UAB
Universitat Autònoma de Barcelona

Dipòsit digital de documents de la UAB

This is the **published version** of the bachelor thesis:

Cladera Sensat, Grau. *Data Lake analític per a dades obertes (Open Data).* Treball de Final de Grau (Universitat Autònoma de Barcelona), 2026 (Grau en Enginyeria de Dades)

This version is available at https://ddd.uab.cat/record/337349

under the terms of the CC BY-SA license.

Data Lake Analític per a Dades Obertes

Grau Cladera Sensat

Resum

Resum– Aquest Treball de Fi de Grau presenta el disseny i implementació d'un Data Lake analític al núvol d'AWS, construït amb tecnologia Serverless i desplegat íntegrament mitjançant Terraform (IaC). Seguint l'arquitectura de medalló, el sistema ingereix de forma autònoma dades financeres (CoinGecko) i xarxes socials (Bluesky) via AWS Lambda (Capa Bronze). A continuació, s'utilitza AWS Glue i PySpark per netejar-les i convertir-les a Parquet (Capa Silver), i s'aplica Processament de Llenguatge Natural (Amazon Comprehend) per correlacionar l'opinió pública amb el preu de les criptomonedes (Capa Gold). Tota la infraestructura està protegida sota un model Zero-Trust amb AWS Lake Formation i controls de qualitat en línia (DQDL). Finalment, les dades s'exploten amb Amazon Athena i es visualitzen en quadres de comandament interactius a Grafana, aconseguint una plataforma escalable i amb un cost operatiu mínim (FinOps).

Paraules clau– Data Lake, Arquitectura Serverless, Infrastructure as Code (IaC), ETL, Governança de Dades, Anàlisi de Sentiment.

Abstract– This Bachelor's Thesis presents the design and implementation of an analytical Data Lake in the AWS cloud, built with Serverless technology and deployed entirely using Terraform (IaC). Following the medallion architecture, the system autonomously ingests financial data (CoinGecko) and social media posts (Bluesky) via AWS Lambda (Bronze Layer). Next, AWS Glue and PySpark are used to clean and convert the data into Parquet format (Silver Layer), and Natural Language Processing (Amazon Comprehend) is applied to correlate public opinion with cryptocurrency prices (Gold Layer). The entire infrastructure is protected under a Zero-Trust model using AWS Lake Formation and in-line data quality controls (DQDL). Finally, the data is efficiently queried with Amazon Athena and visualized in interactive dashboards in Grafana, achieving a highly scalable platform with minimal operational costs (FinOps).

Keywords– Data Lake, Serverless Architecture, Infrastructure as Code (IaC), ETL, Data Governance, Sentiment Analysis.

1 Introducció - Context del treball

Aquest Treball de Final de Grau presenta el disseny i la implementació d'un Data Lake i una canonada de dades ETL (Extract, Transform, Load) automatitzada, escalable i governada al núvol, basada íntegrament en tecnologia Serverless. El projecte s'ha concebut per ingerir informació financera de criptomonedes i extraccions de xarxes socials, sota la premissa analítica que el sentiment de l'opinió pública afecta el valor de mercat d'aquests actius digitals.

Per a la seva construcció, s'ha utilitzat l'ecosistema d'Amazon Web Services (AWS). El cicle de vida de la dada s'estructura seguint l'arquitectura de medalló (capes Bronze, Silver i Gold). Aquest flux integra serveis com AWS Lambda i EventBridge per a la ingesta reactiva, i Apache Spark (via AWS Glue) per al processament distribuït i la conversió cap a formats columnars optimitzats (Parquet). Addicionalment, s'ha incorporat Amazon Comprehend per extreure'n el sentiment mitjançant Processament de Llenguatge Natural (NLP), preparant així les dades per a la seva explotació visual a Grafana i en motors de bases de dades de grafs.

Des de la perspectiva de l'operació, tota la infraestructura s'ha desplegat com a codi (Infrastructure as Code, IaC) a través de Terraform, garantint-ne la replicabilitat. Així mateix, s'ha establert un marc de seguretat Zero-Trust mitjançant AWS Lake Formation per assegurar una governança de dades rigorosa, aplicant un control d'accés granular sota el principi de mínim privilegi.

Aquesta solució evidencia el potencial del paradigma Serverless per desplegar ecosistemes d'alt rendiment i elasticitat amb un cost operatiu mínim (FinOps), superant les limitacions estructurals dels models on-premise tradicionals. Tota l'execució s'ha dut a terme amb la metodología Kanban sota el marc metodològic d'ITIL 4, assegurant un pas a producció segur i enfocat en aportar valor a l'usuari.

• E-mail de contacte: graucladerasensat@gmail.com

• Treball tutoritzat per: Esteve Alquézar Mora (Departament d'Arquitectura de Computadors i Sistemes Operatius)

• Curs 2025/2026

2 ESTAT DE L'ART

L'evolució accelerada de la tecnologia exigeix una flexibilitat financera inassolible per a les infraestructures tradicionals on-premise. Es requereixen grans inversions inicials (model CapEx) i alts costos de manteniment (TCO) que actuen com a barrera per a la innovació. L'aparició dels proveïdors de núvol (hyperscalers com AWS) ha transformat aquesta dinàmica cap a un model de despeses operatives (OpEx) basat en el pagament per ús. Mitjançant el paradigma Serverless i l'adopció de pràctiques FinOps, s'eliminen els costos fixos de servidors infrautilitzats. Actualment, serveis d'emmagatzematge com Amazon S3 redueixen els costos a fraccions de cèntim per gigabyte, mentre que motors com Amazon Athena facturen exclusivament pel volum de dades escanejat, democratitzant l'accés a còmput d'alt rendiment per processar dades.

Paral·lelament, aquest abaratiment ha propiciat la retenció massiva d'informació (Big Data), posant de manifest la rigidesa dels sistemes tradicionals d'ETL i Data Warehouses. Aquests models històrics pateixen limitacions d'escalabilitat i són ineficients gestionant dades no estructurades (com textos lliures de xarxes socials). Per superar-ho, l'estat de l'art ha convergit en el Data Lake, una arquitectura que fusiona la flexibilitat amb el rendiment analític del Data Warehouse.

Implementat habitualment sota el patró de medalló (capes Bronze, Silver i Gold) i avalat per canonades de processament distribuït (com Apache Spark), el Lake aplica l'esquema només en el moment de la lectura (Schema-on-Read). Aquesta transició tecnològica és el pilar fonamental que permet netejar conjunts de dades dispars de forma elàstica i alimentar models d'Aprenentatge Automàtic i Processament del Llenguatge Natural (NLP), requisits indispensables per a l'extracció avançada de coneixement que està tenint lloc en l'actualitat.

2.1 Tecnologies: AWS i Terraform

Per al desenvolupament del treball, s'ha seleccionat un conjunt d'eines tecnològiques modernes i emprades actualment per les empreses del sector.

Amazon Web Services (AWS). L'adopció d'AWS com a proveïdor de Cloud Computing es justifica, en primer lloc, perquè és l'eina de serveis al núvol més utilitzada a nivell mundial. El seu ecosistema ha permès dissenyar una arquitectura totalment Serverless on s'abstrau per complet la gestió, el manteniment i l'aprovisionament de servidors físics tradicionals. Aquesta decisió és ideal per a les tasques de processament per lots (batch) d'aquest projecte, ja que els recursos de còmput s'encenen només durant l'execució i escalen a zero de manera automàtica quan el pipeline s'atura. Financerament, aquest comportament s'alinea de manera estricta amb la filosofia FinOps, ja que el model de pagament per ús redueix dràsticament el Cost Total de Propietat (TCO) respecte a un entorn on-premise.

Terraform. El desplegament de tota la infraestructura mitjançant Terraform es justifica per la seva condició de programari de referència i estàndard de la indústria per a la gestió d'Infraestructura com a Codi (IaC). Aquesta eina permet orquestrar de manera centralitzada i declarativa la totalitat dels recursos del Data Lake, des de les polítiques de seguretat d'IAM i Lake Formation fins a les xarxes privades de la VPC. L'enfocament basat en codi garanteix la replicabilitat absoluta de l'entorn, actua com una documentació viva del sistema i elimina errors humans, a més d'una ràpida capacitat de recuperació. En adició, s'alinea amb la metodologia de desenvolupament DevOps per agilitzar el pas a producció sota els estàndards del marc de governança d'ITIL 4. Finalment, és una peça clau de l'estratègia FinOps, ja que facilita la creació de recursos efímers que es destrueixen de forma programada per estalviar costos.

3 OBJECTIUS

L'objectiu principal d'aquest Treball de Final de Grau és dissenyar, implementar i avaluar una arquitectura de Data Lake analítica i governada al núvol, aplicant les millors pràctiques de l'enginyeria de dades i el paradigma Serverless per garantir un alt rendiment amb un cost operatiu mínim.

