrow next to the Relationship id. Plus you can jump directly to the TO used on either side of the relationship by clicking on the arrow next to the TO name you want to view.

And you can go to a list of all of the relationships by clicking the arrow in the portal header. This takes you to a list view, with the found set being every Relationship (or TO if you use that arrow button) that is in the portal. This is the core advantage of using the built in FileMaker user interface items for displaying the data. You can, for example go to a layout, and then view a list of every field on that layout with one click.

Windows

One of the most useful things to remember in BaseElements is that you can put up as many windows as you need. You can open a new window at any time and switch between multiple windows at will. New windows are a great way of quickly and easily storing a found set for use later on. That list of fields you got from the portal above, you can put that up in a new window and go back to the layout and keep looking at the details before you go through the field list.

4.03 Useful Keyboard Shortcuts

Almost all of the built in FileMaker keyboard shortcuts remain in place. Command-F for find mode etc. Plus we've added a couple of extra ones to make navigation easier :

Command-N : Is mapped to the "New Window" command. Instead of creating a new record, which doesn't make sense in BaseElements, this will put up a new window.

Command-Shift-H : Home. This is the same as clicking the Home button in the navigation toolbar.

Command-Shift-← (Left arrow) : Back Button.

Command-Shift-→ (Right arrow) : Forward Button.

As well as that, some existing Keyboard shortcuts have extra behaviours in certain situations.

Command-C : Copy. This will copy field data when inside a field. However, when on the certain layouts, and not in any field it will copy the current element(s) to the clipboard [45], ready to paste into FileMaker Pro.

Command-P : Print : When on list views, or when viewing any of the reports, this will take you to a special Print version of the current layout and then run a script preview and print process.

4.04 Modifier Keys

The Navigating the UI [17] section details how to use the portals to view related data, and view either a related element directly or a list of all of the related elements.

In BaseElements there are two extra modifier keys that make this feature even more useful.

Option / Alt

Whenever there is a link to another element, you can cause that link to open in a new window by holding down the option key. For example a list of related Fields will open in a new window and the current window will remain where it is. This works for single items inside portals, entire related found sets using the arrow in the portal header, the links in the details area that are underlined, and the main navigation bar across the top.

Shift

The shift key has an extra function when you're clicking the arrow at the top of the portal, or an underlined item on the details area. In this case, it will do a Go To Related Record using found set. So for example, you can Go to a list of layouts, then hold shift and go to a list of all of the buttons on all of the layouts in the found set. Then go to another list of all of the scripts called by all of the buttons in the found set.

Combinations

The best feature is that both option and shift work together, so you can use both or either one at any time.

4.05 Searching and locating

There are two methods to finding things within BaseElements : the built in Find functions and the integrated fmSearchResults [57] that is tied to the QuickFind.

Built in Find options

The built in Find mechanisms in BaseElements aren't restricted at all so you have full access to all of the find functionality like in any other unlocked FileMaker file. The find options within the menu bar is available, as well as the extra options available in the contextual menus.

The ability to do normal find requests within BaseElements becomes especially handy when linked to the extra GTRR option available in the portals. So you can, for example, go to a list of scripts, and then omit some of those records. You can then do a GTRR using found set [18] (hold down the shift key) to view all the steps for those scripts. You can then constrain the found set to a single step type by right clicking on the step name field. So in a couple of steps you can produce a list of all of the import script steps for a limited set of scripts. You can then see all of the fields used in all those steps.

Also when finding within BE, the related fields can be searched on, and included in the criteria.

Smart Find

There is a setting available on the "Setup" tab of the home screen for "Smart Find". This checkbox is on by default. The Smart Find option will use privileges to limit your access to the items for a single Analysis. Whatever the last Analysis you viewed on the Analysis layout, it will keep track of this Analysis, and it uses privilege to automatically limit your found sets to this one Analysis.

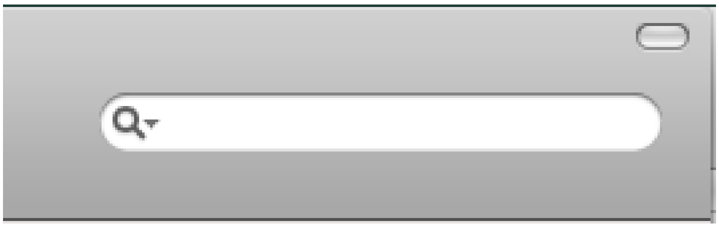
This allows you to have many different solutions or many different versions of the one solution or even both, all within the one copy of BE and to only be working on a single Analysis at a time.

If you want to search across lots of solutions, or to compare the changes within different versions of a single solution you can turn this setting off in the Setup tab at any time. Turn it back on to revert to a single Analysis.

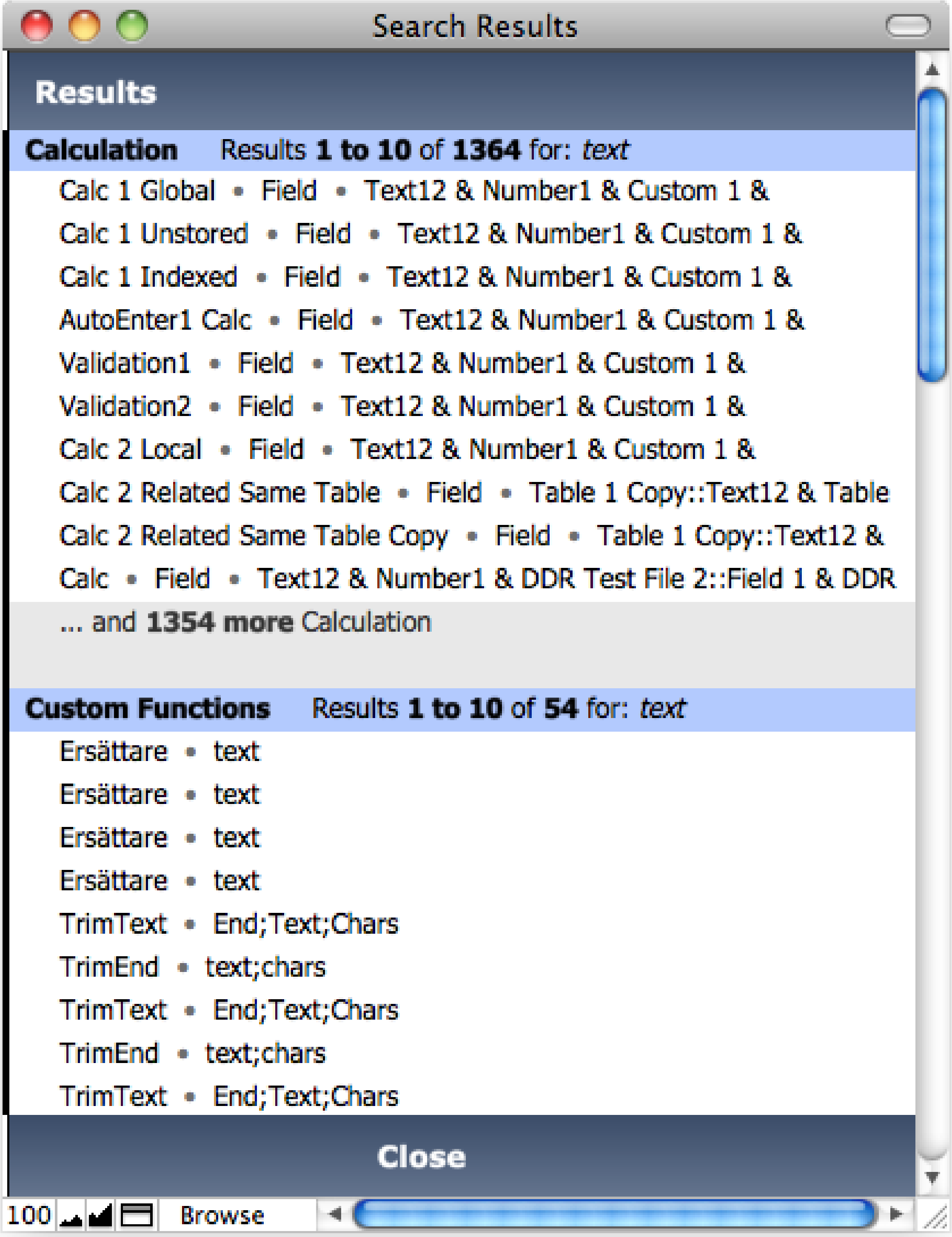
fmSearchResults

The final extra options for finding in BaseElements is the integration of the excellent fmSearchResults [57]. fmSearchResults is a solution available from SeedCode [58] that you can integrate yourself into any FileMaker solution.

