

Syncing FileMaker Systems

Presenter Katherine Russel from Nightwing

author of FileMaker Sync Guide <https://community.filemaker.com/docs/DOC-2795>

originally published in 2012

some updates for more recent versions, but mostly intact

What is sync?

- consolidation of divergent data sets
- sync actions
 - Add record exchange
 - Edit record processing
 - Delete record deletion

Who syncs?

- service personnel
- sales people
- health care professionals

How to sync?

- 3rd-party tools
 - FM EasySync
 - GoZync
 - MirrorSync
 - SyncServerPro
- roll your own

Why build your own?

- to customize or optimize the sync process
- increased control of rules and implementation, including security
 - version control & self-updating
- reduce dependencies or reliance on 3rd parties
- because FileMaker can exchange and process records very effectively

Business rules

- & sync requirements
 - Exchange upload and/or download
 - Modification field-level updates
 - Deletion if applicable
- & process mapping
 - where are the files located?
 - requirements for moving records
 - logic for decision-making
 - resolution of conflicts (which edits win?)
 - connectivity & transactional issues

If remote users...	Clear sync process should:
Read records from the host	Download records
Create new records	Upload new records to the host
Modify records and take precautions over conflicting records on the host	Edit or replace existing hosted records

If remote users...	Clear sync process should:
Modify records and do not take precautions over conflicting records on the host	Modify records directly into a FileMaker host and not download or upload records
Create records	Create records with caution (relationships that are needed from the host sync relationship, where the records exist in other systems, and the data)

Timestamps & Unique IDs

Nightwing uses a base36 UUID custom function

File Management

use a utility file to perform the sync

avoids connectivity problems, so that the local file (which is initiating the sync) has a EDS ref to the hosted file, but no other dependencies

If the hosted file...	The hosted file will open if...	The hosted file will close if...
Has an external data source (e.g., a table in a hosted file)	Open the database that contains the record (or open the record's parent record)	Close the database that contains the record (or close the record's parent record)
Has an external data source (e.g., a table in a hosted file)	Follow action or any other steps that are needed to open the record (e.g., open the record's parent record)	Follow action or any other steps that are needed to close the record (e.g., close the record's parent record)

If the hosted file...	The hosted file will open if...	The hosted file will close if...
Has an external data source (e.g., a table in a hosted file)	Open the database that contains the record (or open the record's parent record)	Close the database that contains the record (or close the record's parent record)
Has an external data source (e.g., a table in a hosted file)	Follow action or any other steps that are needed to open the record (e.g., open the record's parent record)	Follow action or any other steps that are needed to close the record (e.g., close the record's parent record)

Connectivity issues

- causes
 - hosted file unavailable
 - network connection lost
 - FMG or iPhone hibernates/sleeps
- what to do?
 - notify the user
 - construct your process so that it is TRANSACTIONAL

Transactional Integrity

- use relationships from a single context to update multiple records BEFORE committing the current record
- if something goes wrong, can never the current record and ALL child records will also be reverted

Logging while syncing

- Log key steps and iterations in the sync process, including locked records
 - key fields
 - unique IDs of records you're working with
- Use the log as a resource for troubleshooting and sync management
 - identify where the sync halted and resume sync in case of loss of connectivity
 - review locked records
- Use the log to address requirements for audit of the sync process

Building a sync process

- sync construction
 - identifying requirements & business rules
 - map the process flow based on business rules and file locations
 - build scripts (and a utility file if needed) to manage the sync process
 - integrate sync start, processing, and error handling into existing files if needed
 - test and deploy
- main tasks
 - initiate sync - performed by local file
 - exports and opens utility file if needed
 - sync script loops through tables - performed by utility file
 - identifying records to be added, modified, or deleted
 - importing records or looping through them
 - looping through fields, if needed
 - deleting records if needed
 - cleanup - performed by local file
 - close utility file
 - close hosted file
 - local file deletes utility file (if needed)
 - report results
- minimize hard-coding
 - design functions
 - "TableNames()"
 - "FieldNames()"
 - name variables dynamically
 - "Evaluate ("Let(\$\$" & YourVarName & " = " & YourVarValue & "; \"")")"
 - use "Set Field by Name" script step
 - "ExecuteSQL()" to identify records meeting sync criteria
 - but this can have performance implications
 - Managing CUD via utility relationships

Demo

(see session recording for video)

Sync script (in utility file)

- Error Capture On & Allow Abort Off
 - some controversy about exactly where to put it
- creates log entries as it goes, but you can only see it in the demo because it's using Commit Records in a lot of places
 - you DON'T WANT to Commit Records until your transaction is complete, so DO NOT use that method outside of troubleshooting
- timestamp comparison
 - is from last SUCCESSFUL sync's START timestamp (not the FINISH timestamp)
 - because any records changed/modified while that sync was happening may not have been captured in that sync
- sets a variable with list of IDs that match the criteria
 - uses a relationship based on a timestamp
 - log records
 - cartesian join
 - sorted by timestamp
- portal on your layout does not need to have any fields on it
 - is how you manage deletes
- If users are able to edit directly on host system while you are syncing, may want to implement a locking methodology
 - e.g.,
 - tickling the record to lock it (so hosted user can't lock it)
 - [PREFERRED] use "Open Record" script step
 - will throw an error if record is already locked
 - change a value in a field
 - record-level permissions

My Questions

- What does "Managing creation, modification, and deletion via utility relationships" mean? How does that limit hard-coding?
- What is the purpose of the "Companies_hosted_pivot" table?
 - really, why are all the tables suffixed with "pivot"?

Q & A

- If using local time that is different from hosted time, how do you figure out what the "latest" one is?
 - Multiple fields
 - UTC Milliseconds
 - Get (CurrentHostTimestamp)
 - Have script translate the time
- If there's an error, couldn't you just run the sync again?
 - In most cases, that's an option
 - If your sync is slow, may not be