Per assolir aquesta fita principal, s'han definit els següents objectius específics:

- Extracció i Ingesta de Dades (Capa Bronze): Desenvolupar un mòdul d'extracció automatitzat capaç de consumir dades obertes i heterogènies des de diferents APIs (mercats financers via CoinGecko i textos de xarxes socials via Bluesky), emmagatzemant-les en el seu format natiu.
- Processament i Anàlisi de Sentiment (Capa Silver i Gold): Implementar canonades de processament distribuït per netejar, estandarditzar a un format columnar eficient (Parquet) i enriquir les dades mitjançant models de Processament del Llenguatge Natural (NLP).
- Orquestració i Automatització: Construir un flux de dades (Data Pipeline) basat en esdeveniments (event-driven) que garanteixi l'execució autònoma i resilient de tot el cicle de vida de la dada sense intervenció humana.
- Governança i Seguretat Zero-Trust: Aplicar els mecanismes necessaris per protegir la infraestructura, establint controls de qualitat intermedis i un control d'accés granular mitjançant AWS Lake Formation.
- Explotació Analítica i Visualització: Desplegar una capa de consum d'alta disponibilitat que permeti traduir les dades estructurades en quadres de comandament interactius (Grafana) i exploracions topològiques (bases de dades de grafs) per a la intel·ligència de negoci.
- Infraestructura com a Codi (IaC): Desplegar la totalitat de l'arquitectura de forma declarativa mitjançant Terraform per assegurar-ne la replicabilitat, auditabilitat i alineació amb els principis FinOps.

4 METODOLOGIA

El desenvolupament d'aquest projecte s'ha dut a terme seguint la metodología Kanban sota el marc metodològic d'ITIL 4, dividint el cicle de vida en fases de disseny, transició

i operació per garantir un pas a producció segur i auditable. Aquesta metodologia consisteix en cinc fases consecutives que garanteixen que el flux de dades sigui eficient. D'altra banda, Kanban permet dividir les tasques tot assignant un nivell de progrés (per fer, en procés, acabat). A nivell tècnic, la solució es materialitza en un Data Lake al núvol d'Amazon Web Services (AWS), dissenyat sota els principis de desacoblament, automatització i seguretat des del disseny (Security by Design).

- Demanda: es tracta de l'etapa inicial on es gestiona el plantejament del problema, la definició d'objectius i la formulació de les hipòtesis de treball.
- Disseny: fase dedicada a transformar els requisits en una arquitectura de cloud real i funcional mitjançant les eines de l'ecosistema d'AWS. Aquí es troben les limitacions del sistema respecte el plantejament inicial.
- Transició: consisteix en la construcció i el desplegament de la solució mitjançant Terraform. També es creen els scripts necessaris de PySpark y Python per crear no només una arquitectura sinó un sistema funcional complet.
- Operació: es treballa l'execució del sistema i del pipeline de dades, monitorant la ingesta i assegurant que el procés funciona correctament.
- Millora continua: on s'avaluen les mètriques de rendiment i qualitat de les dades.

Finalment, per coordinar l'evolució del projecte, gestionar les dependències i assegurar el compliment dels terminis d'entrega, s'ha utilitzat el diagrama de Gantt. La planificació del treball s'ha definit en 5 parts que consisteixen en les següents fases: definició del projecte, recerca, arquitectura, anàlisi de resultats i redacció.

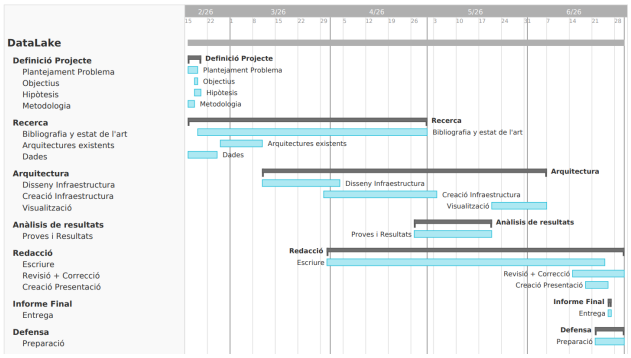

Fig. 1: Diagrama de Gantt

5 Disseny i Implementació

5.1 Justificació del disseny

La decisió arquitectònica més rellevant del projecte ha estat prescindir completament de servidors dedicats o clústers permanents, adoptant un model d'arquitectura 100% *Serverless* i desplegat íntegrament mitjançant Terraform com a Infraestructura com a Codi (IaC). El flux de dades s'estructura en tres capes principals: l'extracció s'executa mitjançant l'orquestració de funcions efímeres d'AWS Lambda, la transformació i neteja es delega a treballs distribuïts d'Apache PySpark sobre AWS Glue, i l'explotació analítica es realitza sobre les dades dels objectes a Amazon S3 amb el motor de consultes interactiu Amazon Athena. Com a punt final d'interacció, el consum visual d'aquestes dades recau sobre un clúster de Grafana allotjat en contenidors gestionats sota Amazon ECS amb AWS Fargate, recolzat en un sistema d'arxius elàstic Amazon EFS per garantir la persistència de l'estat.

Aquesta topologia respon directament a l'aplicació dels principis d'enginyeria *FinOps*, necessaris per mantenir la viabilitat econòmica del projecte. L'objectiu de desacoblar totalment el còmput de l'emmagatzematge permet que la plataforma escali a zero durant els períodes d'inactivitat del cicle d'ingesta (*batch*), facturant pel temps d'execució en Lambda o Glue, i per les dades escanejades per Athena. De forma paral·lela, l'ús d'IaC (Terraform) justifica i garanteix l'auditoria, la immutabilitat i la replicabilitat exacta de la solució, permetent recuperar l'entorn de manera idèntica davant de possibles desastres sense intervenció manual.

Com a alternatives tecnològiques, es va avaluar el disseny clàssic d'un *Data Warehouse* centralitzat (com Amazon Redshift) o l'ús de màquines virtuals aprovisionades (Amazon EC2) per a l'execució d'scripts i el desplegament de Grafana. Aquestes opcions es van descartar, atès que generen un cost fix operatiu les 24 hores del dia, un model oposat a l'estratègia efímera seleccionada. Addicionalment, l'ús de consoles d'administració manuals en comptes de Terraform hauria impedit el seguiment de versions (*version control*) de la topologia física del sistema.

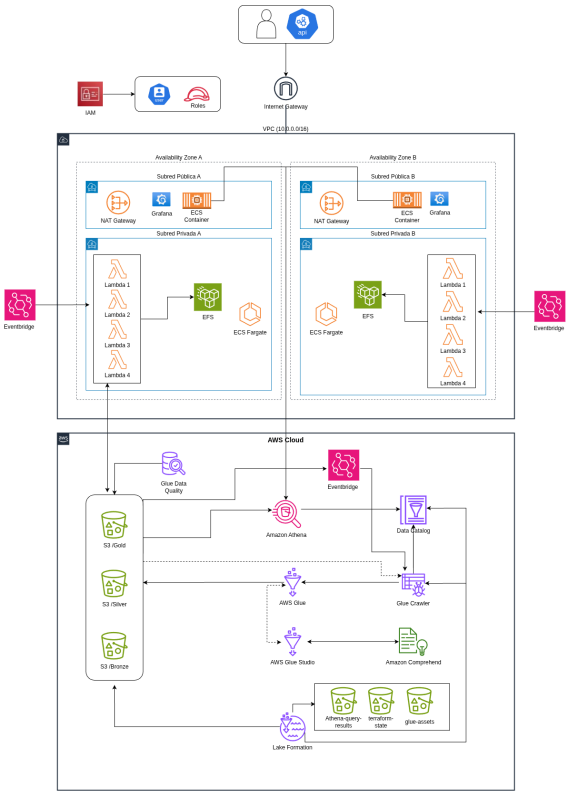

Fig. 2: Diagrama Infraestructura

5.2 Justificació dels components

El disseny de l'arquitectura d'aquest Data Lake s'ha fonamentat en la recerca d'un equilibri òptim entre rendiment, cost econòmic (FinOps) i minimització de la càrrega operativa. Per garantir la idoneïtat de l'entorn tecnològic, cada component clau ha estat seleccionat després d'avaluar les principals alternatives del mercat i de l'ecosistema d'Amazon Web Services:

