MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
DE LA RECHERCHE SCIENTIFIQUE ET DE L'INNOVATION

UNIVERSITÉ SULTAN MOULAY SLIMANE

ECOLE NATIONALE DES SCIENCES APPLIQUÉS DE KHOURIBGA

Mémoire de fin d'études

pour l'obtention du

Diplôme d'Ingénieur d'État

Filière : Informatique et Ingénierie des Données

Présenté par :
Ziyad **RAKIB**

Titre du mémoire

Sous la supervision :
Abdelghani **Ghazdali**
Nidal **Lamghari**

Membres du jury :

Nom et prénoms du président	Entité	Président
Nom et prénoms de l'examinateur	Entité	Examinateur
Nom et prénoms du rapporteur	Entité	Rapporteur
Nom et prénoms du rapporteur	Entité	Rapporteur

Année Académique : **2023**

ehrig2006graph

Sommaire

Dédicace

“

À mLorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus,

À Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus,

À Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus,

À tous ceux qui me sont chers, à vous tous

Merci.

”

- Abdelghani

Remerciements

Nos remerciements

Résumé

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Mots clés : Optimisation des itinéraires, Problème du voyageur de commerce avec contrainte périodique, Apprentissage automatique, Classification.

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin posuere euismod neque, non semper nibh viverra sed. Praesent ut varius magna. Fusce ipsum ante, semper nec interdum at, semper et lacus. Nulla ultrices magna a fringilla finibus.

Keywords : Route optimization, Periodic traveling salesman problem , Machine learning, Classification.

ملخص

في هذا التقرير، وردت إشارة إلى تنفيذ نظام كشف الوجه. والهدف من ذلك هو الكشف عن وجوه الناس المهتمين في منشور معين. لقد إستعملنا لهذا الغرض لوحة الكترونية Raspberry pi 2 بالإضافة إلى وحدة الكاميرا الخاصة بها لملاحقة الوجوه و العين. يستخدم تطبيق كشف الوجه خوارزمية قوية لفيولا جونز مع تدريب المصنف للكشف عن الوجه الإنساني والعينين، عن طريق اختيار أفضل مصنف ممكن. سيتم حفظ البيانات لكل إعلان في قاعدة بيانات MySQL . وقد تم إنشاء تطبيق ويب لهذا الغرض والذي سيمكن المستخدم من معرفة عدد الأشخاص والمدة الإجمالية للمشاهدة في الفترة (ساعة، يوم، شهر) على شكل رسم بياني.

كلمات مفتاحية : ك ك

Table des matières

Table des figures

Liste des tableaux

Liste des sigles et acronymes

API	Application programming interface
VSC	Visual Studio Code
DL	Delta Lake
REST	RepresEntational State Transfer
AWS	Amazon Web Services
GCP	Google Cloud Platform
AZR	Microsoft Azure
S3	Simple Storage Service
IAM	Identity and Access Management
MinIO	Minimal Object Storage
SQL	Structured Query Language
DB	Database
ACID	Atomicity, Consistency, Isolation, Durability
OLTP	Online Transaction Processing
OLAP	Online Analytical Processing
ReactJS	React JavaScript
SPA	Single Page Application
JS	JavaScript
CSS	Cascading Style Sheets
HTML	Hypertext Markup Language
UI	User Interface
API	Application Programming Interface
Spark	Apache Spark
Hadoop	Apache Hadoop
HDFS	Hadoop Distributed File System
YARN	Yet Another Resource Negotiator
MapReduce	MapReduce Programming Model
Metadata	Data about Data
ETL	Extract, Transform, Load
ELT	Extract, Load, Transform
CRM	Customer Relationship Management
RDBMS	Relational Database Management System

Introduction Générale

bla bla bla Def acronyme (acronyme) puis Acroglo et enfin glossaire

Chapitre 1

fknznefkljebngflkjen

Introduction

1.1 -

blablabla

1.2 -

blablabla

Conclusion

Chapitre 2

Delta Lake

Introduction

Data warehouses and data lakes are the most common central data repositories employed by most data-driven organizations today, each with its own strengths and tradeoffs. For one, while data warehouses allow businesses to organize historical datasets for use in business intelligence (BI) and analytics, they quickly become more cost-intensive as datasets grow because of the combined use of compute and storage resources. Additionally, data warehouses can't handle the varied nature of data (structured, unstructured, and semi-structured) seen today.