In BaseElements 3, fmSearchResults is built into the QuickSearch field in the toolbar. We've replaced the default QuickSearch options with a search linked to fmSearchResults. To do a search, just type your request into the QuickSearch field in the toolbar and hit enter :



This will search for the find criteria across every part of the solution, in every area. So this includes all text on layouts and used in calculations or the names of objects. The results appear in a window like the following :



Calculation Results **1 to 10** of **1364** for: *text*

Calc 1 Global • Field • Text12 & Number1 & Custom 1 &
Calc 1 Unstored • Field • Text12 & Number1 & Custom 1 &
Calc 1 Indexed • Field • Text12 & Number1 & Custom 1 &
AutoEnter1 Calc • Field • Text12 & Number1 & Custom 1 &
Validation1 • Field • Text12 & Number1 & Custom 1 &
Validation2 • Field • Text12 & Number1 & Custom 1 &
Calc 2 Local • Field • Text12 & Number1 & Custom 1 &
Calc 2 Related Same Table • Field • Table 1 Copy::Text12 & Table
Calc 2 Related Same Table Copy • Field • Table 1 Copy::Text12 &
Calc • Field • Text12 & Number1 & DDR Test File 2::Field 1 & DDR

... and **1354 more** Calculation

Custom Functions Results **1 to 10** of **54** for: *text*

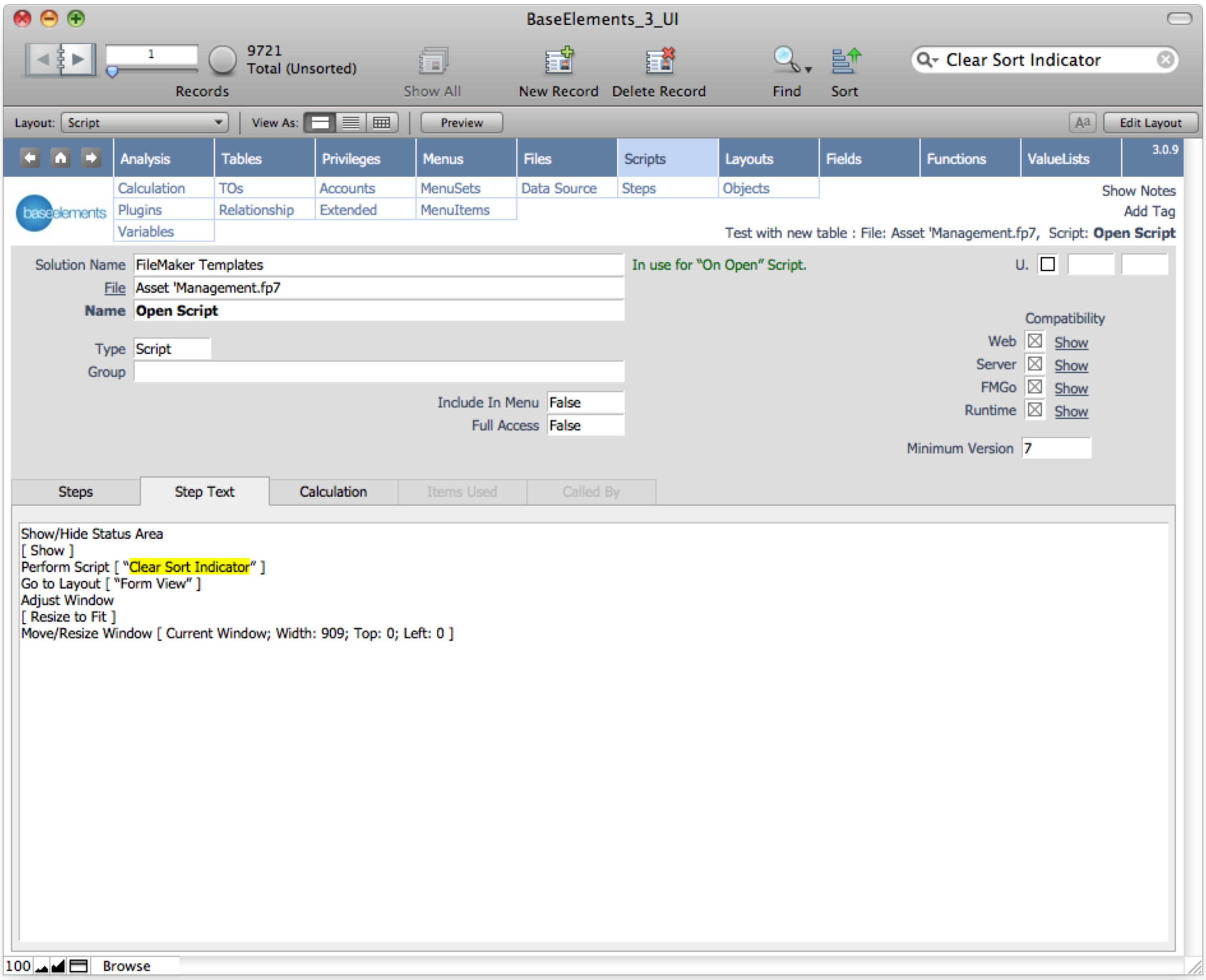
Ersättare • text
Ersättare • text
Ersättare • text
Ersättare • text
TrimText • End;Text;Chars
TrimEnd • text;chars
TrimText • End;Text;Chars
TrimEnd • text;chars
TrimText • End;Text;Chars

You can click on any one item to view that item in it's normal Form layout. Plus you can click on any header (the light blue row at the top of a section) or any footer (the grey totals at the bottom) to view all of the respective items for that type.

The fmSearchResults integration is great for locating individual items, like a section of text within a specific dialog, or for finding the use of any item quickly.

Highlighting

Also introduced in version 3.0.9 is the ability for the search text to show up in Calculations, Scripts or ScriptSteps as a yellow highlight. This is best demonstrated by the screenshot below :



1
9721
Total (Unsorted)

Clear Sort Indicator

Test with new table : File: Asset 'Management.fp7, Script: **Open Script**

Solution Name: FileMaker Templates
File: Asset 'Management.fp7
Name: **Open Script**
Type: Script
Group:
Include In Menu: False
Full Access: False

In use for "On Open" Script.

U. ☐

Compatibility
Web ☒ Show
Server ☒ Show
FMGo ☒ Show
Runtime ☒ Show

Minimum Version: 7

```
Show/Hide Status Area
[ Show ]
Perform Script [ “Clear Sort Indicator” ]
Go to Layout [ “Form View” ]
Adjust Window
[ Resize to Fit ]
Move/Resize Window [ Current Window; Width: 909; Top: 0; Left: 0 ]
```

Any text you put in the quick find box will be highlighted automatically in yellow. This makes it easy to locate text within a large calculation or script. The bonus is that you don't even have to perform a search, by just typing your highlight into the field you can then click out of the quick find in the toolbar and the highlight will display without any searching.

Overall

There is the combination of fmSearchResults, the ability to use the the powerful built in FileMaker find options, the advanced options available for GTRR inside portals, the ability to put up new windows at any time to retain a found set, and the quick back and forward, all combine to give the user lots of options for locating data within your solution.

4.06 Reports

BaseElements is designed to work within the data structure of your actual data, so the reports available are deliberately setup for functions you can't get directly from the data. There are four reports :

- Comparison Report
- Consolidation Report
- Error and Unreferenced Report
- FileMaker Go Compatibility Report

All reports are available from the Home screen on the reports tab. You can also see what reports have been created with any particular Analysis on the main Analysis layout.

Viewing Reports

Once the report has been run or selected, all of the records generated for the report are available to view as either a standard FileMaker List or Table. You can search within the report, or even across multiple reports. As well there are additional fields called "Assigned To", "Complete" and "Notes". These are designed for users who want to use the reports as an action or todo list, for example when working through a list of errors and fixing the items within.

The Assigned To field uses a drop down list, based on previous entries in the field to make re-use easier. The Complete is a single checkbox field that is handy for marking items off once actioned. And the Notes field is a free form text entry field if you need to track any useful information after the report is generated.

