E2.1 – INVESTIGACIÓN Y ELECCIÓN DE LA TECNOLOGÍA

BD+ML Apply Confirming

CONTENIDO

1. INTRODUCCIÓN

En este documento se exponen diversas tecnologías y arquitecturas sugeridas como solución a los desafíos planteados en el análisis funcional para el desarrollo del nuevo sistema ***BI+ML apply Confirming*** de Business Intelligence (BI) y Machine Learning (ML) de la Plataforma de Financiamiento Comercial (PFC) de Confirming.

A través del estudio y análisis de estos desafíos, se han investigado y propuesto tecnologías y arquitecturas adecuadas para abordar el desarrollo de los nuevos sistemas de Business Intelligence y Machine Learning.

A continuación, se presentarán tecnologías de vanguardia actualmente disponibles para el diseño y desarrollo de soluciones de Business Intelligence (BI) y Machine Learning (ML). Estas tecnologías abarcan desde la extracción de datos del sistema origen hasta la obtención de modelos de predicción.

Por último, se procederá a la selección e implementación de las tecnologías propuestas para cada problema.

2. INVESTIGACIÓN Y DISEÑO DE LA ARQUITECTURA DE LA SOLUCIÓN

En los próximos apartados se detallarán las tecnologías analizadas para la creación del nuevo sistema *BD+ML apply Confirming*, para la Plataforma de Financiación Comercial de Alvantia.

2.1. Herramientas ETL

En palabras simples, ETL es un proceso de **integración de datos** en el que una empresa trae datos de múltiples fuentes y los recopila en un único almacén de datos unificado por lo general llamado **Data Warehouse**. Estas herramientas ayudan a las empresas a habilitar estrategias de integración de datos y toma de decisiones, al recopilar diferentes datos y consolidarlos en un único almacén de datos.

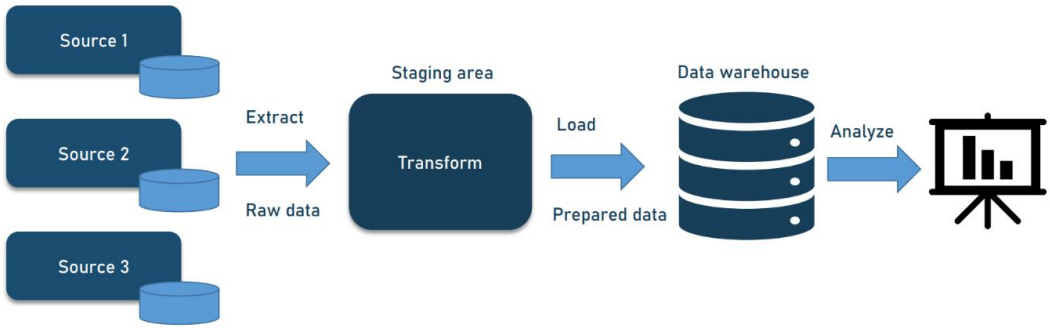
ETL se resume en:

- **Extracción de datos**: necesita extraer datos de varias fuentes y almacenarlos en una única base de datos centralizada.

- **Transformación de datos**: La fase de transformación de un proceso de ETL aplica una serie de reglas de negocio o funciones sobre los datos extraídos para convertirlos en datos que serán cargados.

- **Carga de datos**: En esta fase, los datos procedentes de la fase anterior (fase de transformación) son cargados en el sistema de destino. Dependiendo de los requerimientos de la organización, este proceso puede abarcar una amplia variedad de acciones diferentes. Por ejemplo, en algunas bases de datos será necesario sobrescribir la información antigua con nuevos datos mientras que, en otras, bastará con resumir las transacciones y almacenar un promedio de la magnitud considerada.

ARQUITECTURA ETL

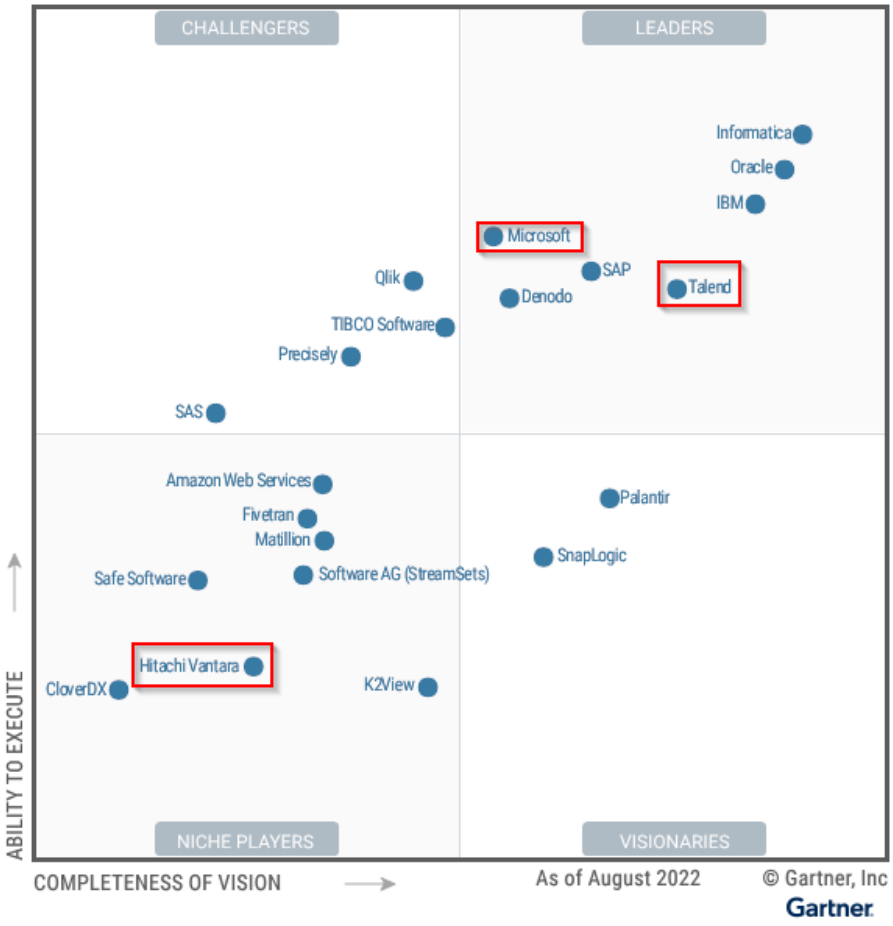


2.1.1. Características de una herramienta ETL

En un proceso ETL no todo vale, por eso hay que saber elegir. Para hacerlo con conocimiento de causa es necesario conocer las características más importantes que debe incluir una herramienta ETL. Según **Gartner**, son las siguientes:

- **Conectividad / capacidades de adaptación (con soporte a orígenes y destinos de datos)**. Se refiere a la capacidad de establecer conexiones con una amplia gama de fuentes de datos, que pueden incluir: bases de datos tanto relacionales como no relacionales, diversos formatos de archivos (CSV, XML, etc.), sistemas ERP (Enterprise Resource Planning), CRM (Customer Relationship Management), entre otras.
- **Capacidades de entrega de datos**. Suponen la habilidad para proporcionar datos a otras aplicaciones, procesos o bases de datos en varias formas, con capacidades para programación de procesos *batch*, en tiempo real o mediante lanzamiento de eventos.
- **Capacidades de transformación de datos**. Capacidad para la transformación de los datos, desde transformaciones básicas (conversión de tipos, manipulación de cadenas o cálculos simples) o transformaciones intermedias (agregaciones, cruces, uniones) hasta transformaciones complejas, como análisis de texto en formato libre o texto enriquecido.
- **Capacidades de Metadatos y Modelado de Datos**. Recuperación de los modelos de datos desde los orígenes de datos o aplicaciones, creación y mantenimiento de modelos de datos, mapeo de modelo físico a lógico, repositorio de metadatos abierto (con posibilidad de interactuar con otras herramientas), sincronización de los cambios en los metadatos en los distintos componentes de la herramienta, documentación, etc.
- **Capacidades de diseño y entorno de desarrollo**. Representación gráfica de los objetos del repositorio, modelos de datos y flujos de datos, soporte para test y, capacidades para trabajo en equipo, gestión de flujos de trabajo o *workflows* de los procesos de desarrollo, etc.
- **Capacidades de gestión de datos (calidad de datos, perfiles y minería)**.
- **Adaptación a las diferentes plataformas hardware y sistemas operativos existentes**.
- **Operaciones y capacidades de administración**. Capacidad para la gestión, monitorización y control de los procesos de integración de datos, como gestión de errores, recolección de estadísticas de ejecución, controles de seguridad, etc.
- **Arquitectura e integración.** Grado de compactación, consistencia e interoperabilidad de los diferentes componentes que forman la herramienta de integración de datos (con un deseable mínimo número de productos, un único repositorio, un entorno de desarrollo común, interoperabilidad con otras herramientas o vía API).

Gartner vuelve a posicionar a **Talend** y a **Microsoft SSIS** entre las herramientas líderes de **procesos de extracción, transformación y carga (ETL).** No obstante, en nuestro análisis tecnológico al tratarse de una herramienta open source o de código abierto, hemos incluido a Hitachi Ventara la cual desarrolla Pentaho Data Integration (PDI), también posicionada en el Cuadrante Mágico de Gartner por séptimo año consecutivo.



2.1.2. Pasos para elegir una herramienta ETL

Conviene considerar algunos pasos que es recomendable seguir antes de elegir una herramienta ETL. Los más importantes son los siguientes:

- Preguntarse por la **cantidad de datos** (en GB o TB) para un cierto período de procesamiento de la herramienta.
- Comprobar la conectividad con el tipo de **fuentes de datos** en la herramienta ETL.
- Comprobar el **formato de datos** que solicitado ya sea texto o CSV, XML, las bases de datos como Oracle, PostgreSQL, SQL Server, MySQL, DB2, Sybase, cualquier longitud fija formato o cualquier otro formato.
- Hacer cumplir las reglas de **validación de datos** mediante la especificación de estas en los procesos.
- Documentar y formalizar los flujos de datos y reglas de asignación.
- Preguntarse cuáles son las funciones de registro y control de las cargas y la forma de manejar condiciones de error.

2.1.3. Creación de un Data Warehouse mediante una herramienta ETL

Una herramienta ETL, por lo general, proporciona una interfaz gráfica para los usuarios a la hora de crear y gestionar las transformaciones. A continuación, destacamos las ventajas y desventajas de desarrollar un Data Warehouse con una herramienta ETL:

2.1.3.1. Ventajas

- La obtención de una interfaz gráfica para crear y mantener los procesos ETL.
- El almacenamiento centralizado de los procesos.
- El control de versiones de los programas
- La simplicidad relativa de la personalización de transformaciones.
- El adecuado soporte de metadatos que proporciona.
- La rápida implementación de transformaciones.
- El sistema de depuración integrado en la mayoría de los productos.
- La posibilidad de transformar la programación.
- La posibilidad de transformar la auditoría.

2.1.3.2. Desventajas

- El alto coste inicial que implica.
- El conocimiento de usuario limitado de la mayoría de los productos.
- La optimización, que a veces está limitada debido a los métodos de programación genéricos.

2.1.4. Propuestas de herramientas ETL *on-premise*

Una herramienta o software ETL *on-premise* se caracteriza por la ubicación de sus recursos, los cuales se encuentran alojados y administrados dentro de los propios centros de datos de la empresa, en contraposición a estar ubicados en la nube o en instalaciones remotas.

2.1.4.1. Talend Open Source (TOS)

Talend es una plataforma de **código abierto** basada en Java, es la **única herramienta** del mercado hecha desde cero y basada en estándares, que abarca tareas como la **Integración de Datos,** Integración de aplicaciones, API, análisis y gobernanza de datos.

Talend es más que un entorno de desarrollo de aplicaciones informáticas, **es una plataforma de Data Integration (ETL),** que gestiona e implementa procesos digitales**,** lo que proporcionará una ventaja competitiva frente a otras soluciones que se encuentran actualmente en el mercado.

2.1.4.1.1. Principales características de Talend

- **Integración de Datos.** Talend permite la integración de datos de diversas fuentes, como bases de datos, aplicaciones empresariales, servicios web, archivos y más.
  Facilita la creación de flujos de trabajo de integración mediante un enfoque basado en gráficos, lo que significa que puedes diseñar y visualizar tus procesos de integración.

- **ETL (Extracción, Transformación y Carga).** Talend es conocido por sus capacidades ETL, lo que significa que puedes extraer datos de diferentes fuentes, transformarlos según sea necesario y cargarlos en el destino deseado.

- **Conectividad.** Ofrece una amplia gama de conectores predefinidos para una variedad de fuentes y destinos de datos, facilitando la conexión con diferentes sistemas.

- **Calidad de Datos.** Incluye funcionalidades para el perfilado y limpieza de datos, lo que mejora la calidad de los datos en el proceso de integración.

- **Gestión de Datos Maestros.** Permite la gestión centralizada y la consolidación de datos maestros para garantizar la coherencia y la precisión de los datos en toda la organización.

- **Código Abierto.** Talend es una plataforma de código abierto, lo que significa que puedes acceder al código fuente y personalizarlo según tus necesidades. También hay una versión comercial con características adicionales y soporte.

- **Diseño y Desarrollo Colaborativo.** Facilita la colaboración entre equipos de desarrollo, permitiendo que múltiples usuarios trabajen en un mismo proyecto de integración.

- **Monitorización y Gestión.** Proporciona herramientas para monitorear y gestionar los flujos de trabajo de integración, permitiéndote supervisar el rendimiento y la ejecución de los procesos.

- **Compatibilidad con la Nube.** Talend es compatible con entornos en la nube, lo que significa que puedes integrar y gestionar datos en plataformas en la nube como AWS, Azure y Google Cloud.

2.1.4.2. Microsoft Server Integration Services (SSIS)

SSIS son las siglas de SQL Server Integration Services, una parte integral del software de gestión de datos de Microsoft SQL Server, ampliamente utilizado para llevar a cabo una variedad de tareas de migración de datos. Se trata de una herramienta ETL incluida en el conjunto de herramientas de Business Intelligence de Microsoft, cuyo propósito principal es lograr la integración de datos.

Esta plataforma está específicamente diseñada para abordar desafíos relacionados con la integración de datos y flujos de trabajo de aplicaciones. Incluye una herramienta de almacenamiento especialmente diseñada para el proceso ETL.

2.1.4.2.1. Principales características de SSIS

- **Extracción y Transformación:**

SSIS inicia el proceso desde un almacén de datos operacionales, una base de datos especialmente diseñada para integrar información proveniente de múltiples fuentes, permitiendo así operaciones adicionales sobre los datos.

- **Carga (ETL):**

Luego, se lleva a cabo el proceso de Extracción, Transformación y Carga (ETL), que permite la preparación y optimización de los datos para su posterior almacenamiento y uso.

- **Almacenamiento en el Data Warehouse:**

El Data Warehouse captura datos desde diversas fuentes, asegurando que estén accesibles y listos para ser utilizados de manera efectiva.

- **Gestión y Respuesta Empresarial:**

Los datos se almacenan en el Data Warehouse para su consolidación y gestión eficiente. Esto permite responder a preguntas empresariales cruciales, apoyando así la toma de decisiones informadas.

2.1.4.2.2. Otras funcionalidades de SSIS:

Además de las funciones ETL, SSIS ofrece una amplia gama de capacidades adicionales:

- **Limpieza, Agregación y Fusión de Datos:**

SSIS facilita la limpieza y preparación de datos, así como la capacidad de combinar y resumir información de diferentes fuentes.

- **Transferencia de Datos y Extracción Multifuncional:**

Permite la transferencia fluida de datos entre diferentes bases de datos y la extracción de información de una variedad de fuentes, como bases de datos de SQL Server, archivos de Excel, bases de datos Oracle y DB2, entre otros.

- **Herramientas Gráficas y Asistentes para Workflows:**

SSIS incluye herramientas visuales y asistentes que facilitan la creación de flujos de trabajo. Esto incluye funcionalidades como el envío de correos electrónicos, operaciones de FTP, y la gestión de fuentes y destinos de datos.

En resumen, SQL Server Integration Services (SSIS) se destaca por su competencia en la extracción, transformación y carga de datos, proporcionando además un conjunto integral de herramientas para abordar diversas necesidades de integración y gestión de datos en ambientes empresariales. Su habilidad para interactuar con múltiples fuentes y su enfoque en la limpieza y preparación de datos, convierten a SSIS en una herramienta esencial para optimizar los procesos de integración de datos.

2.1.4.3. Tabla comparativa de Talend vs SSIS

A continuación, se muestra la tabla comparativa entre Talend y SSIS.

	SSIS	Talend Data Integration
Fabricante	Microsoft	Talend
Objetivo	Extracción, transformación y carga de datos de múltiples fuentes con diferentes formatos	Realizar trabajos ETL o fuentes de datos de múltiples ubicaciones y necesita integrarlos para una mejor compatibilidad de datos
Pros	Permite ejecutar muchos procesos en paralelo.	Interfaz fácil de usar
	Proporciona muchas herramientas para transformar datos durante el proceso de migración.	Diseñar trabajos es muy sencillo
	Una vez desarrollados, sus paquetes son muy estables y requieren relativamente poco mantenimiento, lo que ahorra mucho tiempo de trabajo.	Talend Data Integration ha optimizado la gestión del almacén de datos, lo que puede reducir tanto los costes como el tiempo
Contras	No puede copiar y pegar anotaciones, por lo que, si tiene varias similares, debe escribir cada una de ellas desde cero	La sincronización con Git es un poco difícil.
	A veces, la conversión de tipos de datos no funciona implícitamente y debe hacerlo explícitamente.	Puede enfrentar problemas con ciertos problemas de Big Data.
Margen de mejora	SSIS puede mejorar el manejo de diferentes tipos de datos. También conectividad con diferentes fuentes de datos como conectividad Salesforce, conectividad Oracle Cloud, etc. será un gran desafío	La versión de código abierto debe incluir características como el control de versiones de código fuente y la ejecución paralela.

	para SSIS. Así que esperamos mejoras en estas áreas.	
Problemas de escalabilidad	Ninguno	Requiere un poco de ajuste antes de llegar al rendimiento óptimo.
Soporte	El soporte es excelente. También puede obtener tareas de script prescritas con una variedad de funcionalidades que ahorran mucho tiempo.	Buen soporte, especialmente cuando se relaciona con el entorno PROD.
Coste	Según licencia Microsoft	Open Source sin soporte

2.1.4.4. Pentaho Data Integration (PDI)

La Plataforma de Integración de Datos Pentaho (PDI) es una aplicación de escritorio que ofrece las capacidades de Extracción, Transformación y Carga (ETL) que simplifican la tarea de extraer, transformar y cargar datos utilizando un formato uniforme y coherente, accesible y pertinente tanto para los usuarios finales como para las tecnologías de Internet de las Cosas (*IoT*).

2.1.4.4.1. Principales características de PDI

- **ETL (Extracción, Transformación y Carga).** PDI se utiliza principalmente para realizar procesos de ETL, lo que implica extraer datos de diversas fuentes, transformarlos según sea necesario y cargarlos en un destino deseado.

- **Diseño Gráfico de Flujos de Trabajo.** Al igual que Talend, PDI utiliza una interfaz gráfica para diseñar flujos de trabajo de integración de datos. Esto permite a los usuarios diseñar y visualizar procesos de ETL de manera intuitiva.

- **Conectividad Versátil.** PDI admite una amplia variedad de conectores para diversas fuentes de datos, lo que facilita la integración con bases de datos, archivos, servicios web y otros tipos de datos.

- **Código Abierto.** Pentaho Data Integration es de código abierto, lo que significa que puedes acceder al código fuente y personalizarlo según tus necesidades.

- **Transformaciones y Limpieza de Datos.** Proporciona una gama de operaciones de transformación y limpieza de datos que pueden aplicarse a medida que los datos se mueven a través del proceso de ETL.

- **Orquestación y Planificación.** Permite programar y orquestar los flujos de trabajo, lo que facilita la automatización de tareas de integración de datos.

- **Integración con otras Herramientas Pentaho.** PDI se integra con otras herramientas de Pentaho, como Pentaho Reporting y Pentaho Analysis, para proporcionar soluciones integrales de BI y análisis.

- **Escalabilidad** y Despliegue. PDI es escalable y puede implementarse en entornos distribuidos para manejar grandes volúmenes de datos.

2.1.4.4.2. Ventajas de Pentaho Data Integration

- Carga de conjuntos de datos extensos en bases de datos aprovechando al máximo entornos de procesamiento en la nube, en clústeres y de procesamiento masivamente paralelo.
- Limpieza de datos con pasos que van desde transformaciones muy simples hasta muy complejas.
- Integración de datos, incluida la capacidad de aprovechar ETL en tiempo real como fuente de datos para *Pentaho Reporting*.
- Carga de almacenes de datos con soporte incorporado para dimensiones cambiantes de forma gradual (SCD) y creación de claves subrogadas.
- Migración de datos entre diferentes bases de datos y aplicaciones.

2.1.4.4.3. Desventajas de Pentaho Data Integration

- Es un conjunto de herramientas diseñadas por varios desarrolladores, lo que significa que nos encontraremos con menús, diseños y visualizaciones diferentes, lo que puede dificultar un poco el aprendizaje de las nuevas herramientas de Pentaho.
- Al tratarse de una versión de la comunidad, también carece del respaldo oficial de Hitachi, la empresa creadora de Pentaho.
- La documentación, al ser de una versión de la comunidad, es limitada y a veces no resulta suficiente, por lo que a menudo necesitamos recurrir a otros usuarios que hayan experimentado el mismo problema o necesidad.

2.1.5. ETL Cloud - Integración de datos en la nube

La integración de datos reúne información de distintos orígenes de datos con el fin de almacenarlos en un en Data Warehouse o almacén de datos y finalmente analizarlos para la toma de decisiones. A continuación, veremos las ventajas y ventajas de implementar un entorno de integración de datos en la nube.

2.1.5.1.1. Ventajas de ETL Cloud

- **Ahorro en infraestructura:** no es necesario contar con un servidor y el software puede ser tanto local e integrado con la nube como a modo de servicio.
- **Escalabilidad:** tanto en almacenamiento como en procesamiento de datos se permite hacer desarrollos escalables de forma rápida y a bajo costo.
- **Velocidad:** en respuestas al analizar datos, sacar conclusiones, tomar decisiones *data-driven*, gracias a una visión global.
- **Agilidad:** incorporación de nuevas fuentes de datos para responder con rapidez y a un costo eficiente a los cambios externos o internos de la organización.
- **Facilidad de uso:** los usuarios manejan el sistema sin complicaciones ni pérdidas de tiempo en implementaciones gracias a las facilidades de la escalabilidad.
- **Explotación de los datos:** mediante el uso de estructuras de software de código abierto, tecnologías para un análisis en tiempo real y herramientas de visualización.

2.1.5.1.2. Desventajas de ETL Cloud

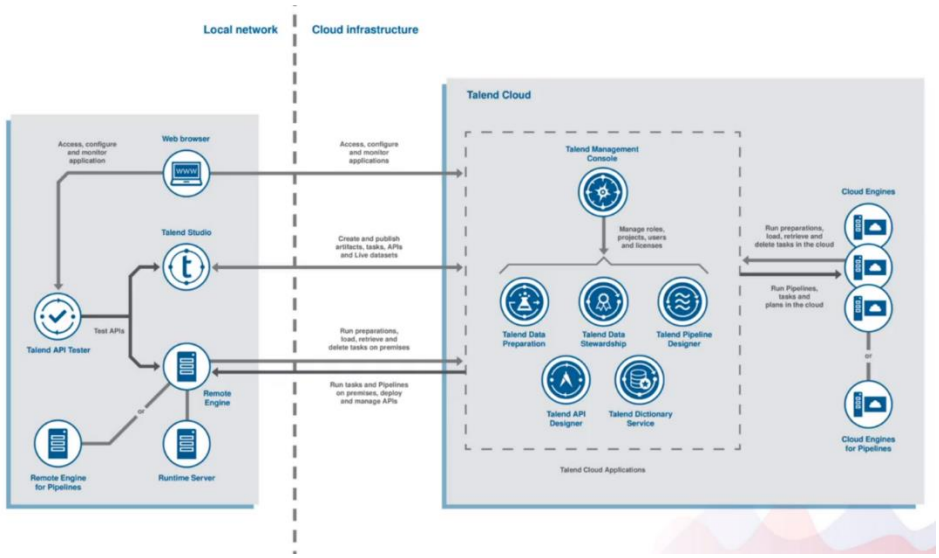
- Dependencia del proveedor de la plataforma.
- Posible control de datos a terceros.
- La salida de un entorno de Cloud pública conlleva importantes costes.

