DESIRED FILE GENERATION USING ETL

A PROJECT REPORT

Submitted by

HARSHITHA M S **113219031051**

In partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

VELAMMAL ENGINEERING COLLEGE, CHENNAI-66.

(An Autonomous Institution, Affiliated to Anna University, Chennai)

April 2023

VELAMMAL ENGINEERING COLLEGE
CHENNAI -66

BONAFIDE CERTIFICATE

Certified that this project report **"DESIRED FILE GENERATION USING ETL"** is the bonafide work of **HARSHITHA M S (113219031051)"** who carried out the project work under my supervision.

Dr. B. MURUGESHWARI
PROFESSOR & HEAD
Department of Computer Science and Engineering
Velammal Engineering College
Ambattur – Red Hills Road,
Chennai - 600066

Dr. A. RAJESWARI
SUPERVISOR
Assistant Professor – III
Department of Computer Science and Engineering
Velammal Engineering College
Ambattur – Red Hills Road,
Chennai - 600066

CERTIFICATE OF EVALUATION

COLLEGE NAME : VELAMMAL ENGINEERING COLLEGE

BRANCH : COMPUTER SCIENCE AND ENGINEERING

SEMESTER : VIII

SI. No	Name of the students who has done the project	Title of the Project	Name of supervisor with designation
1	HARSHITHA M S	DESIRED FILE GENERATION USING ETL	Dr. A. RAJESWARI Assistant Professor - III

This report of Project work submitted by the above students in the partial fulfillment for the award of Bachelor of Engineering Degree in Anna Universitywas evaluated and confirmed to be reports of the work by the above student and then assessed.

Submitted for Internal Evaluation held on …….

Internal Examiner External Examiner

ABSTRACT

In today's data-driven world, organizations deal with vast amounts of data that require efficient and automated processing for reporting and analysis. The automated file generation project aims to streamline data processing and reporting by automating the generation of files from diverse data sources. The project utilizes Extract, Transform, Load (ETL) techniques to extract data from various sources such as databases, APIs, spreadsheets, and other file formats. The extracted data is then transformed and cleaned to ensure consistency and accuracy. Data validation, aggregation, enrichment, and formatting are applied to ensure that the generated files are in a meaningful and standardized format. The automated file generation project also leverages advanced data integration techniques, such as data mapping, data merging, and data transformation, to reconcile and consolidate data from different sources into a cohesive file. The project may also involve the use of business rules and data validation checks to ensure data integrity and quality. The generated files can be used for various reporting purposes, such as financial reporting, operational reporting, regulatory reporting, and management reporting. The project aims to improve data processing efficiency, reduce manual errors, and enable timely and accurate reporting, leading to informed decision-making and better business outcomes. The project is implemented using modern data integration and automation tools, such as ETL platforms, data pipelines, and scripting languages.

ACKNOWLEDGEMENT

I wish to acknowledge with thanks to the significant contribution given by the management of our college **Chairman, Dr.M.V.Muthuramalingam,** and our **Chief Executive Officer Thiru. M.V.M. Velmurugan,** for their extensive support.

I would like to thank **Dr. S. Satish Kumar, Principal** of Velammal Engineering College**,** for giving me this opportunity to do this project.

I wish to express my gratitude to our effective **Head of the Department, Dr. B. Murugeshwari,** for her moral support and for her valuable innovative suggestions, constructive interaction, constant encouragement and unending help that have enabled me to complete the project.

I wish to express my indebted humble thanks to our **Project Coordinators, Dr. P. S. Smitha, Dr. P. Pritto Paul and Dr. S. Rajalakshmi,** Department of Computer Science and Engineering for their invaluable guidance in shaping of this project.

I wish to express my sincere gratitude to my **Internal Guide, Dr. A. Rajeswari, Assistant Professor - III,** Department of Computer Science and Engineering for her guidance, without her this projectwould not have been possible.

I am grateful to the entire staff members of the department of Computer Science and Engineering for providing the necessary facilities and to carry out the project. I would especially like to thank my parents for providing me with the unique opportunity to work and for their encouragement and support at all levels. Finally, my heartfelt thanks to **The Almighty** for guiding me throughout the life.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	iv
	LIST OF FIGURES	ix
	LIST OF SYMBOLS, ABBREVIATIONS	x
1.	**INTRODUCTION**	1
	1.1 Domain Introduction	1
	1.1.1 Extract Transform Load	1
	1.1.2 Data Warehouse	6
	1.1.3 Data Marts	7
	1.1.4 Online Transaction Processing	9
2.	**LITERATURE SURVEY**	11
	2.1 Research works	11
3.	**SYSTEM ANALYSIS**	14
	3.1 Extract Transform Load	16
	3.2 Requirement Analysis	17
	3.2.1 Hardware Environment	19
	3.2.2 Software Environment	20
	3.2.3 POSTGRESQL	20
	3.2.4 Pentaho	22
	3.2.5 Business Benefits	24
	3.2.6 Features of Pentaho	25

CHAPTER NO.	TITLE	PAGE NO.
	3.3 Backend	28
	3.3.1 POSTGRESQL	28
	3.3.2 POSTGRESQL Architecture	29
	3.3.3 Storage Engines	30
	3.3.4 POSTGRESQL Cluster Overview	31
4.	**SYSTEM STUDY**	33
	4.1 Automated File Generation	33
	4.2 Data Validation	34
	4.3 Existing System	35
	4.3.1 Disadvantages	35
	4.4 Proposed System	35
	4.4.1 Advantages	35
5.	**DESIGN & IMPLEMENTATION**	36
	5.1 System Architecture	36
	5.2 Modules	39
	5.3 Module Description	39
	5.3.1 Client Request Filtration	39
	5.3.2 Detail Level Filtration	39
	5.3.3 Query Generation	39
	5.3.4 File Output	40
	5.3.5 Update the status	40

CHAPTER NO.	TITLE	PAGE NO.
6.	**CONCLUSION**	41
	6.1 Conclusion	41
	APPENDICES	42
	A.1 Sample Code	42
	A.2 Tables	44
	A.3 File Name Output	46
	REFERENCES	48

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
3.3.2	POSTGRESQL Architecture	29
3.3.4	POSTGRESQL Cluster	32
5.1	System Architecture	36

LIST OF ABBREVIATIONS

ELT	Extract transform Load
OLTP	Online Transaction Processing
API	Application Programming Interface
SQL	Structured Query Language
JSON	Javascript Object Notation
XML	Extensible Markup Language

CHAPTER 1
INTRODUCTION

1.1DOMAIN INTRODUCTION

1.1.1 ETL

ETL is a critical process used in the field of data integration and data warehousing, which involves extracting data from various sources, transforming it into a format suitable for analysis, and loading it into a target system for storage and retrieval. In today's data-driven world, organizations need to collect, process, and analyze large volumes of data from diverse sources such as databases, data lakes, cloud storage, APIs, and more. ETL plays a crucial role in ensuring that data is accurately and efficiently extracted from these sources, transformed into a consistent and meaningful format, and loaded into a data warehouse or data lake for analysis and reporting. The process of ETL typically involves several stages. The first stage is data extraction, where data is sourced from one or more external systems and brought into a central repository. The next stage is data transformation, where the extracted data is cleaned, validated, enriched, and converted into a format that is consistent and useful for analysis. This stage may also involve data integration, aggregation, and enrichment using various techniques such as data mapping, data validation, data cleansing, and data enrichment. Once the data is transformed, it is loaded into a target system, such as a data warehouse, data lake, or other data storage solutions. This allows organizations to store and manage their data in a centralized and organized manner, making it accessible for various analytical and reporting purposes. ETL is used in a wide range of industries and domains, including finance, healthcare, e-commerce, marketing, supply chain, and more. It enables organizations to gain insights from their data, make data-driven decisions, and drive business outcomes. In summary, ETL is a crucial process that helps organizations extract, transform, and load data from various sources into a central repository for analysis and reporting. It plays a fundamental role in data integration, data warehousing, and data management, enabling organizations to harness the power of data for better decision-making and business success.

In the context of ETL (Extract, Transform, Load) for file generation, there are several domain concepts that are relevant. These concepts help in understanding the overall process of extracting data from various sources, transforming it into a suitable format, and loading it into files for further processing or analysis. Let's explore some of these domain concepts:

Data Sources

Data sources are the systems or applications from which data is extracted for further processing. Data sources can include databases, flat files, APIs, web services, spreadsheets, and other sources of structured or unstructured data. Data sources may be internal or external to the organization, and they can vary in terms of data formats, structures, and complexities.

Data Extraction

Data extraction is the process of retrieving data from the data sources for further processing. This may involve techniques such as data integration, data mapping, data profiling, and data validation to ensure that the data is accurately and efficiently extracted from the data sources. Data extraction may also involve scheduling and automation to perform regular and repetitive data extraction tasks.

Data Transformation

Data transformation is the process of converting data from its original format to a format that is suitable for further processing or analysis. This may involve data cleansing, data validation, data enrichment, data aggregation, data normalization, and data formatting, among other transformation techniques. Data transformation ensures that the data is consistent, accurate, and relevant for the intended use.

File Generation

File generation is the process of creating files from the transformed data for further processing or analysis. This may involve creating files in different formats such as CSV, Excel, JSON, XML, or other custom formats depending on the requirements of the downstream systems or applications. File generation may also involve naming conventions, file splitting, file compression, and file delivery mechanisms to ensure that

the generated files are organized, manageable, and efficiently delivered to the intended recipients.

Data Validation

Data validation is the process of verifying the integrity, accuracy, and consistency of the data after transformation and before file generation. Data validation may involve data quality checks, data integrity checks, data consistency checks, and data reconciliation to ensure that the transformed data meets the defined business rules, data standards, and data quality requirements.

Error Handling

Error handling is an important domain concept in ETL for file generation, as it involves managing and resolving errors that may occur during the ETL process. Error handling may include logging and monitoring of errors, error notifications, error resolution, and error recovery mechanisms to ensure that any issues or errors that arise during the ETL process are identified, resolved, and reported in a timely manner.

Metadata Management

Metadata management is the process of managing the metadata associated with the data being processed in the ETL for file generation. Metadata includes information such as data source details, data transformation rules, data mapping, data validation rules, file generation rules, and other relevant information about the data being processed. Metadata management ensures that the relevant information about the data is captured, stored, and used for documentation, auditing, and data lineage purposes.

