Katz

Katz School of Science and Health

Donors Data Warehouse

ETL Documentation

Version 1.0

Prepared by Jesus Olivera

Table of Content

1. Introduction

ETL broadly refers to the process that extract data from source systems, transform the data to fit into the schema of the destination systems, and load the data into the target systems. For an intuitive read, when this document refers to ETL the document refers to the general extract, transform and load process pattern.

The ETL process includes data validation, process logging, authentication, exception handling, etc. The information in this document applies to all donors' data warehouse architecture. For simplicity when this document refers to the donors dw the document refers to the donors' data warehouse.

2. ETL Donors DW Architecture

The donors dw uses Phyton programing language ecosystem of modules and code libraries to access, transform and load data from source to location. The ETL process is a mix of pure Python code and externally defined functions or objects.

The ETL process extract data from a local folder and send that data to Jupyter Notebooks for ingestion into the pipeline.

The donors dw data transformed by the ETL processes include:

- Business data encoded in a CSV format
- Median income and home value data from the US Census Bureau encoded in JavaScript Object Notation (JSON)

ETL Overview

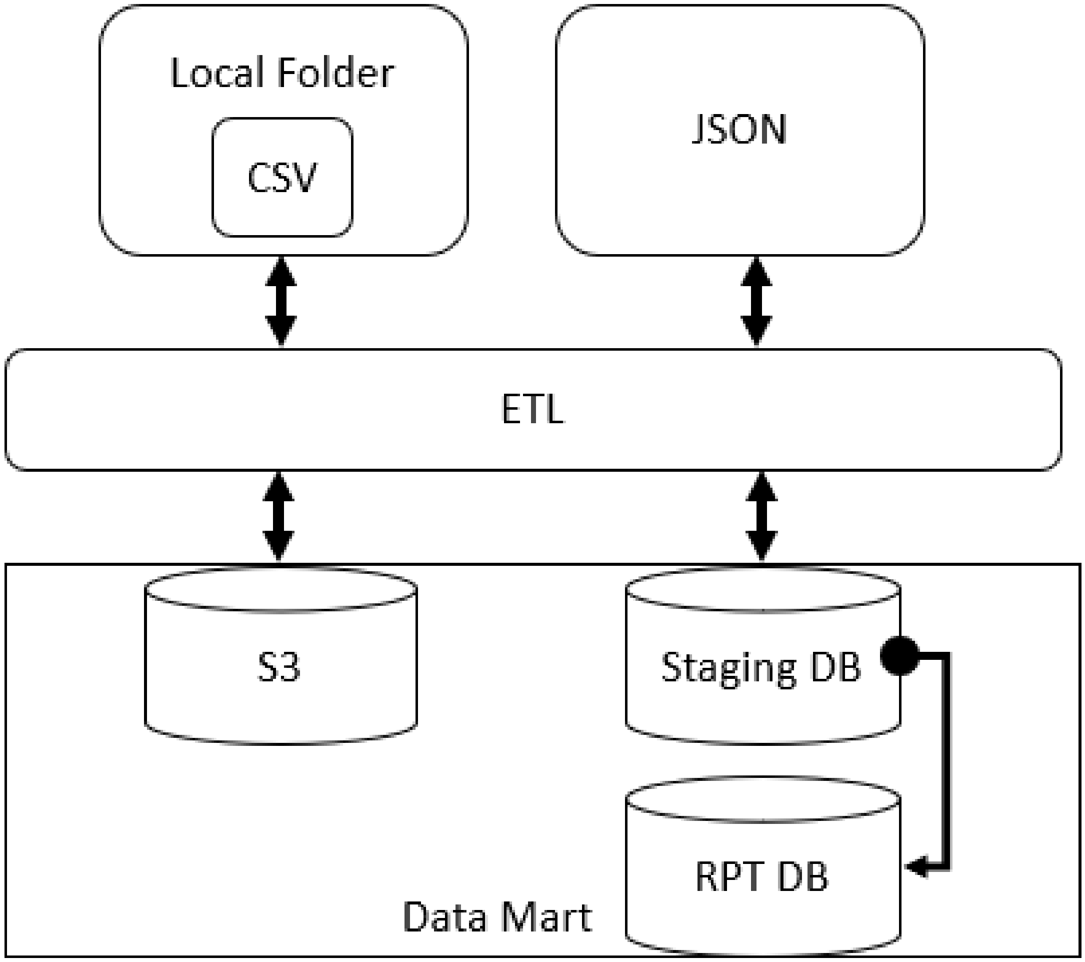


Figure 1. donors dw ETL overview

3. ETL Pattern

With the E-T-L pattern, the ELT process stages and transforms the data and loads it in an intermediate database (not the destination database) before uploading the data to the destination.

The ETL below (Figure 2. ETL steps) displays the steps to extract data from the source and load the data to the staging tables. Data transformation occurs according to the business rules of the process before and within the stagging tables.

Transformed data is loaded into the destination tables and the process is completed.

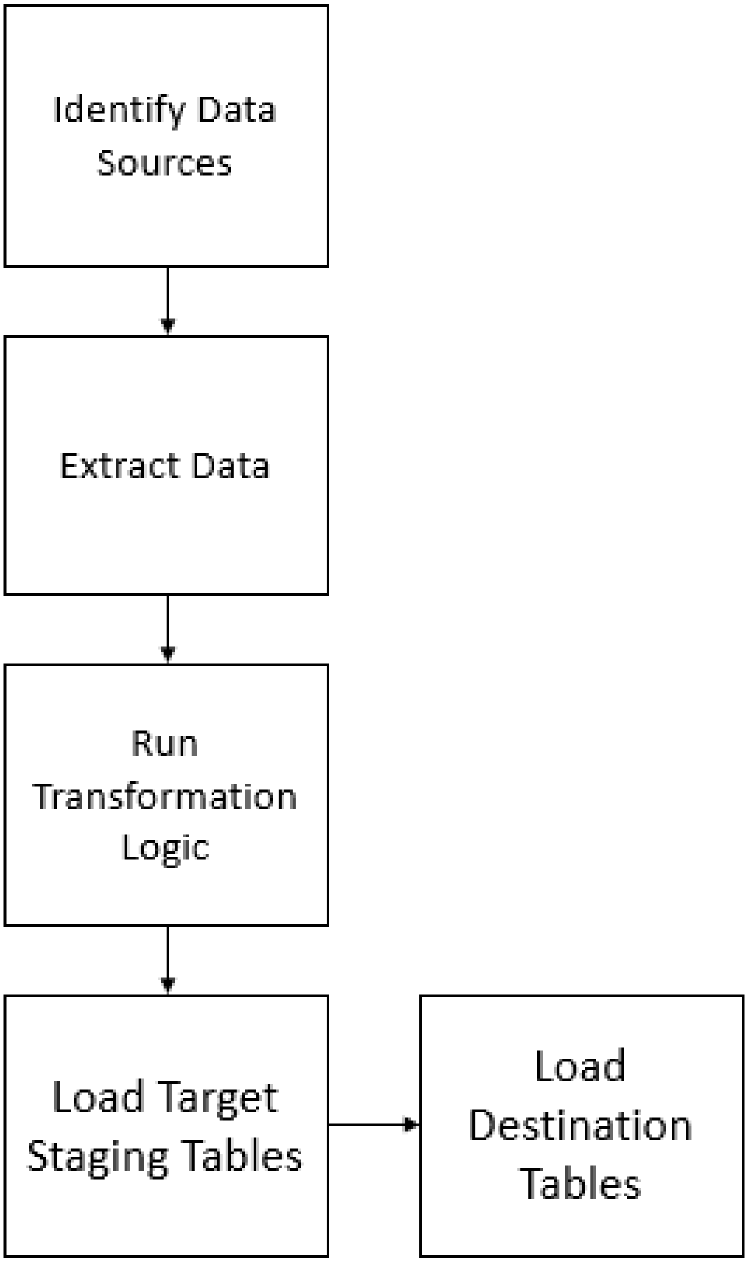


Figure 2. ETL Steps

4. ETL Process Setup

Failure response to scenarios

Exception handling scenarios include the following:

- Detect an error, stop the process, and present the error code

Log Exceptions:

- Version 1.0 of donors dw does not automatically record errors in a log.
- All errors and potential errors will be assigned unique error codes and description.
- All error and potential errors will be entered manually in an error log.

