

## The Co-op Database:

The Excel spreadsheet which the Co-Op uses for their student and family data is a mess. It was built years ago by the old treasurer, but since that member left, the spreadsheet has been taking exorbitant amounts of time from the rest of the group. Very few members understand how to use the spreadsheet, and even fewer can fix it when it breaks. The process of changing one field on one record can take up hours of valuable time.

### The Situation:

So what is this spreadsheet and how does the co-op use it? This spreadsheet stores all the information for every family in the co-op. This includes the names of all the children, and all the classes each individual child is taking. Every year, data is downloaded from Google Forms and manually typed into the database with a macro. As the year goes on, that same macro is used to edit the existing data and update the various schedules, contact lists, and other useful tables for co-op organizers. In a way, the data cannot be said to be stored in one place: the data found in the contact list is not sourced from the data in the “database” in any way. It is its own database, storing its own subset of the data, which receives updates from the same macro which updated all the other databases in the Excel workspace.

### The Problems:

The first problem is that the database does not perform the core function of a database. At the core of the ideal of the database is its function as a Single Source of Truth (SSOT). Any piece of data should be traceable back to a single point of storage. In the Excel sheet, data is loaded into a macro, then various fields are duplicated into various different presentations. This makes the data incredibly fragile and incredibly complex to update and fix.

The second problem with the database is that both the input form and the input macros are inflexible. This is a problem because the number of children in a family is flexible. Every time the standing record for the largest co-op family is broken, the entire database system of macros must be essentially rebuilt from the ground up. This is no bueno.

### My Solution:

The good news is that both of these problems are easily solvable at the database level by building a system around a normalized and multi-table SQL database. My recommendation is to use a cloud platform such as Turso, because it is free and comes with a simple online database editor: LibSQL Studio. The question now comes: how do we get calculate and generate those reports? and do we really have to type everything in manually from the Google Sheet? Is it worth the cost? These are decisions which need to be made.

### The Proof of Concept (WIP)

My current POC consists of a Turso Database as well as a repository of Python scripts to perform all large-volume additions and reads/aggregations from the database. The Co-op currently uses Google Forms for their registration system, so I will write a Python script which, along with some changes to the Google Form, will allow the administrators to use a familiar system, and also automatically load that data, as opposed to the old system of re-typing all the Google Form Responses. Since Registration restarts every year, I will also build a system facilitating local back-ups of the current database and for wiping that database. The biggest challenge is going to be making the application accessible for use by normal people, so I plan to write extensive and easy-to-understand documentation for these scripts as well.