Performance Optimization

Performance optimization is the process of optimizing the performance of the ETL for file generation process to ensure that it runs efficiently and in a timely manner. This may involve techniques such as data indexing, data caching, parallel processing, data partitioning, and other performance tuning techniques to improve the overall performance of the ETL process and reduce processing times.

These are some of the domain concepts that are relevant in the context of ETL for file generation.

ETL, which stands for Extract, Transform, Load, is a comprehensive data integration process used in modern data management and analytics. It involves extracting data from various sources, transforming it into a format suitable for analysis, and loading it into a target system for storage and retrieval. Let's take a closer look at each step of the ETL process:

Extract: The first step in the ETL process is extracting data from one or more sources. This can include structured data from databases, unstructured data from text files or APIs, or semi-structured data from spreadsheets or XML files. Data extraction can be done using various techniques, such as batch processing, real-time streaming, or incremental updates.

Transform: Once the data is extracted, it undergoes a series of transformations to ensure its quality, consistency, and usefulness for analysis. Data transformation involves cleaning, validating, enriching, and aggregating data to create a cohesive and meaningful dataset. This may also include data mapping, data normalization, data enrichment, and data validation to ensure that the data is accurate, complete, and consistent.

Load: After the data is transformed, it is loaded into a target system, such as a data warehouse, data lake, or other data storage solutions. Data loading involves storing the transformed data into the target system in a structured and organized manner. This can include inserting, updating, or appending data, depending on the requirements of the target system.

The ETL process may also involve additional steps, such as data profiling, data cleansing, data deduplication, data integration, and data validation, to ensure data quality and integrity throughout the process. ETL can be performed using various tools, technologies, and platforms, such as ETL software, data integration platforms, cloud-based services, or custom scripts.

The full view of ETL encompasses the entire process of extracting, transforming, and loading data from various sources into a target system for analysis, reporting, and decision-making. It is a critical component of modern data management and analytics, enabling organizations to collect, process, and analyze large volumes of data for gaining

insights and driving business outcomes.

TYPES

There are three main types of ETL (Extract, Transform, Load) processes, which are commonly used in the field of data integration and data warehousing. These types are:

- Batch ETL
- Real-time ETL
- Change Data Captuire

1. Batch ETL: In batch ETL, data is extracted from source systems in batches or chunks at scheduled intervals, typically during off-peak hours when system resources are less utilized. The extracted data is then transformed and loaded into the target system or systems in batches. Batch ETL processes are typically used for large volumes of data that can be processed offline, and are well-suited for scenarios where data can be staged, transformed, and loaded in bulk. Batch ETL processes are often used in data warehousing scenarios where data is loaded into a data warehouse or data mart for reporting, analytics, and other downstream processes.
2. Real-time ETL: In real-time ETL, data is extracted from source systems in near real-time or real-time as it becomes available, and is transformed and loaded into the target system or systems in near real-time or real-time as well. Real-time ETL processes are designed to handle data that needs to be processed in real-time or near real-time, such as data from online transactions, social media feeds, IoT (Internet of Things) devices, and other sources that require immediate processing for decision-making or operational purposes. Real-time ETL processes are typically more complex and require higher levels of performance and scalability compared to batch ETL processes, as they need to handle data in real-time or near real-time.
3. Change Data Capture (CDC): Change Data Capture (CDC) is a specific type of

ETL process that captures and replicates changes to data in real-time or near real-time. CDC processes are used to identify and capture changes to data in source systems, such as changes to records, updates, deletions, and insertions, and replicate those changes to a target system or systems. CDC processes are typically used in scenarios where it is important to maintain real-time or near real-time synchronization of data between source and target systems, such as in data replication, data synchronization, and data consolidation scenarios.

Each type of ETL process has its own advantages and disadvantages, and the choice of ETL type depends on the specific requirements of the data integration project, such as the volume and velocity of data, the latency requirements for data processing, the performance and scalability requirements, and the overall architecture and infrastructure of the data integration solution.

1.1.2 DATA WAREHOUSE

A data warehouse is a central repository that stores data from various sources in a structured and organized format for efficient data management, analysis, and reporting. In the context of ETL (Extract, Transform, Load), a data warehouse plays a crucial role as the target system where the transformed data is loaded after the extraction and transformation steps. Let's take a closer look at the role of a data warehouse in the ETL process:

Data Storage: A data warehouse serves as the ultimate destination for storing the transformed data. Once the data is extracted from various sources and transformed into a consistent format, it is loaded into the data warehouse for long-term storage. Data warehouses are designed to handle large volumes of data and provide efficient storage and retrieval mechanisms for fast and reliable data access.

Data Integration: A data warehouse allows for the integration of data from multiple sources. In the ETL process, data is extracted from various sources, transformed into a standardized format, and then loaded into the data warehouse. The data warehouse acts as a central repository where data from different sources can be integrated and combined

to create a unified view of the data. This enables organizations to perform complex analysis and reporting on integrated data from multiple sources.

Data Modeling: A data warehouse often includes data modeling techniques to optimize data storage and retrieval. Data modeling involves structuring the data in the data warehouse in a way that supports efficient data retrieval, analysis, and reporting. Common data modeling techniques used in data warehouses include dimensional modeling, star schema, snowflake schema, and data cubes, among others. These data modeling techniques enable organizations to structure the data in a way that supports efficient and fast data analysis and reporting.

Data Security: Data warehouses often include robust security features to protect the data stored in the warehouse. Data security is critical in the ETL process as it ensures that the transformed data is stored securely and accessed only by authorized users. Data warehouses typically include features such as access controls, encryption, authentication, and auditing to ensure the confidentiality, integrity, and availability of the data.

Data History and Auditing: Data warehouses may also include features for data history and auditing. Data history enables organizations to track changes in the data over time, allowing for historical analysis and reporting. Auditing features provide visibility into data changes, ensuring data integrity and accountability in the ETL process. These features are particularly useful for compliance and regulatory requirements.

In summary, a data warehouse plays a pivotal role in the ETL process as the target system where the transformed data is loaded for efficient storage, data integration, data modeling, data security, and data history. It serves as a central repository for data management and analysis, enabling organizations to leverage the transformed data for informed decision-making and business success.

1.1.3 DATA MARTS

Data marts are subsets of data warehouses that are designed for specific business units, departments, or functional areas within an organization. They are typically created to address specific analytical or reporting needs of a particular group of users. In the context

of ETL (Extract, Transform, Load), data marts play a key role in the process of extracting, transforming, and loading data for specific business units or departments. Let's explore how data marts are used in the ETL process:

Data Extraction: Data marts are often designed to extract data from the data warehouse or other data sources that are relevant to a specific business unit or department. Data extraction from the data warehouse or other sources is performed using ETL techniques such as data integration, data mapping, and data profiling. The extracted data is then transformed into a format suitable for analysis and reporting.

Data Transformation: Once the data is extracted, it undergoes transformation processes to ensure its quality, consistency, and usefulness for analysis in the context of the specific business unit or department. Data transformation in data marts may involve cleaning, validating, enriching, aggregating, and/or summarizing data to meet the specific requirements of the business unit or department. This may also include data normalization, data enrichment, and data validation to ensure that the data is accurate, complete, and relevant for the specific analytical or reporting needs.

Data Loading: After the data is transformed, it is loaded into the data mart for storage and retrieval. Data loading into data marts involves storing the transformed data in a format that is optimized for analysis and reporting by the specific business unit or department. This may include inserting, updating, or appending data to the data mart, depending on the requirements of the target system.

Data Modeling: Data marts may also include data modeling techniques to structure the data in a way that supports efficient data analysis and reporting for the specific business unit or department. Data modeling in data marts may include techniques such as dimensional modeling, star schema, snowflake schema, and data cubes, among others. These data modeling techniques enable organizations to structure the data in a way that supports efficient and fast analysis and reporting for the specific business unit or department.

Data Security: Data marts may also include security features to protect the data stored in the mart. Data security in data marts may include features such as access controls,

encryption, authentication, and auditing to ensure the confidentiality, integrity, and availability of the data.

In summary, data marts play a crucial role in the ETL process as they are designed to address the specific analytical or reporting needs of a particular business unit or department within an organization. They involve extracting, transforming, and loading data from various sources into a format optimized for analysis and reporting by the specific business unit or department. Data marts provide a focused and efficient way to deliver relevant and timely data for decision-making at the departmental level, while maintaining data integrity and security.

1.1.4 OLTP

OLTP (Online Transaction Processing) is a concept that is typically associated with database management systems and refers to the real-time processing of transactions in a production environment. On the other hand, ETL (Extract, Transform, Load) is a process used to extract data from various sources, transform it into a suitable format, and load it into a target system for further processing or analysis. While OLTP and ETL are related in the sense that they both deal with data processing, they serve different purposes and have different characteristics.

In the context of ETL, OLTP data sources can be one of the sources from which data is extracted. These OLTP systems are typically transactional databases that are used for day-to-day operational activities such as capturing customer orders, processing financial transactions, updating inventory levels, and managing other operational processes. ETL processes may extract data from these OLTP systems to consolidate, transform, and integrate the data with other data sources in a data warehouse or data lake for reporting, analytics, or other business purposes.

The data extracted from OLTP systems during the ETL process may undergo various transformations to cleanse, validate, enrich, aggregate, or otherwise manipulate the data to meet the requirements of the target system. This may involve data mapping, data conversion, data validation, data enrichment, and other transformation techniques to

ensure that the data is transformed into a format that is suitable for further processing or analysis.

Once the data from OLTP systems is transformed, it is loaded into a target system such as a data warehouse, data mart, or other destination systems for further processing or analysis. The loaded data can then be used for various purposes such as reporting, analytics, business intelligence, data visualization, and decision-making.

It's important to note that the ETL process may be designed to run at different frequencies depending on the requirements of the business, such as near real-time, batch, or incremental processing. The goal of ETL in the context of OLTP data sources is to ensure that the data extracted from these systems is transformed, integrated, and loaded into the target system efficiently, accurately, and with appropriate data quality checks in place.

In summary, OLTP systems can be one of the data sources for ETL processes, and the ETL process may involve extracting data from these systems, transforming it, and loading it into a target system for further processing or analysis. ETL plays a critical role in integrating data from OLTP systems with other data sources and making it available for reporting, analytics, and decision-making purposes.

