Contacts Organiser

# What

Contacts organiser is an online program designed to allow cross provider syncing of email contacts. It allows for syncing, tidying and automatic contact updates with a built in scheduler service.

# Why

Syncing contacts across platforms remains a relatively unsolved problem, current solutions tend to have a different business model, often this model involves using a contacts email address and phone number to scrape as much information about that contact as possible to populate the contacts information fields.

# How

### Overview:

The program is a web application with a python backend. The program utilizes the google gmail api [[2]](https://www.zotero.org/google-docs/?qXfZjj) for gmail syncing and tidying services. The Microsoft Graph api is used for the Office 365 integration [[1]](https://www.zotero.org/google-docs/?OVpONh).

### Tidy Service:

The tidy service follows several constraints when ‘tidying’ contacts. This service does not tidy all contacts correctly, sometimes duplicates can pass through unnoticed.

### Sync Service:

The sync service is responsible for syncing contacts from one account to another. It is a one way service meaning it will sync from a master account to a slave account, this is a user controlled event allowing for any account to be master or slave. Accounts can be the same service provider or cross provider for example; you can sync two gmail accounts, you can sync a gmail account and an office 365 account and vise versa.

Syncing will also take into account two contacts of the same account, it will keep the information of both parties but will destroy duplicate information. For example if the master and slaves accounts hold a contact called John Doe but one account only contains Johns email address and the other contains email address, phone number and home address as long as the master account holds the additional information it will be included into the slave accounts information.

High level flow chart of user events:

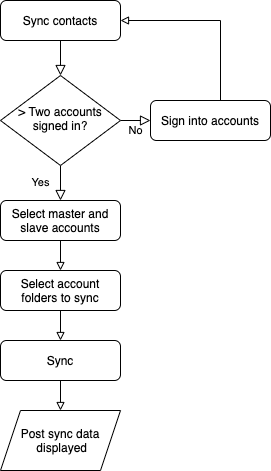


Figure 1

In figure 1, we can see a simplified user event flow chart, as shown before sync can take place at least two accounts need to be signed in.

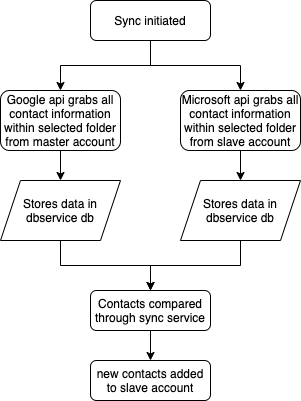


Figure 2, High level view of sync process

Figure 2 shows how the system manages to compare the contacts across services. Note **this service with microsoft office 365 is extremely slow**, the max number of contacts that can be requested through the api at any one time is 10, therefore we have to request 10 contacts, set the current contact number as the next start number and request another 10 until the process iterates through the whole contact list.

Comparison algorithm needs testing. Gmail contacts will sync but there are cases that do not behave as expected.

# Current problems:

**Currently office synchronization contains several bugs:**

The last method in sync\_service class called def sync\_contacts may not assert an error but there are logical errors that cause a failure in the system. This needs investigating.

App.py also contains logical errors, where a decision between office and google accounts is made a try catch is used yet this is not optimal. This needs investigating.

**Tidy:**

Tidy functions but some multiples of contacts can pass through, i.e. duplicate contacts are not filtered. The tidying algorithm could be improved.

[[1] Jackson-Woods. Use the Microsoft Graph API - Microsoft Graph. Retrieved September 6, 2020 from https://docs.microsoft.com/en-us/graph/use-the-api](https://www.zotero.org/google-docs/?neoZiU)

[[2] Gmail API. *Google Developers*. Retrieved September 6, 2020 from https://developers.google.com/gmail/api](https://www.zotero.org/google-docs/?neoZiU)