Changes to the ETL process will adhere to the design framework and will be subjected to version and configuration management. Examples of changes affecting the ETL process include:

- File layout changes
- Database schema changes
- Addition or removal of data elements

5. Direct Database Connections

The designing team provides the following connection details:

- Database host
- Database name
- Port
- Service ID
- Password
- Appropriate drivers

- For the source and target system, the database access will be read and write.
- The database connection details will be setup in the configuration of the ETL script and will under no circumstances be changed at any time without management and ETL designer's approval.
- The table structures of the databases will be managed by the database administrator.

6. File-Based Exchanges

The ETL process uses files as input or produce files as output to its processing. File transfers in the donors dw conform to the standards described by the business requirements.

In addition to file format, the structure of fields in the file is also agreed upon.

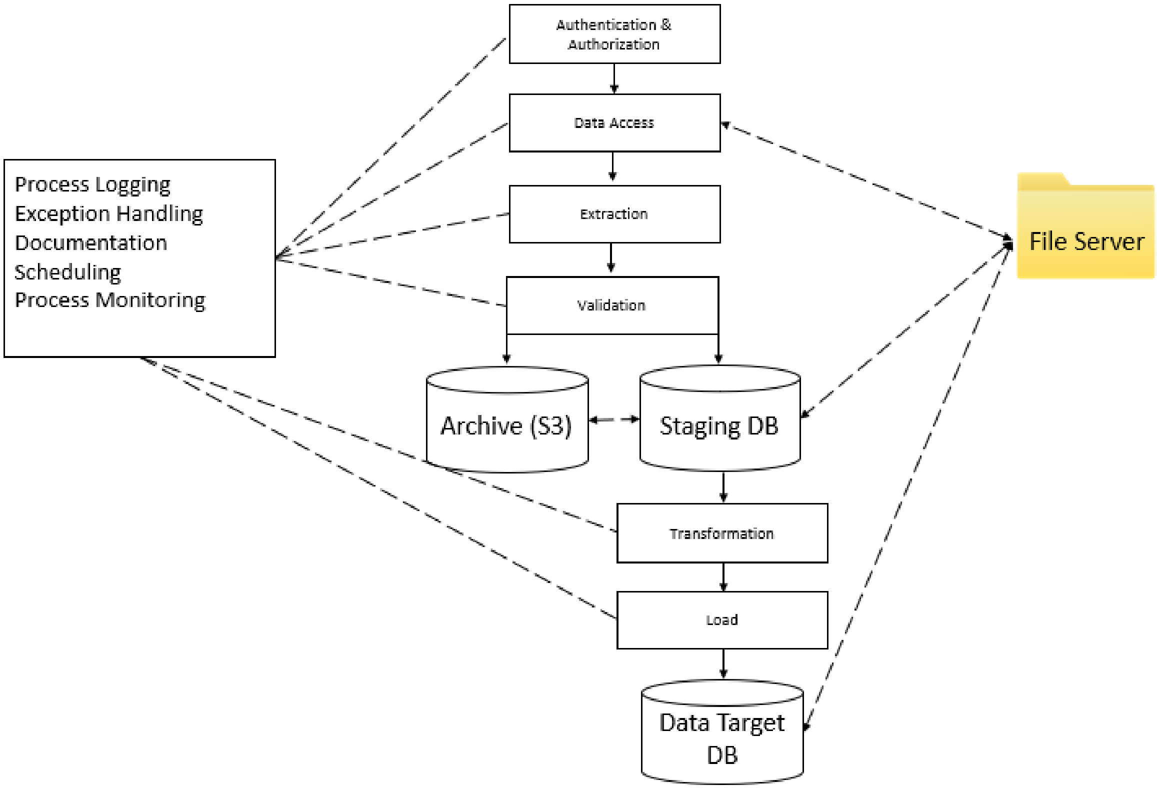
7. ETL Process Standard