CHAPTER 2
LITERATURE SURVEY

Title: Survey on ETL Process

Author: Ahmed Kabiri, Dalila Chiadmi

Year: 2013

In Data Warehouse (DW) environment, Extraction-Transformation-Loading (ETL) processes constitute the integration layer which aims to pull data from data sources to targets, via a set of transformations. ETL is responsible for the extraction of data, their cleaning, conforming and loading into the target. ETL is a critical layer in DW setting. It is widely recognized that building ETL processes is expensive regarding time, money and effort. It consumes up to 70% of resources. By this work we intend to enrich the field of ETL processes, the backstage of data warehouse, by presenting a survey on these processes. Therefore, in current work, firstly (1) we review open source and commercial ETL tools, along with some ETL prototypes coming from academic world, secondly (2) we review the modeling and design works in ETL field. Also, (3) we approach ETL maintenance issue then (4) we review works in connection with optimization and incremental ETL. Finally, (5) we present and outline challenges and research opportunities around ETL processes. Keywords: ETL, Data warehouse, Data warehouse Population, Data warehouse Refreshment, ETL Modeling, ETL Maintenance.

Title: Real –Time Data Warehousing: A tool for continuous actionable knowledge

Author: Ricardo J Santos, Jorge Bernardino

Year: 2012

Technological evolution has redefined many business models. Many decision makers are now required to act near real-time, instead of periodically, given the latest transactional

information. Decision-making occurs much more frequently and considers the latest business data. Since data warehouses (DWs) are the core of business intelligence, decision support systems need to deal with 24/7 real-time requirements. Thus, the ability to deal with continuous data loading and decision support availability simultaneously is critical, for producing continuous actionable knowledge. The main challenge in this context is to efficiently manage the DW's refreshment, when data sources change, to recapture consistency and accuracy with those sources, while maintaining OLAP availability.

Title: Extraction Transformation and Loading (ETL) of Data Using ETL Tools

Author: Manish Manoj Singh

Year :2022

This Research Paper presents the Extract, Transform, Load (ETL) Process and discusses various ETL Tools Available in the Market. A huge piece of BI frameworks is a well-performing Implementation of the Extract, Transform, and Load (ETL) process. In BI projects, implementing the ETL process can be the big task ETL is the core process of Data integration which is associated with Data Warehouse. This paper also focuses on the best ETL Tools and which tool can be the best for the ETL process.

Title: A proposed model for data warehouse ETL processes

Author: Shaker H. Ali El-Sappagh, Abdeltawab M. Ahmed Hendawi, Ali Hamed El Bastawissy

Year :2011

Extraction–transformation–loading (ETL) tools are pieces of software responsible for the extraction of data from several sources, its cleansing, customization, reformatting, integration, and insertion into a data warehouse. Building the ETL process is potentially

one of the biggest tasks of building a warehouse; it is complex, time consuming, and consumes most of data warehouse project's implementation efforts, costs, and resources. Building a data warehouse requires focusing closely on understanding three main areas: the source area, the destination area, and the mapping area (ETL processes). The source area has standard models such as entity relationship diagram, and the destination area has standard models such as star schema, but the mapping area has not a standard model till now. In spite of the importance of ETL processes, little research has been done in this area due to its complexity. There is a clear lack of a standard model that can be used to represent the ETL scenarios. In this paper we will try to navigate through the efforts done to conceptualize the ETL processes. Research in the field of modeling ETL processes can be categorized into three main approaches: Modeling based on mapping expressions and guidelines, modeling based on conceptual constructs, and modeling based on UML environment. These projects try to represent the main mapping activities at the conceptual level. Due to the variation and differences between the proposed solutions for the conceptual design of ETL processes and due to their limitations, this paper also will propose a model for conceptual design of ETL processes. The proposed model is built upon the enhancement of the models in the previous models to support some missing mapping features.

CHAPTER 3

SYSTEM ANALYSIS

File generation using ETL (Extract, Transform, Load) techniques involves a complex process of extracting data from different sources, transforming it into the desired format, and loading it into target files. System analysis is a critical step in designing and developing an efficient and effective file generation system using ETL. Here are some key aspects of system analysis for file generation using ETL:

1. **Data Source Analysis**: The system analysis process begins with understanding the different data sources from which data needs to be extracted for file generation. This includes analyzing the structure, format, and quality of data in various source systems, such as databases, data warehouses, APIs, and external files. The analysis should also consider the volume, velocity, and variety of data in the sources, as it impacts the ETL processes and file generation requirements.
2. **Data Transformation Analysis:** Once the data is extracted from the sources, it needs to be transformed into the desired format for file generation. System analysis involves identifying the data transformation requirements, such as data cleansing, data validation, data enrichment, data aggregation, and data mapping. The analysis should also consider any data quality rules or business rules that need to be applied during the data transformation process.
3. **File Generation Analysis:** The system analysis process should also focus on understanding the requirements for file generation. This includes analyzing the target file format, such as CSV, Excel, XML, JSON, or any other custom format, and identifying the specific file generation requirements, such as file naming conventions, file structure, and file delivery methods. The analysis should also consider any performance, scalability, and error-handling requirements for file generation.
4. **ETL Workflow Analysis:** The ETL workflow is the sequence of tasks and processes involved in extracting, transforming, and loading data for file generation. System analysis should involve analyzing the ETL workflow, including the

sequencing of tasks, dependencies, and interactions between different ETL components. This analysis should consider the optimal flow of data, error handling, performance optimizations, and monitoring requirements for the ETL workflow.

5. **Technology Analysis**: System analysis for file generation using ETL should also involve evaluating and selecting the appropriate ETL tools, technologies, and frameworks. This includes analyzing the capabilities, features, and limitations of different ETL tools or platforms, such as Apache NiFi, Talend, Microsoft SQL Server Integration Services (SSIS), and others, based on the specific requirements of the file generation system.
6. **Security and Compliance Analysis:** System analysis should also consider the security and compliance requirements of the file generation system. This includes analyzing the data security, data privacy, and compliance requirements, such as GDPR, HIPAA, PCI-DSS, and others, and identifying the necessary measures to ensure the confidentiality, integrity, and availability of data during the ETL processes and file generation.
7. **Performance and Scalability Analysis:** System analysis should also evaluate the performance and scalability requirements of the file generation system. This includes analyzing the volume and velocity of data, processing time, and resource utilization during the ETL processes, and identifying any performance bottlenecks or scalability challenges. The analysis should consider the need for optimizations, such as parallel processing, caching, and indexing, to ensure optimal performance and scalability of the file generation system.
8. **Error Handling and Monitoring Analysis**: Lastly, system analysis should consider the error handling and monitoring requirements of the file generation system. This includes analyzing the different types of errors that can occur during the ETL processes, such as data errors, data validation failures, data mapping issues, and others, and identifying the appropriate error handling mechanisms, such as logging, alerts, notifications, and error recovery strategies. The analysis should

also consider the monitoring requirements, such as data lineage, data profiling, and data quality monitoring, to ensure the accuracy.

3.1 ETL:

ETL (Extract, Transform, Load) requirements refer to the specifications and criteria that need to be met in order to successfully implement an ETL process for data integration and processing. ETL requirements typically encompass the following areas:

Data Extraction: This includes defining the sources from which data needs to be extracted, specifying the data formats, structures, and protocols, and determining the frequency and scheduling of data extraction.

Data Transformation: This involves specifying the data transformation rules, business logic, data validation and cleansing processes, data enrichment, data aggregation, and data mapping requirements. It also includes defining data quality standards and data governance policies.

Data Loading: This includes defining the target systems or data repositories where data needs to be loaded, specifying the data loading methods (such as batch processing or real-time streaming), data loading frequency, data storage and indexing requirements, and data synchronization requirements.

Data Integration: This includes defining the integration requirements for integrating data from multiple sources, including data consolidation, data integration logic, data reconciliation, and data merging requirements.

Data Security: This involves specifying the security requirements for protecting data during the ETL process, including data encryption, access controls, data masking, and data privacy requirements.

Performance and Scalability: This includes specifying the performance and scalability requirements for the ETL process, including data processing speed, data volume handling capabilities, and resource utilization requirements.

Error Handling and Exception Management: This involves defining the requirements

for handling errors, exceptions, and data quality issues during the ETL process, including error logging, error notification, data recovery, and data reconciliation requirements.

Metadata Management: This includes defining the requirements for capturing and managing metadata, including data lineage, data cataloging, data profiling, and data documentation requirements.

Documentation and Reporting: This involves specifying the requirements for documenting the ETL process, including data flow diagrams, data transformation rules, data lineage, and data mapping documentation. It may also include reporting requirements for tracking ETL performance, data quality, and data lineage.

It's important to thoroughly define and document the ETL requirements before starting the implementation process to ensure that the ETL process meets the business and technical objectives, and to avoid potential issues during implementation and operation. Clear and comprehensive ETL requirements help guide the development and testing of the ETL process, and serve as a reference for ongoing maintenance and enhancement efforts.

3.2 REQUIREMENT ANALYSIS

Data Source Requirements:

The first step in requirement analysis is to understand the data sources from which data needs to be extracted. This includes analyzing the structure, format, and quality of data in various source systems, such as databases, data warehouses, APIs, and external files. The requirement analysis should consider the volume, velocity, and variety of data in the sources, as it impacts the Pentaho ETL processes.

Data Transformation Requirements:

Once the data is extracted from the sources, it needs to be transformed into the desired format. Requirement analysis should identify the data transformation requirements, such as data cleansing, data validation, data enrichment, data aggregation, and data mapping. The analysis should consider any data quality rules or business rules that need to be

applied during the data transformation process, and map them to Pentaho's data integration capabilities.

Data Loading Requirements:

Requirement analysis should also consider the data loading requirements, which involve loading the transformed data into target systems or files. This includes analyzing the target system or file format, such as databases, data warehouses, CSV, Excel, XML, JSON, or any other custom format, and identifying the specific data loading requirements, such as data mapping, data validation, and data delivery methods. The analysis should consider any performance, scalability, and error-handling requirements for data loading.

ETL Workflow Requirements:

Pentaho provides a graphical user interface (GUI) for designing and managing ETL workflows. Requirement analysis should involve analyzing the ETL workflow requirements, including the sequencing of tasks, dependencies, and interactions between different ETL components. The analysis should consider the optimal flow of data, error handling, performance optimizations, and monitoring requirements for the Pentaho ETL workflows.

