Entre los componentes que ofrece AWS Glue para los flujos de transformaciones de ETL se puede destacar **AWS Glue Studio,** que es una herramienta visual desarrollada por AWS que simplifica y acelera el proceso de preparación de datos y creación de flujos de trabajo ETL. A continuación, tienes una descripción detallada de AWS Glue Studio y sus principales funcionalidades:

- **Interfaz gráfica de usuario intuitiva.** AWS Glue Studio proporciona una interfaz gráfica de usuario intuitiva que les permite a los usuarios diseñar y construir flujos de trabajo ETL de manera visual. Utilizando una interfaz de tipo «arrastrar y soltar», los usuarios pueden crear y personalizar flujos de trabajo sin necesidad de escribir código.

- **Creación de flujos de trabajo ETL.** Glue Studio permite que los usuarios creen flujos de trabajo ETL para extraer, transformar y cargar datos desde diversas fuentes a destinos como almacenes de datos, *data lakes* o bases de datos. Los flujos de trabajo pueden incluir múltiples pasos de transformación y manipulación de datos.

- **Amplia gama de conectores.** AWS Glue Studio proporciona una amplia gama de conectores predefinidos que facilitan la conexión a diversas fuentes de datos, como Amazon S3, Amazon RDS, Amazon Redshift y más. Esto simplifica el proceso de integración y extracción de datos desde múltiples fuentes.

- **Herramientas de transformación visual.** Glue Studio ofrece herramientas visuales para realizar transformaciones de datos, como filtrado, unión, agregación, y limpieza de datos. Los usuarios pueden arrastrar y soltar operadores de transformación para definir el flujo de datos y aplicar transformaciones a los datos de manera visual.

- **Monitoreo y depuración.** El servicio incluye herramientas integradas para monitorear y depurar flujos de trabajo en tiempo real. Los usuarios pueden ver el progreso de los flujos de trabajo, detectar y solucionar problemas, y optimizar el rendimiento de manera eficiente.

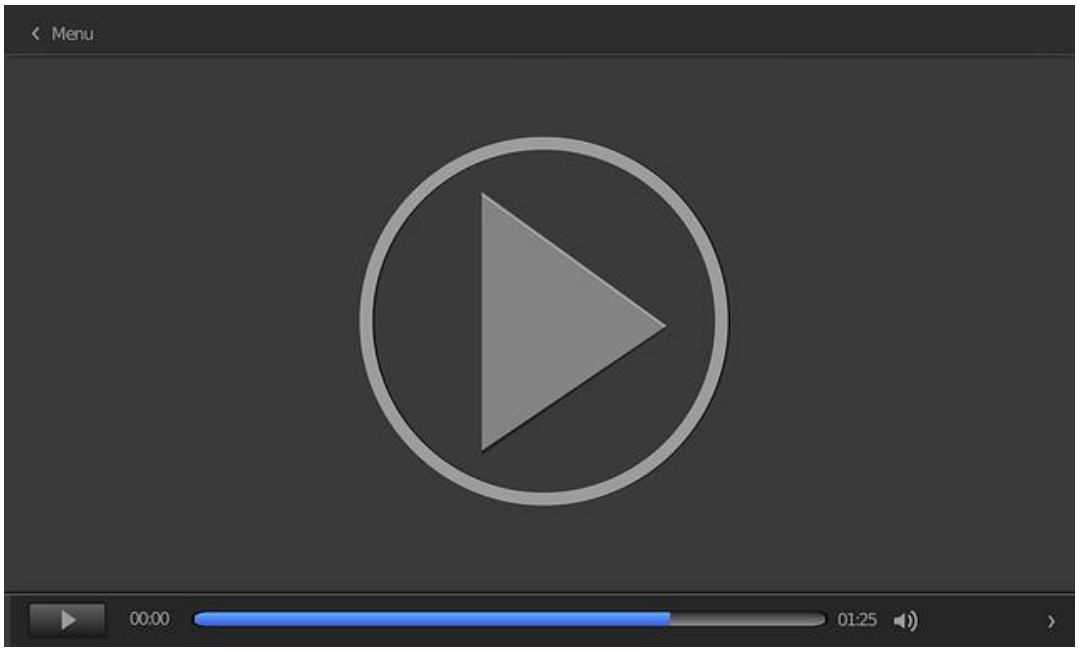
- **Colaboración y versionado.** Glue Studio permite que los equipos colaboren en el diseño y desarrollo de flujos de trabajo mediante la compartición de proyectos y la gestión de versiones. Los usuarios pueden trabajar en equipo para crear y mantener flujos de trabajo de manera colaborativa y controlar los cambios a lo largo del tiempo.