In this chapter, we will explore the key features of Delta Lake, how it works, and why it is a good choice for big data processing. We will also provide examples of how to use Delta Lake with other big data tools, such as Hadoop, Spark, and Trino.

2.1 Definition

Delta Lake is an open-source storage layer built atop a data lake that confers reliability and ACID (Atomicity, Consistency, Isolation, and Durability) transactions. It enables a continuous and simplified data architecture for organizations. A data lake stores data in Parquet formats and enables a lakehouse data architecture, which helps organizations achieve a single, continuous data system that combines the best features of both the data warehouse and data lake while supporting streaming and batch processing.

2.2 How Delta Lake Works

A Delta Lake enables the building of a data lakehouse. Common lakehouses include the Databricks Lakehouse and Azure Databricks. This continuous data architecture allows organizations to harness

the benefits of data warehouses and data lakes with reduced management complexity and cost. Here are some ways Delta Lake improves the use of data warehouses and lakes :

- **Enables a lakehouse architecture :** Delta Lake enables a continuous and simplified data architecture that allows organizations to handle and process massive volumes of streaming and batch data without the management and operational hassles involved in managing streaming, data warehouses, and data lakes separately.
- **Enables intelligent data management for data lakes :** Delta Lake offers efficient and scalable metadata handling, which provides information about the massive data volumes in data lakes. With this information, data governance and management tasks proceed more efficiently.
- **Schema enforcement for improved data quality :** Because data lakes lack a defined schema, it becomes easy for bad/incompatible data to enter data systems. There is improved data quality thanks to automatic schema validation, which validates DataFrame and table compatibility before writes.
- **Enables ACID transactions :** Most organizational data architectures involve a lot of ETL and ELT movement in and out of data storage, which opens it up to more complexity and failure at node entry points. Delta Lake ensures the durability and persistence of data during ETL and other data operations. Delta lake captures all changes made to data during data operations in a transaction log, thereby ensuring data integrity and reliability during data operations.

2.3 Delta Lake Architecture Diagram

Delta Lake is an improvement from the lambda architecture whereby streaming and batch processing occur parallel, and results merge to provide a query response. However, this method means more complexity and difficulty maintaining and operating both the streaming and batch processes. Unlike the lambda architecture, Delta Lake is a continuous data architecture that combines streaming and batch workflows in a shared file store through a connected pipeline.

The stored data file has three layers, with the data getting more refined as it progresses downstream in the dataflow :

- **Bronze tables :** This table contains the raw data ingested from multiple sources like the Internet of Things (IoT) systems, CRM, RDBMS, and JSON files.
- **Silver tables :** This layer contains a more refined view of our data after undergoing transformation and feature engineering processes.
- **Gold tables :** This final layer is often made available for end users in BI reporting and analysis or use in machine learning processes.