Data Integration Requirements:

Pentaho offers a wide range of data integration capabilities, such as data extraction, data transformation, data loading, data validation, data enrichment, data profiling, and data quality monitoring. Requirement analysis should involve identifying the specific data integration requirements based on the organization's data processing needs. This may include defining data integration jobs, transformations, mappings, and other data processing tasks using Pentaho's tools and features.

Security and Compliance Requirements:

Requirement analysis should also consider the security and compliance requirements of the ETL processes. This includes analyzing the data security, data privacy, and compliance requirements, such as GDPR, HIPAA, PCI-DSS, and others, and identifying the necessary measures to ensure the confidentiality, integrity, and availability of data

during the Pentaho ETL processes.

Performance and Scalability Requirements:

Requirement analysis should evaluate the performance and scalability requirements of the Pentaho ETL processes. This includes analyzing the volume and velocity of data, processing time, and resource utilization during the ETL processes, and identifying any performance bottlenecks or scalability challenges. The analysis should consider the need for optimizations, such as parallel processing, caching, and indexing, to ensure optimal performance and scalability of the Pentaho ETL processes.

Error Handling and Monitoring Requirements:

Requirement analysis should consider the error handling and monitoring requirements of the Pentaho ETL processes. This includes identifying the different types of errors that can occur during the ETL processes, such as data errors, data validation failures, data mapping issues, and others, and defining the appropriate error handling mechanisms, such as logging, alerts, notifications, and error recovery strategies. The analysis should also consider the monitoring requirements, such as data lineage, data profiling, and data quality monitoring.

3.2.1 Hardware Environment

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design. It shows what the systems do and not how it should be implemented.

Hard disk	: 500 GB
RAM	: 4 GB
PROCESSOR	: CORE i5/ i7

3.2.2 Software Environment

The software requirements are the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the team and tracking the team's progress throughout the development activity.

Operating system	: Windows 7/8.1
Data Base	: POSTGRESQL
ETL	: Pentaho

3.2.3 POSTGRESQL:

PostgreSQL is a powerful open-source relational database management system (RDBMS) known for its robustness, scalability, and extensibility. It was originally developed at the University of California, Berkeley in the 1980s and has since become one of the most popular and widely used open-source databases in the world.

PostgreSQL, often referred to simply as "Postgres," offers a wide range of advanced features that make it suitable for a wide range of applications, from small-scale applications to large-scale enterprise systems. Some of its key features include:

Advanced Data Types:

PostgreSQL supports a rich set of data types, including standard SQL data types such as numeric, character, and date/time types, as well as advanced data types such as arrays, hstore (a key-value store), JSON, and even custom data types.

Extensibility:

PostgreSQL allows users to define their own data types, operators, functions, and aggregates, making it highly extensible and customizable. This allows developers to tailor PostgreSQL to their specific requirements and easily add new functionality.

Concurrency Control:

PostgreSQL has robust support for concurrent access to the database, allowing multiple users to access and modify the database simultaneously without conflicts. It uses a multi-version concurrency control (MVCC) mechanism to ensure that transactions can be executed concurrently without compromising data integrity.

Full Text Search:

PostgreSQL includes powerful full-text search capabilities, allowing users to perform complex text search queries efficiently. It supports advanced search features such as ranking, stemming, and phrase searching, making it suitable for applications that require sophisticated text search capabilities.

Geospatial Support:

PostgreSQL has built-in support for geospatial data and includes advanced spatial data types, spatial indexing, and spatial functions. This makes it a popular choice for applications that require geospatial data processing, such as GIS (Geographical Information Systems) and location-based services.

Replication and High Availability

: PostgreSQL supports various replication and high availability mechanisms, including streaming replication, logical replication, and built-in high availability features such as failover and automated backups. This ensures data durability and availability in mission-critical applications.

Foreign Data Wrappers:

PostgreSQL allows users to define foreign data wrappers (FDWs) that enable it to interact with external data sources, such as other databases, APIs, or file systems. This allows for seamless integration with external data sources, making it a powerful tool for data integration and data consolidation.

Security Features:

PostgreSQL provides robust security features, including authentication, authorization, and encryption mechanisms to ensure data confidentiality, integrity, and availability. It also supports advanced security features such as SSL/TLS encryption, role-based access control (RBAC), and auditing, making it suitable for applications that require stringent

security measures.

Active Community and Ecosystem:

PostgreSQL has a large and active community of users and developers who actively contribute to its development and provide support through mailing lists, forums, and online resources. It also has a vast ecosystem of extensions, tools, and libraries that extend its functionality and make it even more powerful and versatile.

Overall, PostgreSQL is a mature, feature-rich, and highly extensible open-source RDBMS that is widely used in various applications ranging from small-scale to large-scale enterprise systems. Its advanced features, scalability, and extensibility make it a popular choice for developers and enterprises alike.

3.2.4 PENTAHO:

Pentaho is a widely used open-source business intelligence (BI) and data integration platform that provides a comprehensive set of tools for ETL (Extract, Transform, Load) processes. Pentaho's ETL capabilities are designed to help organizations extract data from various sources, transform it into a suitable format, and load it into a target system for further processing or analysis.

Here are some key aspects of Pentaho's ETL capabilities:

Data Integration: Pentaho provides a visual and user-friendly environment for designing ETL workflows, also known as data integration jobs or transformations. Users can define data sources, data transformations, and data destinations using a drag-and-drop interface, making it easy to design and implement complex ETL processes.

Connectivity: Pentaho supports a wide range of data sources and destinations, including databases, flat files, XML, JSON, web services, big data sources (such as Hadoop and Spark), cloud-based storage (such as Amazon S3 and Google Cloud Storage), and many more. This allows organizations to extract data from various sources and load it into different target systems as needed.

Data Transformation: Pentaho provides a rich set of data transformation functions and capabilities to clean, validate, enrich, aggregate, and otherwise manipulate data during

the ETL process. Users can perform data mapping, data conversion, data validation, data enrichment, and other transformations to ensure that the data is transformed into the desired format for further processing or analysis.

Data Quality: Pentaho also includes data quality tools that allow users to apply data quality checks and validations during the ETL process. These checks can help ensure that data is accurate, complete, and consistent, and can be used to identify and handle data quality issues before the data is loaded into the target system.

Scalability and Performance: Pentaho's ETL capabilities are designed to handle large volumes of data and provide performance optimization features such as parallel processing, data partitioning, and caching. This allows organizations to process data efficiently and at scale, making it suitable for big data scenarios and high-volume data processing requirements.

Monitoring and Management: Pentaho provides monitoring and management tools that allow users to track the progress of ETL processes, monitor performance, and troubleshoot issues. This helps organizations ensure that their ETL processes are running smoothly and effectively.

Integration with Other Pentaho Components: Pentaho's ETL capabilities are tightly integrated with other components of the Pentaho suite, such as reporting, analytics, and data visualization. This allows organizations to seamlessly integrate their ETL processes with other data-related activities, enabling end-to-end data integration and analytics workflows.

In summary, Pentaho is a comprehensive open-source platform that offers robust and feature-rich ETL capabilities for data integration, transformation, and loading. Its visual and user-friendly interface, wide range of data source connectivity, data transformation functions, data quality tools, scalability and performance optimizations, monitoring and management features, and integration with other Pentaho components make it a popular choice for organizations looking to implement ETL processes for their data integration and analytics needs.

3.2.5 BUSINESS BENEFITS

Using ETL (Extract, Transform, Load) in data integration and processing can bring several business benefits, including:

Improved Data Quality:

ETL processes often include data validation, cleansing, and enrichment steps, which can help ensure that data is accurate, consistent, and reliable. Improved data quality leads to better decision-making, reduced errors, and improved operational efficiency.

Enhanced Data Integration:

ETL allows businesses to integrate data from multiple sources, such as databases, applications, and external systems, into a consolidated and meaningful format. This enables organizations to gain a comprehensive view of their data, identify trends, correlations, and patterns, and make informed business decisions.

Efficient Data Processing:

ETL processes can automate the extraction, transformation, and loading of data, reducing the need for manual data handling and processing. This improves data processing speed, reduces the risk of human errors, and increases operational efficiency.

Data Consistency and Standardization:

ETL processes can enforce data standardization and consistency across different data sources, ensuring that data is formatted, structured, and transformed in a consistent manner. This improves data accuracy, reduces data inconsistencies, and facilitates data integration and analysis.

Scalability and Flexibility:

ETL processes can handle large volumes of data and can be scaled to accommodate

growing data needs. ETL also provides flexibility to accommodate changes in data sources, data formats, and business requirements, making it adaptable to evolving business needs.

Data Governance and Compliance:

ETL processes can incorporate data governance practices, such as data lineage tracking, data cataloging, and data profiling, to ensure data governance and compliance with data privacy regulations. This helps organizations maintain data integrity, security, and compliance with regulatory requirements.

Enhanced Decision-Making:

ETL processes provide organizations with timely, accurate, and consistent data for analysis, reporting, and decision-making. This enables organizations to make data-driven decisions, identify business opportunities, and optimize business processes.

Improved Data Accessibility:

ETL processes can consolidate and transform data into a format that is easily accessible and understandable by business users, analysts, and stakeholders. This empowers organizations to leverage data for various purposes, such as reporting, analytics, and visualization.

Cost Savings:

ETL processes can help organizations save costs by automating data integration and processing, reducing manual efforts, and minimizing data errors. Additionally, ETL can enable organizations to optimize data storage and processing resources, resulting in cost savings.

3.2.6 FEATURES OF PENTAHO

Pentaho is a popular open-source data integration and business analytics platform that

offers a wide range of features for data integration, transformation, and visualization. Some of the key features of Pentaho include

Data Integration

Pentaho provides a comprehensive set of data integration tools that allow users to extract data from various sources, such as databases, applications, files, and web services, transform it into the desired format, and load it into target systems. It supports a wide range of data integration techniques, including batch processing, real-time data streaming, and data replication.

Data Transformation

Pentaho offers a visual interface for designing data transformation processes using a drag-and-drop approach, making it easy for users to create and manage complex data transformation logic. It provides a rich set of data transformation functions, such as data cleansing, validation, aggregation, enrichment, and normalization, to manipulate data and prepare it for further analysis.

Data Visualization and Reporting

Pentaho allows users to create interactive and visually appealing dashboards, reports, and visualizations to gain insights from data. It offers a wide range of visualization options, including bar charts, line charts, pie charts, heatmaps, and more, and supports interactive filtering, sorting, and drilling down capabilities to explore data in-depth.

