**Summary**

Addepar lacks many of the features common portfolio management systems, it is necessary to develop some of these features internally. We will begin by developing a reconciliation program. Since Addepar does not have an internal accounting book, this will not be a typical "reconciliation". Instead, "synthetic T-0" position data will be calculated by querying T-1 position data from Addepar and adjusting the holdings in each account with trade data from Flyer Co-Pilot. This will alert the Investment Management team to any issues that occurred while Addepar was loading custodian data or to any errors leading to trades not settling.

Reconciling the holdings daily will require downloading account characteristic data and account holdings data from Addepar, and trade data from Flyer Co-Pilot. This will lay the foundation for developing additional features that rely on this data.

**Value Add**

* Allows for faster recognition when Addepar fails to completely load the custody files
* Confirms trade fills and allocations
* Confirms agreement between Addepar and Flyer Co-Pilot data
* Identifies completed corporate actions
* Facilitates projects that require holdings or account data, including among others:
  + Options collateral tracking project - no point logging collateral at the initiation of a trade if collateral corporate actions are not tracked.
  + Programmatically allocating holdings to strategies at the security level in Addepar - allows for easier reporting on the strategy level.

**Costs**

* Cloud based server - can shut down over night to reduce costs
* MS SQL Server license

**Risks**

* Stores client information on a SQL server
* Data between Addepar and local database can get out of sync leading to investment decisions being made on out of date information
  + Can be mitigated by not providing users access to cached data to make investment decisions - only cached for automated systems
  + Allow data to be updated midday if any discrepancy between local data and Addepar data is discovered

**Prerequisites**

* SQL Server with SSIS Installed

**Requirements for basic functionality**

* Download data daily from Addepar
  + T-1 Addepar position data
  + T-0 Addepar position data
  + Account level descriptive data
* Get data from Co-pilot
  + Flyer Co-pilot trade or allocation file
  + File dropped nightly by Trade team to FTP or file share location
  + Co-Pilot API if integrated
* Load all data to SQL database
* Create a "synthetic" T-0 "projected" holdings using T-1 Addepar position data and updating it with the Co-Pilot trade file
* Compare the "synthetic" T-0 holdings with the actual T-0 holdings and filter out any differences
  + Share differences
  + Positions missing from one file or the other
  + Etc.
* Create a SQL view that can be queried into excel or generate a report and email it
* Runtime
  + Run at 6:30 - 7:00 AM so it's ready when Paul logs in every morning
  + Have the ability to rerun later in the day if necessary

**Future development**

* Integrate with Eikon to project Corporate Actions
* Integration with Options applications for tracking options and collateral - raise alerts if any options or collateral were traded without prior logging
* Programmatic allocation of client holdings into strategy groups
* Internal Security Master based off cached holdings that can be updated periodically

**Flow diagram**

\*Due to the size of data downloaded from Addepar, it is necessary to make use of their "Jobs" API (<https://developers.addepar.com/docs/jobs>). Jobs must be queued in Addepar's system and only downloaded once complete. It is unknown, a priori how long the job will take to run. As a result, the job ID will be stored to the database and not stored as a session object in the SSIS job in case an error occurs before the Addepar job is complete.

* Queue firm holdings Addepar API job - log job ID to database
* Queue accounts Addepar API job - log job ID to database
* … Wait for jobs to complete …
* For each job - once complete:
  + Download the raw API JSON data - save the file
  + Load the data to a dbimport (temp) table as all varchar data
  + Run post-import proc to scrub data and import it to target (final) table
  + Log list of database import errors
* Depending on design of reporting
  + If active reporting - run and/or email report
  + If passive reporting - done. Wait for user to query view of recon exceptions

**Outstanding Questions**

* How do we capture treasury and option trades?
  + Co-pilot allocation file I was provided only has equity trades.  
    [Allocations 11.22.csv](https://lidoadvisors1.sharepoint.com/:x:/s/IM-Ops/EZXcj26_jC5DkxrDqyEXvJ4BZ4pn9_De4ltZs37sUS43rw?e=67GxQo)
  + Ignore this file for now - Paul is developing a new file
    - He will add in options (Cap and Cushion)
  + FI guys don't care about allocations - just exclude from Recon for now
* When to run? Risk of running it too early before position update? 6:00 ET?
  + ~6:30 - 7:00 AM ET
    - Allocation file will be exported EOD
    - Schwab, Fidelity will export holdings to 10:00 PM ET
    - Addepar usually loads in the morning
* Data retention policy?
  + Database data - not the book of record - shouldn't be any regulatory requirements
    - Delete data each time new data is loaded?
    - Retain in database 5 days, 1 month, 1 year, etc.?
  + Raw JSON API files (debugging purposes)
    - 1-2 days?
* Can we email from the SQL server?
* Cache "breaks" in a separate SQL table or always analyze the data / find the breaks in real time?
  + Question will be answered once the project is compete - will depend on how long the data takes to analyze
* One stage ETL (load directly to target table) or two stage ETL (load to dbimport table with all varchar datatype then load to target table with SQL proc
* *TECHNICALLY* shouldn't database CUSIPs and ISINs - requires a license

**Project timeline**

* Create a data view in Addepar for the recon
  + Current view is a Personal view 'Brian\_FA\_Addepar\_Recon\_Test\_Delete20230315'. Can roll into a permanent view (or API view definition) when the project is ready for deployment.
  + Sample output queried by Python  
    [DataViewSample\_20221215.csv](https://lidoadvisors1.sharepoint.com/:x:/s/IM-Ops/EX_MtjDw6ixKtZtyl5GNCiwBEHRpPdTzgEmg6u1TWCq6NA?e=5nSAQk)
  + View definition:  
    [AddeparDataViewDefinition.txt](https://lidoadvisors1.sharepoint.com/:t:/s/IM-Ops/EQoTdzsjxwpBmrYyEjSZMi8B3mHt1Rzan_nbfyvWvlUDkQ?e=Y3C0Qg)
* Create data query / data view for account level data
* Get Addepar API "Job" working in Python
* Create SQL tables
* Create SSIS project to load data
* Create SQL scripts to analyze the data
* Create data view for breaks
* Create method to view data (excel spreadsheet with link to DB, python view, automated CSV report, email, etc.)