- **Integración con servicios de AWS.** AWS Glue Studio se integra estrechamente con otros servicios de AWS, como AWS Glue Data Catalog, AWS Glue Jobs, Amazon S3, Amazon Redshift y más. Esto permite que los usuarios aprovechen el ecosistema completo de AWS para el procesamiento y el análisis de datos.

En resumen, **AWS Glue Studio** es una herramienta poderosa y versátil que permite que los usuarios diseñen, construyan y administren flujos de trabajo ETL de manera visual y colaborativa. Con su interfaz gráfica intuitiva, amplia gama de conectores, herramientas de transformación visual, y capacidades de monitoreo y depuración, Glue Studio simplifica y agiliza el proceso de preparación de datos en la nube.

# Tema 2. Ingesta de datos para big data e inteligencia artificial en cloud

El siguiente vídeo, *Prepara tus datos y administra tus ETLs con AWS Glue – Español* (AWS LATAM, 2021b), es un tutorial que explica cómo preparar y administrar los datos con AWS Glue.




Prepara tus datos y administra tus ETLs con AWS Glue – Español.

Accede al vídeo:

https://www.youtube.com/embed/mw6nu7-_4PI

Los **pasos básicos para crear un trabajo** con AWS Glue Studio son los siguientes:

- **Acceder a AWS Glue Studio.** Inicia sesión en la consola de AWS y navega hasta el servicio de AWS Glue. Una vez allí, selecciona «Glue Studio» en el menú de navegación.
- **Crear un nuevo proyecto.** En la interfaz de Glue Studio, selecciona «Crear un nuevo proyecto» para iniciar la creación de un nuevo proyecto.
- **Crear un flujo de trabajo.** Dentro del proyecto, selecciona «Crear un flujo de trabajo» para comenzar a diseñar tu flujo de trabajo ETL.

- **Seleccionar conectores de origen y destino.** Selecciona los conectores de origen y destino que deseas utilizar para tu flujo de trabajo. Puedes elegir entre una amplia gama de conectores predefinidos para acceder a datos desde diferentes fuentes y cargarlos en diferentes destinos.

- **Configurar los pasos del flujo de trabajo.** Diseña el flujo de trabajo arrastrando y soltando operadores de transformación y manipulación de datos en el lienzo de Glue Studio. Configura los pasos del flujo de trabajo, como la extracción, la transformación y la carga de datos, según tus requisitos específicos.

- **Configurar parámetros y propiedades.** Configura los parámetros y las propiedades de cada paso del flujo de trabajo, como los nombres de las tablas, los tipos de datos, las opciones de partición, las claves de partición, etc. Ajusta las configuraciones según sea necesario para adaptarse a tu caso de uso.

- **Definir las transformaciones de datos.** Define las transformaciones de datos necesarias para limpiar, normalizar, enriquecer y manipular los datos según tus requisitos. Utiliza las herramientas visuales de Glue Studio para diseñar las transformaciones de manera eficiente y efectiva.

- **Guardar y ejecutar el flujo de trabajo.** Una vez que hayas configurado tu flujo de trabajo, guarda tus cambios y ejecuta el flujo de trabajo para procesar los datos. Glue Studio te permite monitorear y gestionar el progreso de la ejecución del flujo de trabajo en tiempo real.

- **Monitorear y depurar.** Monitorea el progreso de la ejecución del flujo de trabajo y realiza cualquier depuración necesaria para solucionar problemas o mejorar el rendimiento. Glue Studio proporciona herramientas integradas para monitorear y depurar flujos de trabajo en tiempo real.

- **Guardar y desplegar el proyecto.** Una vez que hayas completado la creación y la ejecución del flujo de trabajo, guarda tus cambios y despliega el proyecto para hacerlo disponible para su uso en producción. Glue Studio te permite gestionar versiones y controlar los cambios a lo largo del tiempo.