Big Data Integration

Pentaho provides integration with big data technologies, such as Hadoop, Spark, and NoSQL databases, allowing users to process and analyze large volumes of data stored in big data platforms. It supports data ingestion, transformation, and analysis of big data using its visual interface and provides connectors to various big data sources.

Data Quality and Governance

Pentaho includes features for data quality and governance, such as data profiling, data cleansing, and data validation, to ensure data accuracy, consistency, and integrity. It also provides capabilities for data lineage tracking, data cataloging, and metadata management to ensure proper data governance practices.

Collaboration and Workflow

Pentaho offers collaboration and workflow features that allow teams to work together on data integration and analytics projects. It provides version control, scheduling, and notification features, as well as integration with popular collaboration tools, such as Git and JIRA, to facilitate team collaboration and streamline project workflows.

Extensibility and Customization

Pentaho is highly extensible and customizable, allowing users to develop custom data integration and analytics solutions using its APIs, SDKs, and plugins. It supports custom scripting, Java-based plugins, and integration with external tools and systems, enabling users to tailor Pentaho to their specific needs and requirements.

Scalability and Performance

Pentaho is designed to handle large volumes of data and provides scalability and performance features, such as parallel processing, clustering, and caching, to optimize data integration and analytics performance. It can be deployed on-premises or in the cloud, and supports distributed processing for big data workloads.

Security

Pentaho offers robust security features, including data encryption, user authentication and authorization, role-based access control, and auditing, to ensure data security and compliance with data privacy regulations.

3.3 BACK END

3.3.1 POSTGRESQL

PostgreSQL is an open-source object-relational database management system (DBMS) that is known for its robustness, extensibility, and support for advanced data types. It is widely used in various industries for storing and managing large amounts of data.

POSTGRESQL is a Data Base Management System. PostgreSQL acts as a powerful and reliable DBMS, providing robust data storage, retrieval, and management capabilities.

- It supports advanced data types, such as arrays, hstore (key-value pairs), JSON, and JSONB (binary JSON), allowing users to store and query complex data structures.
- PostgreSQL supports transactions with ACID (Atomicity, Consistency, Isolation, Durability) properties, ensuring data integrity and consistency even in the face of failures.
- It provides support for various indexing techniques, including B-tree, hash, GIN (Generalized Inverted Index), and GiST (Generalized Search Tree), for efficient data retrieval and query performance.
- PostgreSQL supports advanced querying features, including support for stored procedures, triggers, views, and full-text search, allowing users to implement complex business logic and perform advanced data analysis.
- It provides support for authentication and authorization, allowing users to define roles, users, and privileges to ensure secure access to the database.
- PostgreSQL is highly extensible and allows users to define custom data types, operators, functions, aggregates, and procedural languages, making it a flexible and powerful DBMS.

3.3.2 POSTGRESQL Architecture

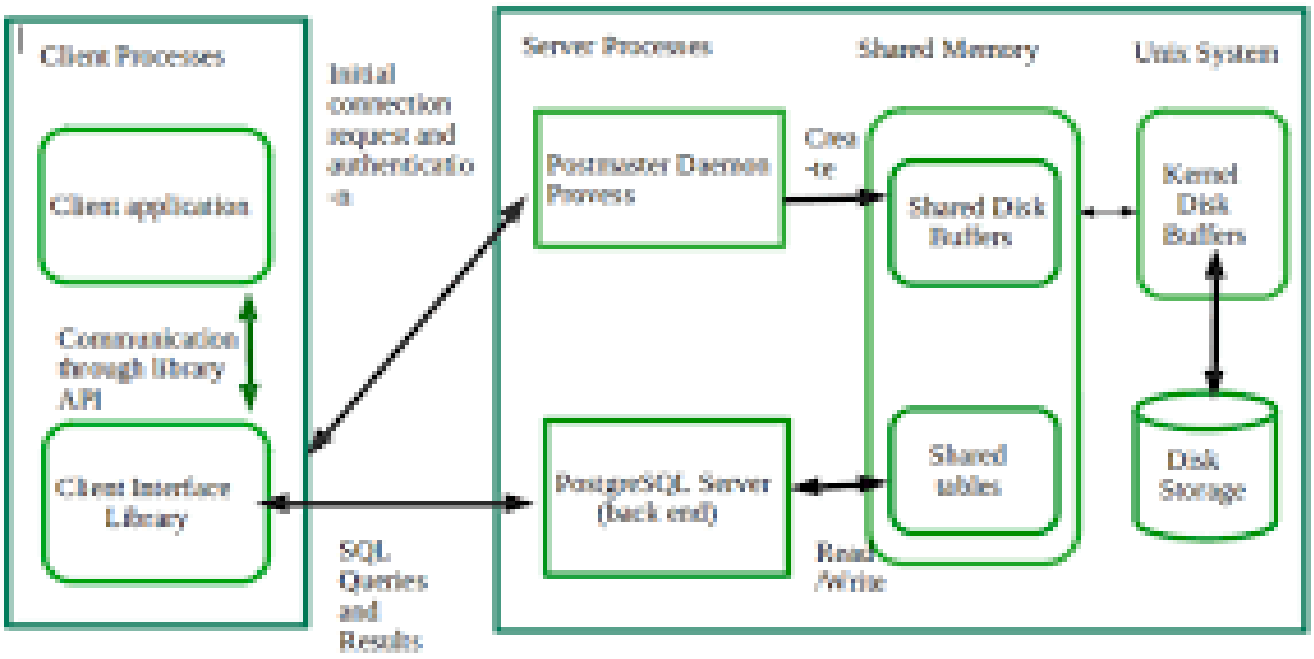


Fig 3.3.2 POSTGRESQL Architecture

- PostgreSQL follows a client-server architecture, where multiple clients can connect to a PostgreSQL server and interact with the database.
- The core of PostgreSQL consists of a set of C language libraries that provide the database engine, query processing, transaction management, and other functionalities.
- PostgreSQL uses a multi-process architecture, where each client connection is handled by a separate server process (backend) that communicates with the main database process (backend).
- The backend processes communicate with each other and with the main database process using inter-process communication (IPC) mechanisms.
- PostgreSQL supports concurrent access to the database, allowing multiple clients to perform transactions and queries simultaneously.

3.3.3 STORAGE ENGINES

In PostgreSQL, the storage engine is responsible for managing the storage of data on disk. It handles the organization of data files, indexes, and free space management. The storage engine in PostgreSQL is designed to efficiently store and retrieve data, providing high performance and data integrity.

Some of the key features of the storage engine in PostgreSQL include:

Data File Organization: PostgreSQL organizes data into tablespaces, which are collections of data files on disk. Each table and index in PostgreSQL is stored as a separate data file within a tablespace. The storage engine manages the allocation and deallocation of space within data files, ensuring efficient storage and retrieval of data.

Indexing: PostgreSQL supports various types of indexes, including B-tree, hash, GIN (Generalized Inverted Index), and GiST (Generalized Search Tree). Indexes are used to accelerate data retrieval by providing a fast lookup mechanism for accessing data based on indexed columns. The storage engine manages the creation, maintenance, and usage of indexes to optimize query performance.

Free Space Management: As data is added, modified, or deleted in PostgreSQL, free space is created within data files. The storage engine is responsible for managing this free space and reusing it efficiently to minimize the overhead of file growth and fragmentation. This includes techniques such as free space map management and automatic vacuuming to reclaim unused space.

Concurrency Control: PostgreSQL supports concurrent access to the database by multiple users and transactions. The storage engine manages concurrent access to data files, ensuring that multiple transactions can read and write data concurrently without conflicts. This is achieved through various techniques, such as multi-version concurrency control (MVCC), which allows multiple versions of a row to coexist in the same data file.

Durability: PostgreSQL ensures that changes made to the database are durable and survive system failures. The storage engine manages transaction logs, also known as write-ahead logs (WAL), which record all changes made to the database. These logs are used for recovery in case of system failures, ensuring that changes made by committed

transactions are not lost.

Compression: PostgreSQL supports various compression techniques, such as table-level and column-level compression, to reduce the size of data on disk and improve storage efficiency. The storage engine handles the compression and decompression of data during storage and retrieval, ensuring that compressed data is transparently processed by the database engine.

Overall, the storage engine in PostgreSQL plays a critical role in managing the storage of data on disk, including data file organization, indexing, free space management, concurrency control, durability, and compression. These features contribute to the high performance, data integrity, and scalability of PostgreSQL as a robust and reliable DBMS.

3.3.4 POSTGRESQL CLUSTER OVERVIEW

Cluster Engine in PostgreSQL:

- PostgreSQL does not have a built-in cluster engine, but it supports various methods for achieving clustering and high availability, such as logical replication, streaming replication, and third-party tools like Pgpool-II and Patroni.
- Logical replication allows replicating specific tables or databases to other PostgreSQL instances, providing flexibility in replicating only the required data.
- Streaming replication is a built-in feature of PostgreSQL that allows replicating the entire database to a standby server in real-time.
- Pgpool-II is a popular open-source tool that provides connection pooling, load balancing, and replication features for PostgreSQL, allowing users to create a clustered setup.
- Patroni is another popular open-source tool that provides automated high availability and clustering for PostgreSQL, allowing users to create a highly available and resilient PostgreSQL cluster.