2.1.5.2. Talend Data Fabric

Talend Data Fabric (TDF), es un conjunto de aplicaciones para cubrir todos los proyectos de integración y cualificación de datos empresariales. TDF centraliza todos tus proyectos de gestión de datos en una única herramienta. Permite a las empresas pasar de un proyecto a otro sin tener que formarse en nuevas aplicaciones.

2.1.5.2.1. Ventajas de Talend Data Fabric

- Un enfoque unificado que aborda todas las facetas de la gestión de datos en una plataforma todo en uno.
- Conectividad nativa a todas las fuentes de información del mercado con más de 900 conectores incluidos.
- Funciones para garantizar rápidamente la pertinencia y fiabilidad de sus datos.
- Una interfaz gráfica intuitiva y fácil de usar
- Un sistema flexible y escalable
- Adecuado para entornos híbridos on-premise y multi-Cloud



Talend Data Integration es el software estrella de Talend. Permite a las organizaciones integrar datos rápidamente mediante la creación de flujos ETL de arrastrar y soltar.

2.1.5.3. Precios de Talend Data Fabric

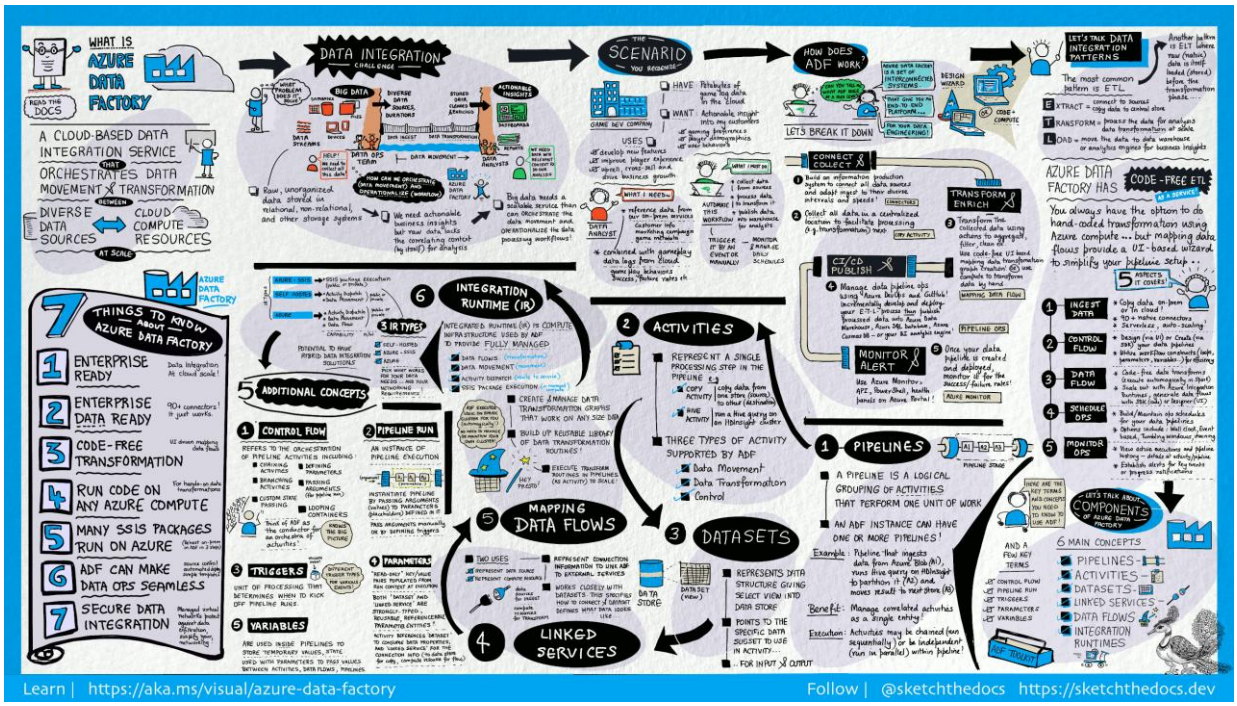
Talend Cloud Data Integration

Users	Description	1 MONTH	12 MONTHS	24 MONTHS	36 MONTHS
Talend Cloud DI	Talend Cloud Data Integration - Per User - includes Gold Support	$1,170	$12,000	$24,000	$36,000

Para más información consultar el siguiente enlace:
https://www.talend.com/pricing/

2.1.6. Microsoft Azure Data Factory

Azure Data Factory es el servicio ETL en la nube de Azure para la integración y transformación de datos sin servidor de escalabilidad horizontal. Ofrece una interfaz de usuario sin código que favorece la creación intuitiva y una supervisión y administración desde un único panel. También se puede realizar la migración mediante lift-and-shift de los paquetes de SSIS existentes a Azure y ejecutarlos con plena compatibilidad en ADF. SSIS Integration Runtime ofrece un servicio totalmente administrado, por lo que no hay que preocuparse de la administración de la infraestructura.



2.1.6.1. Azure Data Factory vs herramientas ETL tradicionales

- Azure Data Factory es una de las mejores opciones a la hora de construir pipelines ETL (o ELT) en la nube e híbridos. Hay algunas características que diferencian a Azure Data Factory de otras herramientas.
- La capacidad de ejecutar paquetes SSIS.
- Adaptación automática según la carga de trabajo dada. De hecho, Azure Data Factory va un paso más allá al asegurar a sus clientes que el precio de Data Factory se basa en el uso. El número de actividades (pasos de procesamiento de datos) al mes y el uso del tiempo de ejecución de la integración se cobra por horas, en función de la máquina y del número de nodos utilizados.
- Enlace entre los sistemas locales y la nube de Azure sin problemas a través de una pasarela.
- Maneja grandes volúmenes de datos que son críticos en la era del Big Data
- Capacidad de conectar y trabajar con otros servicios de computación (Azure Batch, HDInsights) para ejecutar cálculos de datos realmente grandes durante el ETL.
- Por último, una de las mayores ventajas es su rápida y fácil integración con otros recursos de Azure Compute & Storage.

2.1.6.2. Precios Azure Data Factory

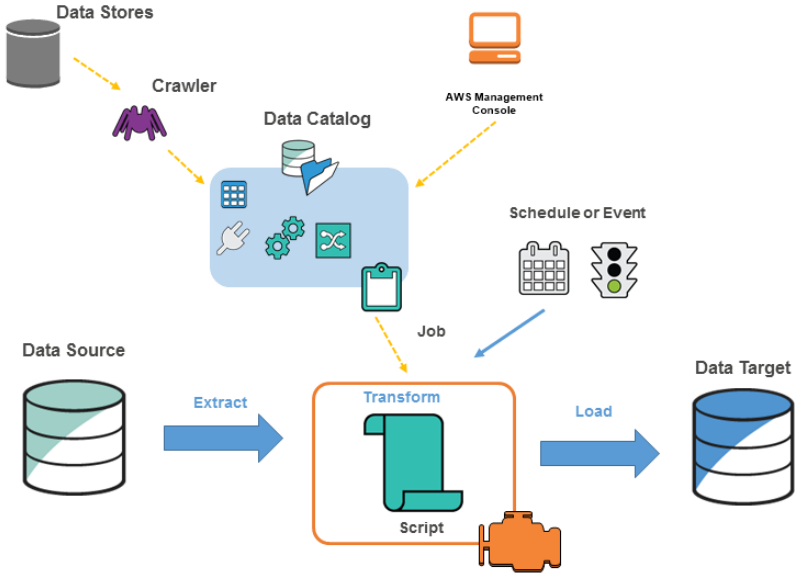
En la siguiente figura se muestra un ejemplo de los precios de Azure Data Factory

Tipo	Precio de Azure Integration Runtime	Precio de un entorno de ejecución de integración con una VNET administrada de Azure	Precio de un entorno de ejecución de integración autohospedado
Orquestación[1]	€0,952 por 1.000 ejecuciones	€0,952 por 1.000 ejecuciones	€1,428 por 1.000 ejecuciones
Actividad de movimiento de datos[2]	€0,238/DIU/hora	€0,238/DIU/hora	€0,096/hora
Actividad de canalización[3]	€0,005/hora	€0,952/hora (Hasta 50 actividades de canalización simultáneas)	€0,001903/hora
Actividad de canalización externa[4]	€0,000238/hora	€0,952/hora (Hasta 800 actividades de canalización simultáneas)	€0,000096/hora

Para más información consultar el siguiente enlace:
https://azure.microsoft.com/es-es/pricing/details/data-factory/data-pipeline/

2.1.7. Amazon AWS Glue

AWS Glue es una plataforma de integración de datos sin servidor que simplifica el proceso para los analistas de datos al descubrir, preparar, migrar e integrar datos provenientes de diversas fuentes. Esta herramienta es versátil y puede ser empleada tanto para análisis, machine learning como para el desarrollo de aplicaciones. Además, ofrece una gama de herramientas suplementarias para mejorar la productividad y optimizar las operaciones relacionadas con los datos, abarcando la creación, ejecución y despliegue de flujos de trabajo empresariales.



2.1.7.1. ¿Por qué usar AWS Glue?

El primer paso de un proyecto de Business Intelligence o ML consiste en preparar los datos para obtener resultados de calidad. **AWS Glue** es un servicio de integración de datos ***sin servidor*** que vuelve más fácil, rápido y barato el proceso de preparación de datos. Puede detectar y conectarse a más de 70 orígenes de datos distintos, administrar los datos en un catálogo centralizado, además de visualizar, crear, ejecutar y supervisar las canalizaciones de ETL para cargar los datos en los lagos de datos correspondientes.

Mediante AWS Glue, se tiene acceso a una extensa variedad de más de 70 fuentes de datos distintas y se facilita la administración de estos datos a través de un catálogo centralizado. Es posible visualizar y administrar de forma gráfica las etapas de extracción, transformación y carga (ETL) para cargar información en los repositorios de datos. Asimismo, se puede realizar búsquedas y consultas instantáneas sobre los datos catalogados utilizando herramientas como Amazon Athena, Amazon EMR y **Amazon Redshift**.

AWS Glue consolida las principales funcionalidades de integración de datos en una sola plataforma. Estas capacidades abarcan desde la detección de datos hasta el proceso ETL moderno, pasando por la limpieza, transformación y catalogación centralizada. Cabe destacar que se trata de un servicio sin servidor, lo que implica que no es necesario preocuparse por la gestión de infraestructuras. Al contar con una flexibilidad amplia para todo tipo de cargas de trabajo, ya sea ETL, ELT o streaming, AWS Glue es apto para una variada gama de usuarios y para diferentes tipos de tareas.

Adicionalmente, AWS Glue facilita la integración de datos en toda la arquitectura. Se integra perfectamente con los servicios de análisis de AWS y con los depósitos de datos almacenados en Amazon S3. Sus interfaces de integración y herramientas de creación de trabajo son intuitivas y accesibles para todo tipo de usuarios, desde desarrolladores hasta profesionales de negocios, ofreciendo soluciones personalizadas adaptadas a diversos niveles de habilidades técnicas.

2.1.7.2. Precios de AWS Glue

- **Trabajo de ETL:** suponga que tiene un trabajo de Apache Spark para AWS Glue que se ejecuta durante 15 minutos y utiliza 6 DPU. El precio de 1 hora de DPU es de 0,44 USD. Dado que el trabajo se ejecutó durante 1/4 de hora y utilizó 6 DPU, AWS le facturará 6 DPU * 1/4 hora * 0,44 USD, o 0,66 USD.
- **Blocs de notas de trabajos de AWS Glue Studio y sesiones interactivas:** suponga que utiliza un bloc de notas en AWS Glue Studio para desarrollar el código ETL de manera interactiva. Una sesión interactiva tiene 5 DPU de forma predeterminada. Si ejecuta la sesión durante 24 minutos o 2/5 partes de una hora, se facturarán 5 DPU * 2/5 de hora a 0,44 USD por hora de DPU, o 0,88 USD.
- **Transformaciones de ML:** de manera similar a las ejecuciones de trabajos de AWS Glue, el costo de ejecutar transformaciones de ML, que incluye usar FindMatches en sus datos, variará según el tamaño y el contenido de los datos y la cantidad y los tipos de nodos que utiliza. En el siguiente ejemplo, utilizamos FindMatches para integrar información de puntos de interés de múltiples orígenes de datos. Con aproximadamente 11 000 000 filas (1,6 GB) en el conjunto de datos, un tamaño de los datos de etiquetas (ejemplos de coincidencias verdaderas o no coincidencias verdaderas) de aproximadamente 8000 filas (641 KB), que se ejecutan en 16 instancias del tipo G.2x, tendría un tiempo de ejecución para la generación de conjuntos de etiquetas de 34 minutos con un costo de 8,23 USD, un tiempo de ejecución para la estimación de métricas de 11 minutos con un costo de 2,66 USD y un tiempo de ejecución para los trabajos de FindingMatches de 32 minutos a un costo de 7,75 USD.

Precios

Región: Europa (España)

- 0,44 USD por hora de DPU para cada **trabajo de Apache Spark** o de **Spark Streaming** que se factura por segundo con un mínimo de 1 minuto (Glue, versión 2.0 o posterior) o de 10 minutos (Glue versión 0.9/1.0)
- 0,44 USD por hora de M-DPU para cada **trabajo de Ray** facturado por segundo, con un mínimo de 1 minuto.
- 0,44 USD por hora de DPU para cada trabajo de **Shell de Python** que se factura por segundo con un mínimo de 1 minuto

Cargos adicionales

Si sus datos de ETL provienen de orígenes de datos como Amazon S3, Amazon RDS o Amazon Redshift, se le cobran tarifas de transferencia de datos y solicitud estándares. Si usa Amazon CloudWatch, se cobran tarifas estándar para los registros y los eventos de CloudWatch.

Ejemplos de precios

Trabajo de ETL: suponga que tiene un trabajo de Apache Spark para AWS Glue que se ejecuta durante 15 minutos y utiliza 6 DPU. El precio de 1 hora de DPU es de 0,44 USD. Dado que el trabajo se ejecutó durante 1/4 de hora y utilizó 6 DPU, AWS le facturará 6 DPU * 1/4 hora * 0,44 USD, o 0,66 USD.

Blocs de notas de trabajos de AWS Glue Studio y sesiones interactivas: suponga que utiliza un bloc de notas en AWS Glue Studio para desarrollar el código ETL de manera interactiva. Una sesión interactiva tiene 5 DPU de forma predeterminada. Si ejecuta la sesión durante 24 minutos o 2/5 partes de una hora, se facturarán 5 DPU * 2/5 de hora a 0,44 USD por hora de DPU, o 0,88 USD.

Transformaciones de ML: de manera similar a las ejecuciones de trabajos de AWS Glue, el costo de ejecutar transformaciones de ML, que incluye usar FindMatches en sus datos, variará según el tamaño y el contenido de los datos y la cantidad y los tipos de nodos que utiliza. En el siguiente ejemplo, utilizamos FindMatches para integrar información de puntos de interés de múltiples orígenes de datos. Con aproximadamente 11 000 000 filas (1,6 GB) en el conjunto de datos, un tamaño de los datos de etiquetas (ejemplos de coincidencias verdaderas o no coincidencias verdaderas) de aproximadamente 8000 filas (641 KB), que se ejecutan en 16 instancias del tipo G.2x, tendría un tiempo de ejecución para la generación de conjuntos de etiquetas de 34 minutos con un costo de 8,23 USD, un tiempo de ejecución para la estimación de métricas de 11 minutos con un costo de 2,66 USD y un tiempo de ejecución para los trabajos de FindingMatches de 32 minutos a un costo de 7,75 USD.

Para más información, consultar el siguiente enlace:
https://aws.amazon.com/es/glue/pricing/

2.1.8. Google Cloud Dataflow

Google Cloud Dataflow es un servicio de procesamiento de datos en tiempo real y por lotes, que también puede ser utilizado para realizar tareas de ETL. Permite la creación de flujos de trabajo para la transformación y el análisis de datos a gran escala utilizando un modelo de programación basado en Apache Beam. Google Cloud Dataflow se integra con otros servicios de Google Cloud Platform, como Google BigQuery y Google Cloud Storage.

2.1.8.1. Características de Google Cloud Dataflow

- **Auto escalado de recursos y restablecimiento dinámico del equilibrio de trabajo.**
  Reduce al mínimo la latencia del flujo de procesamiento, aumenta la utilización de recursos y disminuye los costes de procesamiento de los registros de datos mediante el auto escalado de recursos basado en datos. El sistema realiza particiones automáticas de las entradas de datos que, a su vez, se nivelan constantemente para equilibrar la utilización de recursos de trabajadores y reducir el efecto de las "claves en caliente" en el rendimiento del flujo de procesamiento.

- **Programación y precios flexibles para el procesamiento por lotes.**
  Algunas tareas se pueden programar de forma más flexible, por ejemplo, para ejecutarlas por la noche. En estos casos, el procesamiento por lotes cuesta menos si usas FlexRS, que es la programación flexible de recursos. Las tareas flexibles se ponen en cola, con la garantía de que se ejecutarán en un plazo máximo de seis horas.

- **Patrones de IA en tiempo real listos para usar.**
  Las funciones de inteligencia artificial (IA) en tiempo real que ofrece Dataflow se habilitan mediante patrones listos para usarse, lo que te proporciona un sistema capaz de reaccionar al instante a grandes cantidades de eventos con una inteligencia casi humana. Los clientes pueden elaborar soluciones inteligentes de todo tipo, como análisis predictivos, detección de anomalías, personalización en tiempo real y otros usos de analíticas avanzadas.

2.1.8.2. Precios de Google Dataflow

Las tareas de Dataflow se facturan por segundos, según el uso real de trabajadores de procesamiento por lotes o de streaming de Dataflow. Si utilizas otros recursos, como Cloud Storage o Pub/Sub, se te cobrarán las tarifas correspondientes a cada servicio.

¿Cómo se facturan las unidades de procesamiento de Dataflow?

Se te factura por segundo según la cantidad total de DPU que consume tu trabajo durante una hora determinada. El precio de una sola DPU varía según el tipo de trabajo: por lotes o de transmisión.

Países Bajos (europe-west4)

Job Type (Tipo de trabajo)	**Procesamiento de Dataflow (por DPU por hora)**
Lote	$0.078
Transmisión	$0.090

Para más información, consultar el siguiente enlace:
https://Cloud.google.com/dataflow/pricing?hl=es

2.1.9. Informatica Cloud Data Integration

Basada en los Servicios de Nube Inteligente (IICS) de Informatica para Integración de Datos en la Nube

Con IICS, las empresas tienen la capacidad de aprovechar la innovadora Plataforma de Integración como Servicio (iPaaS) para impulsar su transformación digital centrada en los datos en un entorno de múltiples nubes. Además, se benefician de una solución flexible de gestión e integración de datos basada en la nube, diseñada para expandirse al ritmo de la empresa. IICS proporciona una experiencia de usuario unificada y moderna a través de diversos modelos de integración iPaaS, y soporta una arquitectura de microservicios que es personalizable y de navegación sencilla. La nueva experiencia iPaaS aprovecha el motor CLAIRE™ para ofrecer recomendaciones basadas en metadatos de inteligencia artificial, con el objetivo de automatizar y acelerar los procesos de integración y gestión de datos.

2.1.9.1. Ventajas principales de Informatica Cloud:

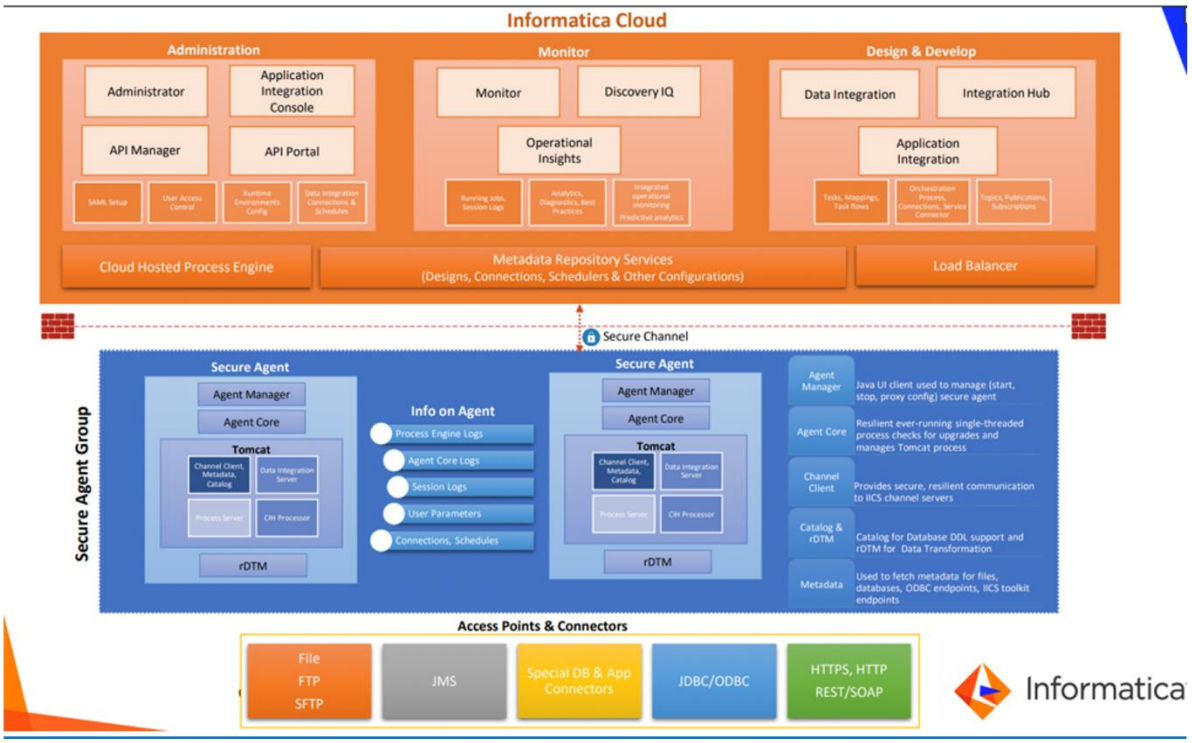
- **Compatibilidad con cargas de trabajo de gran volumen.** La capacidad de incorporación masiva de archivos transfiere activos de datos empresariales en un formato de archivo plano desde el entorno local a los almacenes de datos de Amazon S3 y los Data Warehouses de Amazon Redshift en el Cloud mediante los protocolos estándar FTP, SFTP y FTPS. Los desarrolladores pueden crear fácilmente tareas de incorporación masiva con asistentes y supervisar las ejecuciones de tareas, con una gran precisión tanto en el archivo como en el trabajo, utilizando el mismo servicio de supervisión empleado en otros trabajos de integración de datos. La incorporación masiva de archivos se ha diseñado para manejar miles de archivos al día en un único proceso.

- **Diseño de mappings en Cloud.** Cloud Mapping Designer proporciona un asistente fácil de usar que genera cargas de datos tanto simples como complejas para proyectos de integración de datos en Cloud de gran escalabilidad. Con unos pocos clics, puede crear y conectar varios endpoints en los entornos locales y en Cloud, así como transformar, filtrar y programar cargas de trabajo en el Data Warehouse en Cloud para la generación de análisis e informes.

- **Ejecución de flujos de integración de datos visuales.** Gracias al flujo de tareas avanzado, los usuarios diseñan flujos de tareas tanto simples como complejos mediante la coordinación de tareas de mapping y la sincronización de tareas de manera no linear y paralela. Los usuarios pueden definir una lógica personalizada que implica acciones como tareas en paralelo, bucles, condiciones, decisiones, tiempo de espera, excepciones y el tratamiento de errores y de excepciones. De este modo, logran coordinaciones mucho más complejas que antes. El diseñador

de flujos de tareas avanzado tiene una interfaz de usuario similar a la del diseñador de mappings y se integra a la perfección con el servicio de supervisión.

