

Low impact startup

Overview

As a best practice, for the sake of both network efficiency and enhanced security, it is suggested you create one (1) empty table with no fields and one (1) blank layout and name them such that it identifies the table and layout as the file startup elements.

Background

FileMaker, by nature of being a fully encompassing database platform has made it easy for developers to quickly create solutions to everyday business problems with less development investment as compared to other technologies such as MySQL and PHP based solutions. While this is an incredible benefit, it also opens the door for coding practices that are resource intensive. The "Low Impact Startup" best practice is preventative medicine for the developer before such issues arise.

The fundamental issue is the difference (and implied separation) between the client UI and the data structures used to support the collection of data and application of relevant business rules on both the data and UI. With careful practice a developer can separate these two topics to provide excellent user experience while maintaining the ability of a FileMaker server to process and serve the data used to populate it in a timely manner.

Implementation

The blank layout should be associated to the empty table, and should be the selected layout for the **Switch to layout:** check box within the **File > File Options...** settings.

Using the **Perform script:** setting, within this dialog, is the preferred method for arriving at any default target layout for solution startup.

Both the Table and Layout should be the topmost listed items in **Manage > Database > Tables** and **Manage > Layouts**.

Reasoning

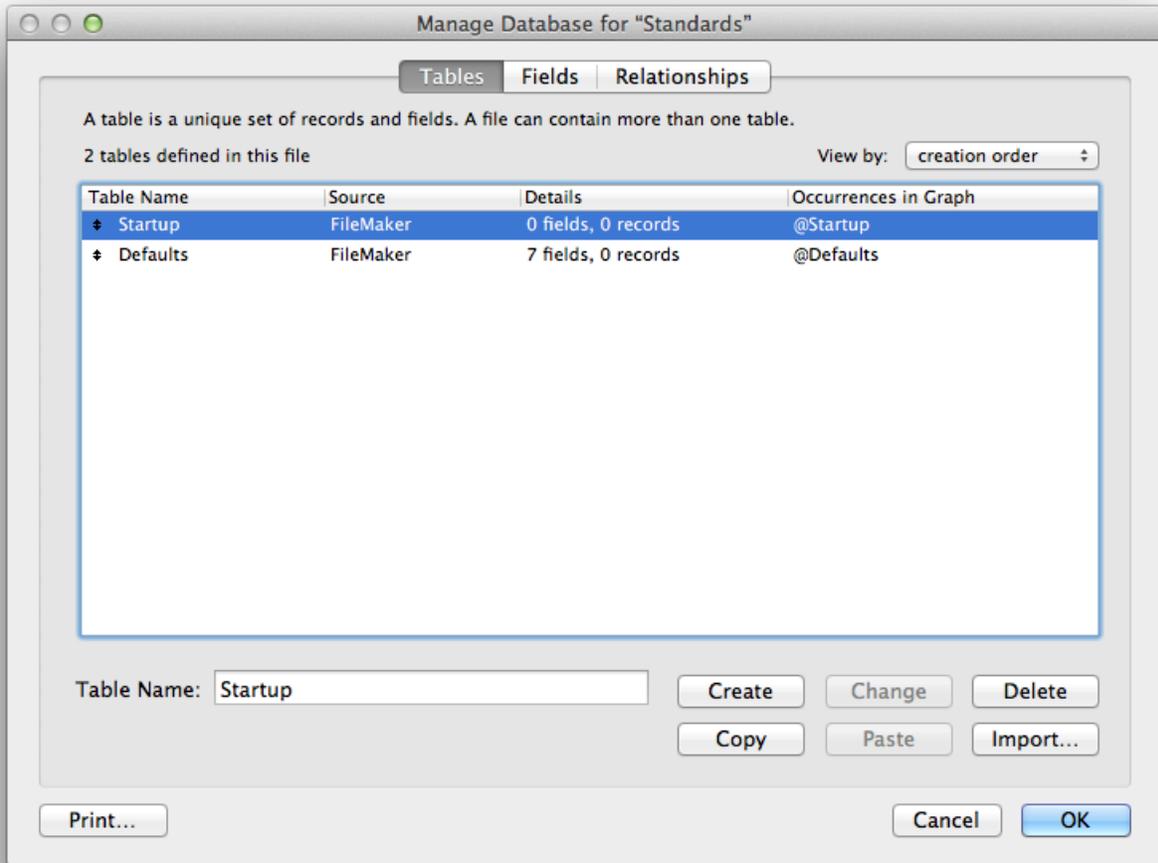
Reduced startup overhead is realized when the layout is ensured to be local to the file (meaning it is not accidentally or intentionally associated to any external files). This also ensures the file is opened to a known location with a low load impact, because graphics and other elements are not loaded until navigation takes the user to the target layout based on the client being used (FileMaker client vs. FileMaker Go).