- **Motor de Consulta: Amazon Athena vs. Amazon Redshift** S'ha optat per Amazon Athena en lloc d'Amazon Redshift degut al paradigma d'emmagatzematge del projecte. Redshift és un Data Warehouse orientat a càrregues de treball contínues i massives que requereix l'aprovisionament de clústers (amb costos fixos associats) i la ingesta de dades en els seus propis discs. Per contra, Athena és un motor de consultes interactives totalment Serverless basat en Trino/Presto, que permet interrogar directament els fitxers Parquet emmagatzemats a Amazon S3 desacoblant el còmput de l'emmagatzematge. Això permet una reducció dràstica de costos, ja que només es factura per les dades escanejades durant el temps de consulta.
- **Processament Distribuït: AWS Glue vs. Amazon EMR** Per a l'execució de les canonades (pipelines) d'Apache PySpark, s'ha descartat l'ús d'Amazon EMR (Elastic MapReduce). Tot i ser l'estàndard per a ecosistemes Hadoop de gran escala, EMR exigeix una elevada sobrecàrrega operativa (gestió d'instàncies, configuració de xarxes i ajust de paràmetres del clúster). AWS Glue, en canvi, ofereix un entorn d'integració de dades purament Serverless que elimina l'administració de la infraestructura i factura només pels segons exactes d'execució. A més, proporciona una integració nativa amb AWS Glue Data Catalog, peça fonamental per al descobriment d'esquemes en aquesta arquitectura.
- **Infraestructura com a Codi (IaC): Terraform vs. AWS CloudFormation** L'automatització del desplegament s'ha materialitzat amb HashiCorp Terraform per sobre de l'eina nativa AWS CloudFormation. Tot i que CloudFormation ofereix una integració profunda amb AWS, genera dependència del proveïdor (vendor lock-in). L'objectiu de l'elecció de Terraform és garantir que la infraestructura pugui ser fàcilment adaptada a altres entorns cloud en el futur.
- **Governança i Seguretat: AWS Lake Formation vs. Polítiques IAM / S3 tradicionals** La gestió del control d'accés s'ha centralitzat mitjançant AWS Lake Formation. Si s'haguessin utilitzat exclusivament les polítiques tradicionals d'AWS IAM i les Bucket Policies d'S3, el control de seguretat hauria quedat limitat a un nivell macro (accés a directoris o fitxers sencers). Lake Formation introdueix un model de governança basat en taules i columnes, permetent definir permisos granulars (Fine-Grained Access Control). Això significa que, des d'un únic punt centralitzat, es pot denegar l'accés a columnes específiques (com informació sensible o mètriques clau) a determinats rols d'usuari assolint un autèntic model de Zero-Trust.

5.3 Seguretat i Governança

5.3.1 Gestió de permisos

La governança i seguretat de les dades s'estructura mitjançant un model de Confiança Zero (*Zero-Trust*). A nivell base, s'aplica xifratge SSE-S3 per a les dades en repòs i s'utilitzen rols d'AWS IAM basats en el Principi de Mínim Privilegi per a l'execució de cada servei. Sobre aquesta base, s'ha implementat AWS Lake Formation com a tallafocs centralitzat de metadades, actuant com a capa principal de governança.

L'ús exclusiu de polítiques descentralitzades d'IAM i *Bucket Policies* d'Amazon S3. S'ha descartat perquè el control de seguretat hauria quedat limitat a un nivell macroscòpic (accés total o nul a directoris o fitxers sencers). Això hauria fet inviable aplicar restriccions a nivell de columna perquè per definir un control d'accessos com el de Lake Formation, és necessari crear de múltiples còpies físiques de les dades per a diferents perfils d'usuari, trencant el principi de font única de veritat (*Single Source of Truth*) del sistema.

5.3.2 Configuració de xarxa

A nivell de xarxa, el processament i l'emmagatzematge es realitzen de forma aïllada dins d'una *Virtual Private Cloud* (VPC) estructurada en múltiples Zones de Disponibilitat (*Multi-AZ*). Els recursos de còmput (les funcions AWS Lambda d'ingesta i els treballs de PySpark a AWS Glue) es despleguen exclusivament en subxarxes privades sense adreces IP públiques. Per permetre que els components de AWS Lambda consumeixin les APIs de CoinGecko i Bluesky, s'encamina el trànsit a través d'un *NAT Gateway* situat a les subxarxes públiques. Addicionalment, el trànsit cap a Amazon S3 transcorre completament per la xarxa interna regional d'AWS, mentre que l'accés a nivell de recurs es restringeix mitjançant *Security Groups* d'estat (per exemple, permetent que el clúster de Grafana a ECS Fargate només es comuniqui amb el sistema EFS i l'API d'Athena).

5.4 Pipeline de Dades

El cicle de vida de la informació s'estructura mitjançant el patró de disseny de l'arquitectura de medalló, dissenyada per refinar les dades de forma progressiva i idempotent. L'orquestració de tot el flux és completament autònoma i reactiva (Event-Driven), coordinada a través d'Amazon EventBridge.

Capa Bronze (Raw): Representa el punt d'entrada immutable del Data Lake. S'ha desenvolupat un mòdul d'extracció automatitzat via AWS Lambda que captura periòdicament la cotització de criptomonedes (CoinGecko) i les interaccions socials (Bluesky). Les dades s'emmagatzemen en el seu format natiu inalterat (JSON/CSV) en una estructura particionada temporalment (year/month/day) per facilitar-ne la traçabilitat.

Capa Silver (Clean): Un cop catalogada la dada bruta, EventBridge desencadena les lambdes silver. En aquest cas, no s'utilitza un ETL job, perquè no està justificat. El treball computacional que es realitza sobre les dades és merament estructural i computacionalment lleuger. Utilitzant codi Python, es realitzen les transformacions estructurals:

s'eliminen columnes redundants, es fa el tipatge (casting) al format correcte i, per afavorir el rendiment futur, les dades es reescriuen en format Apache Parquet. Aquest format columnar millora el temps de les lectures posteriors, reduint la latència i els costos de consulta a Athena gràcies a l'aplicació del paradigma Schema-on-Read.

Capa Gold (Business): La darrera fase del pipeline està dissenyada exclusivament per a l'extracció de valor de negoci mitjançant un ETL Job (Glue job). Mitjançant un Job de PySpark, s'integra el servei d'intel·ligència artificial Amazon Comprehend per aplicar models de Processament del Llenguatge Natural (NLP). Es correlacionen temporalment les mètriques financeres estructurades amb l'anàlisi de sentiment extret dels textos de Bluesky. El resultat s'emmagatzema en taules altament agregades i optimitzades, preparades per alimentar de forma immediata els panells interactius de Grafana i l'exportació topològica cap a bases de dades de grafs.

5.5 Funcionalitat del sistema

Aquesta secció introdueix el flux complet de la informació, desconnectant la infraestructura subjacent per centrar-se en com la dada viatja des de l'extracció fins a la presentació analítica final. Figura 9

5.5.1 Ingesta

Les funcions Serverless d'AWS Lambda, desenvolupades en Python, orquestren l'extracció. S'integra la llibreria boto3 per recuperar de forma segura les credencials a través d'AWS Secrets Manager, evitant exposar claus en el codi font. S'ingereixen dades de mercat financer (CoinGecko) i de xarxes socials (Bluesky mitjançant AT Protocol). A continuació, els DataFrames generats amb pandas es transformen a format columnar Parquet utilitzant el motor pyarrow per optimitzar l'emmagatzematge, i es desen al Data Lake (Amazon S3) particionats cronològicament (any, mes, dia).

Durant la fase d'ingesta, un dels reptes inherents a l'extracció iterativa de dades mitjançant APIs de xarxes socials és l'aparició de registres duplicats. En el cas dels posts, el mateix post pot aparèixer més d'un cop si no s'ha pujat nou contingut a la xarxa social.

Seguint els principis de disseny de l'arquitectura de medalló, la capa Bronze s'ha concebut com un repositori immutable d'informació crua (*raw data*). Per aquest motiu, durant el procés d'ingestió orquestrat per AWS Lambda, no s'aplica cap filtre de descartament; totes les dades extretes, incloses les publicacions repetides, s'emmagatzemen inalterades. Aquesta estratègia garanteix la conservació d'un registre històric exacte, traçable i auditable de les respostes de l'API en el moment exacte de la ingesta.

No obstant això, per garantir la integritat analítica a les fases posteriors i complir amb el principi d'idempotència de les canonades de dades, s'ha implementat un mecanisme de deduplicació basat en identificadors (*hashing*). En el moment de la captura, l'script genera dinàmicament una clau primària subrogada de longitud fixa (utilitzant l'algoritme criptogràfic MD5) a partir de l'identificador únic del contingut de Bluesky (`cid`) i les metadades de la publicació.

Posteriorment, durant la transició de les dades de la capa Bronze a la capa Silver, s'utilitza aquest *hash* identificatiu per descartar els registres redundants. D'aquesta manera, s'assegura que només les publicacions noves siguin netejades i promogudes al format Apache Parquet de la capa Silver. Aquesta tècnica de deduplicació és fonamental per optimitzar l'eficiència econòmica (*FinOps*), ja que evita el malbaratament de recursos computacionals enviant textos duplicats al servei d'intel·ligència artificial (Amazon Comprehend) a la capa Gold, evitant alterar estadísticament el càlcul del sentiment social.

5.5.2 Processament

La fase de processament representa el nucli de transformació del Data Lake, on la informació continguda a la capa Silver es refina, s'enriqueix i s'estructura per promoure-la a la capa Gold, la qual conté les taules de negoci definitives. Aquest procés s'orquestra de forma totalment gestionada i distribuïda mitjançant AWS Glue, executant canonades de dades (ETL) programades en Apache PySpark. Per al descobriment d'esquemes, s'utilitzen AWS Glue Data Catalog, permetent la lectura directa de les taules virtuals emmagatzemades a Amazon S3.