- Sincronización y replicación en batch de gran rendimiento de los datos de las aplicaciones de Cloud y del entorno local.
- Escalado con replicación y sincronización de datos.
- Compatibilidad con cargas de trabajo de gran volumen.
- Ejecución de flujos de integración de datos visuales.
- Activos, proyectos y carpetas reutilizables para escalar la gestión de activos empresariales.



2.1.9.2. Precios de Informatica Cloud Data Integration

	Data Loader— Gratis	Recommended Cloud Data Integration— Gratis	Cloud Data Integration— PayGo	Intelligent Data Management Cloud
	PÓNGASE EN MARCHA	PÓNGASE EN MARCHA	PÓNGASE EN MARCHA	INICIAR CHAT AHORA
Filas/horas	Sin límite	Hasta 20M filas/10 horas de cálculo (al mes)	Número ilimitado de filas/ horas de cálculo	Precios de consumo ilimitados
Usuarios	Unlimited	Unlimited	Unlimited	Unlimited
Experiencia de usuario con asistente	✓	✓	✓	✓
Incorporación	✓	✓	✓	✓
Soporte	Comunidad	Comunidad + Chat en directo	Comunidad + Chat en directo + Básico	Comunidad + Premium + Signature Select
Programación	✓	✓	✓	✓
Transformaciones estándar		✓	✓	✓
Coordinación		✓	✓	✓

Para obtener información detallada de los precios, consultar el siguiente enlace:
https://www.informatica.com/es/free-data-integration.html

2.2. Almacén de Datos o Data Warehouse

Con el objetivo de almacenar la información en un Data Warehouse o almacén de datos, a continuación, explicará qué es, para qué sirve, así como sus ventajas y desventajas.

2.2.1. ¿Qué es un almacén de datos o Data Warehouse?

Un Data Warehouse o almacén de datos es una base de datos diseñada para el almacenamiento de grandes volúmenes de datos, optimizado para la consulta analítica de datos provenientes de diversas fuentes u orígenes de datos.

A diferencia de las bases de datos convencionales o relacionales las cuales están diseñadas para el procesamiento transaccional (OLTP) y la recuperación eficiente de datos individuales, un Data Warehouse está optimizado para procesamiento analítico en línea (OLAP), el análisis de grandes conjuntos de datos, así como para la integración con herramientas de inteligencia empresarial o **Business Intelligence** (BI).

2.2.2. ¿Para qué sirve un Data Warehouse?

Un Data Warehouse es una herramienta fundamental para empresas y organizaciones que necesitan gestionar grandes volúmenes de datos y desean aprovecharlos para tomar decisiones estratégicas y operativas fundamentadas.

2.2.3. Ventajas del uso de un Data Warehouse

- **Soporte para la Toma de Decisiones**: Al contar con datos organizados y listos para el análisis, el Data Warehouse capacita a los líderes y tomadores de decisiones a evaluar información relevante y fundamentada para la toma de decisiones informadas.
- **Generación de Informes:** Facilita la creación de informes personalizados y tableros de control que proporcionan una visión clara y actualizada del estado y rendimiento de la organización.
- **Consolidación de Datos:** Permite integrar información proveniente de múltiples fuentes, como bases de datos operacionales, sistemas de gestión, aplicaciones y más. Esto facilita la obtención de una visión unificada y completa de los datos de la empresa.
- **Historización de Datos:** Permite mantener un historial completo de los datos a lo largo del tiempo. Esto es crucial para analizar tendencias, patrones y cambios en el tiempo, lo que puede ser esencial para la planificación a largo plazo y la evaluación del rendimiento.
- **Mejora del Rendimiento:** Al estar diseñado y optimizado específicamente para consultas y análisis, un Data Warehouse ofrece tiempos de respuesta más rápidos y un rendimiento superior en comparación con bases de datos operacionales tradicionales.

2.2.4. Desventajas del uso de un Data Warehouse

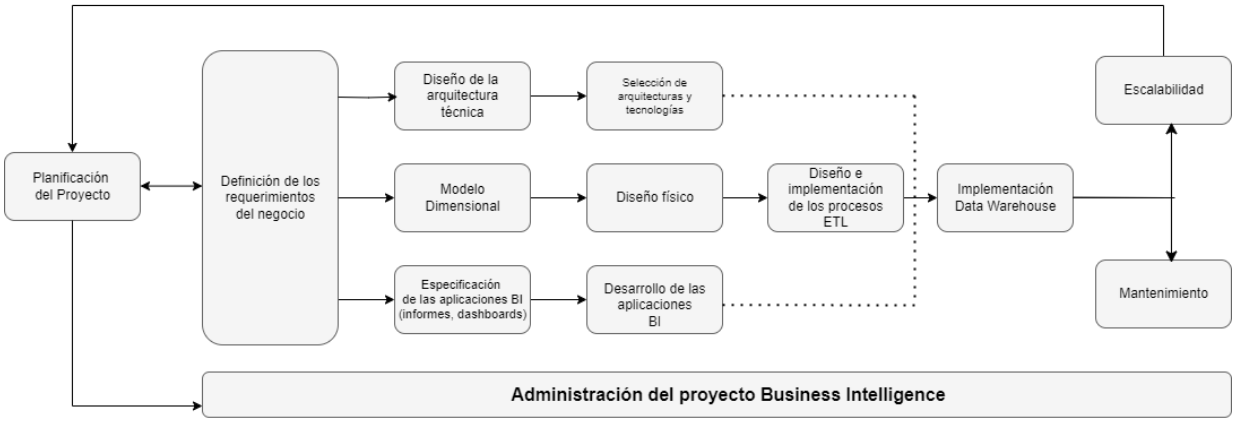
Las principales desventajas de desarrollar un Data Warehouse son:

- **Costes Iniciales Elevados:** La implementación de un Data Warehouse puede requerir una inversión significativa en hardware, software, capacitación y personal especializado. Esto puede ser prohibitivo para organizaciones con presupuestos limitados.
- **Complejidad de Diseño y Modelado:** El diseño adecuado de un Data Warehouse requiere un conocimiento profundo de la arquitectura de datos, la modelización dimensional y las mejores prácticas en la industria. Esto puede ser desafiante y requiere experiencia específica.
- **Sensibilidad a la calidad de los datos de entrada:** La precisión y la consistencia de los datos que se cargan en el Data Warehouse son fundamentales para obtener resultados precisos y confiables.

- **Posibilidad de Redundancia de Datos:** Si no se gestiona adecuadamente, un Data Warehouse puede llevar a la duplicación de datos, lo que puede aumentar los costos de almacenamiento y complicar la gestión de la integridad de los datos.
- **Posibles Desafíos de Escalabilidad:** A medida que la cantidad de datos y la complejidad de las consultas aumentan, puede ser necesario escalar el hardware y los recursos del Data Warehouse. Esto puede requerir una inversión adicional.

2.2.5. Ciclo de vida de un Data Warehouse

A continuación, podemos observar el esquema-resumen que presenta **Ralph Kimball** como propuesta de las etapas del ciclo de vida de proyectos de **Business Intelligence.**



2.2.6. Data Warehouse On-premise

Un Data Warehouse *on-premise* es una base de datos centralizada que se utiliza para almacenar grandes cantidades de datos históricos y actuales, con el propósito de realizar análisis y generar informes de negocio. Un Data Warehouse *on-premise* se encuentra implementado en las instalaciones físicas de una empresa u organización.

Su función primordial es la centralización y conservación de grandes volúmenes de datos procedentes de diversas fuentes internas y externas. Esto facilita la ejecución de análisis detallados y consultas especializadas que culminan en la obtención de información crítica para la toma de decisiones.

No obstante, es importante tener en cuenta que los Data Warehouses o*n-premise* pueden implicar una inversión notable en términos de hardware, software y mantenimiento. A diferencia de las soluciones basadas en la nube o *cloud*, su **escalabilidad** puede tener ciertas limitaciones y pueden requerir más esfuerzo para ajustarse a cambios en los requisitos de almacenamiento y procesamiento de datos. Sin embargo, hay organizaciones que optan por

mantener sus Data Warehouses *on-premise* debido a requisitos de cumplimiento, políticas de seguridad o preferencias estratégicas.

2.2.6.1. PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos relacional de código abierto, altamente escalable y de alto rendimiento. Es conocido por su robustez, extensibilidad y cumplimiento con los estándares ANSI SQL.

En el contexto de un **Data Warehouse**, PostgreSQL puede ser una excelente elección como sistema de gestión de bases de datos subyacente.

Estas son algunas de las razones por las cuales **PostgreSQL** puede ser una buena opción para la creación de un Data Warehouse:

- **Capacidad de manipulación de grandes volúmenes de datos:** PostgreSQL es conocido por su capacidad para manejar grandes conjuntos de datos, lo que es esencial para un Data Warehouse que almacena cantidades masivas de información.

- **Soporte de lenguaje SQL avanzado:** PostgreSQL ofrece una amplia gama de funcionalidades y extensiones de SQL, lo que facilita la manipulación y consulta de datos complejos.

- **Extensibilidad:** Puedes extender las funcionalidades de PostgreSQL mediante la creación de funciones y procedimientos almacenados, lo que es útil para personalizar la base de datos según tus necesidades específicas.

- **Índices y optimización de consultas:** PostgreSQL proporciona una variedad de índices y técnicas de optimización que pueden acelerar la velocidad de las consultas en un Data Warehouse.

- **Compatibilidad con lenguajes de programación:** PostgreSQL es compatible con varios lenguajes de programación, lo que facilita la integración con aplicaciones y herramientas externas.

- **Replicación y escalabilidad:** PostgreSQL ofrece capacidades de replicación y escalabilidad, lo que te permite distribuir la carga de trabajo en múltiples servidores si es necesario.

- **Seguridad avanzada:** PostgreSQL ofrece una amplia gama de características de seguridad, incluidas autenticación avanzada, control de acceso a nivel de fila y soporte para encriptación de datos.

En resumen, PostgreSQL es una opción sólida para la creación de un Data Warehouse debido a su capacidad para manejar grandes cantidades de datos, su soporte de SQL avanzado y su extensibilidad.

2.2.6.2. Microsoft SQL Server

Microsoft SQL Server es un sistema de gestión de bases de datos relacional (DBMS) desarrollado por Microsoft. Está diseñado para administrar y almacenar grandes cantidades de datos de manera eficiente y segura. Es una parte fundamental en la creación y gestión de bases de datos en entornos empresariales y corporativos.

En el contexto de la creación de un Data Warehouse, Microsoft SQL Server desempeña un papel crucial. Un Data Warehouse es un sistema de almacenamiento y gestión de datos que se utiliza para consolidar y organizar información proveniente de múltiples fuentes dentro de una organización. Esta información se estructura de manera que sea óptima para la consulta y el análisis de grandes volúmenes de datos.

Microsoft SQL Server proporciona una amplia gama de herramientas y funcionalidades específicamente diseñadas para respaldar la creación y administración de Data Warehouses:

- **Capacidad para manejar grandes volúmenes de datos:** SQL Server está diseñado para trabajar eficientemente con grandes conjuntos de datos, lo que lo convierte en una elección natural para un Data Warehouse.

- **Integración con herramientas de extracción, transformación y carga (ETL):** SQL Server Integration Services (SSIS) es una herramienta que permite la extracción, transformación y carga de datos desde diversas fuentes hacia el Data Warehouse.

- **Herramientas de análisis y consulta:** SQL Server proporciona un potente lenguaje de consulta llamado *Transact-SQL* (T-SQL) que permite realizar consultas complejas y operaciones de análisis en los datos almacenados.

- **Funcionalidades de optimización de consultas:** SQL Server incluye optimizadores de consultas que ayudan a mejorar el rendimiento de las consultas, lo que es esencial cuando se trabaja con grandes conjuntos de datos.

- **Herramientas de administración y monitoreo:** Ofrece un conjunto de herramientas para administrar y monitorear la base de datos y el rendimiento del servidor, lo que es crucial para mantener la eficiencia y la disponibilidad del Data Warehouse.

- **Capacidad de escalabilidad:** SQL Server puede escalarse tanto vertical como horizontalmente para adaptarse a las necesidades cambiantes de almacenamiento y procesamiento de datos.

- **Seguridad avanzada:** Proporciona mecanismos robustos de seguridad para proteger los datos almacenados en el Data Warehouse contra accesos no autorizados.

- **Integración con otras herramientas y tecnologías de Microsoft:** SQL Server se integra a la perfección con otras soluciones de Microsoft, como **Microsoft Azure**, **Power BI** y Excel, lo que facilita la creación de un ecosistema completo de análisis y visualización de datos.

Tal y como hemos visto, Microsoft SQL Server es una plataforma completa y versátil que proporciona las herramientas y funcionalidades necesarias para diseñar, implementar y administrar un Data Warehouse de manera eficiente y efectiva. Su integración con otras tecnologías de Microsoft y su capacidad para manejar grandes volúmenes de datos lo convierten en una elección popular para proyectos de análisis de datos a gran escala.

2.2.6.3. MySQL

MySQL, aunque es una base de datos popular para soluciones transaccionales (OLTP), no es la opción más común para la construcción de un Data Warehouse a gran escala debido a sus limitaciones en cuanto a escalabilidad y capacidad de procesamiento de datos complejos.

Consideraciones Importantes:

- **Escalabilidad:** MySQL puede no ser la mejor opción si se prevé un crecimiento masivo de datos. En tales casos, es posible considerar otras bases de datos más especializadas para la creación de un Data Warehouse tales como, PostgreSQL, SQL Server, Azure Synapse, Amazon Redshift, Snowflake, Google BigQuery.

- **Particionamiento y Índices:** Dado que MySQL no es especialmente eficiente para el procesamiento de grandes cantidades de datos, es importante definir particionamientos y crear índices adecuados para mejorar el rendimiento de las consultas.

2.2.6.4. Comparativa entre PostgreSQL y Microsoft SQL Server

A continuación, analizaremos algunas de las ventajas y desventajas de implantar un Data Warehouse on-premise, con PostgreSQL y Microsoft SQL Server.

	SQL Server	PostgreSQL
Fabricante	Microsoft	PostgreSQL Global Development Group
Disponibilidad	SQL Server incluye varias herramientas de alta disponibilidad, como el envío de registros, los clústeres de conmutación por error y la replicación. Los grupos de disponibilidad de SQL Server, que funcionan las veinticuatro horas del día, proporcionan una conmutación por error automática cuando se cumplen determinadas condiciones. Sin embargo,	PostgreSQL ofrece un amplio abanico de soluciones para garantizar una alta disponibilidad a los usuarios, como el envío de registros con escritura anticipada, la conmutación por error de disco compartido, la partición de datos y varios métodos de replicación. Herramientas como EDB Postgres Failover Manager proporcionan una conmutación por error

	sólo se puede acceder a esta oferta en la edición empresarial de SQL Server.	automática para garantizar la alta disponibilidad mediante la supervisión e identificación de los fallos de la base de datos.
Estructura de datos y tablas	SQL Server utiliza **T-SQL**, que se parece al SQL estándar. T-SQL incluye soporte adicional para el procesamiento de datos y cadenas, la programación procedimental y las variables locales.	Ofrece a sus usuarios el lenguaje de programación procedimental **PL/pgSQL** para que sea más sencillo. Las funcionalidades adicionales al SQL estándar en PostgreSQL incluyen tipos definidos por el usuario, módulos personalizados, extensiones, soporte de JSON y opciones adicionales para triggers y otras funcionalidades.
	SQL Server utiliza una reserva de memoria intermedia que puede limitarse o aumentarse en función de las necesidades de procesamiento	PostgreSQL aísla los procesos tratándolos como procesos separados del sistema operativo
Desfragmentación	SQL Server proporciona un recolector de basura eficiente que no genera más de un 15-20% de sobrecarga. Técnicamente, los desarrolladores también pueden ejecutar los recolectores de basura de forma continua, porque es así de eficaz. En resumen, SQL Server ofrece más métodos de desfragmentación que PostgreSQL.	PostgreSQL escanea las tablas de una capa de datos para buscar filas vacías y se deshace de los elementos innecesarios. Al hacerlo, el sistema libera el espacio del disco. Sin embargo, este método necesita mucha CPU y puede afectar al rendimiento de la aplicación.
Índices	SQL Server proporciona una rica funcionalidad	PostgreSQL admite la organización de tablas

	automatizada para la gestión de índices. Pueden organizarse en clústeres y mantener el orden adecuado de las filas sin intervención manual. SQL Server también admite índices parciales y búsquedas de índices múltiples	basada en índices, pero las primeras versiones no utilizaban actualizaciones automáticas de los índices. También te permite consultar muchos índices en una sola búsqueda, lo que significa que puedes descubrir mucha información.
Escalabilidad	SQL Server también puede utilizar núcleos, sin embargo, la versión estándar está limitada a veinticuatro núcleos de CPU. La versión empresarial permite el uso de núcleos de CPU ilimitados. SQL Server también posee una función de hiper escala, en la que puedes determinar los límites inferior y superior que te permiten escalar hacia abajo y hacia arriba según las necesidades.	PostgreSQL ofrece muchas características en términos de escalabilidad y puede emplear varios núcleos de CPU para implementar paralelamente las consultas con rapidez.
Seguridad	En MS SQL Server, hay dos características de mejora de la seguridad a nivel de servidor: El modo de autenticación de Windows y el modo mixto, que incluye un proceso de autenticación tanto de Windows Server como de MS SQL Server. El modelo de seguridad de MS SQL Server es una estrecha integración entre el modo de	En el lado del servidor, PostgreSQL ofrece métodos avanzados de autenticación, como el protocolo ligero de acceso a directorios (LDAP) y el módulo de autenticación enchufable (PAM), que reducen potencialmente la superficie de ataque de los servidores de bases de datos PostgreSQL. Otras mejoras de seguridad a nivel de

	autenticación de Windows Server y la base de datos.	servidor en PostgreSQL incluyen la dirección de escucha del servidor PostgreSQL, la autenticación basada en el host y la autenticación de certificados.
Almacenamiento	SQL Server 2016 y superiores pueden utilizar un máximo de cien ordenadores o máquinas virtuales con un máximo de cinco instancias ejecutadas por ordenador. Sin embargo, esto también depende de su edición. La edición empresarial permite el máximo ancho de banda, por lo que se concederán más capacidades de almacenamiento. Mientras que la última edición estándar permite el uso de hasta 128 GB de memoria, en la versión empresarial puedes utilizar memoria ilimitada.	PostgreSQL ofrece tipos de datos más complejos y permite la herencia de objetos, aunque también hace que trabajar con PostgreSQL sea más complejo. Tiene un único motor de almacenamiento compatible con ACID e inicializa un nuevo proceso del sistema con su asignación de memoria para cada conexión de cliente. Por tanto, con el aumento de conexiones de clientes en los sistemas, es necesario asignar más memoria.
Precios	MS SQL Server se publicó bajo una licencia comercial como parte de los productos de Microsoft. A principios de 2016, la base de datos se puso a disposición de los desarrolladores como herramienta gratuita, pero sólo admite un procesador y 1 GB de memoria máxima. Aunque es gratuita, carece de varias funciones que puedes necesitar para una empresa. Es posible que tengas que pagar 899	PostgreSQL se publicó bajo la licencia PostgreSQL, una licencia liberal de código abierto. El Grupo de Desarrollo Global de PostgreSQL mantiene su compromiso de hacer que PostgreSQL esté disponible como software libre y de código abierto para siempre. No hay planes para alterar o liberar PostgreSQL bajo una licencia diferente.

	dólares por servidor si necesitas más servidores. Últimamente, la edición empresarial de SQL Server cuesta 13.748 dólares	

2.2.6.5. Diferencias entre PostgreSQL y MySQL

PostgreSQL y MySQL son dos sistemas de gestión de bases de datos relacionales (RDBMS) más populares y utilizados en todo el mundo. A continuación, se presentan algunas de las diferencias más importantes entre ambas:

- **Licencia**. PostgreSQL es una base de datos de código abierto con licencia MIT, mientras que MySQL es de propiedad de Oracle Corporation y cuenta con una combinación de licencias comerciales y de código abierto.

- **Extensibilidad.** PostgreSQL ofrece una gran cantidad de extensiones y opciones de personalización, lo que le permite adaptarse a una amplia gama de casos de uso. MySQL, por otro lado, está más orientado a aplicaciones web y es menos personalizable.

- **Lenguaje de programación**. PostgreSQL ofrece soporte completo para PL/pgSQL, su propio lenguaje de programación almacenado, mientras que MySQL utiliza principalmente SQL y proporciona soporte limitado para otros lenguajes de programación almacenados.

- **Funcionalidades avanzadas.** PostgreSQL tiene una serie de características avanzadas que no se encuentran en MySQL, como soporte para arrays, tipos de datos personalizados, transacciones anidadas, subconsultas en DELETE y UPDATE, y soporte para procedimientos almacenados y funciones. MySQL carece de algunas de estas características avanzadas.

- **Escalabilidad.** En general, PostgreSQL se considera más escalable que MySQL, especialmente en aplicaciones que requieren una alta concurrencia.

- **Rendimiento** MySQL es conocido por su alto rendimiento en operaciones simples y de alta velocidad, mientras que PostgreSQL se destaca por su capacidad para manejar grandes cantidades de datos complejos y realizar consultas complejas.

En resumen, **PostgreSQL es una base de datos más avanzada y personalizable**, mientras que MySQL es más simple y se adapta mejor a las aplicaciones web. Sin embargo, ambas bases de datos son excelentes opciones y la elección depende de las necesidades específicas de cada proyecto.

2.3. Data Warehouse en la nube (Cloud)

Un Data Warehouse en la nube es un entorno de almacenamiento y procesamiento de datos escalable, altamente disponible y distribuido a través de la infraestructura de una plataforma de servicios en la nube (PaaS). Su objetivo principal es permitir el almacenamiento y análisis eficiente de grandes volúmenes de datos de diversas fuentes y facilitar el análisis y la generación de cuadros de mando y/o informes para la toma estratégica de decisiones.

2.3.1. Principales características de un Data Warehouse en la nube

- **Escalabilidad:** Los almacenes de datos en la nube permiten aumentar o reducir la capacidad de almacenamiento y procesamiento según las necesidades del negocio. Esto es clave para manejar grandes cantidades de datos de manera efectiva.
- **Integración de Datos:** Facilita la integración de datos desde una variedad de fuentes, incluidas bases de datos relacionales, sistemas en la nube, aplicaciones de terceros, etc.
- **Alta Disponibilidad:** La infraestructura en la nube ofrece redundancia y disponibilidad constante, lo que garantiza que los datos estén accesibles cuando se necesiten.
- **Procesamiento Paralelo:** Pueden procesar consultas y análisis en paralelo, lo que permite tiempos de respuesta más rápidos, incluso con grandes volúmenes de datos.
- **Herramientas de Análisis y Visualización:** A menudo, estos almacenes de datos en la nube están integrados con herramientas de análisis y visualización que facilitan la creación de informes y la toma de decisiones basadas en datos.

2.3.2. Ventajas de un Data Warehouse en la nube

- **Escalabilidad Bajo Demanda:** Los recursos se pueden ajustar según sea necesario, evitando la sobrecarga de infraestructura no utilizada.
- **Reducción de Costos:** Al aprovechar la infraestructura de la nube, las organizaciones pueden evitar los costos de adquisición y mantenimiento de hardware y software.
- **Agilidad:** Permite a las organizaciones adaptarse rápidamente a las cambiantes demandas y necesidades de análisis de datos.

- **Seguridad y Cumplimiento:** Los proveedores de servicios en la nube suelen ofrecer medidas de seguridad avanzadas y cumplimiento de regulaciones para proteger los datos.
- **Acceso Ubicuo:** Los datos están disponibles desde cualquier lugar con acceso a Internet, lo que facilita el trabajo remoto y la colaboración global.

