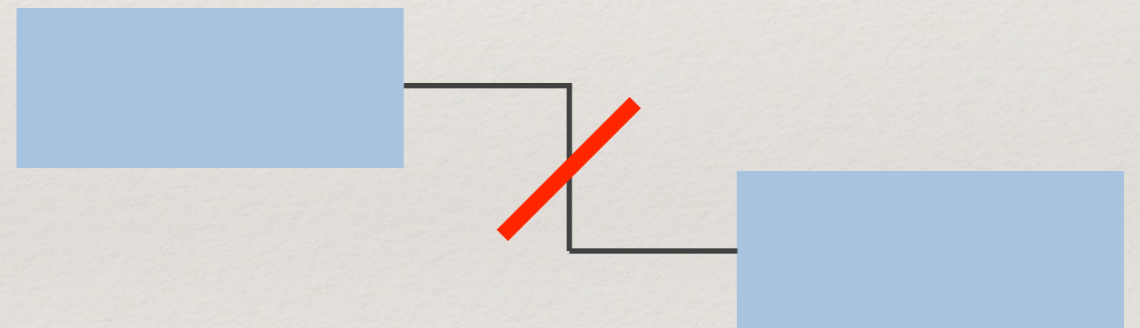


Using the

“Virtual List” Technique

Anything is Possible
When You're **Not in a Relationship**



David Hollander
Combined Fields Consulting

Philly FileMaker Developers Group — February 2015

“Virtual List”: Agenda

1. **Contrast** native relationships to a virtual approach
2. See some **uses** of virtual lists
3. **Dive** “under the hood” of a simple demo database
4. **Eat** pizza

“Virtual List” Technique

- ❖ Neat way to display data in ways that standard FileMaker methods can't naturally do
 - ❖ “situational UI trick”, not a framework or rule
- ❖ Good example of thinking “outside the box”
- ❖ Chance to explore FileMaker's ways of doing things
- ❖ Genius: concept credited to Bruce Robertson

Virtual List: Situations

- ❖ Display data from **more than one table** in the same portal
- ❖ Display data from **different tables at different times** in the same portal
- ❖ Display a set of “potential” records **before** they’re created (or deleted) in their actual table
- ❖ An alternative to a standard **conditional value list**
- ❖ **Control** the number of rows on a report
- ❖ Keep the Relationships Graph **simple** by relying instead on scripting and variables

“Virtual”

“Very close to being something
without actually being it”

<http://www.merriam-webster.com/dictionary/virtual>

“Virtual”

“Going *outside* the **normal database schema**
to accomplish some goal”

– *me*

Why would one want to “bypass”
the normal database schema?

– when it limits your abilities

Normal Database Schema

- ❖ **Structure:** all those elements in a FM Database Design Report

Overview	
Tables	13
Relationships	40
Layouts	13
Scripts	22
Value Lists	0
Custom Functions	5
Accounts	10
Privilege Sets	3
Extended Privileges	7
File Access (in / out)	0 / 0
FileMaker Data Sources	2
ODBC Data Sources	0
Custom Menu Sets	2
Custom Menus	33

- ❖ **Relationships Graph:** the path from one entity to another

Schema in FileMaker

- ❖ **Relationships** are the “heart” of a relational db
- ❖ But relationships in FileMaker have **issues**:
 - ❖ multiple Table Occurrences of the same Base Table
 - ❖ Limit of only one T.O. **context per layout**
 - ❖ Limit of only one T.O. **relationship per portal**
 - ❖ And no dynamic way to **switch** T.O.’s

Relationship Problems

- ❖ “Match field pairs”
 - ❖ **simple** (single pair with equal values)
 - ❖ **complex** (multiple pairs, or comparing fields using non-equal operators, or using calculated match fields)
- ❖ To match successfully
 - ❖ “AND” each pair together — each must find its match
 - ❖ “right side” (target) of a relationship must be **indexed**

Relationship Problems

- ❖ Join fails if a match field is an **unindexed** field
 - ❖ target match fields can't be container, summary, global, or **unstored calculation** fields
 - ❖ any formula that includes a related field, summary field, global field, or a reference to another unstored calculation is an unstored calculation
- ❖ Relationship **Sort Order** can only use that table's fields

VL to the Rescue!