El processament es divideix en tres grans fluxos de treball independents segons la naturalesa de la dada:

1. Transformació de Dades Financeres: L'ETL de dades financeres s'encarrega d'alleugerir i normalitzar les mètriques obtingudes de l'API de CoinGecko. Per optimitzar el volum d'emmagatzematge i el cost de consulta posterior, es seleccionen només les columnes essencials per a l'anàlisi de negoci i s'eliminen columnes que inclouen imatges o dates històriques. A més, s'introdueixen regles de resolució d'errors estructurals, com la imputació de valors nuls: si una moneda no reporta el seu preu actual (current_price), el sistema li assigna automàticament el valor màxim assolit en les últimes 24 hores (high_24h). Finalment, el DataFrame resultant es redueix a una única partició física mitjançant l'operació coalesce(1) i s'emmagatzema en format Apache Parquet, particionat dinàmicament per any, mes i dia segons el moment del procés.

2. Processament de Llenguatge Natural i Anàlisi de Sentiment: El tractament dels textos lliures provinents de la xarxa social Bluesky és el procés computacional més complex de l'arquitectura, ja que integra aprenentatge automàtic en el flux de dades. En primer lloc, el sistema aplica funcions definides (UDFs) per realitzar una neteja del text lliure (text_cleaned): s'eliminen URL, mencions d'usuaris, caràcters especials i una extensa llista personalitzada de stopwords (paraules buides com preposicions o pronoms) per reduir el soroll de fons, però es mantenen els accents i les puntuacions ja que aporta context necessari per al model. A la Figura 14 podem veure que s'aconsegueix reduir a la meitat el nombre de paraules i de caràcters. Per evitar colls d'ampolla i respectar els límits de concurrència (Rate Limits) de l'API d'AWS, el DataFrame s'estructura en blocs i s'utilitza la funció mapPartitions. Això permet instanciar el client de boto3 només una vegada per cada node de treball de Spark (i no per cada fila), enviant

de forma eficient paquets de 25 documents a la funció batch_detect_sentiment. El resultat retorna probabilitats de sentiment (positiu, negatiu, neutral i mixt) que es fusionen (Join) amb el text original.

A continuació, el sistema ha d'identificar de quina criptomoneda específica parla cada publicació. Per fer-ho, l'ETL consulta la partició més recent de la taula financera (que conté el *Top 20* per capitalització de mercat) ubicada a la capa *Silver*. Per evitar que els múltiples nodes de treball (*workers*) realitzin peticions redundants a Amazon S3 per consultar quines són les criptomonedes rellevants, aquest procés es realitza a l'inici de l'execució, abans de crear els workers, aplicant una tècnica d'optimització de memòria d'Apache Spark anomenada *Broadcast Variables*. Aquesta tècnica empaqueta el diccionari de criptomonedes i els seus àlies (per exemple, associant "bitcoin" i "btc") i l'envia a la memòria cau (RAM) de cada node de forma compartida. Una segona UDF escaneja el text netejat contra aquest diccionari i etiqueta la publicació amb els actius esmentats. Els registres que no mencionen cap criptomoneda rellevant són filtrats i descartats de la canonada.

Com en el cas anterior, el resultat es compacta amb coalesce(1) —Amazon Athena processa els arxius entre 128 MB i 1 GB de forma molt més ràpida i econòmica que múltiples arxius petits— i es desa a la capa Gold.

3. Modelatge Topològic de Relacions Socials (Grafs): Per donar suport a l'anàlisi de comunitats i usuaris influents, el tercer ETL s'encarrega d'estructurar les dades com un model de nodes i arestes, preparat per ser exportat a bases de dades de grafs. Utilitzant la potència de Spark, el codi converteix la columna d'objectes JSON complexos que conté els seguidors (follows) utilitzant un esquema de lectura anidat. A continuació, aplica l'operador explode per desenrotllar la llista, creant una fila independent per a cada interacció social. A partir d'aquí, es divideix el DataFrame en dos grans blocs estructurals requerits en teoria de grafs:

- Nodes: Extreu els identificadors únics d'usuari (author_did) assignant-los l'etiqueta categòrica estandarditzada :LABEL="User".
- Arestes (Relationships): Evalua la direccionalitat de la relació (si l'usuari segueix o és seguit) mitjançant condicions lògiques booleanes. Finalment, es crea un arxiu amb les columnes :START_ID i :END_ID, afegint-hi la propietat d'acció :TYPE="FOLLOWS". Aquests actius s'exporten a la capa final de forma segregada.

5.5.3 Visualització

L'objectiu de la capa de visualització és validar empíricament la integritat de les dades processades, quedant el seu disseny fora de l'abast central de recerca d'aquest projecte. (Annexe Figura 13)
La interfície per a l'usuari final s'ha implementat desplegant la imatge oficial de Grafana (grafana/grafana:latest) sobre Amazon Elastic Container Service (ECS) utilitzant el motor de còmput AWS Fargate. Per connectar-se nativament al Data Lake, la configuració injecta automàticament el connector grafana-athena-datasource mitjançant variables d'entorn. A més, per dotar el contenidor d'estat, es munta un volum compartit d'Amazon Elastic File System (EFS) directament a la ruta interna /var/lib/grafana del contenidor, aplicant xifratge en trànsit i control d'autorització basat en IAM.

L'explotació del Data Lake es realitza dissenyant consultes SQL, preparades per ser executades per motors distribuïts (com Amazon Athena) i visualitzades a Grafana. Es calculen mètriques com l'evolució temporal de les criptomonedes i la mitjana del sentiment correlacionat. S'apliquen tècniques de processament de text en SQL (amb UNNEST i SPLIT) per extreure les paraules més freqüents, unions (CROSS JOIN i LEFT JOIN) per calcular variacions de mercat respecte a finestres temporals passades, i subconsultes agrupades per estructurar els usuaris més populars en un model topològic de nodes i arestes que representen les relacions socials.

5.6 Automatització

El procés de captura, transformació i càrrega de dades està completament automatitzat, garantint un sistema autònom que no requereix d'intervenció humana per al seu funcionament normal. L'actualització de la informació es realitza de forma contínua, estant subjecta exclusivament al temps d'execució requerit pel pipeline per processar les dades. Aquesta orquestració s'ha implementat mitjançant el servei Amazon EventBridge, a través de la configuració de disparadors (triggers) i fluxos de treball de Glue (workflows) que connecten els diferents serveis de forma seqüencial i per esdeveniments.

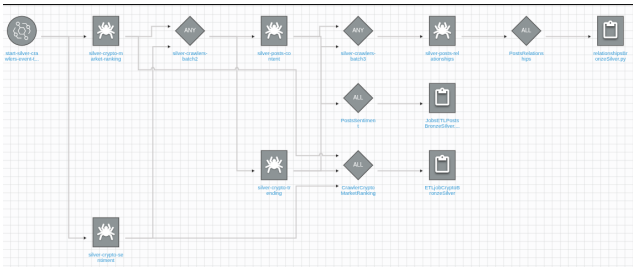

Fig. 3: Diagrama Workflow Silver

El comportament del pipeline segueix una arquitectura seqüencial (Figura 3), que s'inicia de manera periòdica en la capa Bronze mitjançant un esdeveniment programat per temps en Amazon EventBridge, el qual executa la primera funció AWS Lambda. Un cop les dades inicials són dipositades a Amazon S3, una notificació de bucket detecta la ingesta i dispara de forma automàtica la funció Lambda de la capa Silver, la qual s'encarrega de processar, transformar i emmagatzemar les dades resultants en el repositori S3 corresponent.

L'enregistrament de les dades silver al bucket S3 actua com a disparador per executar els treballs de transformació (ETL Jobs) destinats a la capa Gold. Una vegada finalitzada l'execució d'aquests ETL Jobs, l'eina de visualització Grafana realitza consultes periòdiques mitjançant Amazon Athena sobre aquestes dades indexades, actualitzant els panells de control de manera automàtica a mesura que la informació es consolida en el sistema.

Els crawlers, existents a cada capa del procés de les dades, s'executen un cop a l'inici del dia perquè es generen

nous directoris cada dia a causa del particionat del bucket S3.

6 Resultats i Validació

Per garantir la viabilitat i robustesa del Data Lake proposat, s'ha sotmès la infraestructura a una sèrie de proves empíriques dissenyades per avaluar la funcionalitat, el rendiment operatiu, la tolerància a fallades i el compliment de les polítiques de governança de dades.

6.1 Validació de la hipòtesi de negoci

El present projecte neix amb la hipòtesi subjacent que el sentiment expressat a les xarxes socials pot actuar com un indicador predictiu o paral·lel del valor de mercat de les criptomonedes. Un cop desplegat el Data Lake i estabilitzada la ingesta de dades, s'ha procedit a analitzar aquesta relació empírica creuant l'índex de sentiment agregat (modelat amb Amazon Comprehend) amb les fluctuacions de preu d'actius clau com el Bitcoin (BTC) i l'Ethereum (ETH).