Prevents the accidental last layout issue because FileMaker remembers the last viewed layout when the file was closed as a locally opened file. This means the file, when opened over the network, without the **Switch to layout** setting set, would default to the last viewed layout.

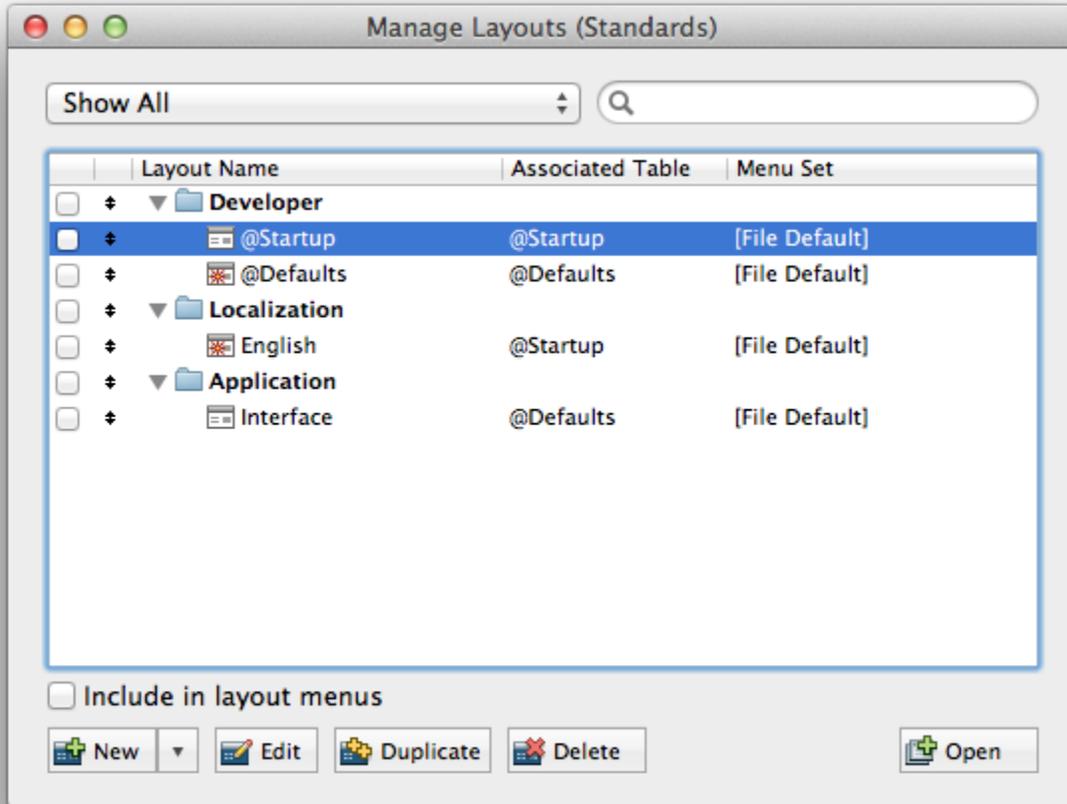
Avoids accidental External Data Sources from being opened if the last viewed layout was based on an external data source.

Provides supplemental security to the file, in that the startup layout and table contain no fields or elements which are immediately accessible.

