

Example File load plan

Thursday, July 11, 2024
1:12 PM

The purpose of this plan is to ingest SomeCompany (OFB) Product data submissions from multiple sources (Stores) with individual schemas.

User Action Description:

- OFB representative (OFBR) will access Sharepoint Website
- OFBR will fill out form and upload file
- OFBR will receive email notice
 - That the file was found
 - That the file was associated with Particular Source
 - File was archived
 - Data had the right number of columns, per Source Schema, once parsed
 - Data was able to fully ingest
 - Data was cleansed
 - Data was incorporated
 - Link to Ingestion report regarding data

LI Processing actions:

- Website logs data from OFBR on Submit
- Submit will trigger Power Automate flow
 - Checks uploaded file size is greater than 30kb?
 - To indicate whether the file has more than just headers
 - If file size is less than minimum
 - send email that file is empty (or nearly empty) and to resubmit with more data
 - Stop processing
 - Checks file naming code matches existing expected file name structure
 - To make sure we can determine correct submission schema
 - If name does not match
 - Send email that file name needs to be corrected and re-submitted
 - Stop processing
 - Append file name with OFBR creds & datetime stamp
 - Timestamp = 202407111530
 - Power Automate copies file to another Sharepoint folder called [OFB Submission Archive]
 - Power Automate moves file to Azure Blob Storage
- Power Automate triggers Azure DataFactory
 - Alternately have any new files in Azure Blob Storage trigger Data Factory
 - Per code in file name, choose processing path
 - Ingest single, first, line of file
 - To check if delimiter is consistent with expectations
 - To check if the number of headers match
 - To check if header fields are in correct order for Schema
 - If either of the above doesn't match

- Send email that file code is not matching schema representative of code
 - Stop Processing
 - Ingest full file into staging table in Sql database - line by line
 - Validate that there are not extra or missing delimiters in each line
 - Validate that data within each column matches expected data types
 - Validate that there are no weird or unusual characters in columns
 - Check for SQL injection
 - For records that meet validation
 - Add to Staging table
 - Append filename, OFBR details, Datetime stamp, matching code ID
 - For matching code ID, join with file code naming table and insert Matching Code ID
 - For records (lines) that do not meet validation
 - Put them in exception table
 - Add Validation error type, record number (from dataset)
 - At end of ingestion
 - Summarize ingestion table
 - Number of records
 - Number of records accepted
 - Summarize exception table for all records meeting Filename
 - Get number of exception records
 - Transform staged data to align with final data expectations
 - This will be done per submission schema
 - Pivot and map data columns to final table alignment
 - Join data to standardize/cleanse
 - Need to know what to do with non-standard data (including blanks ("" or nulls)
 - Drop the record?
 - Move the record to Exceptions table?
 - Put in "Other" Tag?
 - Ignore?
 - Other?
 - Trim
 - Remove or replace data (per specific, standard requests)
 - Append records to final sql tables
 - Run any post ingestion processes
- Send Email to OFBR
 - Summarize
 - Number of records in file
 - Number of records accepted
 - Number of records not accepted
 - Link to Power BI Ingestion report for details
 - Include Filename to reference in report
 - Recommendations on how to correct issues for records not accepted
 - Resubmission

LI Necessary Objects

- Sharepoint website Form
 - External Facing Sharepoint
 - Ms Forms to select local files
 - Folder that lands the files
- Power Automate Flows
- Power BI Ingestion report
 - Include summary of ingestion
 - Number of records ingested
 - Breakdown by Product Type (or other)
 - Include summary of Errors
 - Number of records
 - Group by type of exception
 - Give access to details of exception records
 - Give access to details of accepted records
- Sql Database
 - Exception table
 - Datetime stamp
 - Filename
 - Error Exception type
 - Original file line number
 - Record Data (in single line with delimiters)
 - Pipe delimited (|)
 - Log Table
 - Datetime stamp
 - Filename
 - Total record count
 - Accepted record count
 - Exception record Count
 - OFBR details
 - Ingestion Staging Tables
 - Need to be created per ingestion schema format
 - Must be consistent

Questions:

- What type of file submissions are acceptable?

Concerns:

- How do we avoid processing the file 2x?
 - Can Morgan manage?
- How do we handle exception records
 - Can the user just fix the records and reprocess the file without causing duplication?
- How do we handle duplicate records within a single file?
 - Options:
 - Auto put duplicates into exception table
 - Provide list of duplicate records for user to approve/deny

- Drop duplicate records
 - On the submitter to make sure no duplicates are included
- Need to have validation across all submissions
 - Limit record totals to 3000(?)
 - Other?

<<OFB File Load Plan(1).vsdx>>

<<OFB File Load Plan-1.pdf>>

Logic Intelligence OFB Submission

Joan Crawford
Joan.Crawford@OFB.com

Da

Select File to submit*

Gorge tricounty donation file.csv

[Browse](#)

Submit

* Required for submission