El model de Amazon Comprehend retorna 4 mètriques: Positiu, Negatiu, Neutral i Mixte. La mètrica Mixte s'ha descartat perquè no aportava un valor real per a l'anàlisi que es fa, en conseqüència, s'han normalitzat els valors de les tres variables restants perquè la seva suma doni 100%. Tanmateix, treballar amb tres variables independents per a cada criptomoneda dificulta l'anàlisi visual. Per facilitar la comprensió de la informació a l'usuari final, s'ha implementat una transformació per les tres mètriques utilitzades, de forma que s'aplica una suma ponderada, la fórmula implementada és la següent:

$$Sentiment = \frac{P_{pos} * 1 + P_{neut} * 0.5 + Pneg * 0}{P_{pos} + P_{neut} + P_{neg}}$$

Aquesta transformació matemàtica comprimeix la informació en una variable unidimensional amb valors entre 0 i 1 que representen:

- Un valor de 0 indica un sentiment negatiu absolut
- Un valor de 0.5 indica un sentiment neutre absolut
- Un valor de 1 indica un sentiment positiu absolut

Per avaluar la hipòtesi de negoci s'ha utilitzat el coeficient de correlació de Pearson. Aquest coeficient és el més utilitzat per mesurar la força i direcció de les relacions lineals entre dues variables quantitatives contínues. En aquest cas s'avaluen els preus de les criptomonedes i el sentiment dels posts. El coeficient (r) produeix un valor entre -1 (correlació negativa perfecta) i 1 (correlació positiva perfecta). El valor de 0 indica que no existeix relació lineal. L'expressió matemàtica utilitzada és la següent:

$$r = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n}(X_i - \bar{X})}\sqrt{\sum_{i=1}^{n}(Y_i - \bar{Y})}}$$

On X representa el valor de l'índex de sentiment per cert moment i Y el preu de la criptomoneda en el mateix espai de temps. $\bar{X}$ i $\bar{Y}$ representen les mitjanes de cada variable durant la finestra d'anàlisi de 30 dies. A continuació es detallen els resultats observats:

- Existència i intensitat de la correlació: L'anàlisi de les dades recollides conclou que no existeix cap correlació lineal entre el sentiment de les publicacions a la xarxa Bluesky i el preu dels actius en l'escala temporal analitzada (1 mes). En calcular el coeficient de correlació de Pearson (r) mitjançant el motor d'Amazon Athena sobre la capa Gold, els resultats obtinguts demostren una independència estadística total. La correlació per al Bitcoin (BTC) se situa en $r \approx -1.03 \times 10^{-13}$, mentre que per a l'Ethereum (ETH) és de $r \approx 1.17 \times 10^{-13}$. Aquests valors són estadísticament equivalents a zero i per tant, descarten que les fluctuacions diàries del sentiment dels posts de la xarxa social tinguin un efecte directe en l'evolució del preu de les criptomonedes. El resultat es pot observar gràficament a la Figura 4, a on es mostra la relació entre el valor real de la criptomoneda Bitcoin i el sentiment associat. Es pot observar que el sentiment es manté amb un valor constant proper a la neutralitat, independentment de la variació de la moneda.

- Comportament generalitzat: Aquesta absència total de correlació no és exclusiva dels actius de gran capitalització, sinó que és un patró constant en tot l'ecosistema analitzat. Tant les Altcoins consolidades com Cardano (ADA, $r \approx 2.35 \times 10^{-14}$) o Solana (SOL, $r \approx 1.94 \times 10^{-13}$), com les monedes de caire més especulatiu tipus Dogecoin (DOGE, $r \approx 6.06 \times 10^{-14}$), presenten resultats igualment neutres. P

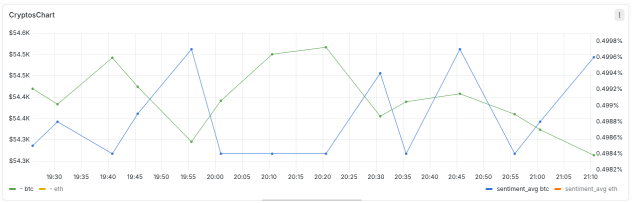

Fig. 4: Correlació entre el preu de la criptomoneda bitcoin i el sentiment dels posts

6.2 Capacitat de processament

Durant la fase d'avaluació i operació del Data Lake, el sistema ha monitoritzat l'entrada i transformació contínua de dades. Per cada ingesta, a nivell de la capa Bronze, l'arquitectura ha capturat una mitjana de 200 publicacions i 134 relacions per a cada usuari provinents de la xarxa social Bluesky i 50 registres mètrics de l'API de CoinGecko corresponents al ranking de les monedes, trending i sentiment, acumulant un volum total de 0.5 GB de dades crues (JSON/CSV) per cada ingesta. Després de travessar la canonada de processament (AWS Glue) i aplicar la neteja, el filtre de stopwords i la inferència de sentiment, la capa final d'explotació (Gold) reduïm el volum total de dades fins a obtenir aproximadament 0.35GB de dades.

Cal destacar que el volum de dades podría ser molt major, però existeixen certes limitacions que impedeixen augmentar-lo. Tanmateix, el volum de dades crues ingerides demostra que la partició dinàmica a Amazon S3 (per any, mes i dia) i la transformació cap al format columnar compressiu Apache Parquet a les capes Silver i Gold son passes necessaries per optimitzar el pipeline i la lògica de

negoci. En aquest cas, es valora en igual mesura la ingesta puntual i la capacitat de recuperar dades històriques àgilment.

6.3 Qualitat de les dades

Per garantir la veracitat, la integritat i la robustesa dels indicadors analítics, el Data Lake integra controls de qualitat automatitzats mitjançant AWS Glue Data Quality. Aquest servei utilitza el llenguatge declaratiu DQDL (Data Quality Definition Language) per avaluar l'estat de la informació en cada etapa de l'arquitectura. Totes les regles s'han desplegat íntegrament com a Infraestructura com a Codi (IaC) amb Terraform, acoblant-se directament a les taules lògiques de l'AWS Glue Data Catalog per garantir l'abstracció de l'emmagatzematge físic.

L'estratègia de validació s'ha dissenyat de manera progressiva, endurint les restriccions a mesura que les dades avancen pel patró de medalló:

Capa Bronze (Control de Disponibilitat): En aquesta fase inicial, l'objectiu és assegurar la recepció efectiva de les dades procedents de les APIs de tercers. S'aplica una regla bàsica però crítica de volumetria ($RowCount > 0$) a totes les taules d'ingesta (market_ranking, sentiment, trending i post_content). Aquesta condició actua com un mecanisme de Fail-Fast: si una API experimenta una caiguda o retorna un fitxer buit, el sistema atura la canonada immediatament, impedint la propagació de dades corruptes o buides cap a les fases de transformació.

Capa Silver (Integritat Estructural i Tipatge): Un cop les dades es netegen, les regles asseguren la coherència de l'esquema i la validesa lògica dels registres.

Per al domini financer, s'exigeix el compliment estricte del tipus de dada (ColumnDataType = "Double") per als preus, i s'apliquen filtres de valors no negatius (ColumnValues ≥ 0.0) per a mètriques com el current_price i el market_cap, evitant anomalies numèriques d'origen.

Per al domini social, es garanteix la presència obligatòria (IsComplete) dels identificadors clau (id, cid, author_handle, author_did), i s'audita que els comptadors d'interacció (like_count, reply_count, repost_count) no assoleixin mai valors inferiors a zero.

Capa Gold (Validació de Regles de Negoci):

```
  description = "Regles de qualitat per a score_by_crypto_gold"
  target_table {
    database_name = aws_glue_catalog_database.schema_database.name
    table_name    = "score_by_crypto_gold"
  }
  ruleset = <<-EOF
    Rules = [
      RowCount > 0,
      IsComplete "sentiment_up_percentage",
      ColumnValues "sentiment_up_percentage" >= 0.0,
      ColumnValues "sentiment_up_percentage" <= 100.0,
      IsComplete "sentiment_down_percentage",
      ColumnValues "sentiment_down_percentage" >= 0.0,
      ColumnValues "sentiment_down_percentage" <= 100.0,
      IsComplete "watchlist_count",
      ColumnValues "watchlist_count" >= 0
    ]
  EOF
}
```

Fig. 5: Exemple de validació de la qualitat de les dades

Com s'observa a la Figura 5, a la capa final, destinada a l'explotació analítica i visual, les regles DQDL certifiquen els indicadors de negoci. S'audita el resultat del model de Processament del Llenguatge Natural (NLP), assegurant que els percentatges de sentiment positiu i negatiu es mantinguin dins de l'escala percentual vàlida (valors ≥ 0 i ≤ 1). Finalment, per a l'exportació topològica cap a bases de dades de grafs, es valida la integritat dels nodes i les relacions, exigint que cap clau primària o identificador d'autor sigui nul, evitant així la creació d'arestes òrfenes en l'anàlisi de xarxes socials.

