Background

The use of data and analytics is fundamental for investment management. It has become industry practice to source data from data provider services. The ability to retain archives allows for a flexible and independent use of such data.

One form of investment data is the stock index. An index is comprised of a curated list of stocks and their accompanying metrics. These metrics vary across providers. There are typically two sets of data for each trading day with a 'snapshot' of the index made at the closing of the market and an adjusted close index which is used the following morning. Typically, such data contains information regarding the company, identification codes for varying platforms and various other metrics relating to the company.

Problem Statement

The aim of this project is to design and implement a solution to mirror, parse and archive data from a rolling window subscription-based service whilst retaining fidelity and robustness.

Details of the rolling window are as follows; with each trading day having approximately 57 files with varying upload times. There is a backlog of 7 days on the server with the oldest files removed from access as new ones are added. Within the 57 files that are released each day there are different file types and contents with each index having files with values on closes and adjusted bases, with other file types relating to corporate actions. The format of the data is typically comma-separated values with differing file extensions.

You are required to upkeep the archive of the most recent files from a sftp server, read and parse the files for its contents and upload to a database. The data of interest is the S&P ASX 300 constituent files on a closed basis. Accuracy of the data from the provider is to be retained. Any changes regarding the contents of the files made by the source whilst they are on the on the server should to be flagged and accounted for, using potential change indicators such as the modified time and the size of the file.

Parsing of the contents of each file and a verification is to be employed to maintain accuracy. The data is then to be uploaded to a database to be able to be used by client applications with varying filter conditions or otherwise directly queries by another application.

Thorough documentation is to be written as a resource both for reference and for review.

Process

- Python as primary development language
- Data to be stored in a relational database
- 2-day time limit for submission

Assessment Criteria

- Design of solution
- Implementation
- Documentation

SFTP connection details

Hostname:

s-025790f19af84c0fa.server.transfer.ap-southeast-2.amazonaws.com

Username:

deuser