Printing Reports

All List view Report layouts (as well as all list layouts across the entire system) have a matching print layout already. You don't need to perform any special actions to print a report layout, when you invoke the Print menu item, either via shortcut key, or via the menu, it will automatically take you to the correct print format layout, and go through a scripted print process. The Print layouts don't display in the Layout Menu by default, but you can edit them yourself or look at the details used by going into Layout mode first.

4.06.1 Comparison Report

The comparison report will compare two different Analysis. The normal use for this would be to compare two version of the same solution at different times, to see what changes have been made since the first version.

To run this report select and Old and New Analysis, and also an Old and New File within that Analysis. There is also an option to compare by name or compare by id.

Compare by Name

Comparing by name will use the name of any layouts, scripts, fields etc, when comparing the Old file to the New file. So for example a script called "Startup" in the Old file will be compared to a script called "Startup" in the New file. If you renamed the script "Startup Process" in the New file, it would be treated as a new script, and the old script considered deleted.

Compare by ID

The compare by ID option uses the internal id of any items as the basis for a comparison. Internal ids are not visible inside FileMaker, but every layout, script and table has a unique id. These are set on creation and can't be changed, except by rebuilding a file from scratch.

Why is there a choice?

Imagine you have an Analysis of a file you worked on last week. You have then made some changes in the same file, and you run a new Analysis and then want to see what changes have been made. In this case you would compare by id. The file before and after is exactly the same, so a renamed obejct would be treated as a name change, not a delete and a new, as the id will not have changed. In most cases the compare by id is the more useful comparison.

However ... there are times when you want to compare by name. Imagine you have a solution that you develop off-line. So you have a live version of the database, and the development version, You make a lot of changes, tests and modifications to the dev version. When you're happy with the final result, you migrate these changes into the live version by repeating the changes.

So, in your dev file, you might create Field1 (id # 100), Field2 (id #101) and then change your mind. You delete Field2, and create Field2New (id #102). When you re-create these new fields in the Live file, you only create Field1 (id # 100) and Field2New (id #101) because there is no need to have the extra field created and deleted. Your field ids (the hidden unique id behind everything) are now out of sync and even if you make the same changes from here on, you will always be out of sync unless you look at the internal id and create dummy fields you delete to get them back into sync. But the same also applies to scripts, layouts, tables etc etc.

In this case you want the files to be "functionally" the same, regardless of the internal ids. In other words, you need there to be a Field1 and a Field2New, and nothing else, and the calculations or other settings for both fields to be exactly the same. In this way, the files will operate exactly the same, data could be transferred from one to the other without issue. If they're the same then you've accurately replicated your changes in the Live file.

In this way, comparing by name will show you that the files are behaving the same regardless of whether the internal ids match.

4.06.2 Consolidation Report

The Consolidation Report is a report for moving sections of one file to another. It was developed primary as a tool to merge a multiple file solution into a single (or at least fewer) file solution. It runs with the data from a single Analysis, and you select a Source File and a Destination File. You can also optionally select only one or more tables if you're moving a table from one file to another.

The consolidation report is a highly optimised process that allows you to do these large scale migrations with a minimum of effort and with minimal renaming or breaking of your existing functionality. There are a lot of built in smarts that optimise when to do certain changes and what changes are dependent on what other sections being done.

All of the various steps are outlined below, and you can perform these manually if you wanted, but the Consolidation Report puts all of these into a simple, easy to follow checklist.

Details

1. Create File References that don't exist in the new file, making sure to match by name, so that a Reference to File A has the same name in both Files.
2. Create Custom Functions. You need to be careful in terms of checking for dependencies in Custom Functions. It's simple to copy and paste, but you also need to be aware that Custom

Functions can refer to fields and you may not have access to the fields yet. Where there are any issues create the Function but leave the calculation empty, or make a note inside the calc to resolve this later.

3. Create Value Lists. If a VL with the same name exists, then : 1) If it's an external VL pointing to the other file and it's custom, make it the same custom values as the other file. 2) If it's based on fields, leave it as is. 3) If it's Custom Values, and is different from the other one, you will need to rename one or both. 4) If it's external and points to another file, you'll need to rename the original one, and create a new VL with the same values and the same name. If any of the VLs to be created need fields, mark them to be fixed later.
4. Check you don't already have a Table with the name of each of the Tables to be created, and also check that there aren't any TOs with the same name as the Table to be moved. If any exist, rename them in the Destination file.
5. You then create TOs in the new file to match the graph that you are moving, for every TO except one - the only one you don't create should be the "base" TO and should have the same name as the Table. Use File References to have all of the TOs pointing to the correct Base Tables. Do not create any relationships between these new TOs, you essentially create the Graph without any joins. It's easier if you put things in the same positions if possible. You also don't need to create a TO for any table that only has a single TO, and where that one TO is the same name as the Table.
6. Copy and Paste all of the Tables in the old file into the New. This will also create a TO for each of the Tables, and a layout. Note, that all of the Calculations will still work, even though the Relationships between fields don't exist.
7. Join up all of the Relationships that were missing in the Graph.
8. Change any of the Base Tables for the TOs you created earlier, to move anything that is coming out of the old file to be TOs for Tables in the current file. Change them to use the local file, not a File Reference.
9. Go back to your ValueLists and repoint any that were using Fields to use the new local Tables.
10. Go back to your CustomFunctions and put back any calculations that used Fields that were left empty before.
11. Create all of the Layouts that were in the Source file, but leave them empty.
12. Create all of the Privilege Sets that exist in the new file. If there are special Privileges for Scripts, leave this section aside until later.
13. Create all of the Custom Menu Sets, but leave them empty if they use Scripts.
14. Copy and paste all of the Scripts.
15. Change the Layouts and setup the Layout parts and sizes.
16. Copy and Paste all of the Layout objects.
17. Add any Layout Triggers.
18. Alter any Privilege Sets that used special Privileges for Scripts.
19. Alter any Custom Menus that use Override actions and call scripts.
20. Change all of the script steps that call scripts in the old file to point to the new local file. These will be local scripts, but also scripts in other files that need to be changed from the Source to the Destination.

The advantage of using the comparison report is that it will outline all the specific ValueLists, Functions or Steps that require special treatment. It knows which ones require that they be altered after creation, or which ones will be affected by the process. You can find these yourself, but the report makes it easier.

Post Migration

Once you've done all of the grunt work of moving the details, you should run a new DDR and import of all of the files (including the Source) into BaseElements. You want to check for errors in the Destination that weren't there before in case any mistakes were made.

You also want to look at any references that point to the Source file. Go to the Source file, and to the "FileRefs" tab. Click on the arrow at the top of the portal with "File References To this file". You want to go through each File Reference and change the places that use this FileRef until each Reference is no longer used anywhere. At that point the Reference can be deleted. Once you've no longer got any

References pointing to the Source file you have detached it from the solution and it can be excluded.

4.06.3 Error and Unreferenced Report

BaseElements by default already marks any items with errors, warnings or that are unreferenced by recording a count of the number of errors, and by altering the text colour of the name of the item. As well as that you can consolidate all of these items into a single list by running the Error and Unreferenced list.

To run this report, go to the Home screen and then to the Reports tab. Select the "Errors" tab and fill in the options there. You must choose a Solution to run this report on, and optionally you can also limit it to a single file within that solution. You can also select any combination of Errors, Warnings and Unreferenced items.

Click the Generate Report button once you've made your selections to see the report.

4.06.4 FileMaker Go Compatibility Report

With the introduction of FileMaker Go for iOS, there is an extra option available for checking your solutions compatibility with this new OS. Go to the Reports tab of the Home screen and then to the "Go Compatibility" tab. Select which Analysis to run the report on, and click the "Generate Report" button to create the new report.

The FileMaker Go compatibility is updated in a new release of BaseElements when there are changes to FM Go, and these changes are reflected in the items, where it will mention the version required for this feature to work. The report is based on the following technotes and information about FM Go from FileMaker :

Calculation and scripting behavioral differences in FileMaker Go [59]
Designing a layout optimized for FileMaker Go [60]
Behavioral differences of FileMaker features when accessed by FileMaker Go [61]
Differences between FileMaker Go and FileMaker Pro [62]

as well as the FileMaker Go Development Guide [63].

The report consists of two main parts. First are the critical notes items from the details above. These are reproduced for information purposes only and should be read before developing on FM Go. The second part is where all of the relevant items from the technotes above are analysed in the solution you selected to see where they occur.

