INFORME INICIAL
TREBALL DE FI DE GRAU

Autor: Ricard Tuneu Font

Data: 10/10/2024

Tutor: Eduardo Cèsar Galobardes

Universitat Autònoma de Barcelona

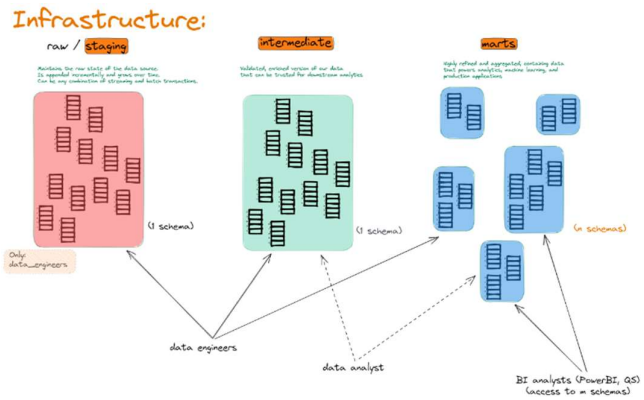
ÍNDEX

1. Context

En conseqüència a realitzar les pràctiques universitàries amb l'empresa Werfen S.A. se m'ha proposat un projecte interessant per a tractar davant l'ampliació del departament de data de la pròpia empresa en els últims anys.

L'empresa està en procés de la generació d'un "datalake" on emmagatzema totes les dades que es generen i que aporten un valor directe en el desenvolupament i creixement de la companyia.

En la següent imatge es pot veure l'estructura d'aquest llac de dades on hi podem veure tres grans divisions.



Imatge 1. Infraestructura base de dades

La estructura està plantejada de forma que en el primer procés, a "staging" s'ingesta les dades provinents de les diferents fonts que utilitza l'empresa, seguidament ve "intermediate" que són taules que s'alimenten generalment de "staging" i que és a grans trets on es formateja les dades obtingudes. Per últim tenim els "marts" que s'alimenten de les "intermediate" on s'agrupen per els diferents dominis que té l'empresa com poden ser "supply chain", finances... i aquestes taules són les que utilitzen directament els analistes de BI en les seves aplicacions per mostrar un anàlisi de les dades.

El volum de dades i la complexitat estructural de la base de dades està contínuament en creixement i ens trobem davant d'un problema a l'hora de recarregar les taules que subministren la informació a les aplicacions que s'utilitzen. Actualment aquestes taules, a excepció d'alguns casos i del grup de "staging" que és on s' ingesta les dades provinents de diverses fonts, es carreguen totes a la vegada. És a dir, totes les taules d' "intermediate" i els

diferents grups de marts es carreguen a la mateixa hora amb un schedule que es va decidir posar en el seu dia.

Antigament quan hi havia poc volum de dades podia tenir un sentit lògic, però actualment això en les hores de càrrega genera un col·lapse en la base de dades ja que es destinen tots els recursos en poder obtenir aquestes dades la qual cosa provoca augment en la latència si es volen realitzar consultes, a banda d'altres problemes de sincronització o inclús de timeouts si el sistema es satura.

Així doncs davant d'aquest problema sorgeix la possibilitat de millorar el flux de càrrega de dades i d'aquesta manera proposar-me una sèrie d'objectius a satisfer amb la realització del treball.

2. Objectius

Després de contextualitzar la raó d'aquest projecte i analitzar la problemàtica que sorgeix m'he plantejat els objectius següents :

- **Objectiu 1 :** Ser capaç de extraure els camins de dependències de cadascuna de les taules de l'esquema "mart" per poder organitzar i sincronitzar l'horari de càrrega de les taules.
- **Objectiu 2:** Optimitzar els fluxos de càrrega de taules i garantir que es tenen les dades actualitzades a l'hora que les aplicacions les necessiten.
- **Objectiu 3:** Generar una visualització o "dashboard" amb els resultats obtinguts per poder captar i entendre la nova proposta amb els fluxos optimitzats.

A banda d'aquests objectius més específics del treball i com a conseqüència de la feina a realitzar em proposo aprendre a utilitzar eines com la API de Github, Prefect (orquestrador) i PowerBI entre d'altres.

3. Estat de l'art

Actualment amb el creixement massiu de dades en empreses com pot ser Werfen fa que es plantegin estructures de bases de dades com la que s'utilitza actualment que com he explicat en el context del treball es basa en un "datalake".

3.1 Estructura Datalake

Per entrar una mica més en detall en quan a com l'empresa emmagatzema i organitza les dades tenim aquest llac de dades que està composat principalment per 2 eines fonamentals. La primera és Amazon Redshift, aquest producte és el que usa l'empresa per a guardar la informació de les taules que es generen. Aquestes taules es guarden fent ús de la segona eina que és Github. És a dir, per un costat tenim els models que llegeixen les taules i que estan en un repositori de Github i per altra banda l'interior de les taules, és a dir les pròpies dades estan en el núvol d'AWS. A més a més el "datalake" consta de 3 nivells de taules com ja he explicat abans. El primer nivell és "staging", aquí es on s' ingesta les dades provinents de fonts com SAP, SharePoints, Excels...