Empty Startup Table



Blank Startup Layout



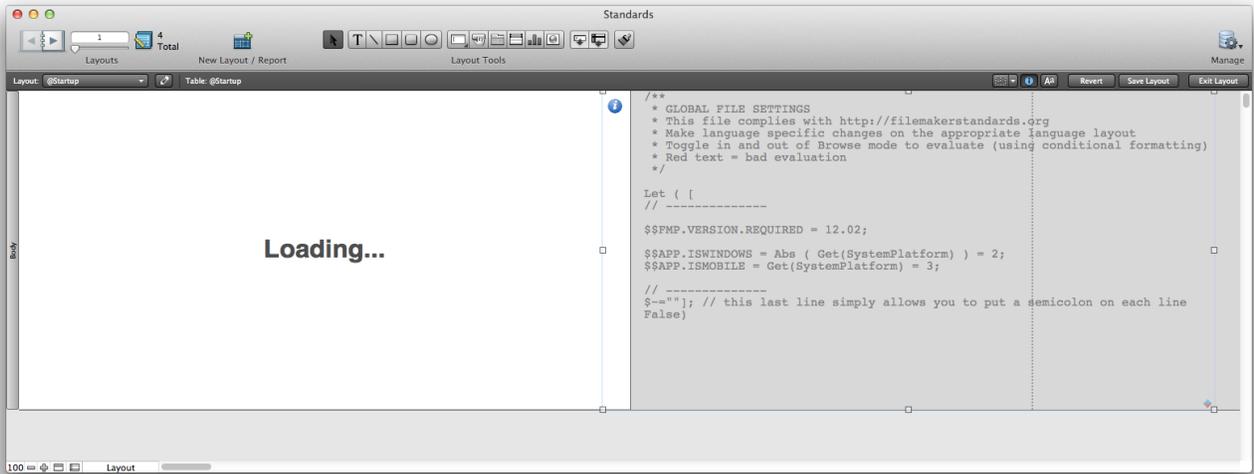
Startup Variables

Using a simple text object on the Startup layout can optionally be used to load data into global variables. By utilizing FileMaker's conditional formatting on this text block, any number of global variables can be initialized. The biggest advantage to this implementation is centralized initialization of global variables - instead of potentially be spread across multiple startup subscripits. The conditional formatting code reads as follows.

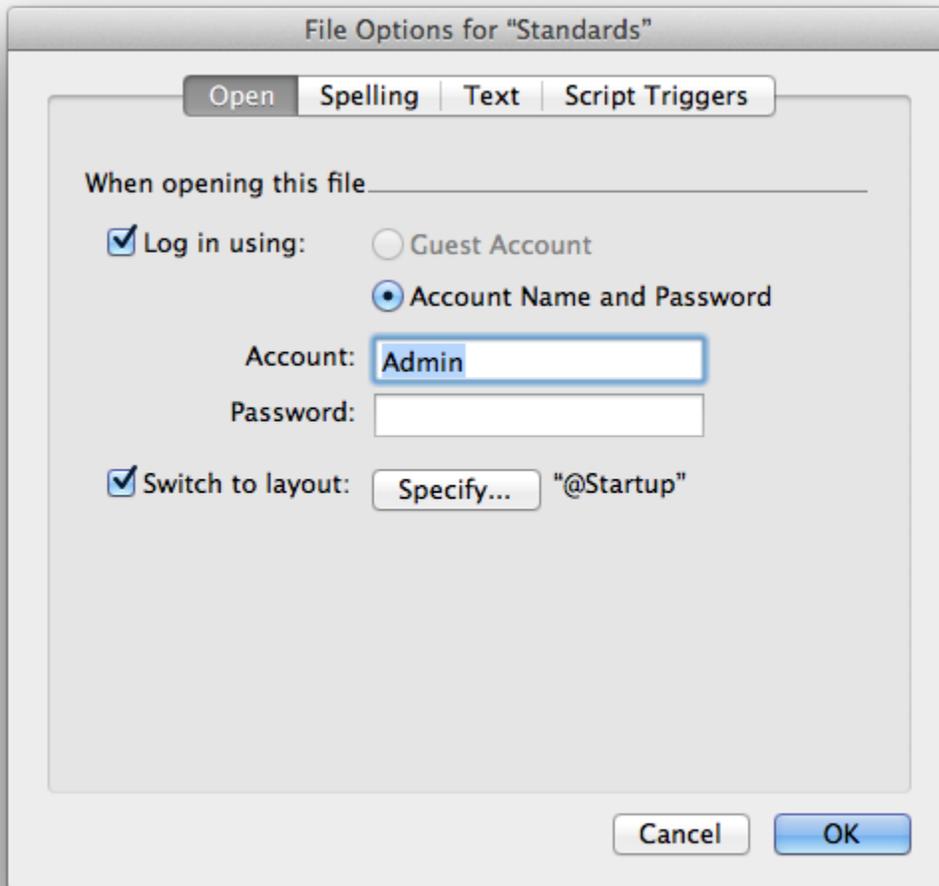
```
EvaluationError ( Evaluate ( Self ) )
```

The text block is preferably situated in the right hand portion of the layout and anchored to the right. By providing a left based indent on the text block, a small portion of the text block should transverse the layout's right most boundary. This will ensure the evaluation of the conditional formatting.

See the [Standards Template File](#) for FileMaker 12



Startup File Options



By using the low impact startup layout within the File's options, the file is always guaranteed to open with minimal impact. It is loading a table with no fields or data. Using the Script Triggers within the File's Options and a dedicated startup script is the preferred method for targeting the proper initial layout for the solution.