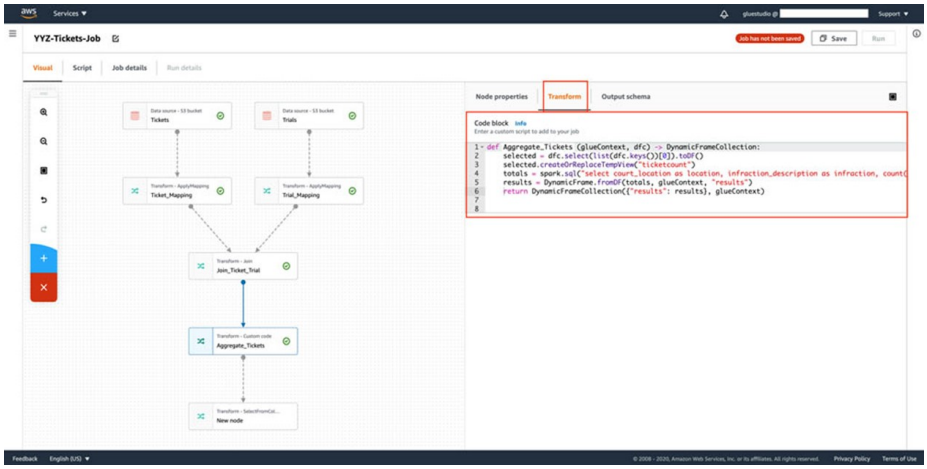
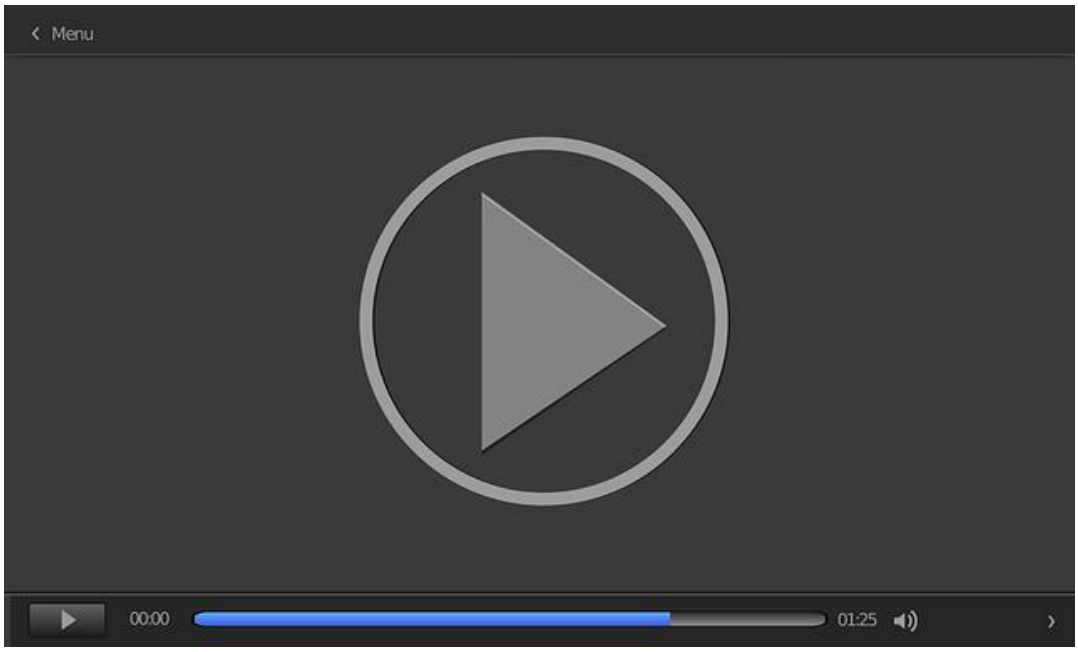



Figura 7. Creación de un trabajo con AWS Glue Studio. Fuente: elaboración propia.

# Tema 2. Ingesta de datos para big data e inteligencia artificial en cloud

El siguiente vídeo, *AWS Glue Studio - Visual data pipeline demo | Amazon Web Services* (Amazon Web Services, 2023a), muestra como realizar un *pipeline* de ETL con AWS Studio.




AWS Glue Studio - Visual data pipeline demo | Amazon Web Services.