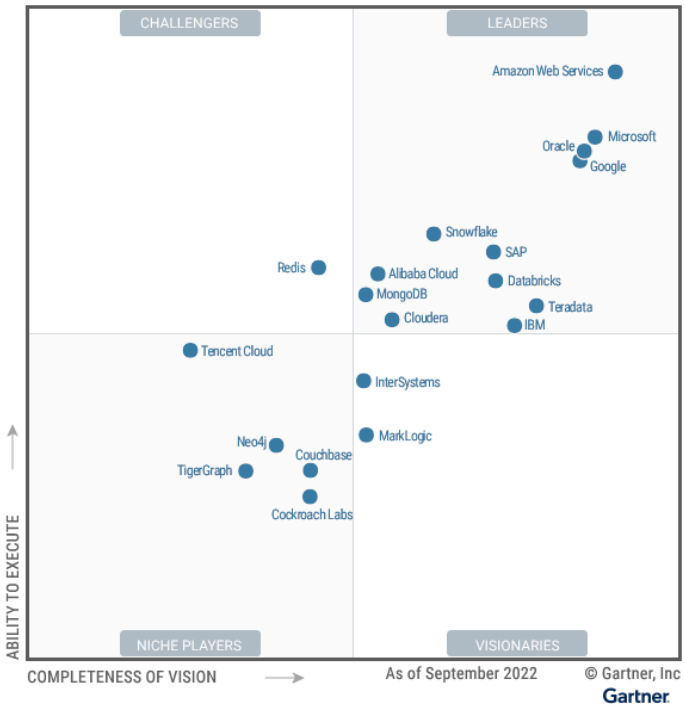
2.3.3. Desventajas o desafíos de un Data Warehouse en la nube

- **Dependencia del Proveedor:** Las organizaciones deben confiar en el proveedor de servicios en la nube para la disponibilidad y seguridad de los datos.
- **Costos de Almacenamiento y Procesamiento:** Aunque puede haber ahorros en hardware, los costos de suscripción a los servicios en la nube pueden acumularse.
- **Latencia:** La latencia de red puede ser un factor en el rendimiento, especialmente cuando se accede a grandes conjuntos de datos.

2.3.4. Análisis tecnológico Data Warehouse Cloud

A continuación, se detallarán las diferentes tecnologías en la nube o *cloud* que existen en el mercado para la implementación de un Data Warehouse.

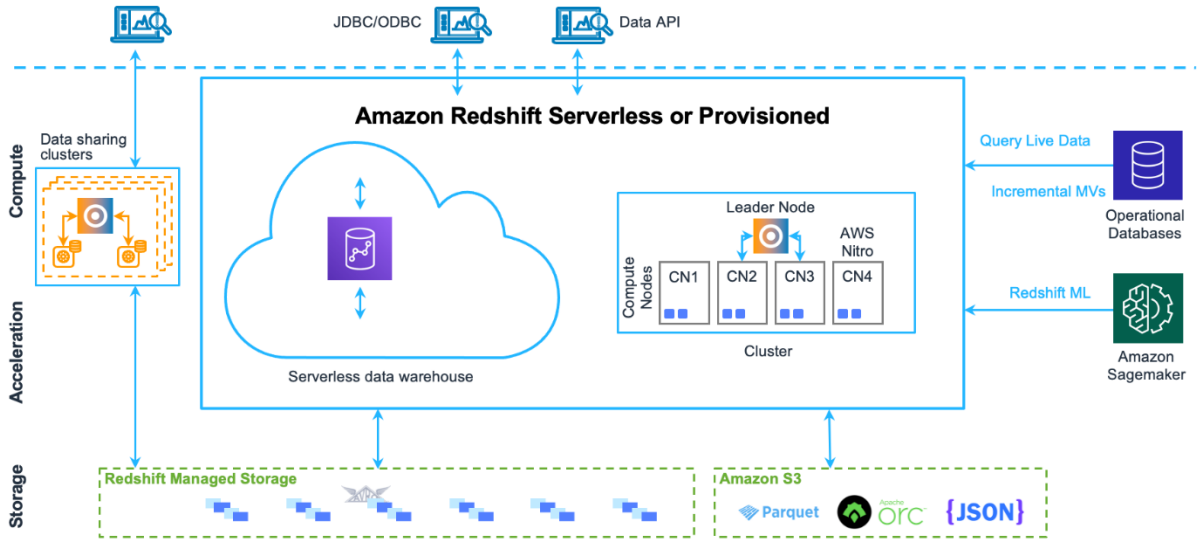
Tal y como se puede observar en la siguiente figura, Gartner posiciona a Amazon Web Services (AWS) como líder en soluciones Cloud, en segundo lugar, a Microsoft Azure y por último a Google Cloud Platform (GCP).



2.3.4.1. Amazon Redshift

Es el servicio de Data Warehouse en la nube de AWS. Es una BBDD MPP (Procesamiento Paralelo masivo), que analiza datos usando múltiples nodos en paralelo, y cada uno de estos ejecuta un subconjunto de una consulta. Además, proporciona funcionalidades necesarias para implementar consultas de procesos de BI.

En la siguiente figura, se presentan los elementos de la arquitectura de Amazon Redshift:



2.3.4.1.1. Ventajas de Amazon Redshift:

- Es uno de los Cloud Data Warehouse más populares.
- Integra otros servicios de AWS como Amazon S3 o a AWS Glue, incluso con otras herramientas de terceros.
- Configuración y gestión de actividades, copias de seguridad y actualizaciones son automáticas. Otras tareas también se pueden automatizar con scripts
- Coste flexible, es decir, pago por uso.
- Permite definir colas para diferentes tipos de procesos mediante Redshift Workload Management (WLM). Funcionalidad muy útil para separar procesos ETL y consultas analíticas de reporting.
- Es una BBDD MPP, lo que la convierte en una de las soluciones más rápidas del mercado.
- Fácil de implementar: Es una de las soluciones más fáciles de implementar que existen hoy en día.
- Ofrece una copia de seguridad coherente con datos y archivos. También los recupera cuando ocurren fallos o daños.

2.3.4.1.2. Limitaciones de Amazon Redshift:

- Solo recibe soporte para la carga paralela con datos específicos. Si los datos no se corresponden con lo especificado se deben insertar mediante scripts o inserciones o usar una solución ETL que cargue datos desde una fuente diferente.
- Tiene su funcionamiento propio, es decir que no funciona como una BBDD de aplicaciones en vivo corriente, y se debe comprender como funcionan las calves de distribución y clasificación que determinan como se indexan y almacenan datos en Redshift
- El rendimiento disminuye a medida que aumentan los grupos, aunque en la mayoría de los casos la potencia y rendimiento son más que suficientes.
- No es una solución multi-Cloud
- No está 100% administrado: Aunque las herramientas que proporciona reducen la necesidad de tener un administrador de la BBDD, no elimina la necesidad de tener uno, pues, por ejemplo, tiene problemas con el eficiente manejo del almacenamiento en entornos propensos a las eliminaciones frecuentes.
- Tiene problemas con ejecución concurrente: en un entorno en el que se ejecutan varias consultas podrían haber errores.

2.3.4.1.3. Precios de Amazon Redshift

Precios del almacenamiento administrado de Amazon Redshift

Se utiliza la tarifa mensual por GB fija de su región para cobrar los datos almacenados en almacenamiento administrado. El almacenamiento administrado viene incluido exclusivamente en los tipos de nodos RA3 y paga la misma tarifa baja para el almacenamiento administrado de Redshift independientemente del tamaño de los datos. El uso del almacenamiento administrado se calcula por hora en función de los datos totales presentes en el almacenamiento administrado (ver ejemplo a continuación acerca de la conversión del uso en GB-hora a cargos en GB-mes). Puede monitorear el volumen de datos en el clúster de RA3 mediante Amazon CloudWatch o la consola de administración de AWS. No deberá pagar cargos por transferencia de datos entre nodos RA3 y el almacenamiento administrado. Los cargos de almacenamiento administrado no incluyen cargos de almacenamiento de copias de seguridad debido a la generación de instantáneas manuales o automatizadas (consulte Almacenamiento de copias de seguridad). Se continúa cobrando la retención de copias de seguridad manuales aunque se apague el clúster.

Región

EU (Stockholm) ▼

  Almacenamiento administrado de Redshift	Precio
Almacenamiento/mes	0,024 USD por GB

2.3.4.1.4. Ejemplo de precios para almacenamiento administrado

Supongamos que almacena 100 GB de datos en almacenamiento administrado, con tipos de nodos RA3, durante los primeros 15 días del mes de abril y 100 TB de datos durante los últimos 15 días de dicho mes.

Primero, calculemos el uso en GB por hora. Durante los primeros 15 días, tendrá el siguiente uso: 100 GB x 15 días x (24 horas/día) = 36 000 GB-hora. Durante los últimos 15 días, tendrá el siguiente uso: 100 TB x 1024 GB/TB X 15 días X (24 horas/día) = 36 864 000 GB-hora.

A finales de abril, el uso en GB por hora total es: 36 000 GB-hora + 36 864 000 GB-hora = 36 900 000 GB-hora.
Convirtamos esto a GB por mes: 36 900 000 GB-hora/720 horas al mes en abril = 51 250 GB-mes.

Si los datos se almacenaron en la región Este de EE. UU. (Norte de Virginia), el almacenamiento administrado se cobrará 0,024 USD/GB-mes. Los cargos de almacenamiento mensual por 51 250 GB-mes serán: 51 250 GB-mes x 0,024 USD por GB-mes = 1230 USD.

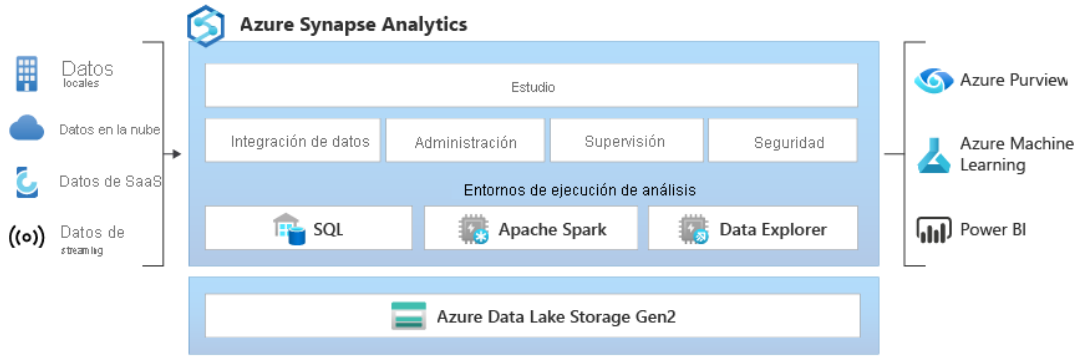
Tarifa total de RMS para el mes de abril = 1230 USD

Para más información, consultar el siguiente enlace:
https://aws.amazon.com/es/redshift/pricing/

2.3.4.2. Microsoft Azure Synapse Analytics

Anteriormente conocido como Microsoft Azure SQL Data Warehouse, Azure Synapse es un servicio de análisis empresarial que optimiza el proceso de obtención de información de sistemas de almacenamiento de datos y macrodatos. Combina lo mejor de las tecnologías SQL empleadas en el almacenamiento de datos empresariales, las tecnologías de Spark utilizadas para macrodatos, Data Explorer para análisis de series temporales y de registro, Pipelines para la integración de datos y ETL/ELT, además de ofrecer una integración completa con otros servicios de Azure como Power BI, CosmosDB y AzureML.

En la siguiente figura se muestran los componentes de la arquitectura de Azure Synapse:



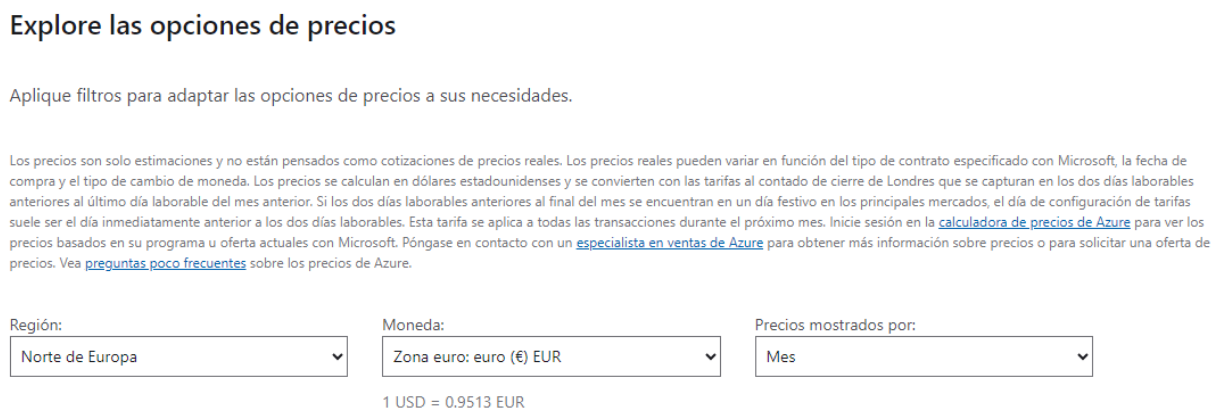
2.3.4.2.1. Ventajas de Azure Synapse Analytics:

- **Escalado ilimitado**. En comparación con Big Query por ejemplo podría ejecutar la misma consulta en un 75% menos de tiempo con un petabyte de datos.
- **Administración de datos centralizada.** Azure usa una tecnología MPP, que permite procesar y gestionar grandes cargas de trabajo y manejar de manera eficiente grandes volúmenes de datos. Lo cual acelera los análisis y los informes, reduciendo esfuerzos al recopilar, cotejar y crear informes, un despliegue del DWH mucho más rápido, que con escalabilidad y flexibilidad se evita tiempos de inactividad con las variaciones de carga de trabajo.
- **Aislamiento de cargas de trabajo:** permite a los usuarios administrar a los usuarios las cargas de trabajo heterogéneas.
- **Integración de ML:** Esto puede ayudar a predecir y puntuar los modelos de ML para generar predicciones dentro del propio almacén de datos.
- **Ofrece un espacio centralizado** (Azure Synapse Studio) para preparar datos, administrarlos, almacenarlos, Big Data y tareas de IA
- **Mejor BI y visualización de datos** gracias a que tiene una integración perfecta y nativa con Power BI.
- **Mayor productividad de TI**, permitiendo al personal automatizar el aprovisionamiento de infraestructuras y tareas administrativas.

2.3.4.2.2. Limitaciones de Azure Synapse Analytics:

- **Costes.** El uso de servicios de nube, incluido Azure Synapse Analytics, puede resultar costoso dependiendo del volumen y la frecuencia de uso. Es importante comprender bien la estructura de precios y planificar adecuadamente para evitar sorpresas en la factura.
- **Nivel de Complejidad Inicial.** Para optimizar plenamente las capacidades de Azure, se requiere un nivel avanzado de conocimientos técnicos y experiencia en su configuración y gestión. Esto puede presentar un desafío inicial para aquellas empresas que carecen de personal especializado en este ámbito.
- "Synapse" agrupa varias tecnologías, con Azure DW como core, requiriendo la gestión de los diferentes servicios.
- Tiene dificultad al gestionar grandes volúmenes de queries concurrentes.
- Requiere tareas complejas de administración de BBDD, incluido ajuste de rendimiento, que otras soluciones Cloud no requieren de administrar.

2.3.4.2.3. Precios de Azure Synapse Analytics



Planes de compra previa | Integración de datos | **Almacenamiento de datos** | Análisis de macrodatos | Análisis de registros y telemetría | Grupo de SQL dedicad

Realice análisis interactivos, por lotes, de streaming y predictivos con una experiencia de T-SQL muy completa. Los usuarios tienen la flexibilidad de elegir entre recursos sin servidor y dedicados. Esto permite a los clientes usar el lenguaje SQL que ya conocen para la exploración de datos en su lago de datos y ejecutar cargas de trabajo exigentes y predecibles, como el almacenamiento de datos, desde el mismo servicio.

A continuación se indican los precios de los modelos de consumo dedicado y sin servidor.

Sin servidor

Esta funcionalidad permite usar T-SQL para consultar los datos de su lago de datos en Azure en lugar de aprovisionar recursos de antemano. Solo paga por las consultas ejecutadas y los precios se basan en la cantidad de datos procesados por cada consulta. Las consultas de metadatos únicamente (instrucciones DDL) no tienen ningún costo. Las consultas conllevan un cargo mínimo de 10 MB y cada consulta se redondea al siguiente MB de datos procesados.

	Precio	Cantidad gratuita
Sin servidor	**€4,757** por TB de datos procesados	1 TB de consultas gratuitas al mes hasta el 30 de noviembre de 2023*

* Las cantidades gratuitas se aplican solo a los tipos de suscripción siguientes: Pago por uso, Microsoft Azure Enterprise, plan de Microsoft Azure, Azure en CSP, Desarrollo/pruebas - Enterprise. Estas cantidades gratuitas incluidas se agregan en el nivel de inscripción para los contratos Enterprise y en el nivel de suscripción para las suscripciones de pago por uso.

Estos precios son específicos de la consulta de datos del lago de datos. Para ver los precios de almacenamiento, consulte la página de precios de Azure Data Lake Storage.

Para obtener más información, visite la página Administración de costos del grupo de SQL sin servidor en Azure Synapse Analytics en la documentación.

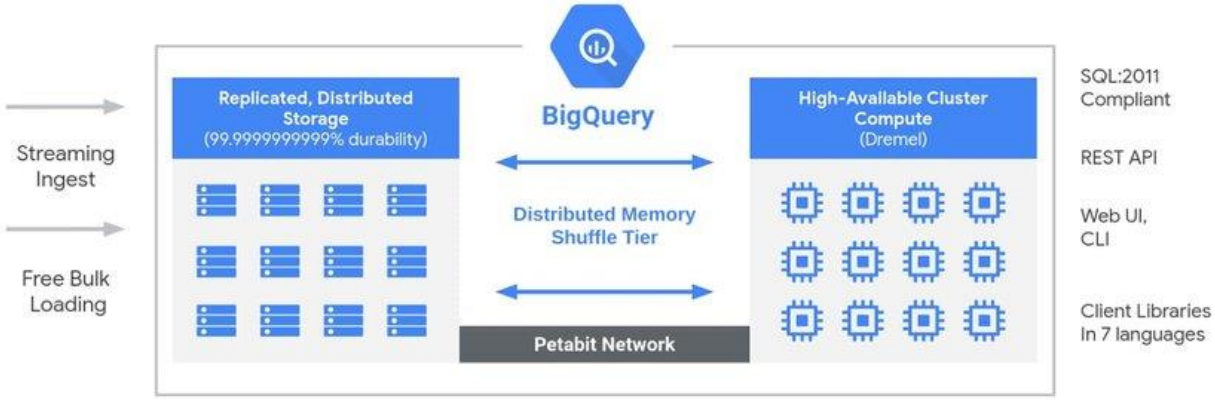
Para más información consultar:
https://azure.microsoft.com/es-es/pricing/details/synapse-analytics/

2.3.4.3. Google BigQuery

Google BigQuery es un almacén de datos empresarial completamente administrado que ofrece una amplia gama de capacidades para la gestión y análisis eficiente de datos. Entre sus funcionalidades destacan el aprendizaje automático, el análisis geoespacial y la inteligencia empresarial. Su arquitectura sin servidores elimina la necesidad de gestionar la infraestructura, permitiéndote abordar las consultas más complejas de tu organización utilizando lenguaje SQL. El motor de análisis distribuido y escalable de BigQuery facilita la consulta de terabytes en cuestión de segundos y de petabytes en minutos, proporcionando así una solución de alto rendimiento para el procesamiento de grandes volúmenes de datos.
El acceso a los datos es por **SQL** estándar, pero antes de las consultas se debe configurar permisos, roles y grupos que se pueden manejar con Identy and Access Management (IAM).

Para profesionales como **analistas de datos**, ingenieros de datos, administradores de almacenes de datos o científicos de datos, la documentación de BigQuery ML proporciona una guía esencial para descubrir, implementar y administrar herramientas de datos críticas en la toma de decisiones empresariales.

En la siguiente figura se muestran los componentes de la arquitectura de Google BigQuery:



2.3.4.3.1. Ventajas de BigQuery

- Ingenieros de datos no emplean tanto tiempo aprovisionando y administrando hardware ni mejorando el uso de recursos, pues estos se calculan dinámicamente. Así las consultas que solían tomar horas o más en las bases de datos tradicionales ahora se pueden procesar en cuestión de segundos.
- BigQuery puede integrar todo tipo de dato y es compatible con las herramientas de Google Cloud y una extensa cantidad de terceros.
- Permite consultar datos nuevos y actualizados de manera periódica (Vistas materializadas).

- BigQuery simplifica el proceso de construir tu propio centro de datos, pues se paga solo por lo que se usa con la capacidad de escalar de forma elástica y flexible.
- Sirve para hacer consultas sobre otras BBDD como MySQL y ficheros de datos como .CSV directamente sobre almacenamiento Cloud.
- BigQuery BI Engine: Es un servicio de analítica en memoria integrado que ofrece tiempos de consulta inferiores a 1 minuto
- BigQuery ofrece varias funciones de ML para generar modelos predictivos y encontrar oportunidades de mejora.
- Permite entrenar modelos de ML de *TensorFlow* directamente con los *datasets* almacenados.

2.3.4.3.2. Desventajas o limitaciones de BigQuery:

- Las consultas cuyo rendimiento no ha sido ajustado y aquellas que devuelven datos redundantes pueden volverse costosas muy rápidamente.
- Funciona mejor con tablas planas, lo cual puede dificultar la gestión de un modelo de datos empresarial.
- El soporte y la compatibilidad de herramientas fuera de GCP falta en comparación con otras plataformas.

2.3.4.3.3. Precios de Google BigQuery

La estructura de precios de BigQuery comprende dos componentes principales:

- **Coste de Análisis.** Este componente abarca el gasto asociado con la ejecución de consultas, incluyendo consultas SQL, funciones definidas por el usuario, secuencias de comandos, así como ciertas instrucciones de lenguajes de manipulación de datos (DML) y definición de datos (DDL) que implican el escaneo de tablas.
- **Coste de Almacenamiento.** Se refiere al gasto relacionado con la retención de los datos cargados en BigQuery.

Además de estos componentes principales, BigQuery aplica cargos por otras operaciones, como inserciones de transmisión y el uso de la API Storage de BigQuery. Se puede encontrar información adicional sobre precios para la ingestión y extracción de datos en las secciones correspondientes.
Es importante destacar que BigQuery ofrece operaciones gratuitas y un nivel de uso sin costos asociados.

Cada proyecto creado está asociado a una cuenta de facturación. Todos los cargos derivados de las operaciones realizadas en BigQuery dentro del proyecto se reflejarán en la cuenta de facturación correspondiente. Los costos de almacenamiento en BigQuery también se facturan

a través de esta cuenta de facturación asociada. Para obtener un detallado seguimiento de los costos y tendencias en BigQuery, se puede consultar la página de informes de facturación de Google Cloud en la consola de Cloud.

Para más información consultar el siguiente enlace:
https://Cloud.google.com/bigquery/pricing?hl=es#overview_of_pricing

2.3.4.4. Snowflake

Snowflake es una plataforma de Data Warehouse en la nube de tipo SaaS. Es compatible con los principales proveedores de servicios Cloud como AWS y Microsoft Azure. En esta plataforma podemos encontrar componentes que facilitan su uso para ingeniería y ciencia de datos, Data Lake y **Data Warehouse**.