For each item, the number of times it appears in your solution is counted and can be referred to directly from the link in the report list. Any items with no occurrences are still in the report, but the count is missing and the text is greyed out. You can then see all of the relevant items at a glance and easily look at only those items that apply to your solution. By clicking on the count you can also view directly the actual items referenced by the report.

The FileMaker Go report is smart enough to recognise areas where there are minor differences in implementation that affect compatibility. For example if you're using the Send Mail script step, the report will only flag situations where the step uses a dialog that FMGo can't display. It won't show items where the step would run without issue in FM Go.

4.07 Runtime vs FileMaker File

BaseElements is available as FileMaker files for use in a copy of FileMaker Pro or Pro Advanced, or as a standalone runtime application. Both have their advantages and disadvantages.

FileMaker Files

The FileMaker files are .fp7 files in BaseElements 3 or .fmp12 files in BaseElements 4 or later. These require a copy of FileMaker Pro / Advanced to run, and also require you to install the plugin [8] yourself in either the Application's Extension folder or the Extension folder of your user folder. BaseElements will attempt to automatically install the plugin for you if it isn't currently installed.

But also the fact that you're running FileMaker means you can modify the BaseElements files. You have complete access to alter the layouts in BE, or to add your own scripting logic to any of the files. As well as that you can create your own external files that reference the Data file and build your own UI or re-use the BE data in any way you see fit.

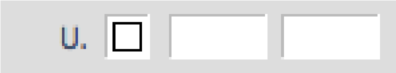
Runtime

The runtime version of BaseElements has the advantages of allowing you to have a separate application for BaseElements and switch between it and your solution running in FileMaker Pro at will. The plugin is already in the correct location, so there is nothing extra to install. Also because BaseElements requires FileMaker Pro 11 to run, if you don't have this version of FMP, you can use the runtime to import a DDR from any version of FMPA.

However, the runtime doesn't allow you to use Layout mode or edit Scripts. You could though build an extra file with your own UI or other details which links to the data in the BE runtime. The runtime BindKey is just "BaseElements", so by building a runtime of your own with the same key you can add extra files into the BaseElements runtime solution.

4.08 Unreferenced

BaseElements will build a list of all of the places that an item is in used, and when there are no references for any one item it marks them as "Unreferenced". This is displayed via a checkbox on the top right of all of the form layouts :



and in list layouts and some portals. The "U." text is an abbreviation for Unreferenced where there is often not enough room to display the full text. Wherever you see this checkbox in the top right, or after the name of an object you can know that it's the unreferenced checkbox. There are three possible states for the unreferenced checkbox :

Referenced - the checkbox is empty, like the screenshot above.
Unreferenced - the checkbox is selected and has a cross in it.
Might be referenced - the checkbox will have a single dash in the middle of the box.

The dash convention is not standard for FileMaker checkboxes and refers to the possibility that the item might be in use, even though there are no references to it.

Referenced

The checkbox is empty and an item considered referenced when there are items that refer to this item. You will be able to look at what those items are by looking on the various tabs and portals on the Form layout for this item.

Might be referenced

This state is invoked in two situations. Firstly is when the item is a Layout that is set to show in the Layouts menu, or a Script set to show in the Scripts menu. In that case the item isn't being referenced directly but may be in use by the user when invoked manually from the menus that display the item.

The second situation is when the item type is being referenced by Indirection, and in that situation, any possible item of the same type may be being used Indirectly via calculations. In this instance it's impossible to know exactly which items are being referred to, so the dash in the Unreferenced checkbox is a warning to be cautious when modifying or deleting this item.

Unreferenced isn't the same as "Safe to Delete"

The Unreferenced checkbox is a very useful tool for cleaning up a solution that has lots of items that are no longer being used. But be aware that there are lots of places that items can be used outside of the list of possible references, as well as reasons why the reference count is incomplete. Firstly the ability to use Indirection restricts what references are generated for any item, and so that limits your access. As well other options for accessing the data inside FMP from outside (things like ODBC or PHP) mean there are lots of external ways to access FileMaker data.

Also be aware that any errors in calculations can affect the reference count for that calculation. When the calculation has errors, FileMaker doesn't generate any references at all. So any other items used in that calculation won't have accurate reference counts.

4.09 Indirection

Indirection in FileMaker is the ability to refer to items "Indirectly". Ordinarily when you build a calculation and use fields or custom functions by name in the calculation, FileMaker links those names to the internal id of the item, and in that way you can close the calculation, change the name of the referenced items, and the calc will also update to match. This ability doesn't exist in many other programming languages and makes life much easier for long term administration of a solution where you can alter items at will and not break your entire solution.

Newer versions of FileMaker have introduced lots of ways of referring to items by their name instead of via this internal ID, and so there now exists the ability to change a name of something and break a calculation. For example you can use GetField ("TextFieldName") and if you change the name of the field from "TextFieldName" to "ChangedField", without updating all of the places you're using this in calculations, you code will no longer work.

As well as that your calculations can be completely dynamic, and build the fieldname from user data. So GetField ("Text" & $Variable) is an equally valid calculation, and the variable data may come from a text field or dialog that the user selects.

BaseElements builds it's list of Indirection functions from any function call that allows you to select a Layout, Field, or Value List or any function where you can include entire calculations that may also refer to Custom Functions.

How this works in BaseElements

BaseElements doesn't try to distinguish between the simple cases such as GetField ("TextFieldName") where it's quite obvious to the user what the field is, and more complex cases such as GetField ("Text" & $Variable) where the resulting field(s) may be unknown to the user. The reason for this is that there are just too many possibilities with the complex options for us to decide which items are being referred to at run time. We could build in a simple case solution that found some of these, but we've taken the decision that we'd prefer not to have a solution that only works half the time and where you would never be sure which half it was being accurate with.

Instead of trying to parse out individual item references, BaseElements keeps track of which items may be using indirection and of what type and lets you make decisions about how this will affect your solution.

How to use this in my solution

Firstly, to make your life easier, use Indirection sparingly. In the later versions of FileMaker you even have options like GetFieldName which allow you to use the item directly and also refer to it by name, so your reference counts will still be included. The ability to Indirectly refer to things is powerful and steps like SetFieldbyName can reduce complexity on a massive scale. Use Indirection where needed and keep track of the sorts of things you're using it for. You can also use naming conventions to mark the items that are being used Indirectly as a warning to not delete or rename them.

Secondly be aware of the various states of the Unreferenced checkbox and don't delete items without being aware of where they may also be used. Don't delete Fields or Custom Functions in a solution when there are calculation errors, and always test your solution before going live after you've deleted items.

4.10 Warnings

As well as Errors and Unreferenced items, BaseElements also has a list of Warnings that can alert you to issues in your solution. Warnings are shown via the orange count field on the top right of any Form layout, as well as the totals on the Analysis and File layouts.

Warnings are any situation where there isn't anything structurally wrong with your solution (these would be considered errors), but where there are situations that may cause unexpected behaviour or just make life difficult. Examples of warnings are :

- Using fields in a relationship join that are of a different field type. You can relate a number field to a text field in FileMaker, and it will work, but occasionally it will give you unexpected results. You can also do it deliberately to achieve specific results.
- Having an Account that requires you to change the password on next login, but the Privilege Set doesn't allow password changes.

Warnings are a useful way to see what might be an issue when a solution doesn't behave as expected. Not all warnings should be fixed, but you should be aware of what the warnings are and what is involved with them.

5.0 Other Features

5.01 Analysis Printout

You can quickly print out an overview report of any one solution, by printing the Analysis printout. This is available from a link in the portal on the Home screen, or by choosing Print from the menu when you're on the Analysis layout.

This report gives you a list of all of the files, with some of the more major counts for the solution, as well as Errors and Unreferenced items, and totals for all Files. This is a quick and easy way to document the size of a solution as well as the possible scope of the errors needing to be fixed or items that are no longer required.

5.02 Notes

Every item in BaseElements has the ability to have notes recorded against it. Notes can be for any purpose and are useful when documenting solutions and changes. To see the notes portals, click the "Show Notes" on the top right of the screen of any Form Layout. The text changes to "Hide Notes" whenever the notes tab is active. When there are already notes for any item then the "Show Notes" text will be bold.

The notes field is intended for general notes about any one item. There is also a "Tag" field in the notes portal that allows you to assign short tags to any item. This tag field also displays on the Form view, so is a quick and easy way of adding a tag that is always visible on Form view of any item. Also by utilising the "Text Color" field, you can colorise any of the tags so that they're easier to see on the Form layout.