Accede al vídeo:

https://www.youtube.com/embed/ckxwnd4BQmk

AWS Glue Studio permite crear trabajos de forma interactiva en una interfaz de cuaderno basada en **cuadernos o *notebooks* de Jupyter.** A través de los cuadernos en AWS Glue Studio, es posible editar *scripts* de trabajos y el código de integración de datos y ver el resultado sin que sea necesario ejecutar un trabajo completo. También es posible agregar un marcado y guardar cuaderno como archivos .ipynb y *scripts* de trabajo. Puede iniciar un cuaderno sin instalar ningún *software* en forma local ni administrar servidores. Una vez que esté satisfecho con el código, AWS Glue Studio puede convertir el cuaderno en un trabajo de Glue con solo hacer clic en un botón.

Algunos de los **beneficios** de utilizar los cuadernos son los siguientes:

- No hay clúster que aprovisionar o administrar.
- No hay que pagar por clústeres inactivos.
- No se requiere una configuración inicial.
- No se requiere instalación de *notebooks* de Jupyter.
- Mismo tiempo de ejecución y plataforma que ETL de AWS Glue.

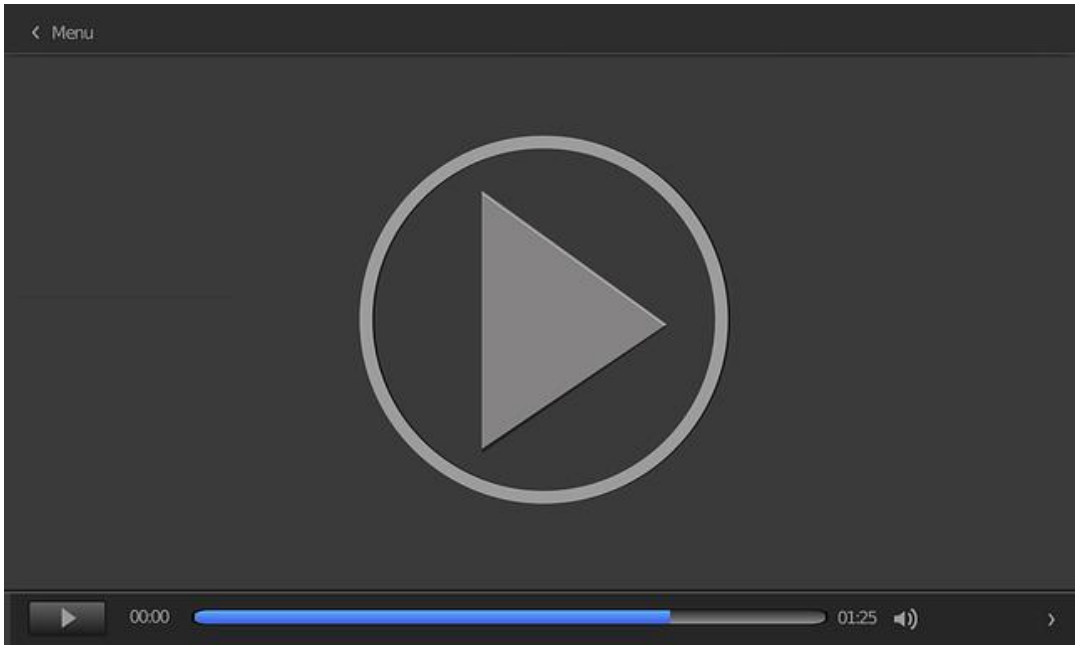
Al iniciar un cuaderno a través de AWS Glue Studio, todos los **pasos de configuración** ya han sido completados para que, apenas después de unos segundos, pueda explorar los datos y comenzar a desarrollar el *script* de trabajo. AWS Glue Studio configura un cuaderno de Jupyter con el *kernel* de Jupyter de AWS Glue. No es necesario configurar VPC (nube privada virtual), conexiones de red ni puntos de conexión de desarrollo para utilizar este cuaderno.

Para crear **trabajos** mediante la interfaz de cuaderno, se deben seguir los siguientes pasos:

- Configure los permisos de IAM (administración de identidad y acceso) necesarios.
- Inicie una sesión de cuaderno para crear un trabajo.
- Escriba código en las celdas en el cuaderno.
- Ejecute y pruebe el código para ver el resultado.
- Guarde el trabajo.