2.3.4.4.1. Ventajas de uso

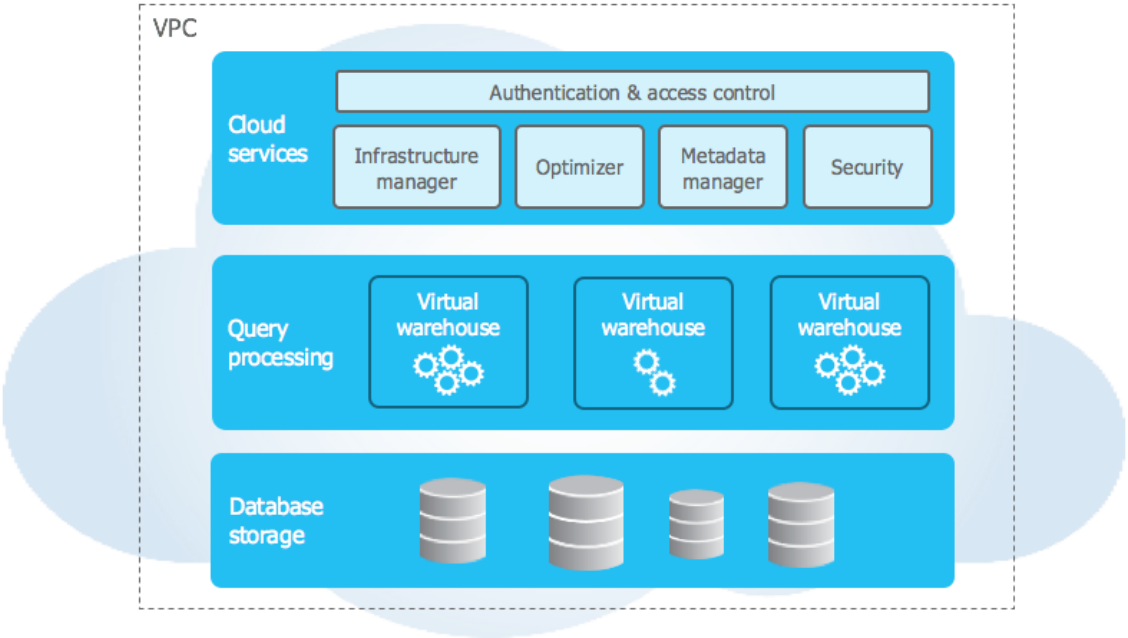
- **Capacidad.** Tiene una alta capacidad de almacenamiento lo que lo hace ideal para negocios que manejan una gran cantidad de datos.
- **Multi-Cloud.** Se puede alojar en diversas plataformas Cloud como AWS, Google Cloud, Microsoft Azure.
- **Capacidad del servidor.** puede escalarse en función de las necesidades de la compañía.
- **Personalización.** permite a los usuarios organizar los datos de la manera que deseen. Además, está diseñado de tal manera que rinde óptimamente por sí solo, por tanto, no hay necesidad de que un especialista esté siempre pendiente.
- **Recuperación tras un desastre.** tiene planes de contingencia para y asegura que muchos centros de datos replican y proveen acceso sencillo a los datos en caso de desastre.
- **Diagrama de estrella.** el esquema de Snowflake es una mejora de la metodología de diseño del diagrama en estrella.

2.3.4.4.2. Limitaciones

- Alto precio con respecto a sus competidores como Redshift, que es 1.3 veces más barato.
- Migrar datos a Snowflake puede ser realmente complicado.
- No hay limitación en los datos lo cual afecta al almacenamiento, al rendimiento y sobre todo al precio.
- Falta de soporte para datos no estructurados, solo tiene para datos estructurados o semiestructurados.

2.3.4.4.3. Arquitectura Snowflake

En la siguiente figura se muestran los componentes de la arquitectura de Snowflake:



2.3.4.4.4. Precios de Snowflake

Snowflake pone a disposición diversas ediciones de su servicio en la Nube de Datos. Si la necesidad son precios flexibles basados en el consumo sin ataduras a largo plazo, es recomendable inscribirse en *Snowflake On Demand*, una opción ágil y sencilla para acceder a Snowflake. Asimismo, al adquirir capacidad de Snowflake de forma anticipada, se puedes aprovechar descuentos en los precios basados en el consumo. El tiempo de cómputo se factura por segundos, con un mínimo de 60 segundos.

Para más información consultar el siguiente enlace:
https://www.snowflake.com/pricing/?lang=es

2.4. Herramientas Business Intelligence

Una herramienta de Business Intelligence (BI) es un software especializado diseñado para ayudar a las empresas a comprender y aprovechar al máximo sus datos. Su principal función es recopilar, organizar y analizar grandes cantidades de información proveniente de diversas fuentes, como bases de datos, hojas de cálculo, sistemas de gestión empresarial entre otras fuentes.

Las herramientas de BI transforman esos datos en visualizaciones claras y fáciles de entender, siendo estas: gráficos, tablas y cuadros de mando. Esto facilita a tomar decisiones informadas basadas en hechos concretos.

En la siguiente figura se muestran las herramientas BI líderes en el mercado, según Gartner:

Figure 1: Magic Quadrant for Analytics and Business Intelligence Platforms

Source: Gartner

2.4.1. Microsoft Power BI

Power BI es una plataforma unificada y escalable de inteligencia empresarial o de negocios (BI) con funciones de autoservicio apta para grandes empresas. Conéctese a los datos, visualícelos e incorpore sin problemas objetos visuales en las aplicaciones que usa todos los días.

Incluye un conjunto de componentes:

- Power Query (para la integración y transformación de datos)
- Power BI Desktop (una herramienta de desarrollo complementaria)
- Power BI Mobile (disponible para teléfonos Android, iOS y Windows)
- Power Pivot (para el modelado de datos tabulares en memoria)
- Power View (para visualizar las representaciones gráficas de datos)
- Power Map (para visualizar datos geoespaciales en 3D)
- Preguntas y Respuestas sobre el poder (para interactuar en lenguaje natural)

2.4.1.1. Arquitectura Power BI

En la siguiente figura se muestra la arquitectura de Power BI:

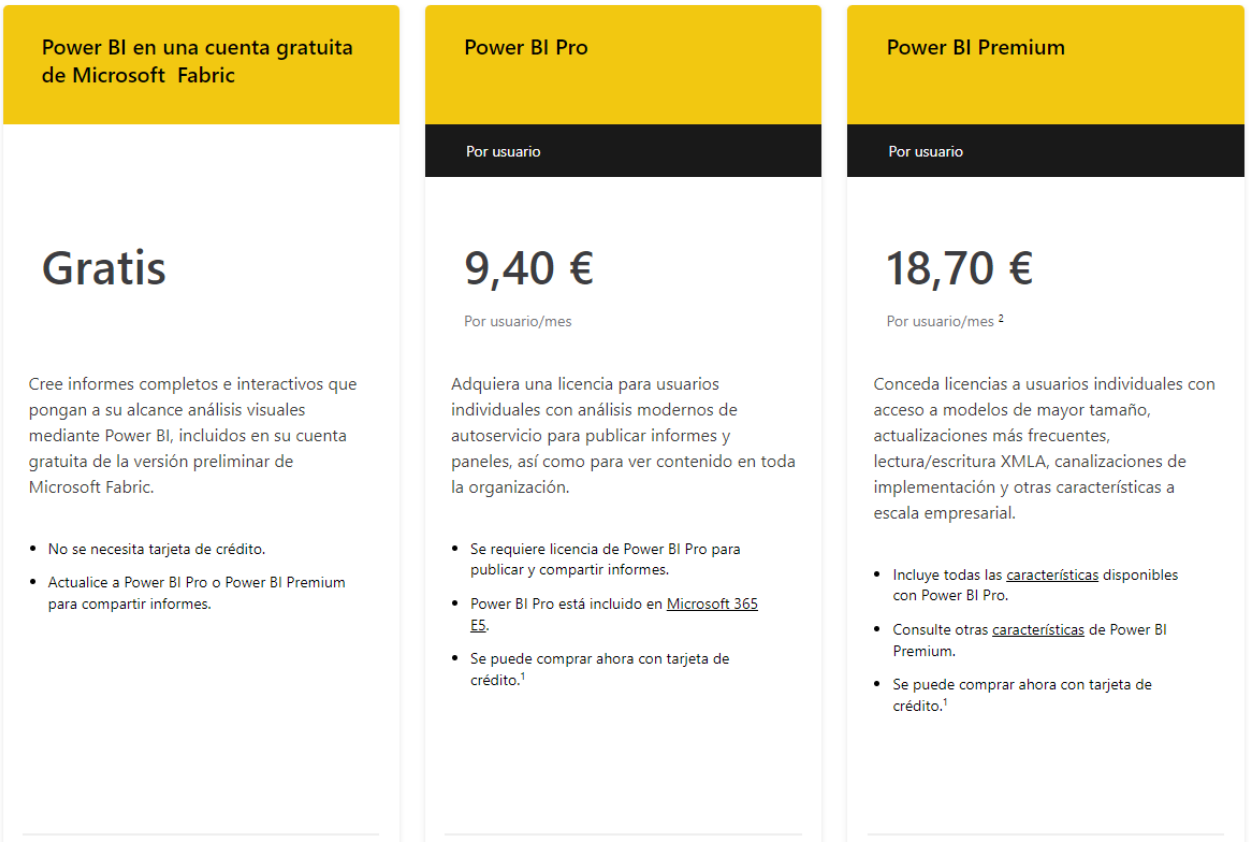
Fuentes de Datos. Un elemento esencial en Power BI es su amplio abanico de fuentes de datos. Pueden importarse datos desde archivos locales, fuentes en línea basadas en la nube o establecer conexiones directas en tiempo real. Si se importa desde fuentes locales o servicios en línea, hay un **límite de 1 GB**. Entre las fuentes de datos comúnmente utilizadas en Power BI se incluyen: Excel, Texto/CSV, XML, JSON, Bases de Datos Oracle, Bases de Datos IBM DB2, Bases de Datos MySQL, Bases de Datos PostgreSQL, Bases de Datos Sybase, Bases de Datos Teradata, Bases de Datos SAP HANA, Servidores SAP Business Warehouse, Amazon Redshift, Impala, Google BigQuery (Beta), Bases de Datos Azure SQL, Informes Salesforce, Google Analytics, Facebook, GitHub, y muchas otras.

Power BI Desktop. Power BI Desktop es una herramienta del lado del cliente reconocida como una herramienta adicional para el desarrollo y creación de informes. Este software de escritorio está repleto de herramientas y funcionalidades para conectarse a fuentes de datos, transformar datos, modelar y crear informes. Puede descargarse e instalarse Power BI Desktop en su sistema de forma gratuita. Con las características de Power BI Desktop, se puede llevar a cabo la limpieza de datos, crear métricas de negocio y modelos de datos, definir relaciones entre los datos, establecer jerarquías, crear visualizaciones y publicar informes.

Servicio Power BI. Power BI Service es una plataforma basada en web en la que se pueden compartir informes creados en Power BI Desktop, colaborar con otros usuarios y construir paneles de control.

2.4.1.2. Precios de Power BI

A continuación, se muestran los precios para los diferentes productor de Power BI:



Power BI en una cuenta gratuita de Microsoft Fabric

Gratis

Cree informes completos e interactivos que pongan a su alcance análisis visuales mediante Power BI, incluidos en su cuenta gratuita de la versión preliminar de Microsoft Fabric.

- No se necesita tarjeta de crédito.
- Actualice a Power BI Pro o Power BI Premium para compartir informes.

Power BI Pro

Por usuario

9,40 €

Por usuario/mes

Adquiera una licencia para usuarios individuales con análisis modernos de autoservicio para publicar informes y paneles, así como para ver contenido en toda la organización.

- Se requiere licencia de Power BI Pro para publicar y compartir informes.
- Power BI Pro está incluido en Microsoft 365 E5.
- Se puede comprar ahora con tarjeta de crédito.[1]

Power BI Premium

Por usuario

18,70 €

Por usuario/mes [2]

Conceda licencias a usuarios individuales con acceso a modelos de mayor tamaño, actualizaciones más frecuentes, lectura/escritura XMLA, canalizaciones de implementación y otras características a escala empresarial.

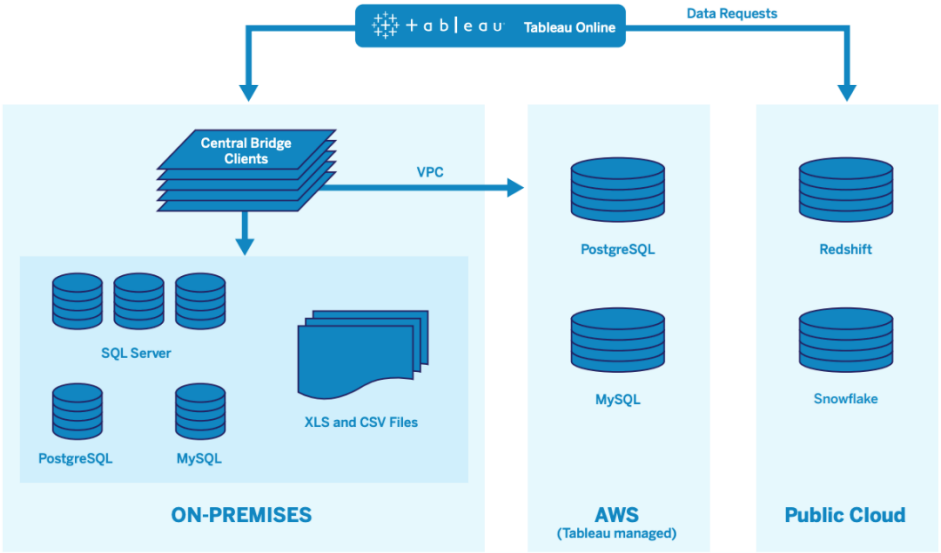
- Incluye todas las características disponibles con Power BI Pro.
- Consulte otras características de Power BI Premium.
- Se puede comprar ahora con tarjeta de crédito.[1]

Para más información consultar el siguiente enlace:
https://powerbi.microsoft.com/es-es/pricing/

2.4.2. Tableau

Tableau es una plataforma de análisis visual que transforma la manera en que se utilizan los datos para resolver problemas. Además, posibilita a las personas y a las organizaciones aprovechar al máximo los datos.

2.4.2.1. Arquitectura de Tableau

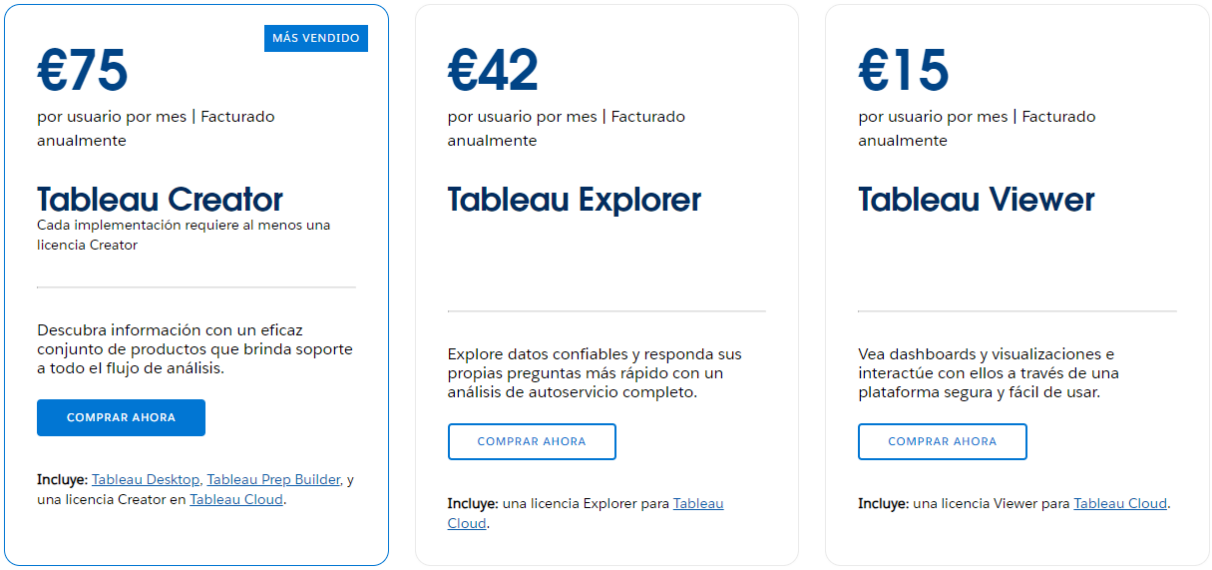


- **Conector de Datos.** Tableau incluye una amplia gama de conectores de datos que permiten a los usuarios conectarse a diversas fuentes de datos, como bases de datos, hojas de cálculo, servicios en la nube y más. Algunos de los conectores populares incluyen Microsoft Excel, MySQL, Salesforce, Google Sheets, entre otros.
- **Tableau Desktop**. Es la herramienta de creación de informes y visualización de datos. Los usuarios utilizan Tableau Desktop para conectarse a diversas fuentes de datos, crear visualizaciones y cuadros de mando interactivos.
- **Tableau Server.** Es una plataforma basada en servidor que proporciona acceso a las visualizaciones y cuadros de mando creados en Tableau Desktop. Permite compartir y colaborar en informes dentro de una organización. También proporciona características de administración, como el control de acceso y la programación de actualizaciones de datos.
- **Tableau Online.** Es una versión hospedada de Tableau Server que se ejecuta en la nube. Ofrece las mismas capacidades que Tableau Server, pero no requiere instalación o mantenimiento locales.

- **Tableau Prep.** Esta herramienta permite a los usuarios preparar y limpiar sus datos antes de cargarlos en Tableau para su análisis y visualización. Facilita la unión, transformación y limpieza de datos.
- **Tableau Bridge.** Esta es una herramienta que facilita la actualización de datos en tiempo real desde fuentes locales a Tableau Online o Tableau Server. Es útil para organizaciones que tienen datos en entornos locales y desean aprovechar las capacidades en la nube de Tableau.
- **Tableau Mobile.** Es una aplicación que permite a los usuarios ver y explorar visualizaciones en dispositivos móviles, como teléfonos inteligentes y tabletas.
- **Tableau Reader.** Permite a los usuarios visualizar y compartir visualizaciones de Tableau de manera gratuita. Sin embargo, no permite la creación o modificación de informes.
- **Tableau Public.** Es una plataforma gratuita de Tableau donde los usuarios pueden cargar visualizaciones y compartir públicamente. Las visualizaciones en Tableau Public son accesibles por **cualquier persona en Internet**.
- **API de Tableau.** Tableau proporciona una serie de interfaces de programación de aplicaciones (API) que permiten a los desarrolladores integrar Tableau con otras aplicaciones y personalizar su funcionamiento.

2.4.2.2. Precios de Tableau

A continuación, se muestran los precios par



Para más detalle consultar el siguiente enlace:
https://www.tableau.com/es-es/pricing/teams-orgs

2.4.2.3. Comparativa de Power BI vs Tableau

Características	Power BI	Tableau
Manejo de fuentes de datos	Admite conexiones con un amplio número de fuentes de datos	Gran cantidad de opciones de ingesta de datos.
Coste	A partir de 9,40€ al mes por usuario	A partir de 15€ al mes por usuario
Velocidad	Rápido, pero sólo cuando maneja pequeñas cantidades de datos.	Tableau es más rápido y escalable, con capacidad para manejar rápidamente grandes volúmenes de datos.
Soporte al cliente	Soporte limitado	Mejor soporte
Soporte de lenguajes	Admite visualizaciones basadas en el lenguaje R y Python	Soporta lenguaje R y Python
Desarrollo de informes o cuadros de mando	Sencillez y facilidad de uso, Interfaz similar a Microsoft Office	Sencillez y facilidad de uso, Drag & Drop
ETL	Power Query.	Tableau Prep
Lenguaje	DAX.	MDX.
Enfoque tipos de usuario	Usuarios con y sin experiencia.	Usuarios experimentados.
Rendimiento	Análisis en tiempo en real.	Rápido y buen rendimiento en todo momento
Curva de aprendizaje	Bajo/Medio nivel.	Alto nivel. Diseñado pensando en un usuario experimentado y se centra en el análisis de datos, brindando una mayor potencia y capacidad analítica.
Escalabilidad	Facilidad de manejar grandes volúmenes de datos.	Amplio potencial de volumen de datos.
Comunidad	Microsoft Power BI Community.	Tableau Public.

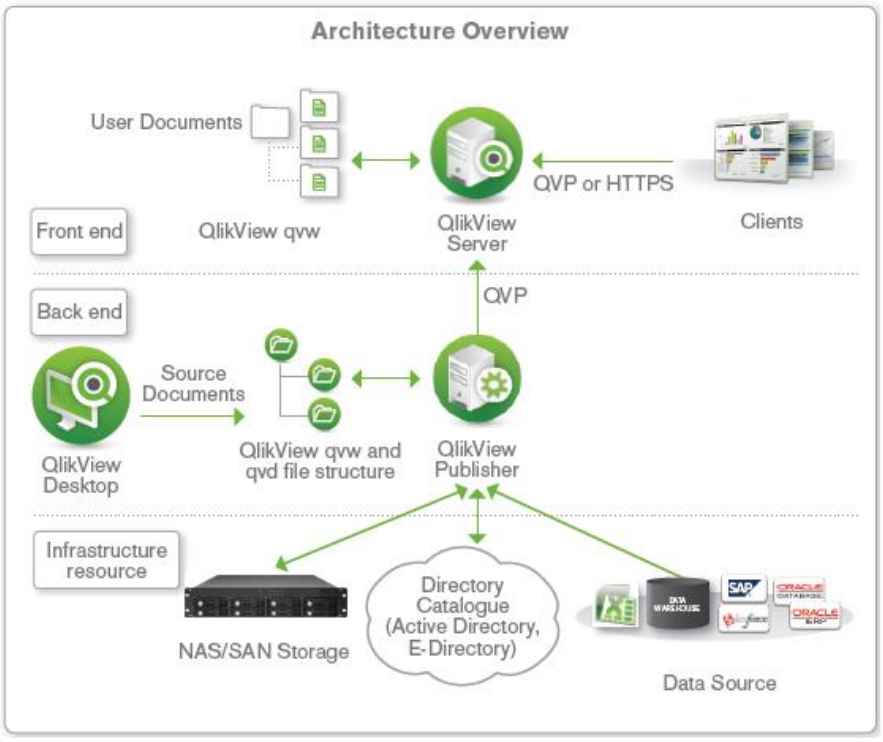
2.4.3. Qlik

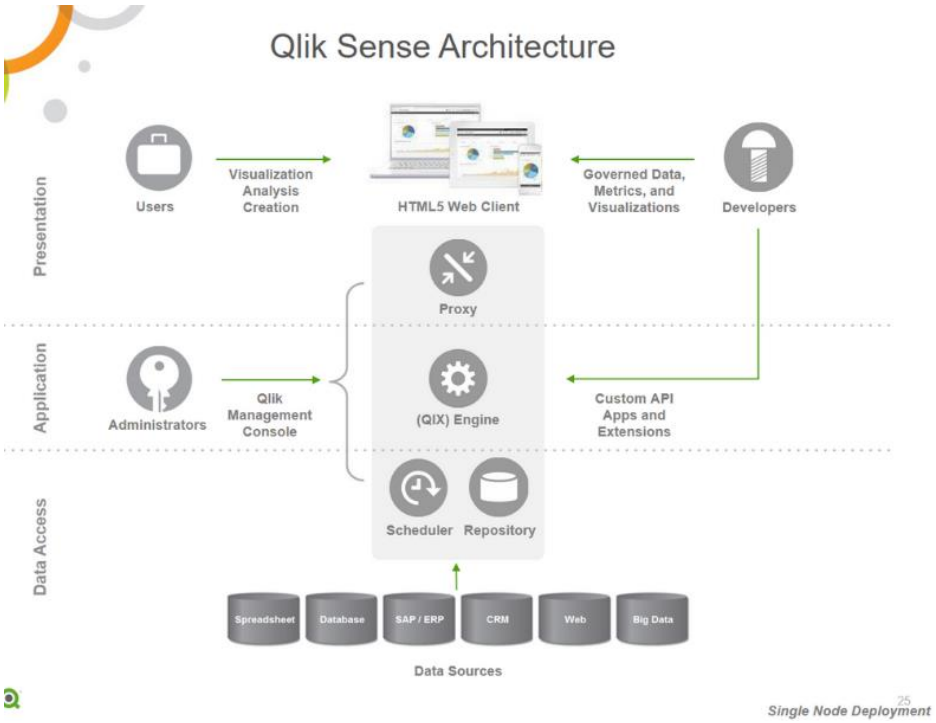
QlikView se destaca como una plataforma de analítica guiada que facilita la creación de aplicaciones por parte de desarrolladores de documentos QlikView. Estos expertos construyen modelos de datos, diseñan la interfaz, generan gráficos y definen fórmulas. Posteriormente, proporcionan la aplicación a los analistas, quienes cuentan con la libertad de explorar, seleccionar y profundizar en los datos. Sin embargo, se encuentran con limitaciones al intentar crear nuevas visualizaciones.