Also notes can be assigned to anyone, and also marked as completed. They are a useful todo list type system as you can look at the notes list layout and find anything not completed and assigned, and work through them one at a time.

The notes tab will also show a separate portal for any error report records that belong to this item. They are only active if the error report has been run for this File though.

5.03 Indented Scripts

BaseElements will automatically show the Scripts, Script Steps, or Layouts in the same level of indentation that is present inside FileMaker. By looking at the use of Script or Layout Groups, or the use of If or Loop steps, it assigns an Indent level to every item and will use this to display in portals or lists where appropriate. The actual step text or name is stored without any Indenting to make searching easier.

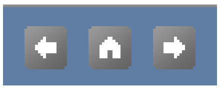
5.04 GTRR from Scripts

When looking at any one particular script there is a portal called "Steps" that shows all of the steps within that Script. There is a link in front of each step that takes you to the primary linked item for that step. So for example, an IF step will link to the calculation used, a Set Field step will link to the field used and a Perform Script step will link to the Script being called.

These links are a great shortcut when going from any Script to the items being used by a step without having to first go to the step in question.

5.05 Back and Forward

At the top left of any of the layouts in BaseElements there is a back and forward button :



BaseElements records a scripted log of all the layouts you visit and the links you click on, so that at any time you can go back (or forwards) through the places you've been. This is very useful for linking deeply into a certain record to find some details and back tracking to your original starting point.

In between those buttons is the Home button which takes you to a File layout when on any detail item layout, or the Analysis layout when on any File layout, or the Home screen from there.

5.06 Plugins and Variables

BaseElements will also sort through all of your calculations and record where you are using Plugins or Variables. There is a portal listing both of these items on the File layout and you can click directly into any one to see exactly where it's used.

Also any Plugins or Variables used by a calculation or any other item will also be listed on the "Items Used" tab.

Plugins are not distinguished by the DDR as separate function calls, so there is extra processing in the import process to break out any internal functions and leave behind only external plugin related functions.

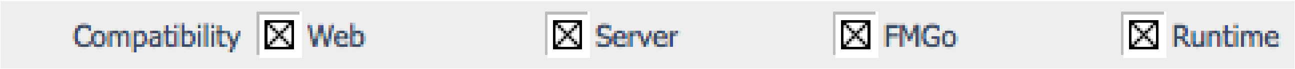
And Variables are not broken out of the calculation at all, so after the calculation import is complete we actually process the text inside the calculations to strip out the variables used. The code to do this is not part of the DDR or of FileMaker's calculation engine, so it may not pick up variables using complex characters or names. Technically it's possible to have all sorts of strange characters in variable names, but we suggest you keep them limited to basic ascii to make things easier.

5.07 Server, FMGo, Runtime and Web Compatible Script steps

BaseElements processes all of the Script Steps in your solution to determine whether or not they are compatible with the various other implementation options. It understands compatibility with

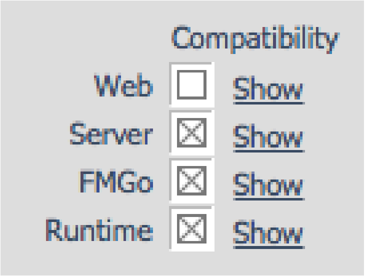
- FileMaker Server scripts
- Scripts run in either Instant Web Publishing or Custom Web Publishing
- Steps not available in Runtimes
- FileMaker Go incompatibilities

This information is available at the Step level, via a set of checkboxes :



Compatibility ☒ Web ☒ Server ☒ FMGo ☒ Runtime

This allows you to see at a glance the compatibility of this individual step. As well as that there is the same option available at the Script level as well :



Compatibility
Web ☐ Show
Server ☒ Show
FMGo ☒ Show
Runtime ☒ Show

The checkboxes in this case indicate that the Script will run on FileMaker Go, a Runtime, or on the Server, but not on the Web (IWP or CWP). There is also the option to "Show" the compatibility for this Script. By clicking the "Show" link, the incompatible steps within that script will turn grey in the same way they do in the Script editor itself. This allows you to see at a glance which steps are issues and would need to be worked around before being used on that platform.

5.08 Smart Find

BaseElements has a setting on the "Setup" tab of the home screen called "Smart Find". It's a simple checkbox you can turn on or off and by default it's on for new users.

The Smart Find option turns on some options in the Privilege Set of BaseElements to limit you to a single user at a time. So when you click on any Analysis from the Home screen, or if you view the Form view of an Analysis record, it stores this record ID and limits you to only records from that Analysis at any one time. This means that your searches are simpler as you're not finding items from other solutions within this one.

However, if you ever want to search across multiple copies of a Solution, (for example to see changes to a Script over time), then you can just turn off this option, and you're back to viewing all available records.

Be aware if you're using Data from the BE Data file in other files that this is a Privilege Set level restriction, so your records will be restricted outside BE as well and you will get No Access warnings for records that are unavailable.

5.09 Growl Support on Mac

Growl [64] is a Mac OS X Preference Pane that allows applications to send notification events. We have linked into this to send events whenever a BaseElements import is complete, or when an Analysis is deleted. To enable these events to be sent, you first need to download and install the FMGrowler [65] plugin. This is a free plugin, so there is no extra registration or setup to be done. If BaseElements detects that the plugin is installed and active then it will send Growl notifications.

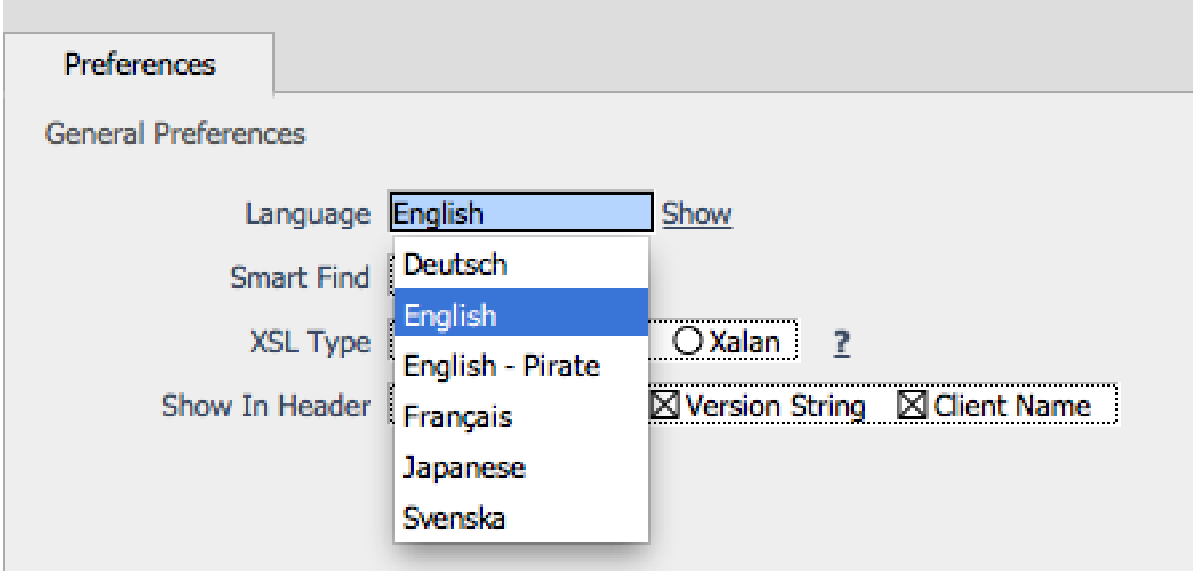
You can administer the notification levels and activity by going to the Growl item in System Preferencences.

At present we are unaware of any similar plugins for Windows, although there is at least one Windows application that performs a similar task on windows. If there is demand and a plugin exists, then we'd be happy to implement it.

5.10 Translations

BaseElements has been kindly translated into other languages by some very dedicated helpers. As of the last writing, there were French, Swedish, Japanese and German translations available. The actual state of the translations is constantly changing, so the best place to find details is in the current Version History [66] details.

To change from one language to another, go to the Setup tab, and to the Language field and choose an alternate language from the drop down list.



Language: English

XSL Type: ○ Xalan

Show In Header: ☒ Version String ☒ Client Name

Also, by clicking the "Show" button you can see and edit the current list of translations inside BaseElements. On that layout is a set of portals with fields that correspond to each of the text items used in BaseElements. By comparing the English record to other records, you can add to, modify or enhance the current set of translations. For example to add a new translation to BaseElements, just duplicate the English record and then change the data in the fields to the alternate language equivalent. Once you also then change the name of the Language it will become available as a choice in the drop down list.