Les dades arriben sense cap tipus de filtre ni format. És a "intermediate", al següent nivell on s'apliquen filtres que interessin a l'empresa, es defineix també el tipus de camps que tenim ja siguin numèrics o de text i també es posa nom als diferents camps ja que solen venir amb codis que no ajuden a comprendre que és el que hi ha a la taula.

Per últim, tenim els "marts". Aquests són taules fetes a partir de les "intermediate" i són pensades ja per a usar-les en les aplicacions o visualitzacions que requereixin fer els Business Inteligence, és a dir que ells simplement s'encarreguin de mostrar de la millor manera les dades obtingudes sense haver de pensar en haver d'afegir lògica a les taules ja que ja ho han fet els analistes de dades prèviament.

3.2 Prefect

A banda de l'estructura de dades que pugui tenir l'empresa es requereix d'una eina que mantingui les taules actualitzades ja que la generació i entrada de dades a la companyia és constant i per tant l'empresa no es pot quedar amb una simple foto del que hi ha a una taula sinó que ha de ser capaç d'actualitzar la informació cada cert temps, en funció de amb quina freqüència es necessiti.

L'eina que fa servir Werfen S.A. és Prefect, és un orquestrador que el que permet és precisament el que esmentava anteriorment, ens ajuda a automatitzar els processos de càrrega de taules amb flexibilitat per indicar les hores en que es vol refrescar la informació a banda d'un munt d'informació extra que és molt interessant per a l'empresa per a monitoritzar el funcionament dels fluxos de treball a través de la detecció i gestió d'errors, temps de càrrega, volums de dades carregats...

3.3 Visualització de Dades

Davant del gran volum de dades que es té avui en dia és important poder plasmar la informació obtinguda i analitzada d'alguna manera i per això fem ús de dashboards.

En l'empresa, tot i utilitzar eines com QlikSense o Tableau en alguns casos, cada vegada s'està derivant més a l'ús de PowerBI establint fins i tot una sèrie de pràctiques per a que totes les visualitzacions que es fan segueixin una mateixa línia d'estil.

PowerBI és una eina capaç de gestionar grans volums de dades per mostra-les en forma de visualització a través d'una interfície molt clara i senzilla. A més a més et permet compartir els dashboards que realitzes fàcilment i actualment també està entrant en el món de la intel·ligència artificial la qual cosa fa que sigui una de les eines més atractives per a treballar i és per això que Werfen S.A. està apostant fort per a ella.

En aquest projecte també serà ideal per projectar els resultats que s'obtinguin i que a banda de ser útil per treball, també pugui ser eficient per a la pròpia empresa.

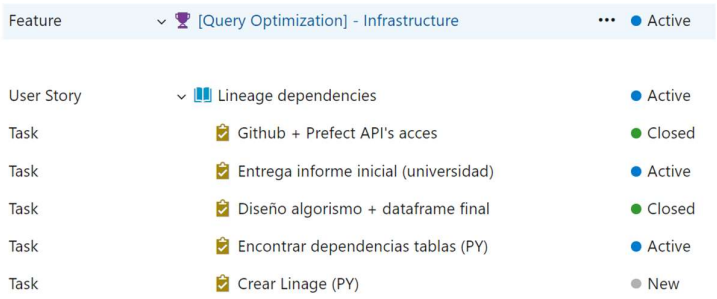
4. Metodologia

La metodologia i organització del projecte es realitzarà a través d'Agile i de l'eina Azure Devops. Aquest projecte dins de l'empresa està situat en el Departament de Laboratori, és allà on es fan aquest tipus de projectes destinats a millores per a la pròpia empresa.

Dins del departament i seguint la metodologia Agile es realitzen Sprints de dues setmanes on el primer dia es fa el sprint planning per assignar les tasques que farà cadascú durant els 10 dies laborables que ocupa. A més d'això cada dia es realitzen reunions d'aproximadament 15 minuts amb tots els membres del departament per poder explicar cadascú en quin estat està la seva feina i si ha sorgit algun problema o no.

Per poder tenir les tasques que fa cadascú a mà i poder ser visibles per a tothom s'utilitza l'eina Azure Devops. Per a aquest projecte dins l'equip de Laboratori s'ha generat una història d'usuari que se li ha anomenat "Linage Dependencies" i dins d'aquesta les diferents tasques que es van fent o que estan pendents de fer.

En la següent captura de pantalla es mostra actualment com està organitzat:



Imatge 3. Estructura tasques Azure Devops

A banda d'això també he fet una mica de planificació sobre el desenvolupament del treball i que constaria de 3 parts així fent referència als 3 objectius que he plantejat anteriorment.

Octubre -> Obtenció del camí de dependències de totes les taules de tipus mart.

Novembre -> Optimització dels fluxos de càrrega a través de les dependències entre taules trobades

Desembre-> Generar "dashboard" que mostri visualment els camins de dependències i a la vegada l'optimització realitzada.

5. Bibliografía / Webgrafia

[1] Què és AWS? Link : https://aws.amazon.com/es/what-is-aws/ (consultat 08/10/2024)

[2] ChatGPT. Link: https://chatgpt.com/ (consultat 08/10/2024)

[3] Documentació Prefect. Link: https://docs.prefect.io/3.0/develop/index (consultat 10/10/2024)

[4] Azure Devops. Link: https://dev.azure.com/ (consultat 10/10/2024)