Qlik Sense, es una solución Business Intelligence de vanguardia desarrollada por la firma sueca Qlik, emerge como una plataforma líder en el ámbito del análisis y visualización de datos. Su capacidad para explorar y analizar conjuntos de datos complejos proporciona una ventaja distintiva soluciones para la toma de decisiones. Este análisis detalla las características y funcionalidades fundamentales que hacen de Qlik Sense una herramienta esencial en el ámbito de tecnologías de Business Intelligence.

2.4.3.1. Arquitectura Qlik

En la siguientes figuras se muestran las arquitecturas de Qlik View y Qlik Sense respectivamente:





2.4.3.2. Comparativa Qlik View y Qlik Sense

Característica	QlikView	Qlik Sense
Cliente		
- Cliente unificado HTML 5	Si	Si
- Interfaz de usuario táctil	Si	
- Diseño receptivo	Si	
Desarrollo		
- Codesarrollo de contenido	Si	
- Desarrollo de aplicaciones	Si	
- Creación de aplicación de autoservicio	Si	
- Preparación de datos de autoservicio	Si	
- Creación avanzada	Si	
- Desarrollo de escritorio	Si	Si
- Desarrollo sin conexión	Si	Si
- Desarrollo del lado del servidor	Si	
Plataforma		

- Amplia conectividad de datos	Si	Si
- Integración de datos - ETL	Si	Si
- Capacidades de Big Data	Si	
- Amplio ecosistema y comunidad	Si	Si
Analítica		
- Exploración asociativa	Si	Si
- Análisis basado en búsquedas	Si	Si
- Visualizaciones modernas	Si	
- Aplicaciones de análisis guiado/paneles	Si	Si
- Integración de análisis avanzado	Si	Si
Colaboración		
- Centro de colaboración	Si	
- Narración de datos	Si	
- Informes gestionados	Si	Si
- Móvil sin conexión	Si	Si
Despliegue		
- Gestión centralizada	Si	Si
- Bibliotecas gobernadas	Si	
- Seguridad basada en reglas	Si	Si
- Opciones de implementación de SaaS	Si	Si
- Motor de indexación asociativo		

2.4.3.3. Precios de Qlik

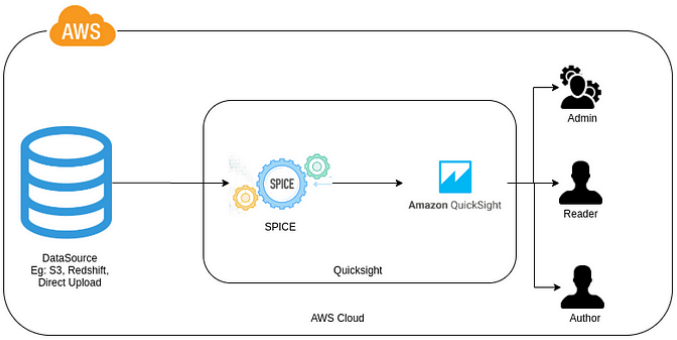
Para más información consultar el siguiente enlace:
https://www.qlik.com/es-es/pricing

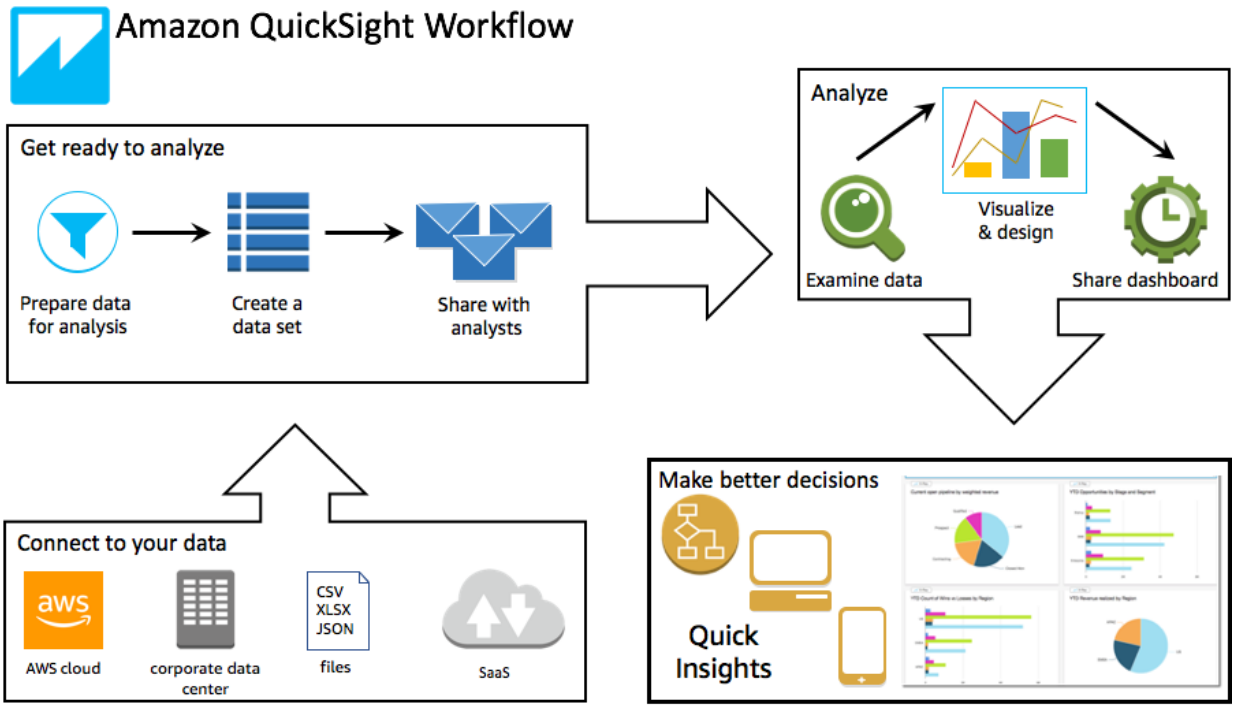
2.4.4. Amazon QuickSight

Amazon QuickSight es una solución de inteligencia de negocios (BI) basada en la nube, la cual permite diseñar y desarrollar informes y dashboards, sin importar su ubicación. QuickSight se integra con sus datos en la nube y agrega información de diversas fuentes. En un único cuadro de mando o dashboard, se pueden combinar datos de AWS, fuentes externas, big data, hojas de cálculo, aplicaciones SaaS, entre otras. Al ser un servicio totalmente administrado en la nube, Amazon QuickSight garantiza seguridad de nivel empresarial, alta disponibilidad global y escalabilidad.

2.4.4.1. Arquitectura de Amazon QuickSight

La siguiente figura describe la arquitectura de Amazon QuickSight:





2.4.4.2. Precios de AWS QuickSight

Para obtener más información sobre los precios de AWS QuickSight, consultar el siguiente enlace:
https://aws.amazon.com/es/quicksight/pricing/

2.4.4.3. Comparativa de AWS QuickSigth vs Power BI

Característica	Power BI	Amazon QuickSight
Fabricante	Microsoft	Amazon
Integración con la nube	Si	Si
Conexión fuentes de datos	Amplia variedad de conectores	Amplia variedad de conectores
Visualizaciones	Amplia variedad de visualizaciones	Variedad reducida de visualizaciones
Precios	Versión gratuita y de pago	Uso solo con suscripción
Seguridad	Azure AD	AWS IAM
ETL	Power Query	Preparación de datos con AWS Glue
Lenguaje	DAX	No
Escalabilidad	Si	Limitada con grandes volumen de datos
Integración con ML	Azure ML	QuickSight ML
Curva de aprendizaje	Media - Baja	Media - Baja

2.5. Lenguajes de programación

2.5.1. Python 3

Python 3 es un lenguaje de programación versátil y de alto nivel que proporciona un conjunto robusto de bibliotecas y herramientas para la manipulación de datos, análisis estadístico, machine learning, y la construcción y evaluación de modelos predictivos.

En proyectos de machine learning y data science, Python 3 se utiliza para tareas como la limpieza y preprocesamiento de datos, la exploración y visualización de datos, la implementación y entrenamiento de modelos de machine learning, y la evaluación del rendimiento de estos modelos. Python 3 es ampliamente apreciado en este contexto debido a su facilidad de uso, una comunidad activa de desarrolladores y una amplia variedad de bibliotecas populares como NumPy, Pandas, Matplotlib, Scikit-Learn, TensorFlow y PyTorch que brindan las herramientas necesarias para el análisis de datos y la construcción de modelos de machine learning.

En resumen, Python 3 es el lenguaje de programación de elección en proyectos de machine learning y data science debido a su versatilidad, una amplia gama de bibliotecas y herramientas disponibles, y su capacidad para abordar eficazmente los desafíos y tareas específicos de estas disciplinas.

2.5.1.1. Ventajas

- **Legibilidad y claridad de código**: Python tiene una sintaxis clara y legible, lo que facilita la escritura y el mantenimiento del código.
- **Curva de aprendizaje moderado**: es muy accesible y fácil de aprender y usar, pues se centra en la legibilidad del código, es muy versátil y es un lenguaje bien estructurado.
- **Amplia comunidad y soporte**: Python tiene una gran comunidad de usuarios y desarrolladores que ofrecen un amplio soporte, bibliotecas y recursos en línea.
- **Amplio ecosistema de bibliotecas**: Python cuenta con una gran cantidad de bibliotecas especializadas en ML y data science, como Scikit-Learn, TensorFlow, PyTorch, Pandas, NumPy, Matplotlib y muchas otras, que facilitan el desarrollo y la implementación de modelos y análisis de datos.
- **Rápido desarrollo de prototipos**: Python es adecuado para el desarrollo rápido de prototipos y pruebas de concepto, lo que permite a los equipos de ML y data science experimentar y probar ideas de manera eficiente.
- **Integración con otras tecnologías:** Python se integra bien con otras tecnologías y bases de datos, lo que facilita la conexión con sistemas de almacenamiento y bases de datos de datos en proyectos de data science.

2.5.1.2. Desventajas

- **Rendimiento limitado:** En comparación con lenguajes de programación de bajo nivel como C++ o Java, Python puede ser más lento en términos de rendimiento.
- **Gestión de memoria y recolección de basura:** Python utiliza un sistema de recolección de basura, lo que puede afectar el rendimiento en aplicaciones de alto rendimiento que requieren una gestión de memoria más precisa.
- **Concurrencia limitada:** Python tiene limitaciones en términos de concurrencia debido al Global Interpreter Lock (GIL), lo que puede dificultar la ejecución eficiente de múltiples hilos en paralelo.
- **Escalabilidad limitada:** A medida que los proyectos de ML y data science crecen, Python puede enfrentar desafíos en términos de escalabilidad y manejo de aplicaciones muy grandes y complejas.

2.5.2. R

R es un lenguaje especializado en estadísticas y análisis de datos que proporciona una amplia gama de paquetes y herramientas diseñados específicamente para la manipulación de datos, la visualización y el modelado estadístico.

En proyectos de data science y machine learning, R se utiliza para tareas como la carga y manipulación de conjuntos de datos, el análisis exploratorio de datos, la creación de visualizaciones informativas, y la implementación de algoritmos de machine learning y modelos estadísticos. R es apreciado en esta área debido a su fuerte enfoque en estadísticas y análisis de datos, que hacen que sea una opción poderosa para la investigación y el análisis de datos.

En resumen, R es un lenguaje de programación ampliamente utilizado en proyectos de data science y machine learning debido a su enfoque especializado en estadísticas, sus herramientas de análisis de datos de alto nivel y su capacidad para realizar análisis estadísticos avanzados y construir modelos predictivos con eficacia.

2.5.2.1. Ventajas

- **Especialización en estadística**: R está altamente especializado en estadísticas y análisis de datos, lo que lo hace ideal para proyectos de data science que involucran análisis estadístico y modelado.
- **Amplio ecosistema de paquetes**: R cuenta con un extenso conjunto de paquetes, bibliotecas y herramientas específicas para estadísticas, análisis de datos y machine learning. Paquetes xgboost son ampliamente utilizados en el campo.
- **Visualización de datos de calidad**: R es popular por su capacidad para crear visualizaciones de datos de alta calidad. Paquetes como ggplot2 permiten la creación de gráficos informativos y personalizados.
- **Programación funcional y expresiva**: R permite una programación funcional y expresiva que es adecuada para el análisis de datos y la manipulación de datos en proyectos de data science.
- **Soporte excelente para la manipulación de datos**: R tiene paquetes como dplyr, readr que son capaces de transformar datos desordenados en datos estructurados.

2.5.2.2. Desventajas

- **Manejo de datos**: en R, los objetos se almacenan en memoria, lo que contrasta con otros lenguajes como Python, por lo que requiere mucha más memoria, y no es muy buena opción cuando nos enfrentamos a problemas con muchísimos datos.

- **Curva de aprendizaje**: es un lenguaje que puede resultar complicado de aprender.
- **Menos velocidad**: R es mucho más lento que otros lenguajes como Python o Matlab.
- **Menos versátil**: Aunque R es excelente para estadísticas y análisis de datos, es menos versátil que Python en términos de aplicaciones fuera de estas áreas.
- **Escalabilidad limitada**: A medida que los proyectos de data science crecen en tamaño y complejidad, R puede enfrentar desafíos en términos de escalabilidad y manejo de proyectos muy grandes.
- **Rendimiento limitado**: en cálculos intensivos, R puede ser menos eficiente en proyectos de machine learning que requieren cálculos intensivos debido a su naturaleza interpretada.

Debido a la versatilidad y popularidad de Python, el proyecto se realizará en dicho lenguaje de programación, pues es muy fácil de entender incluso para quien no sepa programar en Python, y existe muchísima información sobre este lenguaje, lo cual es importante a la hora de buscar documentación y en foros de discusión.

2.6. Gestor de entornos virtuales

2.6.1. Anaconda

Anaconda es una distribución libre y abierta de los lenguajes de programación Python y R, es ampliamente utilizada en ciencia de datos y aprendizaje automático. Incluye procesamiento de grandes volúmenes de información, análisis predictivo y computación científica.

Contiene **gestión de entornos virtuales** de Python mediante Conda. Gracias a esto, es posible correr aplicaciones con diferentes configuraciones de paquetes y dependencias en específicas versiones de Python, cada entorno virtual es totalmente independiente y hace una analogía a los contenedores.

Además, incluye entornos de investigación y desarrollo como Jupyter y Spyder. Gracias a estos entornos con Anaconda ya no tenemos que instalar paquetes Python mediante el sistema de paquetes Pip nativo de la distribución de Linux específica. Esta práctica puede llevar a la corrupción de paquetes o dependencias y un posible mal funcionamiento del sistema operativo.

2.6.1.1. Ventajas

- **Gestión de entornos virtuales**: Anaconda facilita la creación y gestión de entornos virtuales, lo que permite aislar proyectos y sus dependencias, evitando conflictos entre versiones de paquetes.

- **Distribución de paquetes**: Proporciona una amplia gama de paquetes precompilados, lo que simplifica la instalación y actualización de bibliotecas y herramientas.
- **Ideal para ciencia de datos**, pues trae una gran cantidad de paquetes preinstalados de análisis y procesamiento de datos.
- **Conda**: Anaconda utiliza Conda, un sistema de gestión de paquetes y entornos, que es rápido, eficiente y permite la instalación de paquetes tanto de Python como de otros lenguajes, además de permitir la creación de entornos personalizados.
- **Multiplataforma**: Anaconda es compatible con Windows, macOS y Linux, lo que facilita la portabilidad de proyectos entre diferentes sistemas operativos.
- **Integración con IDEs**: Puedes integrar Anaconda con varios entornos de desarrollo como Jupyter Notebook, Visual Studio Code y otros, lo que facilita la creación y ejecución de código.
- **Comunidad activa**: Anaconda tiene una comunidad activa de usuarios y desarrolladores que proporcionan soporte, documentación y recursos en línea.

2.6.1.2. Desventajas

- **Tamaño**: La instalación de Anaconda puede ocupar una cantidad significativa de espacio en disco debido a la gran cantidad de paquetes precompilados incluidos en la distribución.
- **Rendimiento**: Algunos usuarios han informado que Conda, el administrador de paquetes de Anaconda puede ser más lento que otras alternativas en términos de velocidad de descarga e instalación de paquetes.
- **Dependencia de Anaconda**: Algunos usuarios pueden encontrar que la dependencia de Anaconda para la gestión de entornos y paquetes limita su flexibilidad, especialmente si ya están familiarizados con otras herramientas de gestión de paquetes de Python.
- **Actualizaciones de paquetes**: A veces, los paquetes en Anaconda pueden no estar tan actualizados como las versiones más recientes disponibles en PyPI (Python Package Index), lo que puede requerir una instalación adicional.

En general, Anaconda es una herramienta muy útil para la gestión de paquetes y entornos en Python, pero es importante considerar estas ventajas y desventajas para determinar si es la elección adecuada para el proyecto.

2.6.2. Poetry

Poetry es una herramienta de código abierto que **simplifica la gestión de dependencias y el empaquetado** en proyectos de Python. Es ampliamente adoptado en la comunidad de desarrollo de Python, ofreciendo una solución elegante para la gestión de paquetes y entornos de Python.

Una de las características destacadas de Poetry es su capacidad para **definir y gestionar entornos virtuales de forma transparente**. Esto permite ejecutar aplicaciones con configuraciones de paquetes específicas y dependencias en versiones de Python particulares. Cada entorno virtual es independiente, lo que se asemeja a la idea de contenedores, asegurando que los proyectos no entren en conflicto entre sí.

Poetry también proporciona entornos de desarrollo integrados, lo que simplifica la creación y ejecución de código. Además, su integración con PyPI (Python Package Index) facilita la gestión de paquetes y sus dependencias sin preocuparse por la corrupción de paquetes o problemas de dependencias que pueden surgir al utilizar herramientas de gestión de paquetes nativas de sistemas operativos.

2.6.2.1. Ventajas

- **Simplicidad en la gestión de dependencias**: Poetry simplifica la gestión de dependencias de proyectos de Python al permitir definirlas en un archivo pyproject.toml con una sintaxis clara y concisa.
- **Entornos virtuales integrados**: Poetry crea automáticamente entornos virtuales para proyectos, lo que facilita el aislamiento de las dependencias de diferentes proyectos y evita conflictos.
- **Facilidad de publicación en PyPI**: con Poetry es más fácil publicar tus paquetes en el índice de Python (PyPI) y gestionar las versiones de tu proyecto.
- **Empaquetamiento sencillo**: Puedes utilizar Poetry para generar distribuciones de proyectos con un solo comando, lo que facilita la distribución de aplicaciones.
- **Integración con herramientas populares**: Poetry se integra bien con herramientas como pytest, flake8 y otros, lo que facilita la realización de pruebas y asegura la calidad del código.

2.6.2.2. Desventajas

- **Curva de aprendizaje**: para los principiantes, Poetry puede tener una curva de aprendizaje, ya que requiere familiarizarse con su sintaxis y estructura de archivos específicos.
- **Compatibilidad limitada con algunos proyectos heredados**: Puede haber casos en los que proyectos más antiguos o complejos no se integren fácilmente con Poetry, lo que puede requerir un trabajo adicional.

- **Requiere que los colaboradores lo utilicen**: Para aprovechar al máximo Poetry en un proyecto, es importante que todos los colaboradores utilicen la herramienta, lo que puede requerir una adaptación en equipos que ya utilizan otras herramientas.
- **No es tan conocido como otras herramientas**: Aunque Poetry ha ganado popularidad en la comunidad de Python, aún no es tan ampliamente adoptado como pip y conda, lo que puede llevar a una menor disponibilidad de documentación y soporte.

Considerando todos los pros y contras de estas dos herramientas de gestión de entornos virtuales, nos hemos decantado por **utilizar Anaconda**, pues tiene unos pros y contras similares a los de Poetry, pero, además, **tiene preinstalados una serie de paquetes** destinados al análisis y transformación de datos, lo que nos ahorra bastante tiempo de estar instalando estos paquetes. Otro factor clave a la hora de decidirnos por Anaconda y no Poetry es que **Anaconda es mucho más popular**, lo que implica que existe mucha más documentación / información acerca de esta que de Poetry.

2.7. Entorno de desarrollo Integrado (IDE)

2.7.1. Jupyter Notebook

Jupyter Notebook es una **aplicación web** open-source que permite crear, editar y ejecutar cuadernos a través de cualquier navegador web. Puede instalarse en un servidor y servir como aplicación a varios clientes en red. Estos cuadernos pueden estar compuestos por:

- Código fuente
- Salida producida por la ejecución de cada celda de código
- Visualizaciones
- Ecuaciones
- Explicaciones y anotaciones

Jupyter es ideal para el proceso de **análisis** de datos, **investigación** del modelo de aprendizaje automático adecuado, **ajuste** de modelos y **visualización** de datos y resultados.

2.7.1.1. Ventajas

- **Interactividad**: Jupyter Notebooks permite la ejecución de código en celdas individuales. Esto facilita la exploración de datos y el desarrollo paso a paso, ya que puedes ejecutar y ver resultados inmediatamente.
- **Documentación enriquecida:** Puedes combinar código, texto, gráficos, ecuaciones y visualizaciones en un solo documento. Esto facilita la creación

de informes y presentaciones interactivas que incluyen tanto el código como la narración.

- **Entornos de cómputo heterogéneo**: Puedes utilizar Notebooks en una variedad de entornos, desde tu máquina local hasta servidores en la nube y supercomputadoras, lo que facilita la escalabilidad y el acceso a recursos de cómputo más potentes.
- **Comunidad activa**: La comunidad de Jupyter es grande y activa, lo que significa que hay una constante evolución y mejoras en la plataforma.

2.7.1.2. Desventajas

- **No es un IDE completo:** Aunque los Jupyter Notebooks son útiles para ciertos tipos de desarrollo y análisis, no son un entorno de desarrollo integrado completo como PyCharm o Visual Studio Code. Carecen de algunas características avanzadas, como la depuración de código y la gestión de proyectos.
- **Dificultad en proyectos grandes**: Para proyectos muy grandes y complejos, los Jupyter Notebooks pueden volverse difíciles de manejar, ya que la organización y la estructura del código se vuelven complicadas en documentos largos.
- **Control de versiones limitado**: Aunque se pueden utilizar sistemas de control de versiones como Git con Notebooks, a menudo se presentan desafíos al rastrear cambios en los documentos JSON subyacentes, lo que puede llevar a conflictos y problemas de colaboración.
- **Rendimiento limitado:** Los Notebooks pueden tener problemas de rendimiento en tareas computacionalmente intensivas debido a la naturaleza interactiva y al hecho de que los cálculos son ejecutados en un solo núcleo por defecto.
- **Complejidad de mantenimiento a largo plazo**: Con el tiempo, los Notebooks pueden volverse difíciles de mantener, especialmente si no se siguen prácticas de organización y documentación adecuadas.

2.7.2. Visual Studio Code

Visual Studio Code (VS Code) es un entorno de desarrollo de código abierto y altamente personalizable desarrollado por Microsoft. Es ampliamente utilizado en el mundo de la programación para una amplia variedad de lenguajes y tecnologías.

Tiene disponible una extensión de Jupyter Notebooks que permite a los desarrolladores y científicos de datos trabajar con Notebooks de Jupyter directamente dentro de la interfaz de Visual Studio Code.

En otras palabras, la extensión de Jupyter Notebooks integra la funcionalidad de Jupyter directamente en el entorno de desarrollo de VS Code, lo que facilita la creación, edición y ejecución de Notebooks de Jupyter en un entorno de desarrollo que ya es conocido y cómodo para muchos desarrolladores. Esto es especialmente beneficioso para aquellos que trabajan en proyectos de Machine Learning y Data Science, ya que pueden aprovechar las capacidades de Visual Studio Code y Jupyter Notebooks en un solo entorno para desarrollar y documentar proyectos de manera eficiente.