The ETL process steps may overlap and repeat several times within the execution of a business process. For example, an ETL process may extract source data, transform the data, load data into staging tables for cleaning, and then transformed the data to load data to final destination table.

The ETL process includes the following structure and considerations:

- Process Steps:
  - Authentication and authorization
  - Data access
  - Data extract
  - Data validation
  - Data transformation
  - Data load
- Operational Considerations:
  - High availability
  - Scheduling
  - Monitoring

Figure 3. Detailed ETL Process



7.1 Authentication and Authorization

Authentication refers to the process by which the username and password (or authentication token) is validated. Authorization is the process by which the donors dw scripts determines which activities the logged in user can perform.

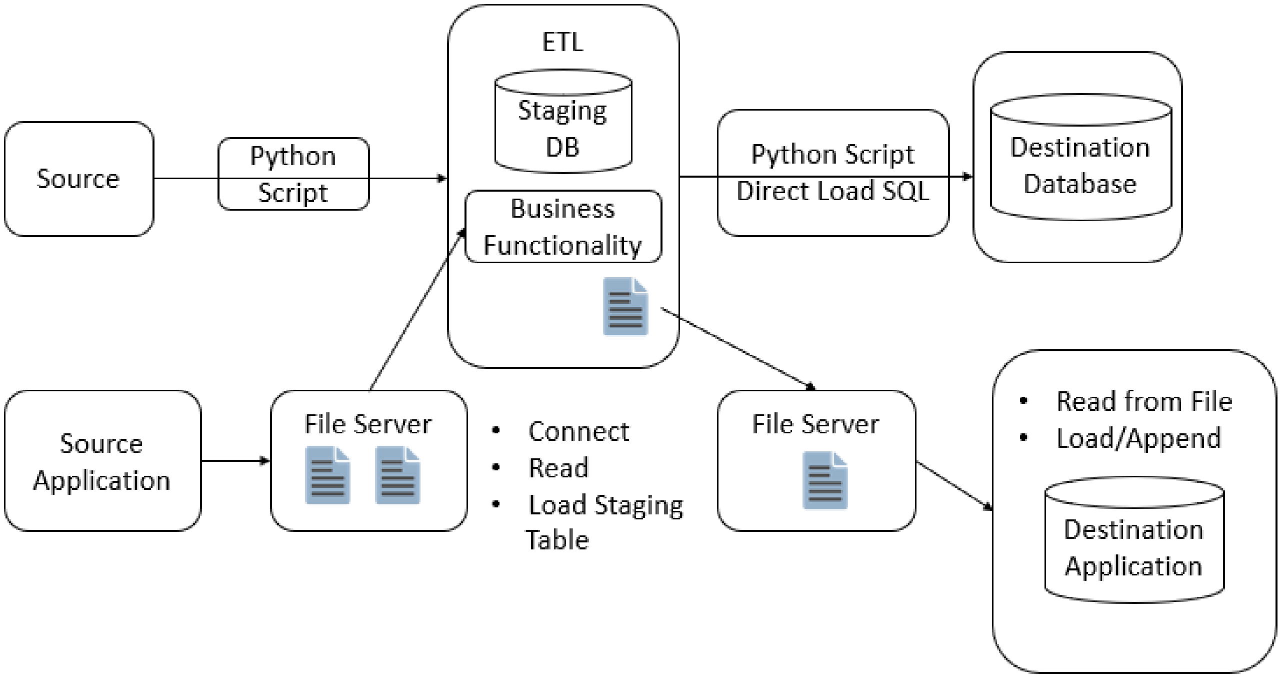
During the execution of ETL process activities, the ETL processes access files, databases, and web services (referred to collectively as end-points). Authentication and access to end-points are performed.