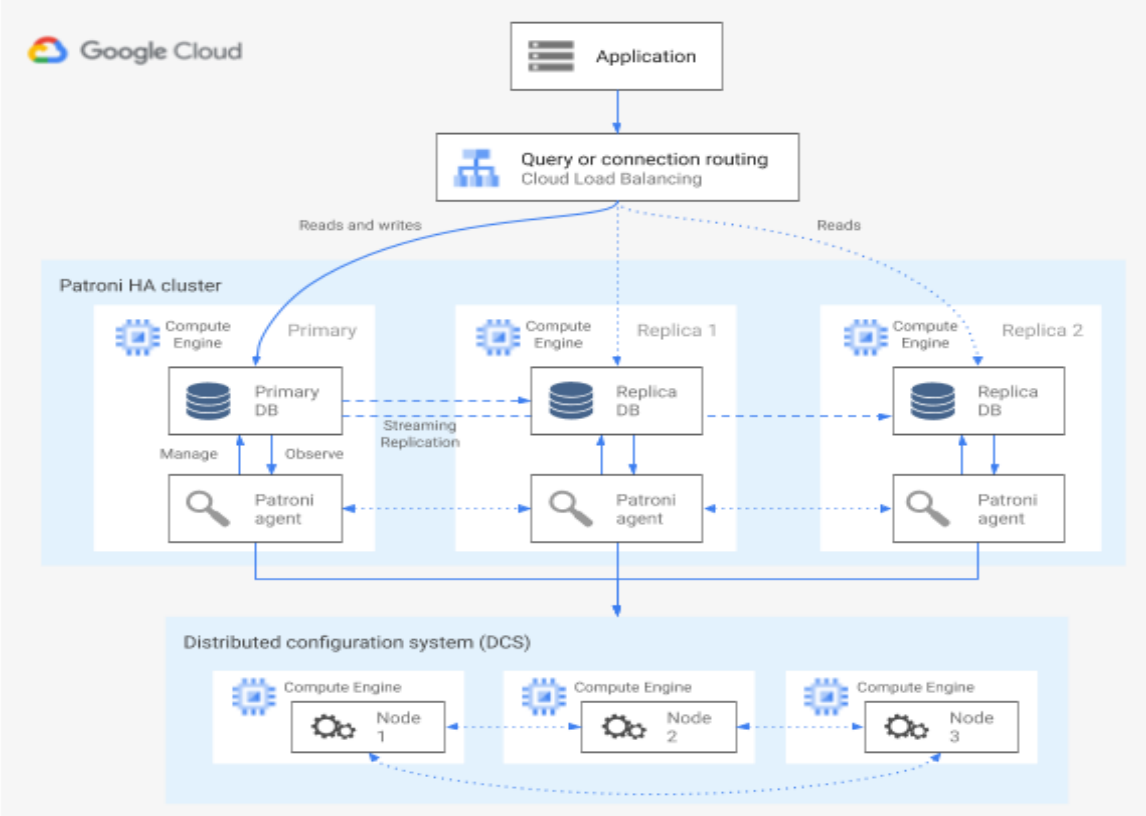


Fig 3.3.4 POSTGRESQL Cluster

CHAPTER 4

SYSTEM STUDY

4.1 AUTOMATED FILE GENERATION:

Automated file generation is a critical aspect of modern data processing and information management. It involves the use of software tools, techniques, and processes to automatically create files, such as reports, spreadsheets, documents, and other data files, without manual intervention. Automated file generation can significantly streamline and accelerate data processing workflows, reduce human error, and improve efficiency and accuracy in generating files for various business and operational needs.

A literature survey on automated file generation would typically involve a comprehensive review of existing research, studies, and publications related to this field. Some potential topics that could be covered in an automated file generation study include:

Techniques and methodologies for automated file generation: This could include a review of different approaches, algorithms, and methods used in automated file generation, such as template-based generation, data-driven generation, rule-based generation, and machine learning-based generation. The study could evaluate the strengths, limitations, and suitability of different techniques for various file types and data processing requirements.

Tools and technologies for automated file generation: This could involve a review of software tools, libraries, frameworks, and technologies that are commonly used for automated file generation, such as programming languages (e.g., Python, Java, R), reporting and data visualization tools (e.g., Tableau, Power BI), document generation tools (e.g., DocuSign, Adobe Sign), and ETL (Extract, Transform, Load) tools (e.g., Apache NiFi, Talend).

Applications and use cases of automated file generation: This could involve an exploration of different domains and industries where automated file generation is extensively used, such as finance, healthcare, marketing, logistics, and e-commerce. The study could review the specific use cases, benefits, and challenges of automated file generation in these domains, including how it has improved business processes, decision-

making, and data-driven insights.

Evaluation and benchmarking of automated file generation: This could involve a review of existing evaluation metrics, benchmark datasets, and performance comparisons for automated file generation. The study could analyze the accuracy, efficiency, scalability, and usability of different automated file generation techniques and tools and identify gaps and areas for further improvement.

Challenges and future directions in automated file generation: This could involve a discussion of the current challenges, limitations, and open research questions in automated file generation, such as handling complex data structures, ensuring data privacy and security, handling large-scale data, and supporting real-time or near-real-time file generation. The study could also highlight potential future directions, emerging trends, and opportunities for research and development in this field.

Overall, an automated file generation study could provide insights into the state of the art, challenges, and opportunities in this field, and contribute to the advancement of automated file generation techniques, tools, and applications in various domains.

4.2 DATA VALIDATION:

Data validation is a critical step in the data processing and data management lifecycle. It refers to the process of verifying and validating data to ensure its accuracy, consistency, integrity, and reliability. Data validation is typically performed at various stages of data processing, such as during data entry, data integration, data transformation, and data loading, to ensure that data meets specific quality standards and conforms to predefined rules or business requirements. The importance of data validation cannot be overstated, as inaccurate or inconsistent data can have significant implications on business decision-making, operations, and performance.

4.3 EXISTING SYSTEM

The existing system of file generation was using PHP scripts which seemed to be tedious task. Writing scripts for each band every client requirement is such a time consuming and requires lot of effort.

4.3.1 DISADVANTAGES:

- Time consuming
- Requires lot of effort

4.4 PROPOSED SYSTEM:

Using 4 tables the entire data is set to be fetched and using the ETL functions the SQL statements is said to be produced. The Logic is set it such a way that it picks the file requirements along with the data formats, quotes etc., and produces the file and updates the process status to success.

4.4.1 ADVANTAGES:

- Most convenient way of file generation
- Consumes less time

CHAPTER 5

DESIGN AND IMPLEMENTATION

5.1 SYSTEM ARCHITECTURE

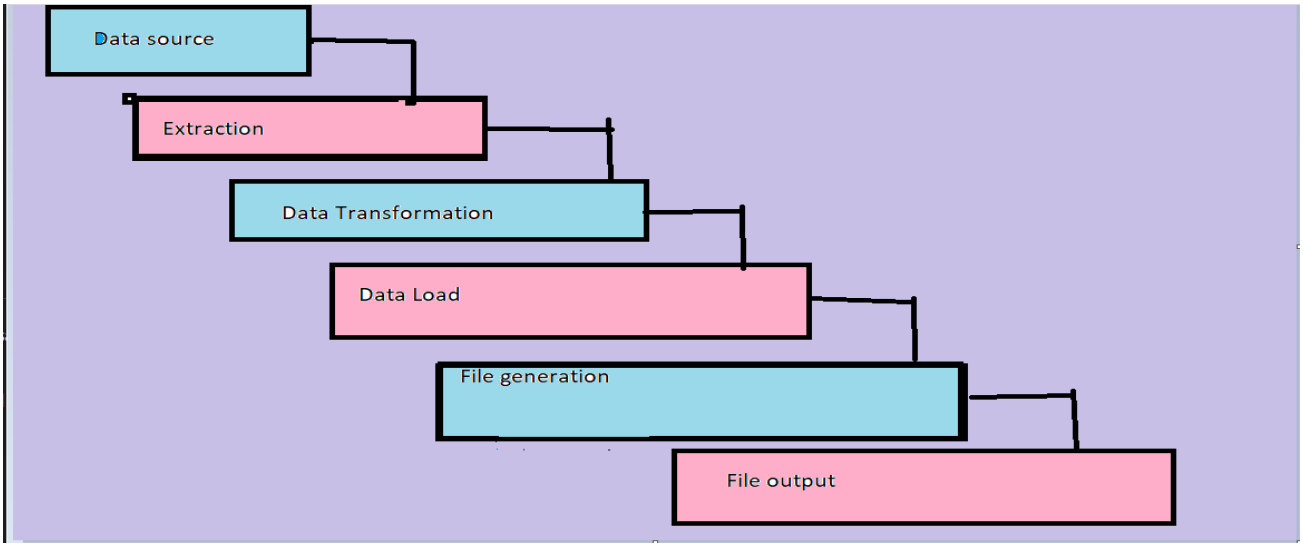


Fig 5.1 System Architecture

Here is a simplified system architecture diagram for file generation using ETL (Extract, Transform, Load) process:

Data Sources:

Data sources can include various types of data, such as structured, semi-structured, or unstructured data, from different sources such as databases, files, APIs, or external systems.

Extraction:

In the extraction phase, data is extracted from the data sources and loaded into a staging area. This can involve various techniques such as batch processing, real-time streaming, or event-driven extraction, depending on the requirements of the file generation process.

Data Transformation:

In the data transformation phase, data from the staging area is transformed into the desired format or structure for the file generation process. This can involve data cleansing, data enrichment, data aggregation, data normalization, data validation, and other data transformation operations, depending on the specific requirements and

business rules.

Data Load:

In the data load phase, the transformed data is loaded into the target system, which can be a file storage system or a file generation tool. This can involve various techniques such as batch loading, real-time loading, or incremental loading, depending on the requirements of the file generation process.

Generation:

In the file generation phase, the transformed and loaded data is used to generate the desired file format, such as CSV, Excel, XML, JSON, or any other custom format, based on the business requirements.

File Output:

The generated file is then stored in the file storage system or delivered to the desired destination, such as a file server, FTP server, cloud storage, or any other target system, as per the requirements of the file generation process.

Monitoring and Logging:

Throughout the ETL process, monitoring and logging mechanisms are implemented to track the progress of the data flow, detect errors or anomalies, and generate logs or alerts for monitoring and troubleshooting purposes.

ETL Tools:

ETL tools, such as Pentaho, Informatica, Microsoft SQL Server Integration Services (SSIS), Apache NiFi, or other custom-built ETL frameworks, are used to automate and streamline the extraction, transformation, and loading processes, as well as to provide various data integration, data transformation, and data management functionalities.