VL: Three Ingredients

1. Utility table

- ❖ reusable, simple and small
 - ❖ just a few unstored calculation fields = no maintenance

2. Global variable (“\$\$_Data1”)

- ❖ temporarily holds a return-delimited (¶) list “taken” from other tables — *no relationship required to take data*

3. Scripting

- ❖ find and gather data from desired records into global variable


VL Ingredients: Utility Table

- ❖ Very few fields
 - ❖ one indexed **number field** (“RecordNum”: 1, 2, 3...)
 - ❖ **calculation field(s)** which derive value by **parsing** lists (arrays) kept in global variables (making the calculation **unstored**)
 - = GetValue (\$\$_Data1 ; RecordNum)
- ❖ Finite number of total records (100 might do)
 - ❖ No adds, deletes, or direct edits of records

VL Ingredients: Parsing a List

\$\$_Data1 (an array)

Global variable with three values



John Doe	123 Main Street	Philadelphia PA
Mickey Mouse	19 Magic Castle	Anaheim CA
1928	“Steamboat Willie”	\$4,986 (estimated)

\$\$_Data1 gets displayed by parsing each row into a record of the VL Utility table

DisplayText1 = GetValue (\$\$_Data1 ; RecordNum)

Utility Table with two fields, 100 records

RecordNum	DisplayText1
1	John Doe 123 Main Street Philadelphia PA
2	Mickey Mouse 19 Magic Castle Anaheim CA
3	1928 “Steamboat Willie” \$4,986 (estimated)
4	