6.4 Validació de Seguretat i Qualitat

La premissa d'un Data Lake governat exigeix garantir que les dades de caràcter sensible (com els identificadors d'usuari i els textos lliures de Bluesky) estiguin protegides d'accessos indeguts. Per validar l'eficàcia del model Zero-Trust, s'han realitzat proves d'accés lògic.

Prova de filtratge de metadades: S'ha intentat realitzar una consulta SQL des d'Amazon Athena cap a la Capa Bronze intentant extreure columnes restringides, fent servir un rol d'usuari (IAM) analista sense privilegis d'administrador. Resultat: En lloc de permetre la lectura, AWS Lake Formation ha interceptat la petició i ha retornat de forma immediata un error de tipus COLUMN_NOT_FOUND o INSUFFICIENT_ACCESS_PERMISSIONS (Figura 8). Aquesta prova demostra el correcte funcionament de la seguretat granular a nivell de columna, garantint l'anonimització i complint amb els estàndards normatius de privacitat i el principi de mínim privilegi.

A més, s'ha validat la integritat del flux de dades mitjançant la injecció de regles de Data Quality (DQDL). El sistema ha demostrat ser capaç de detectar anomalies estructurals (com l'aparició de valors nuls en les mètriques financeres de CoinGecko) abans de la seva promoció a la Capa Gold, assegurant que les visualitzacions finals estiguin basades únicament en conjunts de dades fiables. Figura 11.

6.5 Mètriques de Rendiment i Costos

La validació del rendiment s'ha centrat a comprovar l'eficiència de la canonada de dades (Data Pipeline) i l'impacte directe de les decisions arquitectòniques en els costos (FinOps). Atesa la naturalesa Serverless i totalment gestionada de la infraestructura desplegada (utilitzant serveis com AWS Lambda, AWS Glue i Amazon Athena), s'ha determinat que l'execució de proves de càrrega o d'estrès tradicionals no aporta un valor crític a l'avaluació del projecte. En aquest tipus d'entorns, la responsabilitat de l'escalabilitat i el dimensionament s'abstrau i es delega completament al proveïdor del núvol (AWS), l'arquitectura del qual ja posseeix uns límits operatius i de capacitat àmpliament demostrats a la indústria. En conseqüència, la quantitat d'informació injectada no representa un factor limitant ni un coll d'ampolla per a l'arquitectura, ja que els serveis son capaços d'escalar a mida que incrementa el volum de dades. L'objectiu fonamental d'aquestes proves de rendiment es centra exclusivament en analitzar el temps efectiu que triga el sistema a processar i transformar els lots d'informació, i avaluar com la decisió arquitectònica del sistema es tradueix directament en eficiència a nivell de costos.

6.5.1 Cost per execució

A nivell econòmic, cada execució unitària del pipeline (incloent-hi l'ús de Lambda, les hores DPU de Glue i les consultes d'Athena) té un cost d'aproximadament 0.02 €. Això es complementa amb un cost estructural o "en repòs"d'aproximadament 33 € mensuals, atribuït exclusivament al NAT Gateway i a l'emmagatzematge objectual acumulat (Amazon S3). Mantenir actiu node ECS tot el dia té un cost mensual de 8.64€. Per tant, el manteniment i ús del sistema té un cost aproximat mensual de 41.67€.

Query	Parquet	CSV	Dades Escanejades (Mb)	Temps (s) Execució
1	X		426	6.76
1		X	920	8.27
2	X		129.8	1.29
2		X	763.8	7.63
3	X		1320	13.6
3		X	9740	15.4
4	X		156	1
4		X	346.6	2

Taula 1: Comparació d'execució de consultes d'Athena sobre fitxers CSV i Parquet

La decisió d'emmagatzemar les dades transformades en format columnar (Apache Parquet) i particionar-les temporalment a Amazon S3 ha estat sotmesa a prova mitjançant consultes a Amazon Athena. Els resultats demostren que el model Schema-on-Read combinat amb Parquet redueix significativament el volum de megabytes escanejats per consulta respecte a un format pla tradicional (CSV o JSON).

Parquet ocupa de mitjana un 70% menys que un arxiu CSV, per tant, donat que S3 factura per GB emmagatzemat al núvol, s'aconsegueix un estalvi directe de la mateixa proporció.

Observem que, de mitjana, el format parquet supera al CSV, perquè requereix escanejar menys dades i ofereix uns temps d'execució més ràpids. A partir dels resultats observats, parquet escaneja, de mitjana un 70% menys de dades amb un temps d'execució un 41% més ràpid. A nivell de costos, com que Athena factura exclusivament per la quantitat de dades escanejades (5,00$ per terabyte) i els serveis Serverless poden escalar a zero quan no hi ha ingesta, el cost d'operació i manteniment del Data Lake en repòs és pràcticament nul, complint amb l'objectiu econòmic del projecte i validant l'avantatge davant d'un entorn on-premise.

6.5.2 Rendiment per capa i servei

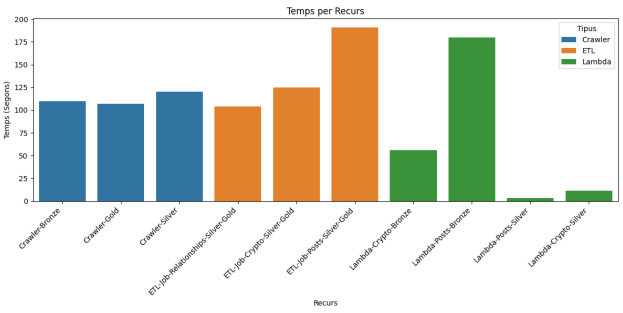

Fig. 6: Temps execució per servei

A la Figura 6 s'observa que durant la fase d'operació, el mòdul d'ingesta (Capa Bronze) ha demostrat una alta eficiència, amb funcions AWS Lambda que s'executen en una mitjana de pocs segons per petició per extreure les respostes JSON de les APIs de CoinGecko i Bluesky. Pel que fa al processament massiu (Capa Silver i Gold), l'ús de treballs distribuïts amb PySpark a AWS Glue ha permès reduir dràsticament el temps de transformació de les dades.

Per tant, si tenim en compte tots els serveis que participen en el correcte funcionament del pipeline de dades, el temps que triga el sistema és d'aproximadament 16 minuts i 45 segons. A la figura 7 podem observar la proporció del temps que triga cada fase del pipeline. D'aquest temps del cicle de les dades, el 34,8% és a dir, 5 minuts i 47 segons corresponen a l'obtenció de les dades de les APIs (Capa Bronze) i el 33,5% és a dir, 5 minuts i 37 segons corresponen als crawlers. Tanmateix, els crawlers només s'executen un cop al dia en hores de mínima demanda teòrica. En conseqüència, podem dir que el temps dedicat exclusivament al processament de la informació és de 5 minuts i 22 segons. Per tant en la gran majoria de casos, el temps que triga la informació en arribar a l'usuari final, és de 10 minuts i 59 segons.

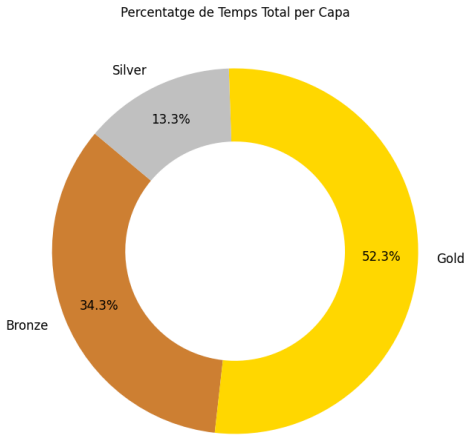

Fig. 7: Percentatge temps total execució per capa

6.6 Consideracions i limitacions

- Restriccions de les APIs públiques (Rate Limiting): Tant CoinGecko com Bluesky imposen polítiques que restringeixen les peticions per minut als usuaris de la capa gratuïta. Aquesta restricció d'origen fa inviable adoptar un patró de captura de dades en streaming segon a segon, ja que provocaria el bloqueig immediat de les adreces IP d'ingesta.

- Latència del model Batch davant de Streaming: Com a conseqüència directa del rate limiting i sota una estricta filosofia FinOps, s'ha optat per un model d'ingesta per lots (batch) de caràcter reactiu. Tot i que aquesta decisió optimitza el cost introdueix una latència temporal inherent que impedeix l'anàlisi de sentiment en temps real.

- Dependència de serveis gestionats i crèdits acadèmics: La viabilitat i el disseny d'aquesta arquitectura s'han vist condicionats per una restricció pressupostària estricta de 200$ en crèdits acadèmics d'AWS. Això limi-

ta la capacitat d'executar proves de volumetria massiva a gran escala i a llarg termini, forçant l'ús estricte de la capa gratuïta (Free Tier) i de serveis Serverless capaços d'escalar a zero durant els períodes d'inactivitat del pipeline.