If you use any of the other languages and find items missing or think they're inaccurate, please feel free to submit suggestions or alternatives.

5.11 Importing From Previous Versions

BaseElements includes the ability to import from another set of BE files.

It's not a highly advertised function but it is there. The reason it's not more prominent is that every version of BaseElements can include changes to the core import code or other functionality that may render the import incomplete. For example often during even minor updates there will be changes to the way the main data fields work, or changes in the import process that mean there is a different data structure between versions. When you import from one version to another, almost all of the data will come across. However we cannot guarantee that the behaviour or data will be the same between two versions. Differences will likely be small and not significant though.

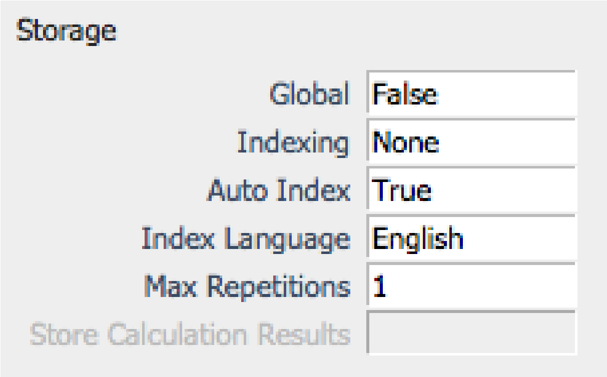
We do recommend that instead of importing from older versions of BaseElements, you either re-import the XML into the new copy, or retain the old versions of BE (you can keep multiple copies on a single machine) for reference purposes.

Obviously for some people that isn't always a preferred option, so if you do want to import from older versions into the latest BE, we suggest you label the Analysis as such and use them for cross reference purposes only. To import from a previous version :

1. Use the Window menu to switch to the BaseElements_4_Data file.
2. Go to the Scripts Menu and select the script "Import From Previous Version".
3. This will ask you to select the previous BaseElements_3_Data or BaseElements_4_Data file and will import from that.

5.12 Finding Indexed Fields

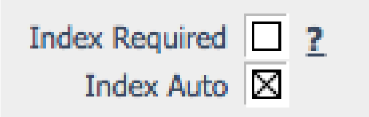
BaseElements imports all of the settings for whether or not a field is indexed. These are available on the Detail tab under "Storage" :



Storage

Global	False
Indexing	None
Auto Index	True
Index Language	English
Max Repetitions	1
Store Calculation Results	

This will tell you if a field currently has an index or not, and whether it's set to Auto Index when FileMaker decides if an index is required. As of BaseElements 3.0.9, there are also options for whether or not an index is needed for the current field :



Index Required ☐ ?
Index Auto ☒

Index Required

An index is required when :

- Used in a relationship. (On the "right" side in FileMaker, but we can't determine context in BE, so either side is used.)
- The fields has the settings for Unique or Existing Validation turned on.
- There are Value Lists that use this field.
- There are layout items that have the "Auto Complete using existing values" option turned on.

Any one of these options, when accessed, will generate an index for this field, and the feature won't work if the index cannot be created. If you've got the option for indexes turned off, then there is a warning generated to say that the field will need an index, but it's not available. There is the exception for fields that are only used on the left side of a relationship, so will never have an index created for them.

Index Auto

The Index Auto checkbox will be on whenever the field is used in a way that will auto generate an index. This will be any of the required options above, plus :

- Layout fields where the "Allow Enter in Find mode" option is on.
- Layout fields where the "QuickFind" option is on.

It is also possible to have scripts that perform finds on the field, or that bypass the field options above, so it's entirely possible to have FileMaker generate an index for a field where none of these items are set. However this checkbox gives you a good indication of the user accessible parts of the indexes, and what end users might be able to change within the system.

Useful Tip

You can use these fields to find fields that have an index that can be turned off. For example, find all fields that currently have an index (Indexing field set to "All" or "Mininal"). Then within that list, look at the "Index Required" and "Index Auto" fields to see why they might be turned on, and turn them off. You might also want to optimise and reduce file size by turning off the auto index for fields with lots of content or in large record sets.

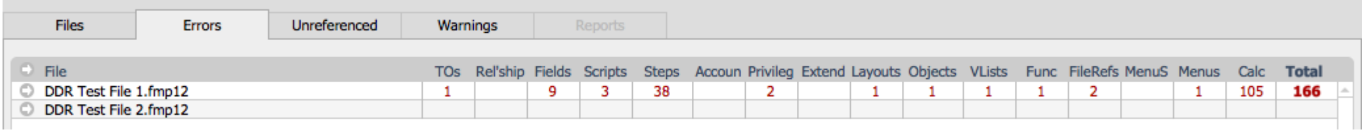
5.13 Marking Errors as fixed

New feature in version 4.0.0

As you're looking at and fixing errors in your solution, you can also mark the errors as fixed in BaseElements. This means they no longer highlight in red and also the counts for total errors reduce on the File and Analysis pages.

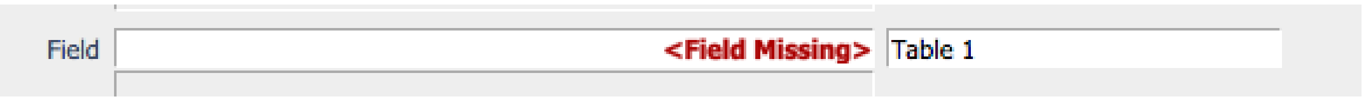
Finding Errors

Errors are marked as total counts on the main Analysis layout. The number is a count of items with errors, not a count of total errors. For example in the screenshot below, there are 38 Script Steps with errors and these steps are spread across 3 Scripts in total.



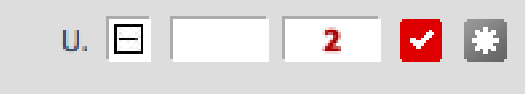
File	TOs	Rel'ship	Fields	Scripts	Steps	Accoun	Privileg	Extend	Layouts	Objects	VLists	Func	FileRefs	MenuS	Menus	Calc	Total
DDR Test File 1.fmp12	1		9	3	38		2		1	1	1	1	2		1	105	**166**
DDR Test File 2.fmp12																	

You can click on any item and see the error in more detail like this example of a Go To Field Script Step with a Field missing. The original Table is still there, but at some point the field was deleted.



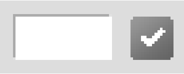
Field <Field Missing> Table 1

The error will count once for this Step, and again once for the Script that this Step is in. If you look at the Script itself, there will be an error count matching the number of Steps with errors. This appears in the top right of the main form layouts.

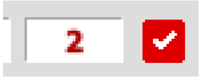


From left to right, these items are the Unreferenced checkbox, the Warning count, the Error count, the button to mark the Error as fixed, and the button to Highlight an item.

When the item doesn't have any errors, the button is grey :

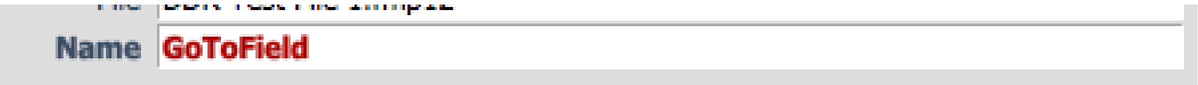


When there are errors, it turns red :