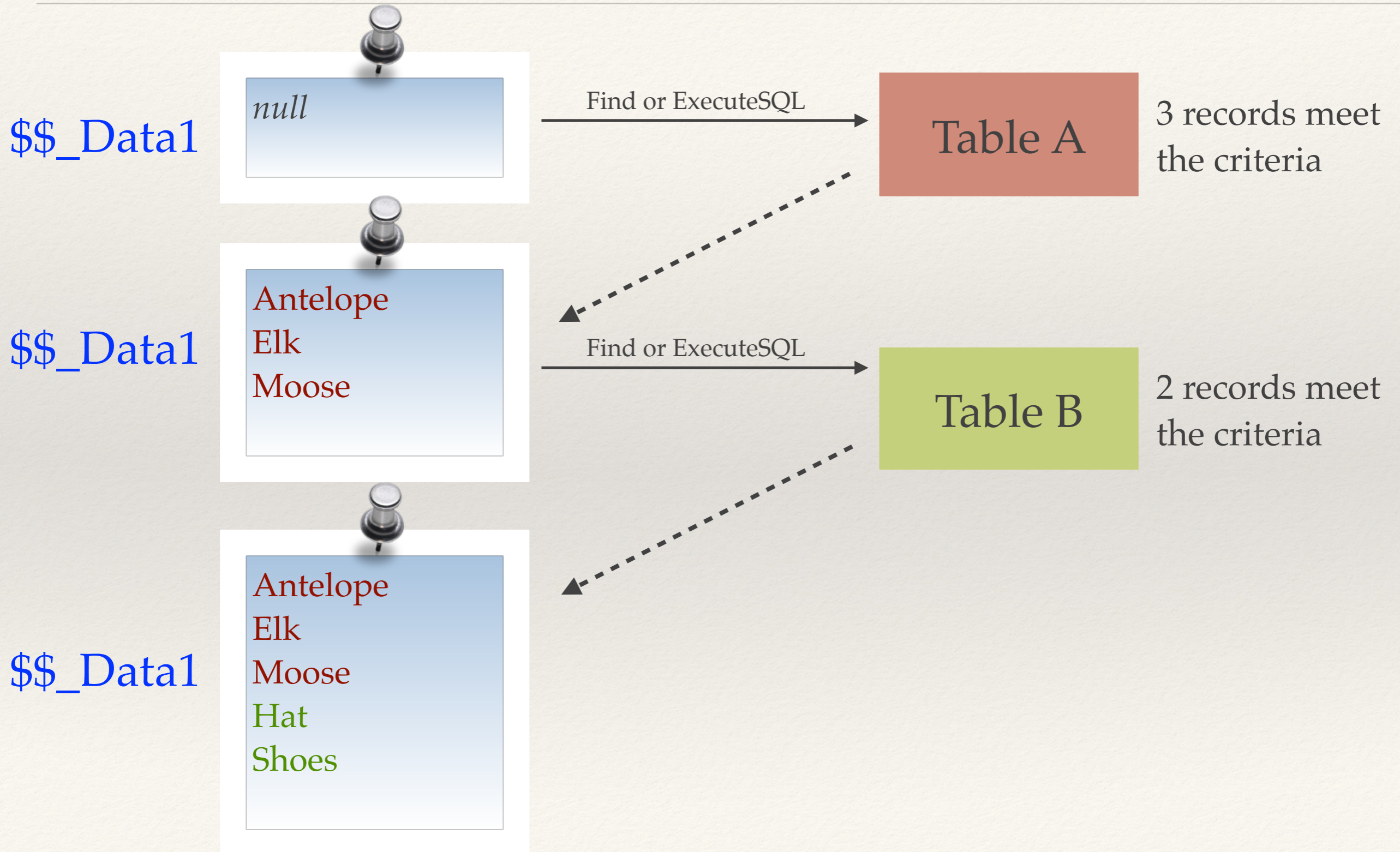
VL Ingredients: List into Table

- ❖ Substituted using a \$\$_variable as a “data source”
 - ❖ \$\$_var in place of a real table: it’s a “virtual table”
 - ❖ each row / value could be completely disconnected
 - ❖ it’s up to you to remember the order of the contents
 - ❖ no discrete fields, types, or easy formatting options

VL Ingredients: Table(s) into List

- ❖ How does the \$\$_var get populated in the first place?
 - ❖ decide what needs to be retrieved
 - ❖ build a list of the desired data, from whatever table(s) they reside, adding each “set” as a row to \$\$_var
- ❖ Methods to find, sort and retrieve data
 - ❖ **Find Request(s)**: requires layout context and looping
 - ❖ SELECT statement in **ExecuteSQL**

Building the Global Variable



One Relationship to a VL

- ❖ To control the number of VL rows to display
 - ❖ Out of all the “n” records, show this many
- ❖ Needs just **one global Number field**, either:
 - ❖ Calculation, **ValueCount (\$\$_var)** or
 - ❖ After Find, “Set Field” to **Get (FoundCount)**

A Portal View of the VL Table

Global variable with three values

`$$_Data1` (an array)

Utility Table with two fields, 100 records

RecordNum	DisplayText1
1	John Doe 123 Main Street Philadelphia PA
2	Mickey Mouse 19 Magic Castle Anaheim CA
3	1928 "Steamboat Willie" \$4,986 (estimated)
4	
5	
... to 100	

global field
`RowsToShow`

`3` \geq `RecordNum`

Portal on some layout

John Doe	123 Main Street	Philadelphia PA
Mickey Mouse	19 Magic Castle	Anaheim CA
1928	"Steamboat Willie"	\$4,896 (estimated)

VL Table: Downsides

- ❖ “~~Go to Related Record~~” step is out
 - ❖ must script navigation to the “real” record
- ❖ Effort required to **script everything**
- ❖ Manually managing the **appearance** of displayed data
 - ❖ inserting separation between fields / columns
 - ❖ \$\$_var is only type Text: you need to format number, date, time values
- ❖ Any user **interaction** with VL rows **requires maintenance**
- ❖ Global variables **not shared across multiple files**
- ❖ Needing to see past “n” VL rows because “n” **was set too low**

VL Table: Upsides

- ❖ Modifications **don't** require changing DB schema
 - ❖ usually just modify scripting
- ❖ Can avoid Relationships Graph **T.O. clutter**

Four Examples and a Demo

- ❖ “Google-like” search of different entities, same portal
- ❖ Calendar using unstored Date field[7]
- ❖ Portal display of matching names instead of a value list
- ❖ Another conditional value list, with conditional formatting