- Biaix en els models d'NLP preentrenats: L'ús d'Amazon Comprehend com a servei d'intel·ligència artificial gestionat ofereix una gran agilitat d'integració i un baix cost operatiu. No obstant això, com que és un model preentrenat genèric, el processament de textos extrets de xarxes socials pot presentar petits biaixos en la classificació de sentiments a causa de l'ús d'expressions col·loquials o ironies altament específic de l'ecosistema de les criptomonedes.
- Neutralitat del text: L'algorisme d'IA demostra que la immensa majoria de publicacions sobre criptomonedes són informatives (bots que publiquen preus o notícies). Aquestes publicacions disimulen la importància dels posts que sí aporten sentiment, els quals són una minoria.

7 Conclusions i Línies de Futur

La finalització d'aquest Treball de Final de Grau permet extreure un conjunt de conclusions satisfactòries sobre el disseny, implementació i avaluació d'infraestructures de dades al núvol. S'ha assolit amb èxit l'objectiu principal del projecte: construir una arquitectura ETL robusta, escalable i governada, aplicant els estàndards i les bones pràctiques de l'enginyeria de dades actual (Data Lake, Infraestructura com a Codi i filosofia Serverless).

7.1 Assoliment d'objectius

L'arquitectura dissenyada ha complert amb èxit tots els requisits tècnics i funcionals establerts inicialment:

Ingesta desacoblada de fonts heterogènies: S'ha dissenyat un mòdul d'ingesta en format batch capaç d'interactuar amb dues APIs de naturalesa completament diferent: l'ecosistema financer (CoinGecko) i la xarxa social descentralitzada (Bluesky). L'ús d'AWS Lambda ha permès aïllar aquesta complexitat de xarxa i abocar les dades en el seu estat pur a la capa Bronze.

Processament distribuït i NLP: La transició cap a una arquitectura de medalló ha demostrat ser el patró òptim per refinar la informació. Mitjançant AWS Glue i PySpark, s'han netejat les dades i s'han transformat a format columnar Apache Parquet. L'èxit diferencial ha estat la integració nativa d'algorismes de Processament del Llenguatge Natural amb Amazon Comprehend, enriquint els textos per convertir-los en mètriques de sentiment creuables amb indicadors financers.

Automatització i resiliència: El sistema s'ha desplegat com una canonada de dades (Data Pipeline) totalment autònoma. L'orquestració mitjançant Amazon EventBridge garanteix l'execució diària sense intervenció humana, mentre que l'aplicació del principi d'idempotència assegura la resiliència davant de fallades tècniques, permetent el reprocés de lots sense corrompre l'històric.

Explotació visual i analítica: S'ha aconseguit un sistema de visualització d'alta disponibilitat (HA). El desplegament de Grafana sobre contenidors efímers (Amazon ECS Fargate) amb persistència d'estat (Amazon EFS), connectat a Amazon Athena, permet a l'usuari final extreure conclusions sobre la correlació social i financera, exportant també estructures cap a bases de dades de grafs. Figura 10

Governança i Seguretat: S'ha establert un marc de seguretat Zero-Trust aplicant el Principi de Mínim Privilegi amb AWS IAM i centralitzant el control d'accés lògic amb AWS Lake Formation. La implementació de proves de qualitat (DQDL) i catalogació automàtica (Glue Crawlers) converteixen l'entorn en un actiu auditable.

Infraestructura com a Codi (IaC): L'ús de Terraform ha permès tractar la infraestructura del núvol amb el mateix rigor metodològic que el programari. Centralitzar l'arquitectura mitjançant IaC garanteix la traçabilitat absoluta dels canvis, permet versionar l'estat del sistema amb eines de control de codi font (com Git) i assegura una replicabilitat exacta de l'entorn, minimitzant els riscos operatius i facilitant la recuperació davant de desastres.

7.2 Treball futur

Com a evolució natural d'aquest Treball de Final de Grau, es proposen les següents línies de desenvolupament per expandir les capacitats tècniques i analítiques de la plataforma:

Evolució cap a l'arquitectura de Streaming: Implementar una arquitectura complementària per treballar amb eines per la visualització en temps real (com OpenSearch i Kinesis) i combinar-les amb les eines per la visualització de dades històriques.

Extracció avançada de publicacions: Es proposa desenvolupar mecanismes per detectar i separar els posts merament informatius generats per bots dels posts humans que sí donen context de la situació i del sentiment cap a les criptomonedes. D'aquesta forma, es dona protagonisme als posts minoritaris però que sí aporten una informació real. En aquest treball, aquest efecte s'ha vist augmentat pel fet que la xarxa social BlueSky no té una activitat d'usuaris prou alta en comparació amb altres plataformes del sector.

Anàlisi de la volatilitat: El coeficient de Pearson busca si el sentiment fa pujar o baixar el preu de forma directa. Tot i això, a vegades un augment brusc del sentiment (molta eufòria o molt de pànic) no prediu si el preu pujarà o baixarà, sinó que prediu volatilitat o volum de comerç. Per tant, es podria creuar l'índex de sentiment amb el Volum de transaccions.

7.3 Ús de la IA generativa

D'acord amb les polítiques d'integritat acadèmica, es declara l'ús d'eines d'intel·ligència artificial generativa com a suport en l'elaboració de codi i per la revisió gramatical, i de coherència i cohesió del text.

La lògica de negoci, les decisions d'arquitectura, la metodologia de treball, l'anàlisi de resultats i la totalitat dels continguts d'aquest document són autoria de l'estudiant, qui ha supervisat i validat tots els resultats generats per aquestes eines per assegurar-ne la veracitat i l'adequació al projecte.

AGRAÏMENTS

Aquest treball representa el final d'una etapa on he après i gaudit en tots els àmbits. Al llarg d'aquests quatre anys, són moltes les persones que m'han acompanyat i que m'han ajudat a convertir-me en enginyer però que resultarien impossible d'enumerar en una extensió tan curta. Per tant, vull agraïr en primer lloc a la meva familia, pare i mare, per tot el seu amor i per les facilitats que em donen per construir el meu futur. A continuació, vull agrair a la meva parella perquè m'ha ajudat a ser qui soc avui i sempre ha estat una peça clau de suport per mi al llarg d'aquests anys. Finalment, vull agrair al meu tutor, qui m'ha guiat per aconseguir aquest treball el qual culmina la meva etapa d'estudiant d'Enginyeria de Dades.

REFERÈNCIES

[1] Amazon Web Services. (s. f.). *¿Qué es Apache Spark?*. Recuperat de https://aws.amazon.com/es/what-is/apache-spark/

[2] Amazon Web Services. (s. f.). *Amazon Comprehend*. Recuperat de https://aws.amazon.com/es/comprehend/

[3] Amazon Web Services. (s. f.). *AWS Well-Architected*. Recuperat de https://aws.amazon.com/es/architecture/well-architected/

[4] Amazon Web Services. (s. f.). *Servicio ETL - Integración de datos sin servidor - AWS Glue*. Recuperat de https://aws.amazon.com/es/glue/

[5] Amazon Web Services. (s.f.). *What is AWS Lake Formation?*. Recuperat de https://docs.aws.amazon.com/lake-formation/latest/dg/what-is-lake-formation.html

[6] Amazon Web Services. (s.f.). *What is Data Governance?*. Recuperat de https://aws.amazon.com/what-is/data-governance/

[7] Apache Spark. (s. f.). *PySpark Overview — PySpark 4.1.1 documentation*. Recuperat de https://spark.apache.org/docs/latest/api/python/index.html

[8] Atlassian. (s.f.). *¿Qué es ITIL?*. Recuperat de https://www.atlassian.com/es/itsm/itil

[9] Bluesky. (s. f.). *Bluesky Documentation*. Recuperat de https://docs.bsky.app/

[10] Clurman, R. (2026, 2 de abril). *EC2 vs. Fargate: Understanding the Differences and Choosing the Best for ECS*. ProsperOps. Recuperat de https://www.prosperops.com/blog/ec2-vs-fargate/

[11] CoinGecko. (s. f.). *GitHub - coingecko/coingecko-python*. GitHub. Recuperat de https://github.com/coingecko/coingecko-python/tree/main

[12] Deloitte Insights. (2019, 17 de junio). *Digital industrial transformation: Reinventing to win in Industry 4.0*. Recuperat de https://www2.deloitte.com/insights/us/en/focus/industry-4-0/digital-industrial-transformation-industrial-internet-of-things

[13] HashiCorp. (s.f.). *AWS: aws_lambda_function*. Recuperat de https://registry.terraform.io/providers/hashicorp/aws/latest/docs/resources/lambda_function

[14] Hotz, N. (2024, 9 de diciembre). *What is CRISP DM?*. Data Science PM. Recuperat de https://www.datascience-pm.com/crisp-dm-2/