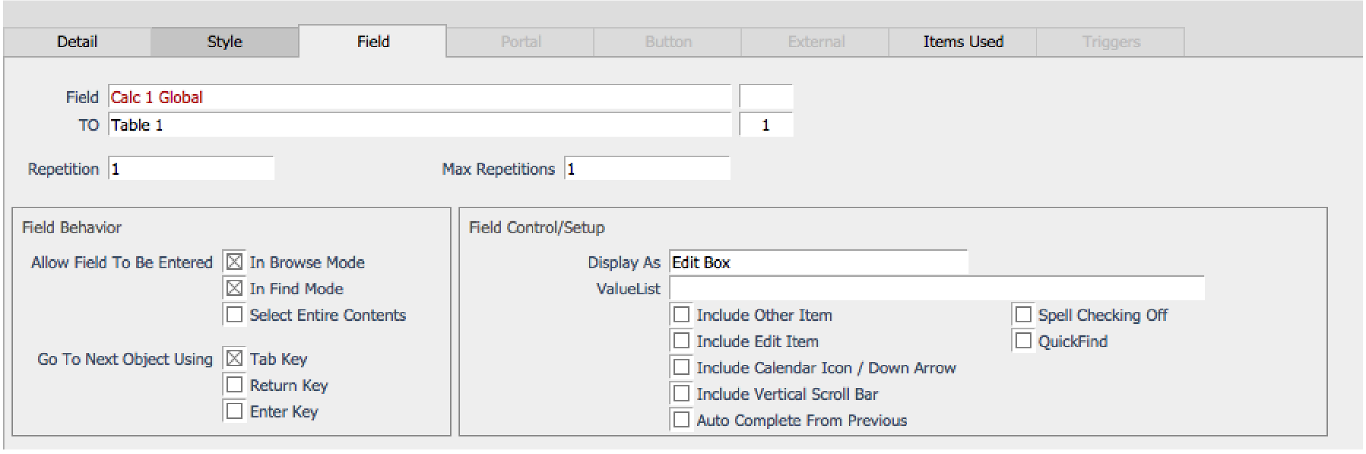
2

Also, the item's name will be red and bold :



Name GoToField

This is useful because it means that it's easy to locate this item if it's used elsewhere. So when you're looking at another item that uses this one, you can see straight away that there might be issues because it's dependent on something that has errors. Below is a Layout Object that uses a field with an error in it. The red highlight makes it obvious there's an issue :



Field Calc 1 Global
TO Table 1 | 1

Repetition 1
Max Repetitions 1

Field Behavior

Allow Field To Be Entered
☒ In Browse Mode
☒ In Find Mode
☐ Select Entire Contents

Go To Next Object Using
☒ Tab Key
☐ Return Key
☐ Enter Key

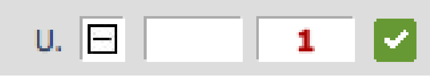
Field Control/Setup

Display As Edit Box
ValueList

☐ Include Other Item
☐ Include Edit Item
☐ Include Calendar Icon / Down Arrow
☐ Include Vertical Scroll Bar
☐ Auto Complete From Previous
☐ Spell Checking Off
☐ QuickFind

Fixing Errors

You can then click on the error button and it will turn green to indicate that the error is fixed.



U. 1

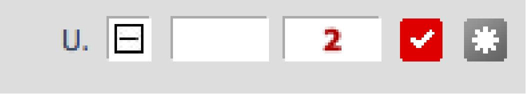
Also, the error counts are updated at the File level, and the name no longer highlights in red.

5.14 Highlighting items

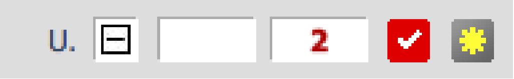
New feature in version 4.0.0

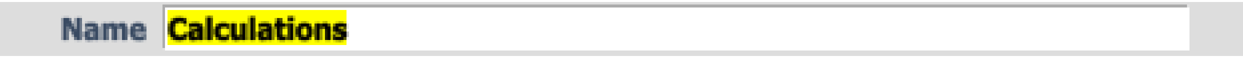
When you're looking at items in BaseElements it can be useful to highlight some items that you're working on so you're alerted whenever these items are used elsewhere. That way, a field that needs to be replaced, or a Script that is being updated will jump out at you if it's referenced by another item.

This feature is activated by clicking the "Highlight" button, next to the Error counts on the top right of any form layout :



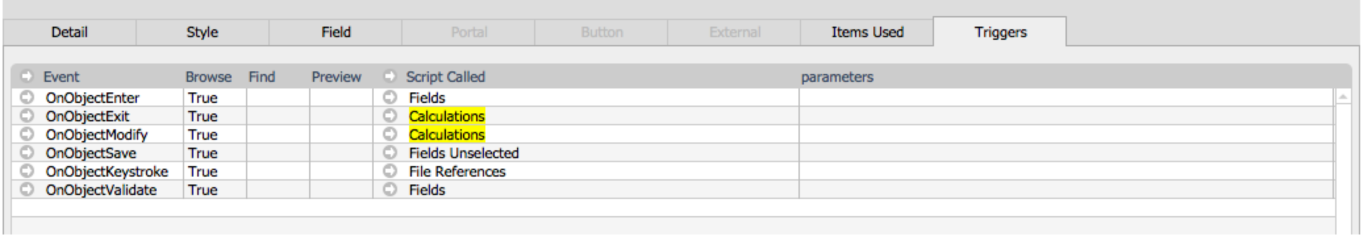
The highlight is the right most button. When clicked, it becomes yellow, and the name of the item highlights using the build in "Highlight" text function.





Name Calculations

Then when this item is used elsewhere, it will show the same highlight :



Event	Browse	Find	Preview	Script Called	parameters
OnObjectEnter	True			Fields	
OnObjectExit	True			Calculations	
OnObjectModify	True			Calculations	
OnObjectSave	True			Fields Unselected	
OnObjectKeystroke	True			File References	
OnObjectValidate	True			Fields	

5.15 Copy to Clipboard

BaseElements overrides the default copy to clipboard with some extra actions based on what element you are looking at and which layout you're on.

The standard action of copying text when in a field remains. However, if you're not in a field, BaseElements can copy elements from your solution so that you can paste them into an active FileMaker solution. BaseElements understands FileMaker's unique clipboard format for Custom Functions, Scripts, Steps, Fields and Tables.

When you're on a form layout for any of those items, using Command-C or the Copy item in the Menubar BaseElements will set the clipboard to that specific item. When you're on a list layout for any item BaseElements sets the clipboard to all of those items.

So you can, for example, copy a list of Scripts or Fields out of BaseElements, and then paste them into another FileMaker file with a standard copy and paste.

6.0 Customising and Extending BaseElements

6.1 Modifiable layouts and scripts

BaseElements ships as two optional file types, .fmp12 files and a runtime application. But in both situations if you open the BaseElements files with FileMaker Pro or Pro Advanced, you have full access to modify Layouts, and add your own Scripts.

In order to do this with the runtime files, you will first need to open the main BaseElements_4.DDR file with FileMaker Pro.

When you modify layouts, you can add your own versions of layouts, or duplicate layouts and change the formats. If you want to, you can add your own Fields, Buttons or Tabs to any layout.

As well, the BaseElements_4_Data file is not locked for access by other tools. So you can also create separate layouts in another file that have Table Occurrences referring to Base Tables in the BE Data file. This allows you to have separate User Interfaces for your BaseElements data.

6.2 Integration with other solutions

Accessing the BaseElements Data

All of the data in the BaseElements Data file is accessible by other FileMaker files. You can add TOs to your other solution files that point to the Base Tables in the Data file, and do whatever you want with the data.

Going to an Analysis from another solution

BaseElements also has some hooks for integrating other solutions that link into the data in BE. In the BaseElements UI file, there is a script called "Event: Go to Analysis" that you can call from other solutions and reference an Analysis directly. For example if you have a project management solution where you track all of the projects you're developing, you can use this script to refer to an Analysis that represents on of the projects in your project management database.

To use this script, call the script "Event: Go to Analysis" with the Analysis ID of the record that you want to refer to. This script does a GTRR to find a matching record with the same ID.

Going to any other data in the UI File

You can also go to any record in any of the Tables, inside the main UI file by calling a script with the parameters. The script does a find, so it can find either individual records or a found set.

The script is in the main BaseElements UI file, it's called "External: Go to Related Records". You can call this script with parameters to set which layout to go to and which fields to search, along with which values to find in those fields. This script is modifiable, so you can see for yourself what it does, and make changes if required. The bonus is that it won't be changed in future releases, so you can call this script from other files and it will remain the same.

The parameters are a list of values you want to use. The first item is the layout name. All of the remaining values are in name-value-pair format according to our previously published examples [67]. So for example, to go to the layout called "MyReport" and search for "abc" in the field called "widgetname", you would use the following as your parameter :

MyReport
widgetname="abc"

There is a blog posting that documents this [68] and includes links to sample files for you to use.

6.3 Automating Imports

If you wanted to automate a BaseElements import that runs unattended, you can automate the imports by calling the appropriate script in the BE Data file. Once the solution is opened - by opening the main BE file - you can call a script to run the import automatically.

In the BaseElements Data file, there is a script called "Event: Import New DDR". You need to call this script with the correct parameters in order to start an import. Parameters for the script are in the form of name value pairs, separated by return characters.