2.7.2.1. Ventajas

- **Integración de lenguajes y tecnologías:** VS Code es conocido por su versatilidad y admite una amplia variedad de lenguajes de programación y tecnologías. Esto facilita la integración de herramientas y bibliotecas específicas para machine learning y data science.
- **Entorno de desarrollo unificado:** Puedes trabajar con tu código Python y Notebooks de Jupyter en un solo entorno, lo que simplifica la gestión de proyectos y archivos.
- **Personalización:** VS Code es altamente personalizable a través de extensiones. Puedes adaptar tu entorno de desarrollo a tus necesidades específicas agregando extensiones para machine learning, visualización de datos, depuración y más.
- **Depuración de código:** VS Code, a diferencia de Jupyter Notebooks, ofrece capacidades de depuración avanzadas que pueden ser útiles al desarrollar algoritmos de machine learning y resolver problemas de data science.
- **Control de versiones:** Puedes utilizar herramientas de control de versiones como Git directamente en VS Code para rastrear y colaborar en proyectos de machine learning y data science.
- **Integración con Jupyter**: La extensión de Jupyter Notebooks permite crear, editar y ejecutar Notebooks de Jupyter en un entorno familiar y altamente personalizable.

2.7.2.2. Desventajas

- **Requisitos de recursos**: Proyectos de machine learning y data science a menudo requieren recursos computacionales significativos, y la ejecución de Notebooks en VS Code puede ser intensiva en términos de memoria y CPU.
- **Complejidad en proyectos grandes**: Para proyectos muy grandes y complejos, mantener la organización y la estructura del código y los Notebooks puede ser un desafío.
- **Posible incompatibilidad con algunas extensiones**: Aunque hay muchas extensiones útiles disponibles en VS Code, puede haber casos de incompatibilidad o conflictos entre ellas.

- **Interfaz de usuario para data science:** Aunque VS Code es un entorno de desarrollo general, no es tan especializado para data science como otras herramientas.

2.7.3. PyCharm

PyCharm es un IDE open-source para desarrollar aplicaciones en Python. Aunque no viene integrado con Anaconda, se puede configurar sobre qué entorno virtual de Conda correr las aplicaciones.

2.7.3.1. Ventajas

- **Asistencia y análisis** de codificación, sintaxis y detección de errores.
- **Facilidad de navegación** en proyectos y códigos.
- **Refactorización**, soporte para frameworks web como Django, Flask y Web2py.
- **Depurador de Python** nativo integrado y test unitarios.
- **Integra controladores** de versiones con interfaz para Git, Subversion, etc.

2.7.3.2. Desventajas

- **Consumo de recursos**: PyCharm tiende a consumir más recursos de memoria y CPU en comparación con entornos de desarrollo más ligeros, lo que puede ser problemático cuando se trabajan con grandes conjuntos de datos o modelos complejos de machine learning que requieren una gran capacidad de cómputo.
- **Coste**: Aunque existe una versión gratuita (Community Edition), la versión Professional con características avanzadas relacionadas con data science y machine learning, como el soporte de Python Scientific Tools, requiere una licencia de pago, lo que puede ser costoso para algunos usuarios.
- **Entorno menos flexible** para Jupyter Notebooks: Aunque PyCharm tiene soporte para Jupyter Notebooks, algunas personas pueden encontrar que su entorno de desarrollo no es tan flexible y ágil como en las otras herramientas
- **Requisitos de hardware**: Para un rendimiento óptimo, PyCharm en proyectos de data science y machine learning puede requerir hardware más potente, lo que puede ser una desventaja para usuarios con recursos limitados.
- **Menos enfocado en ciencia de datos**: Aunque PyCharm es versátil, su enfoque principal es el desarrollo de software en Python, por lo que puede carecer de algunas características específicas de data science y machine learning presentes en herramientas más especializadas como Jupyter Notebooks y herramientas de análisis de datos.

Considerando todas estas ventajas y desventajas, podemos concluir que la mejor opción de todas es VS Code con la extensión de Jupyter Notebooks, pues resulta un término medio entre

las otras dos opciones, mezclando los aspectos positivos de ambos y deshaciéndose de la mayoría de las desventajas que tienen ambos.

2.8. Frameworks y librerías de Machine Learning

2.8.1. SciKit-Learn

Librería de machine learning escrita en Python, proporciona una amplia selección de algoritmos de aprendizaje supervisados y no supervisados. Está desarrollada sobre varias bibliotecas Python orientadas al manejo y visualización de datos como NumPy, SciPy, Matplotlib, IPython y Pandas.

En la fase de investigación de la detección de clientes en situaciones de riesgo, se debe establecer qué algoritmo es el más adecuado mediante la visualización de los datos y resultados utilizando gráficas. Esta librería es básica para este propósito, su manejo y facilidad de aprendizaje conlleva a una rápida solución.

2.8.1.1. Ventajas

- **Fácil de aprender y usar**: SciKit-Learn tiene una interfaz simple y consistente que facilita el aprendizaje y la aplicación de algoritmos de aprendizaje automático. Es una excelente opción para principiantes y usuarios avanzados por igual.
- **Amplia variedad de algoritmos**: SciKit-Learn ofrece una amplia gama de algoritmos de aprendizaje automático supervisado y no supervisado, lo que permite abordar una variedad de tareas, desde clasificación y regresión hasta clustering y reducción de dimensionalidad.
- **Integración con otras bibliotecas**: SciKit-Learn se integra bien con otras bibliotecas de Python, como NumPy, Pandas y Matplotlib, lo que facilita la manipulación de datos y la visualización.
- **Rendimiento sólido**: Aunque no siempre es la opción más rápida en términos de rendimiento, SciKit-Learn ofrece un rendimiento sólido para la mayoría de las aplicaciones de aprendizaje automático y es eficiente para conjuntos de datos de tamaño moderado.
- **Implementación eficiente en Python**: SciKit-Learn está implementado en Python, lo que permite una fácil integración con otros proyectos y bibliotecas de Python. Además, ofrece compatibilidad con múltiples núcleos y se beneficia de la comunidad de desarrollo de Python.

2.8.1.2. Desventajas

- **Escasa optimización para conjuntos de datos masivos**: SciKit-Learn no está optimizado para trabajar con conjuntos de datos extremadamente grandes y puede ser menos eficiente en aplicaciones de big data.

- **Alcance limitado**: Aunque SciKit-Learn es una biblioteca versátil, tiene un alcance limitado en comparación con plataformas más amplias como TensorFlow, que abarcan una variedad de aplicaciones de aprendizaje automático y aprendizaje profundo.
- **Desafíos con características categóricas**: La manipulación de características categóricas puede ser menos intuitiva en SciKit-Learn en comparación con algunas bibliotecas de aprendizaje automático especializadas como CatBoost.

2.8.2. XGBoost

XGBoost (Extreme Gradient Boosting) es un algoritmo de aprendizaje automático supervisado que se utiliza para tareas de clasificación, regresión y otras aplicaciones. Es una implementación optimizada de Gradient Boosting, que es una técnica de ensamblado de modelos que combina múltiples modelos más simples, generalmente árboles de decisión, para crear un modelo más poderoso y preciso.

La característica distintiva de XGBoost es su capacidad para manejar de manera eficiente problemas de alto rendimiento y conjuntos de datos grandes. Utiliza técnicas como la regularización, la poda de árboles y la paralelización para mejorar el rendimiento y la precisión del modelo, por lo que es ampliamente utilizado para resolver una variedad de problemas en el campo de la ciencia de datos y el aprendizaje automático.

2.8.2.1. Ventajas

- **Alto rendimiento**: XGBoost posee un alto rendimiento y eficiencia en términos de velocidad de entrenamiento y precisión de predicción, pues está optimizado para el rendimiento y puede aprovechar múltiples núcleos de CPU y GPUs, lo que lo hace rápido y eficiente en conjuntos de datos grandes. Es una de las bibliotecas más rápidas para implementar algoritmos de Gradient Boosting.
- **Regularización**: XGBoost incorpora técnicas de regularización que ayudan a prevenir el sobreajuste (overfitting) y mejoran la capacidad de generalización de los modelos.
- **Flexibilidad**: XGBoost se puede utilizar tanto en problemas de clasificación como de regresión, lo que lo hace versátil y aplicable a una amplia variedad de tareas de aprendizaje automático.
- **Configuración de hiperparámetros**: Aunque la capacidad de ajustar los hiperparámetros es una ventaja, también puede ser un desafío, ya que la elección de los valores óptimos puede requerir experimentación y conocimiento previo.

2.8.2.2. Desventajas

- **Mayor consumo de recursos**: Dado su rendimiento y optimización, XGBoost puede consumir más recursos de memoria y CPU en comparación con algoritmos más simples, lo que puede ser una desventaja en entornos con recursos limitados.
- **Complejidad**: es un algoritmo complejo que requiere cierto grado de experiencia técnica para implementarlo y optimizarlo eficientemente. El ajustar tantos hyperparámetros puede ser un desafío lo que puede hacer que consuma mucho tiempo.
- **Overfitting**: aunque XGBoost tiene regularización para prevenir el overfitting, puede seguir sucediendo.
- **Falta de transparencia**: a menudo se le considera una caja negra, pues es difícil de comprender e interpretar como llega a sus predicciones.

Entre estas dos librerías no nos decantamos por ninguna de las dos, pues tenemos que ver cuál de las 2 es la más precisa a la hora de realizar una regresión o una clasificación. Es por ello que probaremos con los modelos de regresión / clasificación tanto de Scikit-Learn como de XGBoost. Además, usaremos Scikit-Learn también para otras funciones como crear pipelines, escalar y transformar features.

2.8.3. Feature Engine

Feature-Engine es una biblioteca de Python especializada en ingeniería de características diseñada para simplificar y optimizar la preparación y transformación de características en proyectos de aprendizaje automático y minería de datos. Proporciona una amplia gama de transformadores y herramientas que permiten el manejo eficiente de características categóricas y numéricas, la selección de características, el tratamiento de valores faltantes y la normalización, entre otras tareas comunes en la ingeniería de características. Feature-Engine es una valiosa adición a las bibliotecas de aprendizaje automático y facilita el proceso de preparación de datos para mejorar el rendimiento y la precisión de los modelos de aprendizaje automático.

2.8.3.1. Ventajas

- **Facilita la ingeniería de características**: Feature-Engine proporciona una variedad de transformadores y herramientas que simplifican tareas comunes de ingeniería de características, como el manejo de outliers, la codificación de características categóricas, la selección de características, etc. Pues Feature-Engine está diseñado específicamente para abordar problemas comunes de ingeniería de características, lo que lo hace una herramienta

especializada en este aspecto crítico del proceso de modelado acelerando el proceso de preparación de datos.

- **Integración con SciKit-Learn**: Feature-Engine se integra fácilmente con SciKit-Learn, lo que facilita su uso en flujos de trabajo de aprendizaje automático existentes.
- **Amplia variedad de transformadores**: Ofrece una amplia gama de transformadores que permiten realizar tareas de ingeniería de características avanzadas, como el manejo de outliers, la selección de características y la normalización.
- **Mucha documentación e información**: la librería tiene una página web donde tiene toda la información necesaria, con múltiples ejemplos para entender su funcionamiento.

2.8.3.2. Desventajas

- **Enfoque limitado**: Feature-Engine se enfoca principalmente en la ingeniería de características, por lo que no proporciona funcionalidad completa de modelado y evaluación de modelos, lo que puede requerir el uso de otras bibliotecas.
- **No es la mejor opción para tareas más generales**: Para tareas más generales de análisis de datos y limpieza, puede que sea más eficiente utilizar bibliotecas más amplias como Pandas, que ofrecen una funcionalidad más amplia.

2.8.4. Feature Tools

Featuretools es una biblioteca de código abierto en Python diseñada para la ingeniería de características automatizada en conjuntos de datos estructurados. Su enfoque principal es simplificar y acelerar el proceso de creación de nuevas características a partir de datos existentes, permitiendo la generación automática de características a partir de múltiples tablas relacionadas y la agregación de datos a diferentes niveles de granularidad. Esta capacidad de automatización y escalabilidad hace que Featuretools sea una herramienta valiosa para profesionales de la ciencia de datos y la minería de datos que desean acelerar el proceso de preparación de datos y centrarse en la construcción y evaluación de modelos de aprendizaje automático. La biblioteca se integra con Pandas y es ampliamente utilizada en competiciones de ciencia de datos y aplicaciones empresariales.

2.8.4.1. Ventajas

- **Automatización de la ingeniería de características**: Featuretools automatiza gran parte del trabajo tedioso y manual asociado con la creación de características, lo que ahorra tiempo y esfuerzo en proyectos de análisis de datos.

- **Gestión de datos relacionales**: Es capaz de manejar datos que se encuentran en múltiples tablas relacionadas y crear características que exploten esas relaciones, lo que es especialmente útil en bases de datos relacionales.
- **Creación de características de alto nivel**: Permite la creación de características de alto nivel a partir de datos detallados, lo que facilita la construcción de modelos más descriptivos y predictivos.
- **Escalabilidad**: Featuretools es escalable y puede manejar conjuntos de datos grandes y complejos, lo que lo hace adecuado para proyectos de ciencia de datos en el mundo real.

2.8.4.2. Desventajas

- **Requerimientos de memoria y recursos computacionales**: El procesamiento de datos grandes puede requerir una cantidad significativa de recursos de memoria y computación, lo que puede limitar su aplicabilidad en entornos con recursos limitados.
- **Posible complejidad en proyectos simples**: En proyectos de análisis de datos simples, la automatización completa de la ingeniería de características puede ser excesiva y generar características innecesarias.
- **Dependencia de la calidad de los datos**: Como en cualquier proceso de preparación de datos, la calidad de los datos de entrada es fundamental. Featuretools puede generar características basadas en datos ruidosos o incorrectos, lo que puede llevar a resultados erróneos.

Para llevar a cabo el Feature Engineering, se usará Feature Engine, pues no se dispone de una cantidad enorme de datos para este proyecto, por lo que Feature Tools puede no ser la opción más adecuada. Además, el acceso a la información para Feature Engine es más fácil, y esta información es muy intuitiva, por lo que esto también se ha valorado a la hora de seleccionar Feature-Engine.

2.8.5. Shap

SHAP (Shapley Additive explanations) es una biblioteca de Python utilizada para proporcionar explicaciones interpretables sobre los resultados de modelos de aprendizaje automático, permitiendo una comprensión detallada de cómo cada característica o variable influye en las predicciones del modelo. Basado en la teoría de juegos, SHAP calcula los valores SHAP (Shapley values) que representan la contribución relativa de cada característica en una predicción específica o en el modelo en su conjunto. Esta herramienta es esencial para la interpretación y la transparencia de los modelos de aprendizaje automático, ayudando a los profesionales a comprender y comunicar los factores detrás de las decisiones de un modelo, así como para identificar posibles sesgos y equidad en las predicciones.

2.8.5.1. Ventajas

- **Interpretación precisa**: SHAP proporciona una interpretación precisa y matemáticamente fundamentada de cómo cada característica afecta las predicciones de un modelo, lo que ayuda a comprender el "por qué" detrás de las decisiones del modelo.
- **Explicabilidad en modelos complejos**: Funciona bien con una variedad de modelos de aprendizaje automático, incluidos modelos basados en árboles, redes neuronales y otros, lo que permite la interpretación en modelos complejos.
- **Interpretación global y local:** Permite la interpretación tanto a nivel global como a nivel local, lo que significa que puedes entender el impacto de las características en predicciones individuales o en todo el modelo.
- **Diagnóstico de sesgos y equidad**: Se utiliza para evaluar y diagnosticar el sesgo en los modelos de aprendizaje automático, lo que es crucial en aplicaciones sensibles a la equidad y la justicia.
- **Compatibilidad con diversas bibliotecas de aprendizaje automático**: SHAP es compatible con varias bibliotecas populares de Python, como SciKit-Learn, XGBoost y LightGBM, lo que amplía su utilidad en diferentes entornos y aplicaciones.
- **Visualización de explicaciones**: SHAP ofrece herramientas de visualización para representar gráficamente las explicaciones, lo que facilita la comunicación de los resultados a partes interesadas no técnicas.

2.8.5.2. Desventajas

- **Tiempo de cómputo y recursos**: Calcular los valores SHAP puede ser computacionalmente costoso, especialmente en modelos grandes y conjuntos de datos extensos.
- **Necesidad de conocimiento previo**: Para aprovechar al máximo SHAP, es útil tener un conocimiento sólido de aprendizaje automático y comprender los conceptos subyacentes.
- **Posible sobrecarga de información**: SHAP puede proporcionar una gran cantidad de información detallada, lo que puede abrumar a los usuarios si no se utiliza adecuadamente.

2.8.6. LIME

LIME (Local Interpretable Model-Agnostic Explanations) es una biblioteca de Python diseñada para brindar explicaciones interpretables de modelos de aprendizaje automático, independientemente del algoritmo subyacente. Su objetivo es proporcionar una comprensión detallada de cómo un modelo toma decisiones en datos específicos, lo que facilita la interpretación y la explicación de sus predicciones. LIME utiliza técnicas de

muestreo y generación de datos artificiales para aproximar el comportamiento de un modelo en regiones locales de datos, lo que permite explicaciones a nivel de instancia y ayuda a los usuarios a comprender el razonamiento detrás de las decisiones del modelo. Esto es esencial en aplicaciones donde la transparencia y la interpretabilidad son fundamentales, como el crédito, la atención médica y la detección de sesgos en los modelos de aprendizaje automático.

2.8.6.1. Ventajas

- **Modelo agnóstico**: LIME es "agnóstico" en el sentido de que puede utilizarse con cualquier modelo de aprendizaje automático, lo que lo hace altamente versátil y aplicable a una amplia gama de algoritmos y modelos.
- **Interpretación local**: LIME se enfoca en proporcionar interpretaciones locales, lo que significa que explica el comportamiento de un modelo para una instancia de datos específica, lo que puede ser útil para comprender decisiones individuales.
- **Visualización de explicaciones**: Ofrece herramientas de visualización que permiten representar gráficamente las explicaciones, lo que facilita la comunicación de resultados a partes interesadas no técnicas.
- **Compatibilidad con diversas bibliotecas de aprendizaje automático**: LIME se puede usar con varias bibliotecas populares de Python, lo que facilita su incorporación en flujos de trabajo de aprendizaje automático existentes.

2.8.6.2. Desventajas

- **Dependencia del muestreo**: La calidad de las explicaciones de LIME depende en gran medida de la calidad del muestreo y la generación de datos artificiales, lo que podría resultar en explicaciones inexactas en casos difíciles.
- **Interpretación local**: Aunque LIME se centra en explicaciones locales, esto significa que no proporciona una visión general del modelo en su totalidad, lo que puede limitar la comprensión de su comportamiento global.
- **Posible sobrecarga de información**: Como con cualquier herramienta de interpretación, LIME puede proporcionar una gran cantidad de información detallada, lo que puede abrumar a los usuarios si no se utiliza adecuadamente.

Para la explicaciones interpretables de modelos de aprendizaje automático, elegiremos SHAP, pues sus resultados son más fiables que los de LIME y, además, SHAP puede dar una interpretación local o una global.

2.9. Frameworks y librerías de Deep Learning

2.9.1. Keras

Keras es una **API de alto nivel** desarrollada en Python, capaz de ejecutarse sobre varios motores de deep learning como TensorFlow, Theano y CNTK. Permite un uso rápido a través de su API siendo fácil de utilizar, modular y extensible. Su ejecución soporta tanto CPU como GPU. Abstrae la complejidad de la programación de modelos profundos con TensorFlow, siendo capaz de agilizar la fase de investigación.

Permite la integración de código TensorFlow en los momentos donde la API de Keras sea demasiado restrictiva para modelar algoritmos más complejos.

2.9.1.1. Ventajas

- **Sencillez y facilidad de uso**: Keras se destaca por su API simple y fácil de entender, lo que facilita la construcción y el entrenamiento de modelos de aprendizaje profundo, lo cual es beneficioso para principiantes y desarrolladores experimentados.
- **Modularidad y backend múltiple**: Keras tiene soporte para múltiples backend, siendo TensorFlow, Theano y CNTK los más comunes. También podemos entrenar el modelo de Keras en un backend y testearlo en otro distinto. Cambiar de backend en Keras es muy sencillo. Keras permite construir modelos de aprendizaje profundo de manera modular, lo que facilita la personalización de la arquitectura de las redes neuronales y la conexión de capas de manera sencilla.
- **Compatibilidad con CPU y GPU**: Keras es compatible con CPU y GPU, lo que permite el entrenamiento de modelos en hardware más potente y acelera el proceso de entrenamiento.
- **Comunidad activa y documentación completa**: Keras cuenta con una comunidad activa de usuarios y desarrolladores, lo que significa que es fácil encontrar recursos, tutoriales y ejemplos en línea. La documentación de Keras es completa y detallada, lo que facilita el aprendizaje y la solución de problemas.

2.9.1.2. Desventajas

- **Simplificación a expensas de la flexibilidad**: Aunque la simplicidad de Keras es una ventaja, puede resultar en cierta falta de flexibilidad en comparación con bibliotecas de más bajo nivel. Para proyectos muy específicos, algunos desarrolladores pueden optar por utilizar bibliotecas más personalizables.
- **Modelos grandes y eficiencia computacional**: El uso de modelos de aprendizaje profundo grandes puede requerir una potencia computacional

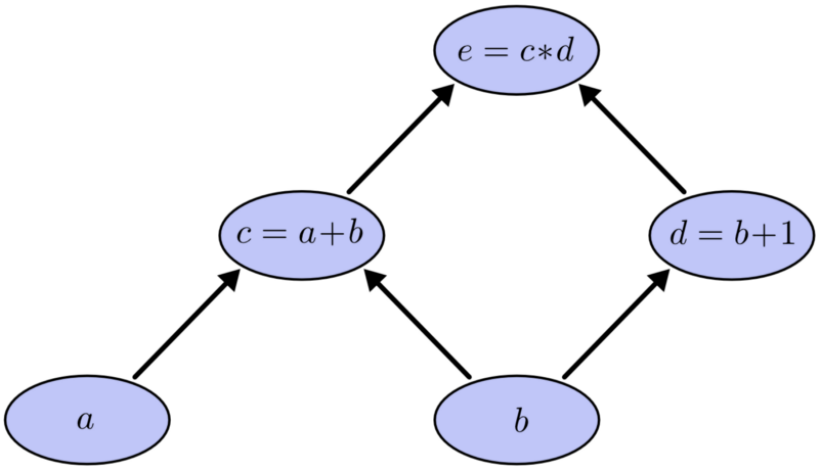
significativa. Para aplicaciones de producción, esto puede ser un desafío en términos de eficiencia y recursos.

- **Limitaciones en aplicaciones especializadas**: Aunque Keras es versátil, algunas aplicaciones altamente especializadas pueden requerir bibliotecas específicas para su dominio, lo que puede no estar completamente cubierto por Keras.

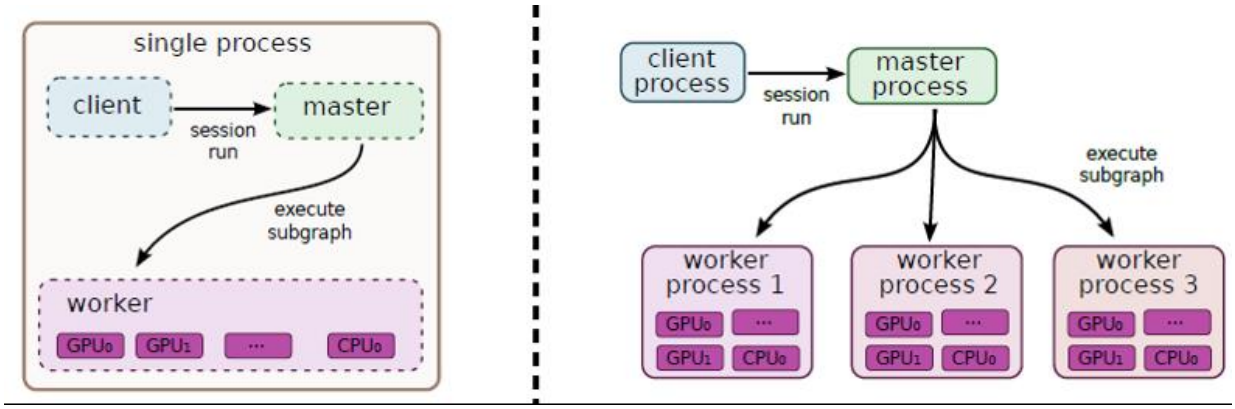
2.9.2. TensorFlow

TensorFlow es una **biblioteca** ampliamente conocida en la investigación e implementación de modelos machine Learning y deep Learning. Desde la versión 2.0, se incorpora modelos y arquitecturas de redes neuronales correspondientes con las últimas investigaciones más recientes por los científicos y data scientist del ámbito.