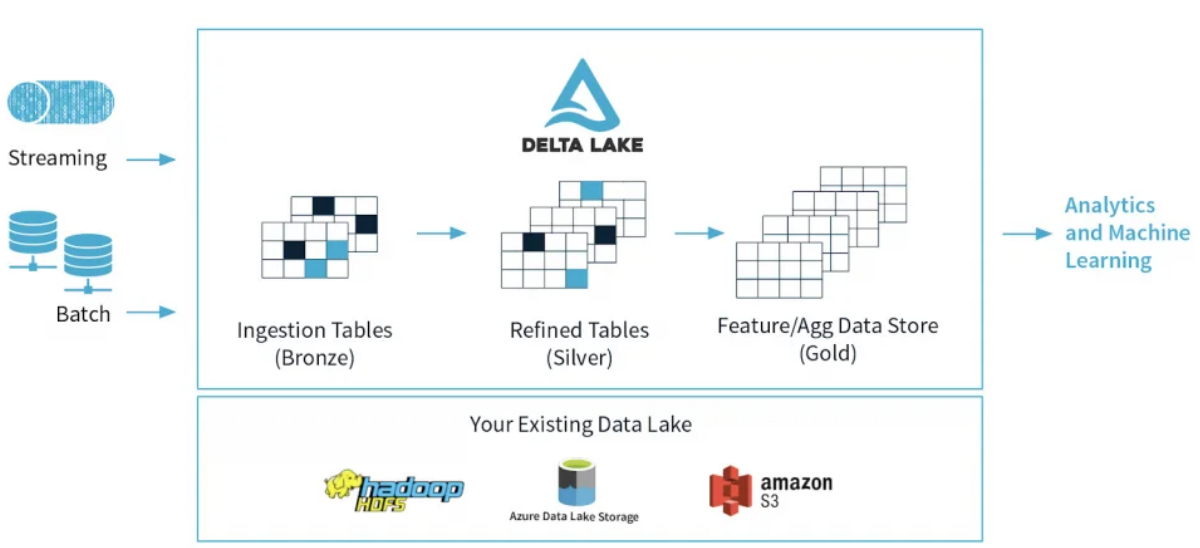

FIGURE 2.1 : Delta Lake multi-hop architecture

2.4 Key benefits and features of Delta Lake

- **Audit trails and history :** In Delta Lake, every write exists as a transaction and is serially recorded in a transaction log. Therefore, any changes or commits made to the transaction log are recorded, leaving a complete trail for use in historical audits, versioning, or for time traveling purposes. This Delta Lake feature helps ensure data integrity and reliability for business data operations.

- **Time traveling and data versioning :** Because each write creates a new version and stores the older version in the transaction log, users can view/revert to older data versions by providing the timestamp or version number of an existing table or directory to the Sparks read API. Using the version number provided, the Delta Lake then constructs a full snapshot of the version with the information provided by the transaction log. Rollbacks and versioning play a vital role in machine learning experimentation, whereby data scientists iteratively change hyperparameters to train models and can revert to changes if needed.

- **Unifies batch and stream processing :** Every table in a Delta Lake is a batch and streaming sink. With Sparks structured streaming, organizations can efficiently stream and process streaming data. Additionally, with the efficient metadata handling, ease of scale, and ACID quality of each transaction, near-real-time analytics become possible without utilizing a more complicated two-tiered data architecture.

- **Efficient and scalable metadata handling :** Delta Lakes store metadata information in the transaction log and leverages Spark's distributed processing power to quickly process and efficiently read and handle large volumes of data metadata, thus improving data governance.

- **ACID transactions :** Delta Lakes ensure that users always see a consistent data view in a table or directory. It guarantees this by capturing every change made in a transaction log and isolating

it at the strongest isolation level, the serializable level. In the serializable level, every existing operation has and follows a serial sequence that, when executed one by one, provides the same result as seen in the table.

- **Data Manipulation Language operations :** Delta Lakes supports DML operations like updates, deletes, and merges, which play a role in complex data operations like change-data-capture (CDC), streaming upserts, and slowly-changing-dimension (SCD). Operations like CDC ensure data synchronization in all data systems and minimizes the time and resources spent on ELT operations. For instance, using the CDC, instead of ETL-ing all the available data, only the recently updated data since the last operation undergoes a transformation.
- **Schema enforcement :** Delta Lakes perform automatic schema validation by checking against a set of rules to determine the compatibility of a write from a DataFrame to a table. One such rule is the existence of all DataFrame columns in the target table. An occurrence of an extra or missing column in the DataFrame raises an exception error. Another rule is that the DataFrame and target table must contain the same column types, which otherwise will raise an exception. Delta Lake also use DDL (Data Definition Language) to add new columns explicitly. This data lake feature helps prevent the ingestion of incorrect data, thereby ensuring high data quality.
- **Compatibility with Spark's API :** Delta Lake is built on Apache Spark and is fully compatible with Spark API, which helps build efficient and reliable big data pipelines.
- **Flexibility and integration :** Delta lake is an open-source storage layer and utilizes the Parquet format to store data files, which promotes data sharing and makes it easier to integrate with other technologies and drive innovation.

Conclusion

Delta Lake is an important tool for big data processing, providing reliable data management and ensuring data integrity at scale. Its ACID transactions, schema enforcement, and data versioning features make it a popular choice for companies that need to process large amounts of data with high accuracy and reliability.

By using Delta Lake, data engineers and data scientists can easily manage data quality, track data lineage, and collaborate on data analysis projects. With its seamless integration with other big data tools, such as Spark, Hadoop, and Trino, Delta Lake provides a powerful solution for big data processing that can help companies gain insights from their data faster and more efficiently.

Chapitre 3

–

Introduction

3.1 -

blablabla

3.2 -

Algorithme 1 : Inverse

```
   Data : x
   Result : r
1  begin
2  |  if x ≠ 0 then
3  |  |  r ← 1/x;
4  |  end
5  end
```

Conclusion

Conclusion Générale

Bla bla bla [1]

Bibliographie

[1] A. C. H. Ehrig, U. M. L. Ribeiro, and G. Rozenberg. Graph transformations. 2006.