7.2 Data Access

Before extracting data from a source or loading data into the target, the ETL process needs to connect to the data source/target. ETL processes in the donors dw architecture supports the following data stores:

- Structured files (csv)
- JSON

Figure 4. donors dw ETL connection architecture



- Each source system ensures that the ETL process has access to the source data.
- The ETL script extracts data from the source location without updating any data. No other updates will be made to the source location during the read process.

7.3 Direct Database Exchanges

The ETL Python script connects to the data source directly using a local path provided by the user. The ETL process identifies the file to import based on date and file name.

7.4 File-Based Exchanges

In addition to the file format, the sequence of fields should conform to the pre-defined structure agreed upon when setting up the ETL process.

File-based exchanges are sensitive to file structure changes and require validations to avoid inadvertent data corruption. Hence, the ETL application will validate that the file format and file encoding are as specified (when setting up the ETL process) before reading and staging the data.

When validation fails, the ETL process will handle the exception and terminate execution.

8. Data Extraction

The ETL process retrieves data according to its business purpose, in the data extraction step. The extracted data may vary from a full-table extract to records that were updated within a specific time period. The scope of data will be defined in the design phase of the ETL process. For the sake of information security, and processing performance, only data directly attributed to the business purpose of the ETL should be accessed and/or extracted.

9. Data Validation

The ETL process performs structural and business validation on data at the source and prior to insertion into the target. These validations depend upon the domain.

Validate the structure of the database or files, according to the data exchange agreement and/or approved design. Failing structural validation means that records are rejected for further processing of transformation and loading. The file shall be rejected after proper error handling.

The ETL process will validate the data to ensure that the data follows certain business rules, for instance:

- Data should be correct data type.
- Values in the field should correspond to the fields meaning.
- The fields should have the correct number of characters.
- Only valid location information is accepted.
- All unique identifiers must be unique and immutable.

10. Data Transformation

The transformation step processes data from source location to the target data schema.

Data transformation includes:

- Transporting extracted data to Jupyter Notebooks.
- Applying transformation according to the business rules of the ETL process design.
- Loading transformed data into archive database, temporary staging tables before loading the data into the target database.

Data transformation functions includes:

- Data mapping form source format to destination format.
- Data aggregation.
- Lookup transformations.

11. Load Step

During the load step of the ETL process, data is loaded directly into the archive, staging database and target database. Once the data is loaded into the desired databases some tables may in-turn the data source for a different ETL process.

Some of the extract step design considerations apply to the loading step design, for example, understanding the target schema and format, the method of connecting to the target, and the means to transport the data.

12. Common Infrastructure

In addition to the standard ETL steps previously discussed, a common infrastructure spans the ETL process end-to-end. These common infrastructure requirements define how process logging, error logging, and exception handling are performed on each ETL process.

13. Process Logging

The ETL logs will be handled manually for donors dw version 1.0. Users will log important events and metrics. Users will decide if increasing or decreasing log details is required.

14. Exception Handling

Any process is bound to encounter unexpected errors. In anticipation of such an event, the process should be designed to handle the errors. The process quits and alert the user an error has been encountered. This error will provide user administrators enough information regarding the occurrence to perform routine fault-analysis.

The user must log all errors encountered in any stage of the ETL process.

The user must capture the full error context.

Element	Attribute	Comments
Process Info		Details about the process logging the error
Process Name		Name of the process in which the error occurred
Process ID		The process id assigned
Purpose		The purpose of the process
Number Of Records		The number of records associated with the error code
Application Type		Jupyter Notebooks, S3, RDS, MySQL, Tableau
Time Stamp		The timestamp at which the error occurred
Error Code		A unique code for each potential error
Error Description		A unique description associated with the error code
Error Message		Actual error message or exception at a specific step in the process
Error Severity		Critical, high, medium, or low
Environment		Development, quality assurance, user acceptance testing, or production

15. Component Documentation

Component documentation is descriptive text that programmers and operations staff can use to analyze code. All components must contain documentation blocks that describe the purpose, description, dependencies, and version control history.