TABLES

Master Table -

This table holds the details of client's/file level requirement details

Master | Detail | Request | Main

Table Name: MASTER (This table holds the details of the client/file level requirement details)		
Column Name	**Data Type**	**Valid Values**
Sno	Number	Surrogate Key
Client_ID	Varchar2	
Client_Code	Varchar2	
Client_Name	Varchar2	
File_Format	Varchar2	XML, EXCEL,DELIMITER, FIXEDLENGTH
Delimiter_Char	Varchar2	|, #, $, , (no muli-byte character)
Qouted_Identifiers_Reqd	Varchar2	Y, N (Applicable only for Delimiter file types)
Qouted_Identifier	Varchar2	", '
Output_File_Folder	Varchar2	
Output_File_Name	Varchar2	abc-DeductionFile.csv

Detail Table -

This table holds the details of the field level requirement details

Master | Detail | Request | Main

Table Name: DETAIL (This table holds the details of the field ils - for both Test and Prod files)	
Column Name	**Valid Values**
Sno	Surrogate Key
DFCM_Sno	Config Master Surrogate Key
Client_ID	
Client_Code	
Field_Sequence	
Field_Name	
Field_Length	
Fixed_Length_Filler	SPACE, 0 (Default SPACE)
Right_Left_Pad	L/R
Data_Field	
Data_Format_Reqd	Y, N
Data_Format	
Qouted_Identifiers_Reqd	Y, N

Request table -

This table holds the request of which client is ready for the process

Master | Detail | Request | Main

Table Name: REQUEST	
Column Name	**Data Type**
Sno	Number
DFCM_Sno	Number
Client_ID	Varchar2
Client_Code	Varchar2
Client_Name	Varchar2
File_Name	Varchar2
Process_Start_Time	Date
Process_End_Time	Date
Process_Status	Varchar2

Main Table -

This table holds the actual data of the clients.

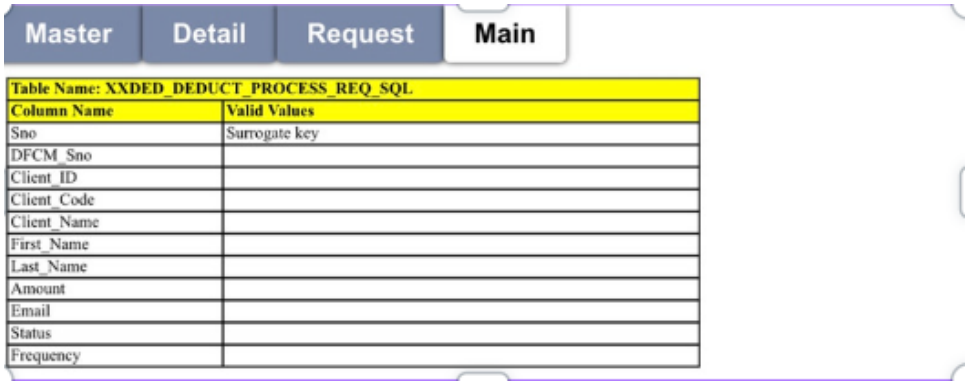


Table Name: XXDED_DEDUCT_PROCESS_REQ_SQL	
Column Name	**Valid Values**
Sno	Surrogate key
DFCM_Sno	
Client_ID	
Client_Code	
Client_Name	
First_Name	
Last_Name	
Amount	
Email	
Status	
Frequency	

5.2 MODULES

1. Client request filtration
2. Detail level filtration
3. Query generation
4. File output
5. Update the success/failure.

5.3 MODULE DESCRIPTION

A module is a separate unit of software or hardware. Characteristics of modular components include portability and interoperability which allows them to function in another system with the components of other systems.

5.3.1 CLIENT REQUEST FILTERATION

In this module, the client level details are filtered on the basis of who requires the file to be generated now the ststus is set to ready. When the status is set to ready that particlar customer client id /client code is a primary key which is set as a paramter to call the next configuration.The process status is updated to in progress.

5.3.2 DETAIL LEVEL FILTERATION

In this module, the length of each and every data filed required is said to be calculated and fed to a new filed. So, that there won’t be any latency in the file level accommodation

5.3.3 QUERY GENERATION

In this module, the parameter that is set in module 1 will call all the details in the detail

table, and the formula and logic is set in such a way it does all the transformations required in the filed level and generated the SQL query.

5.3.4 FILE OUTPUT

In this module, the query reads in the table input step where the table's connection is established to PostgreSQL in the backend for the file to be generated.

5.3.5 UPDATE THE STATUS

In this module, the request table is said to be update once the file is generated and it sends a mail notification regarding the file generation success or failure

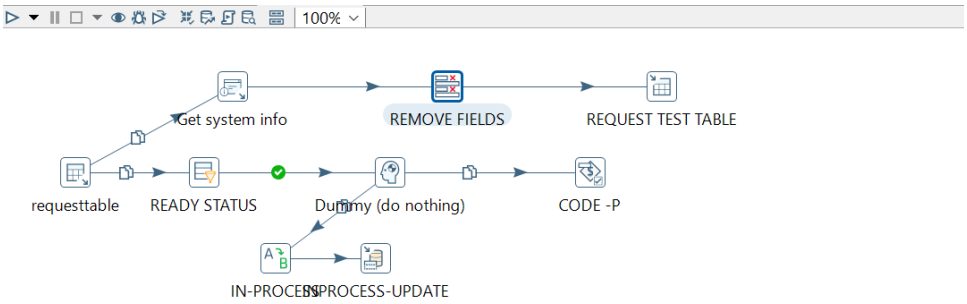
CHAPTER 6
CONCLUSION

6.1 CONCLUSION

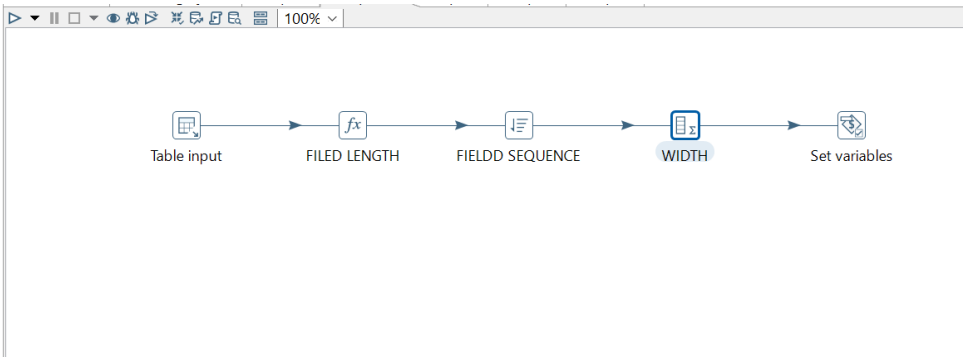
Automated file generation is said to be made easy without any, manual work just by feeding the requirements in the table. It automatically provides you the desired file. A huge amount of data can be transformed and loaded in just few seconds of time. Instead of using a PHP script derived for each and every file, this pentaho job indeed does that only by feeding the requirements in the table. Provided the table requirements and specifications, it is said that the file generation which is automated can be done easily.

APPENDICES
SAMPLE CODE

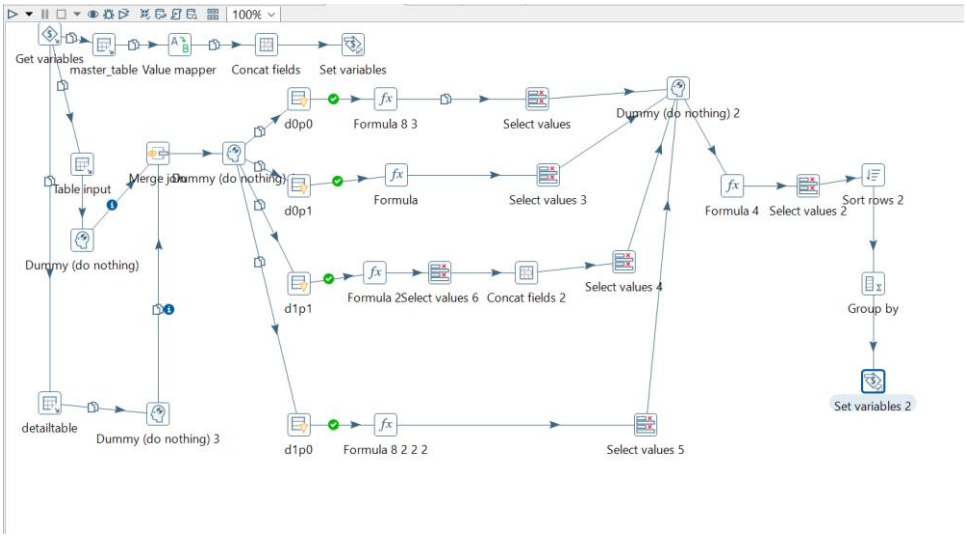
STEP 1



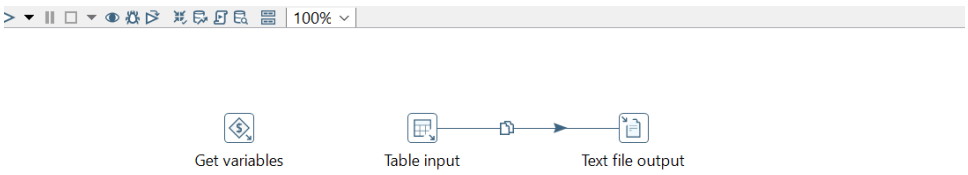
STEP 2



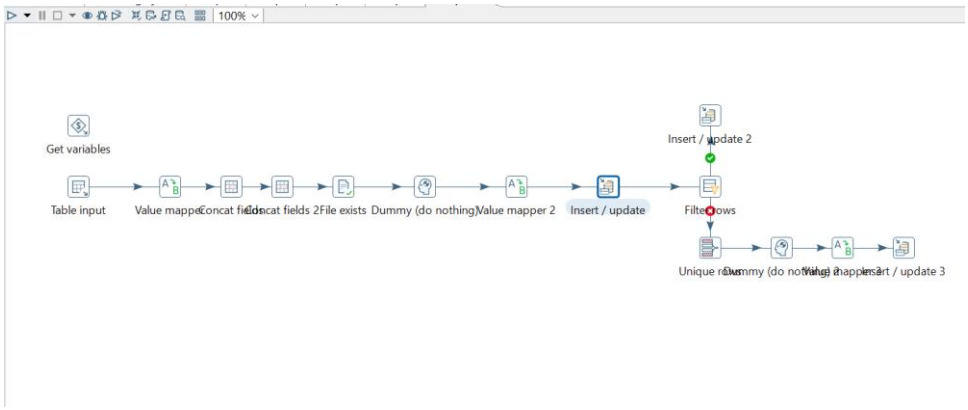
STEP 3



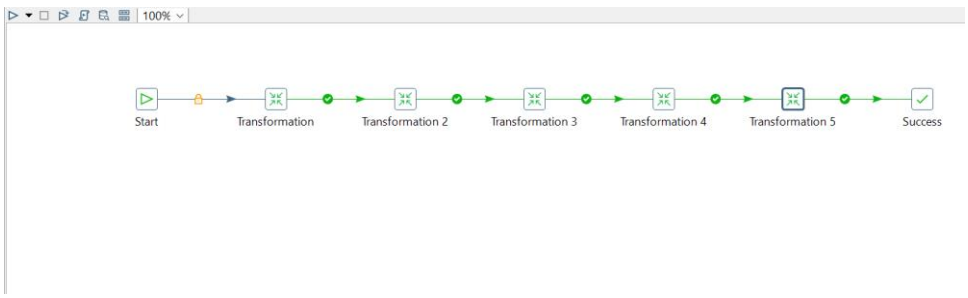
STEP 4



STEP 5



FINAL JOB



TABLES
MASTER TABLE

	sno bigint	client_id bigint	client_code character varying	client_name character varying (8)	file_format character varying (11)	delimiter_char character varying (1)	qouted_identifiers_reqd boolean
1	1	2201	PP01A	renold	DELIMITER	|	true
2	2	2202	PP02A	spunk	DELIMITER	;	true
3	3	2203	PP03A	champion	FIXEDLENGTH	[null]	false
4	4	2204	PP04A	zara	EXCEL	[null]	false
5	5	2205	PP05A	shein	EXCEL	[null]	false