This is useful if you have a large solution with long import times that you want to run regularly for example. You can setup a FMP "Robot" machine and copy your DDR to that machine. When the copy is done, you could use Applescript or a Windows event to automatically import the DDR overnight and have it ready for the next day.

How to use the Auto Import

First, you need to have BaseElements open. So you should open the main BaseElements file and let it run it's opening script. If you're doing this for the first time, it will ask for registration details, so put these in and it will remember them for next time.

Then call a script in the BaseElements Data file. The script to call is called "Event: Import New DDR". You need to call it with some parameters of the details of the DDR to import. The script paramater is formatted as a set of "name value" pairs. An example of the parameters would be :

```
"File=/file/path" & ¶ &
"Solution=My Solution Name" & ¶ &
"Version=Version Number" & ¶ &
"Client=Client"
```

The file paths need to be in the same format that the BaseElements plugin generates. To get an example of the format, use the Data Viewer or a sample file to call the BaseElements plugin function BE_SelectFile (""). This will put up a dialog asking you to select a file. Choose a suitable file, and the function will return the path to that file. You need to use this path format for the parameters to the plugin.

7.0 Purchasing

7.1 Licence options

BaseElements is available in two licence types, Single User and Site Licence.

Both licences allow you to run BaseElements as either a Runtime or FileMaker files on a single machine at a time. You can also put the FileMaker files on FileMaker Server and access them from FMP, although with a Single User licence you can only access the files one user at a time.

The Site Licence also allows you to run BaseElements on every computer at a company or all computers used by employees of that company. As well as that you can run BE on FileMaker Server and access it by other machines in the same company.

7.2 How to Buy

Full purchase details are always available on the web at :

http://www.goya.com.au/purchase [69]

There are links for standard copies as well as upgrades. Purchasing is done via FastSpring who handle payments and processing for us. They offer credit card, paypal and purchase order options.

We do have discounted pricing for competitive upgrades, and for Education or Not For Profit customers. Contact us using the form on the website for more details.

Tax is handled and calculated by FastSpring as the vendor, so only CA, USA related taxes are applied.

7.3 Registration Codes

After purchasing you will receive an email from FastSpring with the purchase details. You also receive a separate email with the registration details that you need to enter into the about screen in BaseElements. Copy the details into the dialog shown below and click "Register" to continue.

Registration codes sometimes take up to 24 hours to process, so you may not receive them straight away. If you have any issues with registration codes or purchase, contact us using the forms on the support page.

8.0 Support

Support is available for BaseElements using the site : http://baseelements.goya.com.au/ [70] There you can view FAQ pages, see questions other people have asked or contact us for help.

About

- About Goya
- About FileMaker
- Our Services
- Staff
- Clients

Products

- BaseElements
- RefreshFM
- BE Plugin
- Purchase
- Support

More

- Blog
- Contact Us

FileMaker 11
CERTIFIED DEVELOPER

FileMaker
BUSINESS ALLIANCE
PLATINUM

Come and see Goya at DevCon 2012 !

FILEMAKER DEVELOPER CONFERENCE
BREAK 2012
THROUGH
MIAMI, FLORIDA - JULY 16-19, 2012

Source URL: http://www.goya.com.au/baseelements/manual

Links:
[1] http://www.goya.com.au/baseelements/manual
[2] http://www.goya.com.au/baseelements/manual/10-about-baseelements
[3] http://www.goya.com.au/baseelements/manual/11-what-baseelements-0
[4] http://www.goya.com.au/baseelements/manual/12-requirements
[5] http://www.goya.com.au/baseelements/manual/13-updates-and-notifications
[6] http://www.goya.com.au/baseelements/manual/14-baseelements-and-filemaker-pro-12
[7] http://www.goya.com.au/baseelements/manual/20-setup
[8] http://www.goya.com.au/baseelements/manual/21-plugins
[9] http://www.goya.com.au/baseelements/manual/22-sharing-filemaker-server
[10] http://www.goya.com.au/baseelements/manual/30-database-design-report
[11] http://www.goya.com.au/baseelements/manual/31-generating-ddr
[12] http://www.goya.com.au/baseelements/manual/32-comparison-be-html-ddr
[13] http://www.goya.com.au/baseelements/manual/33-importing-baseelements
[14] http://www.goya.com.au/baseelements/manual/34-xsl-import-types

[15] http://www.goya.com.au/baseelements/manual/40-baseelements-basics
[16] http://www.goya.com.au/baseelements/manual/41-conventions-and-abbreviations
[17] http://www.goya.com.au/baseelements/manual/42-navigating-ui
[18] http://www.goya.com.au/baseelements/manual/43-useful-keyboard-shortcuts
[19] http://www.goya.com.au/baseelements/manual/44-modifier-keys
[20] http://www.goya.com.au/baseelements/manual/45-searching-and-locating
[21] http://www.goya.com.au/baseelements/manual/46-reports
[22] http://www.goya.com.au/baseelements/manual/461-comparison-report
[23] http://www.goya.com.au/baseelements/manual/462-consolidation-report
[24] http://www.goya.com.au/baseelements/manual/463-error-and-unreferenced-report
[25] http://www.goya.com.au/baseelements/manual/464-filemaker-go-compatibility-report
[26] http://www.goya.com.au/baseelements/manual/47-runtime-vs-fp7
[27] http://www.goya.com.au/baseelements/manual/48-unreferenced
[28] http://www.goya.com.au/baseelements/manual/49-indirection
[29] http://www.goya.com.au/baseelements/manual/410-warnings
[30] http://www.goya.com.au/baseelements/manual/50-other-features
[31] http://www.goya.com.au/baseelements/manual/501-analysis-printout
[32] http://www.goya.com.au/baseelements/manual/502-notes
[33] http://www.goya.com.au/baseelements/manual/503-indented-scripts
[34] http://www.goya.com.au/baseelements/manual/504-gtrr-scripts
[35] http://www.goya.com.au/baseelements/manual/505-back-and-forward
[36] http://www.goya.com.au/baseelements/manual/506-plugins-and-variables
[37] http://www.goya.com.au/baseelements/manual/507-server-fmgo-runtime-and-web-compatible-script-steps
[38] http://www.goya.com.au/baseelements/manual/508-smart-find
[39] http://www.goya.com.au/baseelements/manual/509-growl-support-mac
[40] http://www.goya.com.au/baseelements/manual/510-translations
[41] http://www.goya.com.au/baseelements/manual/511-importing-previous-versions
[42] http://www.goya.com.au/baseelements/manual/512-finding-indexed-fields
[43] http://www.goya.com.au/baseelements/manual/513-marking-errors-fixed
[44] http://www.goya.com.au/baseelements/manual/514-highlighting-items
[45] http://www.goya.com.au/baseelements/manual/515-copy-clipboard
[46] http://www.goya.com.au/baseelements/manual/60-customising-and-extending-baseelements
[47] http://www.goya.com.au/baseelements/manual/61-modifiable-layouts-and-scripts
[48] http://www.goya.com.au/baseelements/manual/62-integration-other-solutions
[49] http://www.goya.com.au/baseelements/manual/63-automating-imports
[50] http://www.goya.com.au/baseelements/manual/70-purchasing
[51] http://www.goya.com.au/baseelements/manual/71-licence-options
[52] http://www.goya.com.au/baseelements/manual/72-how-buy
[53] http://www.goya.com.au/baseelements/manual/73-registration-codes
[54] http://www.goya.com.au/baseelements/manual/80-support
[55] http://www.filemaker.com/products/filemaker-pro/pro-12-specifications.html
[56] http://www.goya.com.au/baseelements/plugin
[57] http://www.seedcode.com/cp-app/ste_cat/fmsearchresults
[58] http://www.seedcode.com/
[59] http://help.filemaker.com/app/answers/detail/a_id/7734
[60] http://help.filemaker.com/app/answers/detail/a_id/7748
[61] http://help.filemaker.com/app/answers/detail/a_id/7743
[62] http://help.filemaker.com/app/answers/detail/a_id/7733
[63] http://www.filemaker.com/products/filemaker-go/docs/fmgo_development.pdf
[64] http://growl.info/
[65] http://www.banks.id.au/software/fmgrowler/
[66] http://www.goya.com.au/VersionHistory
[67] http://www.goya.com.au/blog/using-name-value-pairs-parameters-and-storing-data
[68] http://www.goya.com.au/blog/accessing-baseelements-records-other-solutions
[69] http://www.goya.com.au/purchase
[70] http://baseelements.goya.com.au/