TensorFlow se basa en un grafo computacional estático y tensores que atraviesan los nodos del grafo a través de los bordes. El grafo computacional nunca puede ser cíclico, cada nodo del grafo representa una operación como suma, resta, multiplicación, etc. Y cada operación da como resultado un nuevo tensor, en este caso las operaciones c y d son independientes, es decir, TensorFlow ejecuta nodos en paralelo incluso de forma distribuida.



Los grafos computacionales estáticos (SCG) dan lugar a una programación y ejecución de TensorFlow simbólica, dificultando la depuración de los modelos y algoritmos desarrollados, pero acelerando la computación y optimización de memoria a unos niveles altos.



2.9.2.1. Ventajas

- **Escalabilidad**: TensorFlow es altamente escalable y se puede utilizar en una variedad de entornos, desde máquinas individuales hasta clústeres de servidores, lo que lo hace adecuado para proyectos de diferentes tamaños.
- **Integración con otras bibliotecas y marcos de trabajo**: TensorFlow se integra con otras bibliotecas y herramientas populares de aprendizaje automático, como Tensorboard para visualización y TensorFlow Lite para aplicaciones móviles.
- **Abstracción de alto nivel con Keras**: TensorFlow integra la popular biblioteca Keras, lo que facilita la construcción de modelos de manera más sencilla y legible, especialmente para principiantes en el aprendizaje profundo.
- **Soporte para CPU y GPU**: TensorFlow es compatible con el uso de unidades de procesamiento central (CPU) y unidades de procesamiento gráfico (GPU), lo que permite acelerar el entrenamiento de modelos en hardware más potente.
- **Rendimiento**: su rendimiento es alta, igualando a las mejores en la industria.
- **Amplia comunidad y recursos disponibles**: TensorFlow tiene una gran comunidad de usuarios y desarrolladores, lo que significa que hay una abundancia de recursos, tutoriales y modelos pre-entrenados disponibles.

2.9.2.2. Desventajas

- **Requisitos de recursos computacionales**: El entrenamiento de modelos de aprendizaje profundo con TensorFlow puede requerir una cantidad significativa de recursos de CPU y GPU, lo que puede ser un desafío en entornos con recursos limitados.
- **Posible complejidad en proyectos simples**: En proyectos de aprendizaje automático simples, TensorFlow puede resultar en una sobrecarga innecesaria debido a su versatilidad.

- **Soporte de GPU limitado**: solo NVIDIA y Python son soportados por TensorFlow para programar en la GPU. No tiene ningún otro soporte.
- **Baja velocidad de implementación**: TensorFlow es el que más tarda al entrenar distintos tipos de redes neuronales.

2.9.3. PyTorch

Es una **biblioteca de computación científica** basada en Python para la investigación y desarrollo de modelos de machine learning y deep learning.

Aunque todavía está en una fase muy joven, su curva de aprendizaje es pequeña en comparación con TensorFlow. Su principal ventaja es que se compone completamente de código Pythonic y los grafos de computación generados son dinámicos (DCG). Esta ventaja desencadena en una buena depuración del código.

Dado que al implementar redes neuronales conocer y comprender qué es lo que está sucediendo a nivel de tensor y operación en cada momento es una tarea muy compleja, que PyTorch utilice grafos de computación dinámicos facilita las tareas de depuración.

- Entrenamiento distribuido.
- Desarrollado en C++ integrado en Python: posibilidad de uso con librerías Cython y Numba.
- Anotaciones para generar funciones de computación híbridas entre modo de depuración y modo de optimización de tiempos.
- Exportar modelos ONNX de forma nativa.
- Ofrece una interfaz pura de C++.
- Programación imperativa.

2.9.3.1. Ventajas

- **Trazado dinámico**: La principal ventaja de PyTorch es su trazado dinámico, que permite definir y modificar la estructura de la red neuronal durante el tiempo de ejecución. Esto hace que sea más fácil de depurar y experimentar con modelos.
- **Fácil de debuguear**: puede usar las herramientas de debuguear de Python, usando IDEs de Python para debuguear.
- **Flexibilidad y experimentación**: PyTorch es una elección popular en entornos de investigación y desarrollo debido a su facilidad de experimentación con modelos y la capacidad de modificar modelos en tiempo real.
- **Soporte para GPU y CPU**: PyTorch es compatible con GPU y CPU, lo que permite acelerar el entrenamiento de modelos en hardware más potente.

- **Integración con Python**: PyTorch se integra estrechamente con Python, lo que facilita su uso junto con otras bibliotecas y herramientas de procesamiento de datos.
- **TorchScript**: PyTorch incluye TorchScript, que permite exportar modelos a un formato más eficiente para su ejecución en entornos de producción.

2.9.3.2. Desventajas

- **Ausencia de herramientas de visualización y monitoreo**: como Tensorboard.
- **Otros frameworks tienen mayor comunidad**
- **Menos aplicaciones de producción listas para usar**: Aunque PyTorch es ampliamente utilizado en investigación y desarrollo, es posible que requieras más trabajo de ingeniería para llevar los modelos de PyTorch a entornos de producción en comparación con otras bibliotecas.

Debemos elegir entre PyTorch y TensorFlow. PyTorch parece una mejor opción ya que permite debuguear y definir y modificar la estructura de la red neuronal durante la ejecución, lo que hace que sea más fácil de comprender lo que sucede, por lo que elegiremos PyTorch sobre TensorFlow.

2.9.4. Tensorboard

Es un **software de visualización** para facilitar la comprensión, depuración y la optimización de los modelos entrenados mediante TensorFlow.

Los grafos de computación de TensorFlow se pueden volver extremadamente complejos, esta herramienta ofrece un visualizador del grafo construido por nuestra aplicación. Una ayuda más en la depuración y la comprensión del modelo.

La evolución de los pesos de las capas de una red neuronal durante su entrenamiento o la precisión del modelo, y cualquier otro tensor que necesitemos registrar los datos que fluyen por él durante la ejecución de TensorFlow.

Esta herramienta ofrece gráficas de la evolución de cualquier tensor que queramos depurar de la aplicación. Además, guarda diferentes ejecuciones para una posterior comparación y elección del ajuste óptimo.

Es compatible con Keras, TensorFlow y PyTorch.

2.9.4.1. Ventajas

- **Visualización de modelos**: Tensorboard permite visualizar gráficamente la estructura y arquitectura de los modelos de TensorFlow, lo que facilita la comprensión de su diseño.

- **Seguimiento en tiempo real**: Permite el seguimiento en tiempo real de métricas de entrenamiento, como pérdida, precisión y tasa de aprendizaje, lo que ayuda a ajustar y optimizar los modelos durante el entrenamiento.
- **Diagnóstico de problemas**: Tensorboard proporciona herramientas para diagnosticar problemas en modelos, como gradientes desvanecientes o explosivos, mediante la visualización de histogramas y distribuciones de pesos.
- **Comparación de experimentos**: Puedes comparar diferentes experimentos y modelos en un solo lugar, lo que ayuda a tomar decisiones informadas sobre qué enfoques son más efectivos.

2.9.4.2. Desventajas

- **Dependencia de TensorFlow**: Tensorboard está fuertemente vinculado a TensorFlow, lo que significa que no es tan versátil si estás utilizando otras bibliotecas de aprendizaje automático o marcos de trabajo.
- **Requiere almacenamiento de registros**: Para utilizar Tensorboard, debes almacenar registros de entrenamiento que ocupan espacio en disco, lo que puede ser un inconveniente si se generan grandes cantidades de datos.
- **Requiere recursos de hardware**: La visualización en tiempo real y la generación de gráficos pueden requerir recursos computacionales y de memoria, lo que podría afectar el rendimiento en entornos con recursos limitados.

2.10. Herramientas de procesamiento Big Data

2.10.1. Apache Spark

Apache Spark es una plataforma de computación en clúster diseñada para procesar y trabajar con grandes volúmenes de datos de manera eficiente y rápida.

Spark extiende el clásico modelo MapReduce para aceptar nuevas lógicas de computación, incluidas consultas interactivas y procesos distribuidos en clústeres, además de funcionar con YARN y HDFS para aprovechar el almacenamiento replicado distribuido. Como mejora, Spark acepta lenguajes de programación como Scala, Python, R, SQL y Java.

El potencial de Apache Spark está en el procesamiento paralelo y escalable basado principalmente en el concepto de “Resilient Distributed Dataset” (RDD). Es un nivel superior de abstracción que representa una colección inmutable de objetos distribuidos en diferentes clústers de computación. Este concepto se ha rediseñado en las últimas

versiones añadiendo un nivel aún superior de abstracción mediante DataFrames, donde los datos toman una estructura tabular.

2.10.1.1. Ventajas

- **Alto rendimiento**: Spark está diseñado para ofrecer un alto rendimiento en el procesamiento de datos a gran escala gracias a su capacidad de procesamiento en memoria y al procesamiento en paralelo.
- **Facilidad de uso**: Spark proporciona API en varios lenguajes, lo que facilita su adopción y desarrollo. Las APIs en Scala, Java, Python y R permiten a los desarrolladores trabajar en su lenguaje preferido.
- **Tolerancia a fallos**: Spark ofrece tolerancia a fallos a través de la recuperación de datos, lo que asegura que las aplicaciones continúen funcionando incluso en caso de fallas en el hardware o el software.
- **Compatibilidad con múltiples fuentes de datos**: Spark es compatible con una amplia variedad de fuentes de datos, lo que facilita la ingestión de datos desde diferentes sistemas, como HDFS, bases de datos, sistemas de transmisión y más.
- **Procesamiento en tiempo real**: Spark Streaming permite el procesamiento en tiempo real de datos de transmisión, lo que es fundamental para aplicaciones de análisis en tiempo real.
- **Escalabilidad**: Spark es altamente escalable y puede manejar grandes conjuntos de datos distribuidos en clústeres de computadoras.

2.10.1.2. Desventajas

- **Requisitos de recursos**: Spark puede ser intensivo en recursos, especialmente cuando se utiliza procesamiento en memoria. Esto puede requerir clústeres de servidores potentes y con suficiente memoria.
- **Costos asociados**: Implementar y mantener un clúster de Spark puede ser costoso en términos de hardware y recursos de administración.
- **Compatibilidad limitada con SQL**: Aunque Spark es compatible con SQL, su motor de consulta SQL no es tan avanzado como algunos sistemas de bases de datos dedicados.
- **Limitaciones en el procesamiento de datos estructurados**: Aunque Spark es versátil, algunas tareas específicas de procesamiento de datos estructurados pueden ser más eficientes en sistemas de bases de datos relacionales.

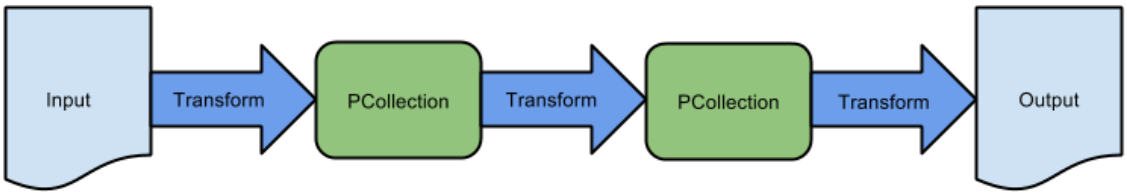
2.10.2. Apache Beam

Apache Beam es una plataforma unificada open-source para definir procesos de streaming y procesamiento distribuido en diferentes clústers de grandes volúmenes de datos. Utiliza el patrón de escalabilidad máster-worker.

Además, es un modelo de programación que se utiliza para definir y ejecutar tuberías de procesamiento de datos que incluyen ETL (Extract, Transform and Load) en el procesamiento de lotes y flujos.

Se compone de varios SDK en diferentes lenguajes de programación: Go, Java y Python. Estos proporcionan un modelo de programación unificado que puede representar y transformar conjuntos de datos de cualquier tamaño. Beam se ejecuta sobre varios motores de procesamiento como Apache Apex, Flink, Spark, Samza, Gearpump y Google Cloud Dataflow.

Además, es posible crear tuberías de transformaciones para procesamiento de datos masivo mediante los objetos PCollection una estructura de datos utilizada por Beam para albergar datos estructurados o no de forma distribuida. Beam utiliza objetos ParDo como transformación de un procesamiento genérico paralelo de datos, es un paradigma similar al MapReduce de Spark.



El testing de las tuberías se desarrolla de manera fácil gracias a los procesos y objetos DoFnTester de la librería integrada.

Soporta conectores de entrada/salida:
https://beam.apache.org/documentation/io/built-in/

2.10.2.1. Ventajas

- **Modelo de programación unificado**: Apache Beam proporciona un modelo de programación unificado que permite a los desarrolladores expresar pipelines de procesamiento de datos de manera consistente en varios motores de procesamiento.
- **Portabilidad**: Las aplicaciones de Apache Beam son portables y pueden ejecutarse en diferentes motores de procesamiento, lo que facilita la migración y la elección de la infraestructura.
- **Integración con ecosistemas de Big Data**: Apache Beam se integra con otros proyectos de Big Data y procesamiento, como Apache Hadoop, Apache Spark y Google Cloud Dataflow, lo que amplía su versatilidad.

- **Optimización de rendimiento**: La biblioteca de Apache Beam se encarga de la optimización y paralelización de las tareas de procesamiento, lo que puede mejorar el rendimiento de las aplicaciones.
- **Manejo de datos en tiempo real y por lotes**: Apache Beam es capaz de manejar tanto el procesamiento en tiempo real como el procesamiento por lotes, lo que lo hace adecuado para una amplia variedad de aplicaciones.
- **Extensibilidad**: Los desarrolladores pueden crear transformaciones personalizadas y adaptar los pipelines según sus necesidades específicas.

2.10.2.2. Desventajas

- **Complejidad adicional**: La abstracción del modelo unificado puede agregar una capa adicional de complejidad a las aplicaciones de procesamiento de datos en comparación con soluciones específicas para un motor.
- **Documentación limitada en algunos SDKs**: Aunque los SDKs principales como Java y Python tienen una amplia documentación, los SDKs menos comunes pueden tener una documentación más limitada.
- **Desarrollo de transformaciones personalizadas**: La creación de transformaciones personalizadas puede requerir un conocimiento más profundo del modelo de Apache Beam y puede ser un proceso más complejo.
- **Requisitos de recursos**: Al igual que otras soluciones de Big Data, Apache Beam puede ser intensivo en recursos, lo que puede requerir una infraestructura adecuada.

2.10.3. Apache AirFlow

AirFlow es una plataforma para crear, programar y monitorear flujos de trabajo mediante programación y creación de grafos acíclicos de tareas DAG.

Permite definir, programar, ejecutar, monitorear y administrar sus flujos de trabajo en código Python puro, al mismo tiempo que proporciona las herramientas y la interfaz de usuario para manejar esas operaciones de flujo de trabajo.

- Dinámico: Las tuberías de AirFlow están configuradas mediante Python y permiten generación dinámica de tareas dentro de ellas mediante instancias codificadas con Python.
- Extensible: Definir operadores, ejecutores e incluso extender la librería en un nuevo entorno se vuelve sencillo.
- Parametrización: Desarrollo de scripts con el uso Jinja como motor de plantillas.
- Escalable: Está desarrollado en una arquitectura modular y utiliza una cola de mensajes para orquestar cualquier número de workers configurados en el sistema principal.

El scheduler de AirFlow ejecuta sus tareas en una matriz de workers mientras sigue las dependencias especificadas. Las herramientas de la línea de comando facilitan la realización de tareas complejas en DAGs. La interfaz de usuario facilita la visualización de las tuberías que se ejecutan en producción, el monitoreo del progreso y la resolución de problemas cuando sea estrictamente necesario.

Airflow proporciona las siguientes herramientas:

- Administración de colas de trabajo
- Sistema de agrupación de recursos
- Funciones de relleno
- Historial completo de métricas y estadísticas
- Registro y parámetros de configuración para las ejecuciones de tareas
- Utiliza una base de datos central
- Interfaz de línea de comandos
- Interfaz de usuario web
- Ejecutores de tareas
- Programador de tareas

2.10.3.1. Ventajas

- **Orquestación y programación de tareas**: Airflow permite la orquestación y programación de tareas de manera flexible, lo que facilita la automatización de flujos de trabajo, la visualización de dependencias y la gestión de tareas, gracias a los DAGs (Grafos Acíclicos Dirigidos).
- **Monitoreo y registro**: Airflow proporciona un panel de control web que permite el monitoreo en tiempo real de tareas y la visualización de registros detallados, lo que facilita la resolución de problemas.
- **Reintentos y recuperación**: Airflow admite la reejecución de tareas en caso de fallos, lo que garantiza la integridad de los flujos de trabajo.
- **Extensibilidad**: La arquitectura modular de Airflow permite la incorporación de conectores y operadores personalizados para adaptarse a las necesidades específicas del proyecto.
- **Integración con sistemas externos**: Airflow se integra con una amplia variedad de sistemas y servicios, lo que facilita la automatización de procesos que involucran múltiples tecnologías.

2.10.3.2. Desventajas

- **Requerimientos de infraestructura**: Implementar y mantener Airflow puede requerir recursos significativos en términos de infraestructura y recursos de administración.

- **Complejidad adicional**: La abstracción de DAGs puede agregar una capa adicional de complejidad en comparación con soluciones más simples de automatización de tareas.
- **Gestión de dependencias**: Si los flujos de trabajo tienen muchas dependencias complejas, la administración de DAGs puede volverse complicada.

2.10.4. Dask

Dask es una biblioteca totalmente flexible para la computación paralela en Python. Se compone de la programación dinámica de tareas optimizada para el cálculo y las colecciones de "Big Data" como matrices paralelas, marcos de datos y listas que extienden de interfaces comunes como NumPy, Pandas o iteradores de Python a entornos con sistemas distribuidos.

- Dask es un componente del ecosistema Python, se acopla y mejora otras bibliotecas como NumPy, Pandas y Scikit-Learn.
- El modelo interno de Dask es inferior a Spark y, por lo tanto, carece de optimizaciones de alto nivel, pero es capaz de implementar algoritmos más sofisticados y construir sistemas a medida más complejos.
- Reutiliza la API de Pandas y el modelo de memoria, no implementa SQL ni optimizadores de consultas, pero es capaz de realizar accesos aleatorios y operaciones indexadas eficientes al estilo Pandas.
- Interactúa con bibliotecas como Scikit-Learn y XGBoost de Python. Aunque se ha desarrollado el proyecto DaskML para integraciones con machine learning.
- Compatible con el modelo NumPy de matrices multidimensionales, muy importante en la implementación de redes neuronales.
- Proporciona una interfaz de características en tiempo real a nivel inferior que Spark, por ello permite casos de uso más creativos y complejos, pero requiere más trabajo para su desarrollo.

Comparación Apache Spark vs Dask: http://docs.dask.org/en/stable/spark.html

2.10.4.1. Ventajas

- **Paralelismo escalable**: Dask permite el procesamiento paralelo y distribuido de datos en clústeres de máquinas, lo que acelera significativamente el procesamiento de grandes conjuntos de datos.
- **Computación perezosa**: Dask utiliza la evaluación perezosa, lo que significa que las operaciones se retrasan hasta que sea necesario, permitiendo la construcción de pipelines de procesamiento de datos eficientes y la optimización de recursos.

- **Interoperabilidad**: Dask se integra bien con bibliotecas populares de Python, como NumPy, Pandas y Scikit-Learn, lo que facilita su uso en aplicaciones de análisis de datos existentes.
- **Manejo de datos más grandes que la memoria RAM**: Dask puede manejar conjuntos de datos que no caben en la memoria RAM de una sola máquina al dividir y distribuir el procesamiento en múltiples nodos.
- **Flexibilidad**: Dask permite a los desarrolladores definir flujos de trabajo personalizados y operaciones, lo que es útil para una variedad de aplicaciones de procesamiento de datos.
- **Escalabilidad en la nube**: Dask se puede configurar para ejecutarse en entornos de nube, lo que facilita la escalabilidad y la administración de recursos en la nube.

2.10.4.2. Desventajas

- **Gestión de clústeres**: La configuración y administración de clústeres para ejecutar Dask en entornos distribuidos puede requerir conocimientos adicionales.
- **Costos asociados**: La escalabilidad en la nube y la administración de recursos en la nube pueden generar costos adicionales.
- **Mayor complejidad en ciertos casos**: Para problemas sencillos que caben en la memoria de una sola máquina, Dask puede ser una solución más compleja que las bibliotecas tradicionales como NumPy o Pandas.

Entre Apache Spark y Dask, seleccionaremos Dask, pues es una excelente opción cuando se trabaja en un entorno basado en Python y se necesita una mayor flexibilidad en la definición de operaciones personalizadas.

3. BIBLIOGRAFIA

[2.1.4.1] Talend Open Source (TOS): https://www.talend.com/products/talend-open-studio/
[2.1.4.2] Microsoft Server Integration Services (SSIS): https://learn.microsoft.com/en-us/sql/integration-services/sql-server-integration-services?view=sql-server-ver16
[2.1.4.4] Pentaho Data Integration (PDI): https://pentaho-public.atlassian.net/wiki/spaces/EAI/overview
[2.1.5.1] Talend Data Fabric: https://www.talend.com/products/data-fabric/
[2.1.6] Microsoft Azure Data Factory: https://azure.microsoft.com/es-es/products/data-factory
[2.1.7] Amazon AWS Glue: https://aws.amazon.com/es/glue/
[2.1.8] Google Cloud Dataflow: https://cloud.google.com/dataflow?hl=es
[2.1.9] Informatica Cloud Data Integration: https://www.informatica.com/es/products/cloud-integration/cloud-data-integration.html
[2.2.6.1] PostgreSQL: https://www.postgresql.org/
[2.2.6.2] Microsoft SQL Server: https://www.microsoft.com/es-es/sql-server/sql-server-2022
[2.2.6.3] MySQL: https://www.mysql.com/
[2.3.4.1] Amazon Redshift: https://aws.amazon.com/es/redshift/
[2.3.4.2] Microsoft Azure Synapse Analytics: https://azure.microsoft.com/es-es/products/synapse-analytics
[2.3.4.3] Google BigQuery: https://cloud.google.com/bigquery?hl=es
[2.3.4.4] Snowflake: https://www.snowflake.com/es/
[2.4.1] Microsoft Power BI: https://powerbi.microsoft.com/es-es/
[2.4.2] Tableau: https://www.tableau.com/es-es
[2.4.3] Qlik: https://www.qlik.com/es-es/
[2.4.4] Amazon QuickSight: https://aws.amazon.com/es/quicksight/
[2.7.1] Python 3: https://www.python.org/download/releases/3.0/
[2.7.2] R: https://www.r-project.org/
[2.8.1] Anaconda: https://www.anaconda.com/
[2.8.2] Poetry: https://python-poetry.org/
[2.9.1] Jupyter Notebooks: https://jupyter.org/
[2.9.2] Visual Studio Code: https://code.visualstudio.com/
[2.9.3] PyCharm: https://www.jetbrains.com/es-es/pycharm/
[2.10.1] Sickit-Learn: https://scikit-learn.org/stable/
[2.10.2] XGBoost: https://xgboost.ai/
[2.10.3] Feature Engine: https://feature-engine.trainindata.com/en/latest/
[2.10.4] Feature Tools: https://www.featuretools.com/

[2.10.5] Shap: https://shap.readthedocs.io/en/latest/
[2.10.6] LIME: https://lime-ml.readthedocs.io/en/latest/
[2.11.1] Keras: https://keras.io/
[2.11.2] Tensorflow: https://www.tensorflow.org/?hl=es-419
[2.11.3] Pytorch: https://pytorch.org/
[2.11.4] Tensorboard: https://www.tensorflow.org/tensorboard/get_started?hl=es-419
[2.12.1] Apache Spark: https://spark.apache.org/
[2.12.2] Apache Beam: https://beam.apache.org/
[2.12.3] Apache Airflow: https://airflow.apache.org/
[2.12.4] Dask: https://www.dask.org/