DETAIL TABLE

Sno	DFCM_Sn	Client_ID	Client_Co	Field_Seq	Field_Nam	Field_Leng	Fixed_Len	Right_Left	Data_Fielc	Data_Forr	Data_Forr	Qouted_Identifiers_Reqd
1	1	2201	PP01A	1	First_Nam	NULL	NULL	NULL	First_Nam	FALSE	NULL	TRUE
2	1	2201	PP01A	2	Last_Nam	NULL	NULL	NULL	Last_Nam	FALSE	NULL	TRUE
3	2	2202	PP02A	1	First_Nam	NULL	NULL	NULL	First_Nam	FALSE	NULL	TRUE
4	2	2202	PP02A	2	Amount	NULL	NULL	NULL	Amount	TRUE	'99.99'	TRUE
5	3	2203	PP03A	3	Amount	5	'0'	RPAD	Amount	TRUE	'99.99'	FALSE
6	3	2203	PP03A	2	First_Nam	15	'0'	LPAD	First_Nam	FALSE	NULL	FALSE
7	4	2204	PP04A	1	FName	NULL	NULL	NULL	First_Nam	FALSE	NULL	FALSE
8	5	2205	PP05A	1	Status	NULL	NULL	NULL	Status	FALSE	NULL	FALSE

REQUEST TABLE

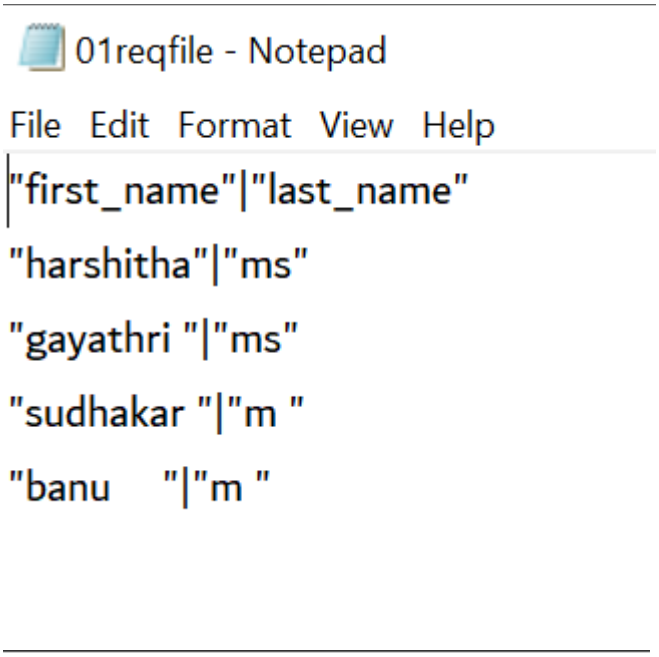
sno	dfcm_sno	client_id	client_cod	client_nan	file_name	process_s	process_e	process_status
1	1	2201	PP01A	renold	01reqfile			Sucess
2	2	2202	PP02A	spunk	02reqfile			Sucess
3	3	2203	PP03A	champion	03reqfile			Sucess
4	4	2204	PP04A	zara	04reqfile			Sucess
5	5	2205	PP05A	shein	05reqfile			Sucess

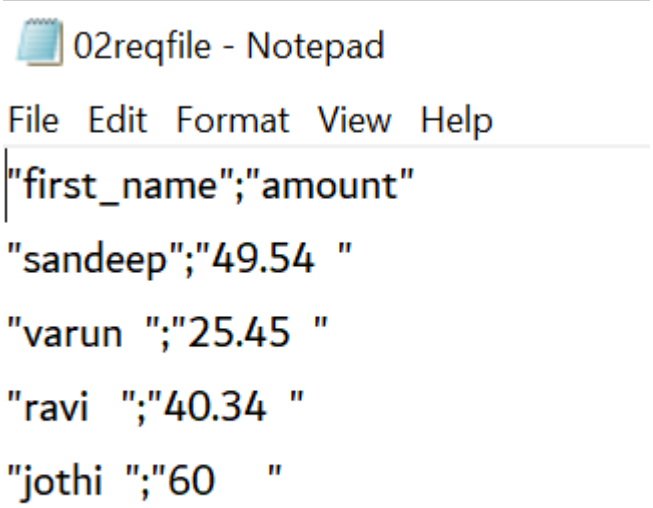
MAIN TABLE

sno	dfcm_sno	client_id	client_cod	client_nan	first_nam	last_name	amount	email	status	frequency
1	1	2201	PP01A	renold	harshitha	ms	20.887	harshithar	Active	2
2	1	2201	PP01A	renold	gayathri	ms	15.456	gaytahrim	Active	4
3	1	2201	PP01A	renold	sudhakar	m	40	sudhakars	Terminate	2
4	1	2201	PP01A	renold	banu	m	35.436	banum@g	Active	2
5	2	2202	PP02A	spunk	sandeep	pr	49.5467	sandeepp	Active	4
6	2	2202	PP02A	spunk	varun	pr	25.453	varunpr@	Active	2
7	2	2202	PP02A	spunk	ravi	p	40.34	ravip@gm	Active	1
8	2	2202	PP02A	spunk	jothi	pr	60	jothipr@g	Terminate	2
9	3	2203	PP03A	champion	sathish	ps	25.345	sathishps(	Active	1
10	3	2203	PP03A	champion	neela	ps	30.3452	neelaps@	Active	1
11	3	2203	PP03A	champion	geethika	ps	45.231	geethikap:	Active	2
12	3	2203	PP03A	champion	sri	ps	56.89	srips@gm	Active	2
13	4	2204	PP04A	zara	kiara	ad	38.123	kiaraad@g	Terminate	1
14	4	2204	PP04A	zara	sid	ml	45.12	sidml@gm	Terminate	2
15	4	2204	PP04A	zara	alia	bh	70.234	aliabh@gr	Active	2
16	4	2204	PP04A	zara	dhruv	v	35.567	dhruvv@g	Active	4
17	5	2205	PP05A	shein	suriya	mh	28.876	suriyamh(	Active	2
18	5	2205	PP05A	shein	nila	mayo	38.67	nilamayo(	Active	2
18	5	2205	PP05A	shein	oviya	lm	45.01	oviyalm@	Terminate	4

FILENAME DERIVATION:

FILENAME OUTPUT





03reqfile - Notepad

File Edit Format View Help

first_nameamount

00000000sathish25.34

0000000000neela30.34

0000000geethika45.23

000000000000sri56.89

fname
kiara
sid
alia
dhruv

status
Active
Active
Terminated
Terminated

REFERENCES

1. A SURVEY ON ETL TOOLS. (2016, March 31). *International Journal of Advance Engineering and Research Development*, *3*(03). https://doi.org/10.21090/ijaerd.030330
2. Astriani, W., & Trisminingsih, R. (2016). Extraction, Transformation, and Loading (ETL) Module for Hotspot Spatial Data Warehouse Using Geokettle. *Procedia Environmental Sciences*, *33*, 626–634. https://doi.org/10.1016/j.proenv.2016.03.117
3. Wojciechowski, Artur. "E-ETL: Framework for Managing Evolving ETL Workflows." *Foundations of Computing and Decision Sciences*, vol. 38, no. 2, 1 June 2013, pp. 131–142, https://doi.org/10.2478/fcds-2013-0005.
4. Kargín, Yağíz, et al. "Lazy ETL in Action." *Proceedings of the VLDB Endowment*, vol. 6, no. 12, 28 Aug. 2013, pp. 1286–1289, https://doi.org/10.14778/2536274.2536297.

PROGRAM OUTCOMES

PO No.	Graduate Attribute	Program Outcomes (POs)
PO1	**Engineering knowledge**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization for the solution of complex engineering problems.
PO2	**Problem analysis**	Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/ development of solutions**	Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural, societal, and environmental considerations.
PO4	**Conduct investigations of complex Problems**	Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
PO5	**Modern tool usage**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools, including prediction and modeling to complex engineering activities, with an understanding of the limitations.
PO6	**The engineer and society**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
PO7	**Environment and Sustainability**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO8	**Ethics**	Apply ethical principles and common to professional ethics and responsibilities and norms of the engineering practice.
PO9	**Individual and team work**	Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	**Communication**	Communicate effectively on complex engineering activities with the engineering community and with the society at large, such as, being able to comprehend and withe effective reports and design documentation, make effective presentations, and give and receive clear instructions.
PO11	**Project management and finance**	Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
PO12	**Life-long learning**	Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Mapping of Program outcomes with the Project titled "**DESIRED FILE GENERATION USING ETL.**"

PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12

PROGRAM SPECIFIC OUTCOMES
B.E COMPUTER SCIENCE AND ENGINEERING

PSO No.	Program Specific Outcomes
PSO1	To analyze, design and develop computing solutions by applying foundational concepts of computer science and engineering.
PSO2	To apply software engineering principles and practices for developing quality software for scientific and business applications.
PSO3	To adapt to emerging Information and Communication Technologies (ICT) to innovate ideas and solutions to existing/novel problems

PSO1	PSO2	PSO3

Signature of Guide