[15] IBM. (s.f.). *Data warehouse vs. Data lake vs. Data Lake*. Recuperat de https://www.ibm.com/think/topics/data-warehouse-vs-data-lake-vs-data-Lake

[16] Microsoft. (s.f.). *Arquitectura de la casa del lago Medallion*. Recuperat de https://learn.microsoft.com/es-es/azure/databricks/Lake/medallion

[17] Neo4j. (s.f.). *Importing CSV files*. Recuperat de https://neo4j.com/docs/getting-started/data-import/csv-files/

[18] Padial Solier, A. (2019, 20 de octubre). *5 claves para crear un buen Data Lake (y 3 errores a evitar)*. LinkedIn. Recuperat de https://www.linkedin.com/pulse/5-claves-para-crear-un-buen-data-lake-y-3-errores-padial-solier/

[19] Paradigma Digital. (s.f.). *AWS Lake Formation: What is it & What is it used for?*. Recuperat de https://en.paradigmadigital.com/dev/aws-lake-formation-what-is-it-what-is-it-used-for/

[20] Snowflake. (2025, 24 de julio). *What Is a Data Lake? Architecture and Use Cases*. Recuperat de https://www.snowflake.com/es/fundamentals/data-lakes-and-data-lake-architecture/

APÈNDIX

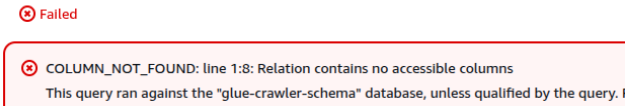

Fig. 8: Error d'usuari amb accés al Data Catalog però sense accés a les taules

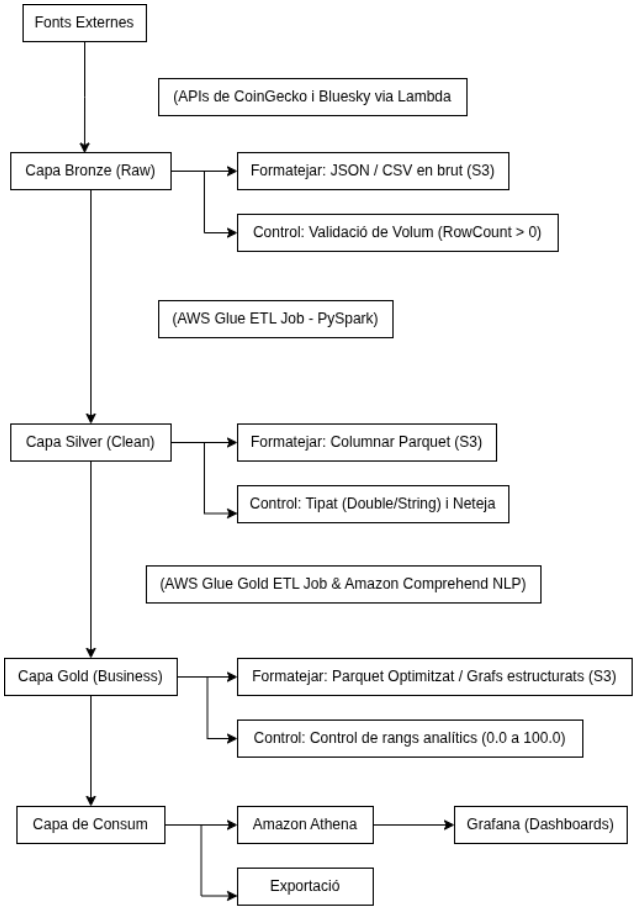

Fig. 9: Flux de dades

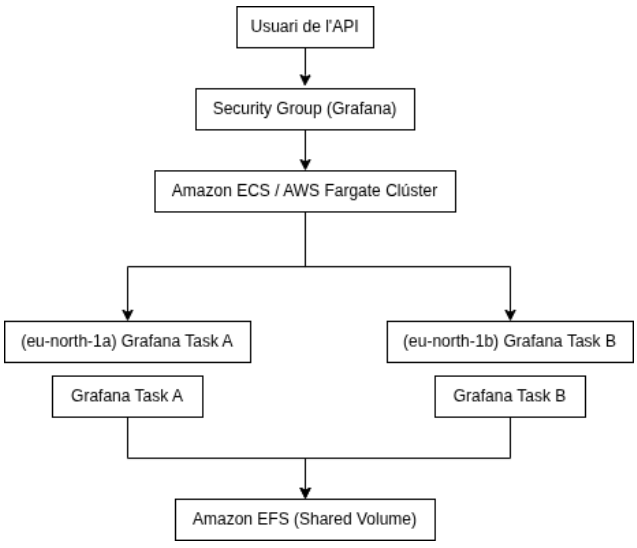

Fig. 10: Arquitectura alta disponibilitat ECS (Grafana)



Fig. 11: Execució DQDL

id	author_did	created_at	timestamp	text_cleaned	crypto_ment	positive	negative	neutral
01536fc	did:plc:6kes	2026-05-22	2026-05-22	5 Performers vs. B	['bitcoin']	0.0037	5.513317	0.99612
01ecc3e	did:plc:lbp5h	2026-06-12	2026-06-12	Forgot mention acc	['tron']	0.4352	0.047336	0.43049
01f59d3	did:plc:3cjbo	2026-05-22	2026-05-22	$2.2K $TRX SHOR	['tron']	0.0043	0.003932	0.99143
02eae27	did:plc:kyxij	2026-06-12	2026-06-12	Market conditions fr	['solana']	0.0524	0.004554	0.94281
03a808c	did:plc:553ee	2026-05-24	2026-05-24	Specific cryptocurr	['solana']	0.7304	0.003779	0.26522
03eafce	did:plc:kpbw	2026-06-01	2026-06-01	Solana payments c	['solana']	0.1685	0.013117	0.81786
04639a8	did:plc:fq3v	2026-06-12	2026-06-12	hourly net stats.	['ethereum']	0.0008	0.000107	0.99907
04da523	did:plc:bd2g	2026-05-23	2026-05-23	Arthur Hayes predi	['bitcoin']	0.0051	0.141288	0.83079
05957f4	did:plc:2yyvx	2026-06-12	2026-06-12	USMNT prepares c	['stellar']	0.3252	0.002103	0.67226
0702430	did:plc:553ee	2026-05-22	2026-05-22	SpaceX revealed h	['bitcoin']	0.0040	0.000313	0.99565
0bf93f69	did:plc:553ee	2026-05-24	2026-05-24	approved Nasdaq p	['bitcoin']	0.0113	0.000224	0.98837
10b6890	did:plc:vqrw	2026-06-03	2026-06-03	Highest waiting tim	['tron']	0.0250	0.000100	0.97476
10d15cc	did:plc:hoyi	2026-06-12	2026-06-12	130 hours Xenobla	['stellar']	0.6121	0.005510	0.35617

Fig. 12: Fragment arxiu de dades dels posts amb sentiment (Capa Gold)

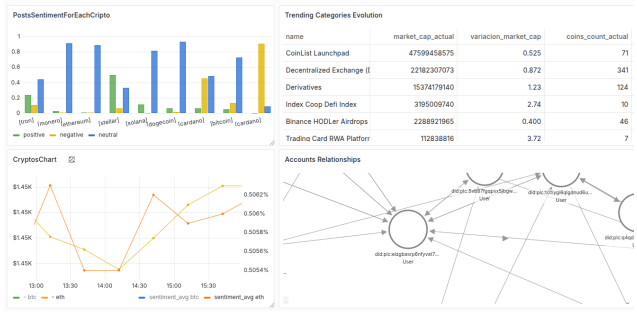

Fig. 13: Visualització de les dades

🚨🔥 BRO, NGL... literally everyone is panicking right now but listen to me, Bitcoin is currently testing crucial support levels. 👀 Like I always say, institutional accumulation quietly continues behind the scenes (IYKYK 🤫 whales are buying!!), setting the stage for the next major macroeconomic
cycle!! 💯💸 While short-term market volatility is actively shaking out weak hands (paper hands gonna
cry 😭📉), the underlying network fundamentals and hash rate remain stronger than ever. 🧠💎🙌 Honestly
just zoom out on the charts, ignore the daily fake news noise, and remember why you built your long-term
conviction in the first place, period. WAGMI!!! LFG!!! 🚀📈🌕 Check my bio for daily VIP signals and next
100x gems 👉 https://t.co/alpha-crypto-signals-vip2026 🚀 #Bitcoin #CryptoMarket #BTC #HODL #Web3 #WAGMI #CryptoAlpha #ToTheMoon

Paraules = 125
Caràcters = 827

BRO NGL... literally panicking listen Bitcoin testing crucial support levels. institutional accumulation quietly continues scenes IYKYK whales buying!! setting stage major macroeconomic cycle!! short term market volatility actively shaking weak hands paper hands gonna underlying network fundamentals hash rate remain stronger ever. Honestly zoom charts ignore daily fake news noise remember built long term conviction place period. Check bio daily VIP signals 100x gems

Paraules = 64
Caràcters = 469

Fig. 14: Neteja Text