Each ETL process job will specify the supported business purpose and the technology components that directly feed data to the ETL job or consume data directly from the ETL job.

16. Scheduling

The ETL process for donors dw will run on a trigger from user to run the Python script.

The ETL process design executes a sequence of standalone tasks as a combination of tasks.

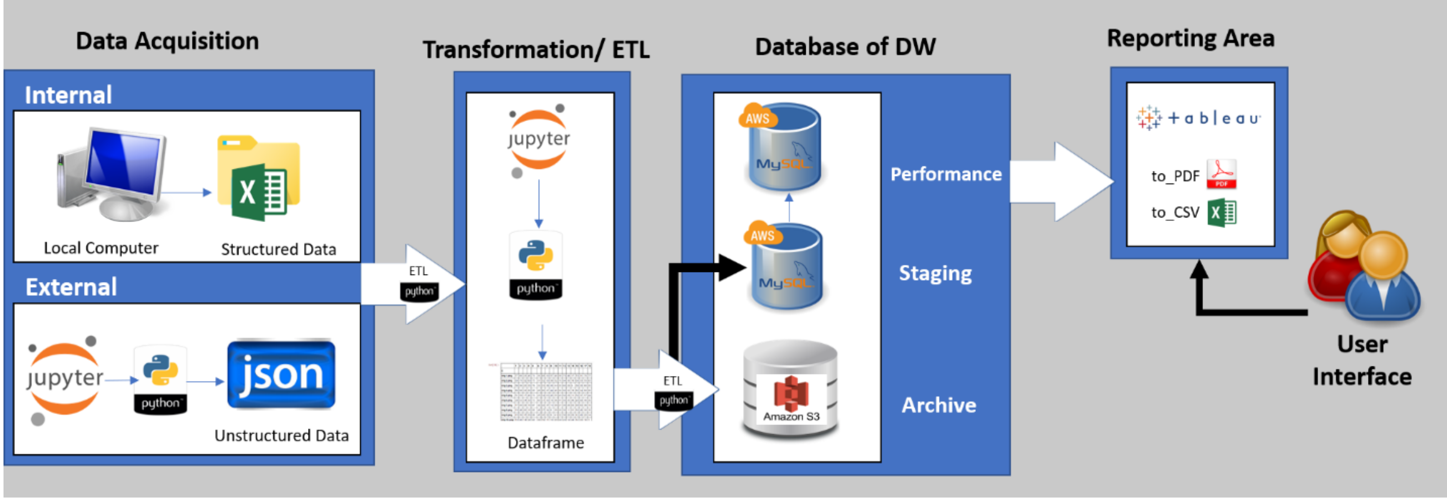
Sequence of tasks:

- Reading and extracting data from the source.
- Data transformation and validation.
- Loading data into the archive database.
- Loading data into the staging database.
- Loading data from the staging database to the target database.

17. Tools and Technologies

Purpose	Tool
Contains Script	Jupyter Notebooks
Programin Language	Python
Query Language	SQL
Archive Database	AWS S3
Staging Database	MySQL Workbench, AWS RDS
Target Database	MySQL Workbench, AWS RDS
Visualization Layer	Tableau

Figure 5. Tools Diagram


Data Acquisition
Internal
Local Computer
Structured Data
External
jupyter
python
json
Unstructured Data
ETL
python
Transformation/ ETL
jupyter
python
Dataframe
ETL
python
Database of DW
AWS
MySQL
Performance
AWS
MySQL
Staging
Amazon S3
Archive
Reporting Area
tableau
to_PDF
PDF
to_CSV
User
Interface