Una vez guardado, el cuaderno es un **trabajo completo** de AWS Glue. Puede administrar todos los aspectos del trabajo, tales como la programación de ejecuciones de trabajos, la configuración de parámetros del trabajo y la visualización del historial de ejecuciones de trabajos justo al lado del cuaderno.

En el vídeo *AWS Tutorials - Interactively Develop Glue Job using Jupyter Notebook* (AWS Tutorials, 2022), se profundiza sobre los pasos de creación y ejecución de un *notebook* con AWS Glue.




AWS Tutorials - Interactively Develop Glue Job using Jupyter Notebook.

Accede al vídeo:

https://www.youtube.com/embed/ckxwnd4BQmk

AWS Glue proporciona **motores de integración de datos** que son fundamentales para el procesamiento y la transformación de datos en la plataforma de AWS:

- El motor de Apache Spark.
- AWS Glue para Ray.

- AWS Glue para Python Shell.

En el caso de **AWS Glue con motor de Apache Spark,** AWS Glue brinda una infraestructura optimizada para el rendimiento y sin servidor destinada a la ejecución de Apache Spark para llevar a cabo trabajos de integración y extracción, transformación y carga (ETL) de datos. AWS Glue para Apache Spark admite el procesamiento de lotes y transmisiones, y acelera la ingesta, el procesamiento y la integración de datos. Puede crear y actualizar su lago de datos y el almacenamiento de datos, y extraer información de los datos con más rapidez.

Las **características clave** de este motor son las siguientes:

- **Procesamiento distribuido.** Apache Spark es un motor de procesamiento de datos distribuido que permite procesar grandes volúmenes de datos de manera eficiente y escalable.

- **Lenguajes de programación.** Soporta varios lenguajes de programación, incluidos Python, Scala, Java y SQL, lo que les permite a los usuarios implementar lógica de transformación compleja utilizando el lenguaje de su elección.

- **Optimización de consultas.** Utiliza un optimizador de consultas avanzado que mejora el rendimiento de las operaciones de ETL al planificar y ejecutar consultas de manera eficiente.

- **Capacidad de paralelización.** Permite la ejecución de operaciones en paralelo en clústeres de computación distribuida, lo que acelera el procesamiento de datos y reduce los tiempos de ejecución.

- **Integración con servicios de AWS.** Se integra estrechamente con otros servicios de AWS, como S3, Redshift, RDS, y más, lo que facilita el acceso a datos y la realización de análisis avanzados.

Por otro lado, **AWS Glue para Ray** es una opción de motor de integración de datos en AWS Glue, ya está disponible para el público en general. AWS Glue para Ray ayuda a los ingenieros de datos y a los desarrolladores de ETL a escalar sus trabajos de Python. AWS Glue es un servicio de integración de datos escalable y sin servidor que se utiliza para descubrir, preparar, trasladar e integrar datos de varias fuentes. AWS Glue para Ray combina esa capacidad sin servidor para la integración de datos con Ray (ray.io), un nuevo y conocido marco informático de código abierto que lo ayuda a escalar las cargas de trabajo de Python.

Al igual que en los motores Apache Spark y Python de AWS Glue, solo paga por los **recursos** que utiliza al ejecutar el código y no necesita configurar ni ajustar los recursos. AWS Glue para Ray facilita el **procesamiento** distribuido de su código Python en clústeres de varios nodos. Puede crear y ejecutar trabajos de Ray en cualquier lugar donde pueda ejecutar trabajos de ETL de AWS Glue. Esto incluye trabajos existentes de AWS Glue, interfaces de línea de comandos (CLI) y API. Puede seleccionar el motor AWS Glue para Ray de forma local o a través de cuadernos en AWS Glue Studio y el cuaderno de Amazon SageMaker Studio. Cuando el trabajo de Ray esté listo, puede ejecutarlo bajo demanda o según un cronograma.

Para trabajar con AWS Glue para Ray, se utilizan los mismos trabajos de AWS Glue y las mismas sesiones interactivas que se usan con AWS Glue para Spark. Los **trabajos** de AWS Glue están diseñados para ejecutar el mismo *script* de forma periódica, mientras que las **sesiones** interactivas están diseñadas para permitir ejecutar fragmentos de código de forma secuencial con los mismos recursos aprovisionados.

En la página «Trabajos» de la consola de AWS Glue Studio, se puede seleccionar una nueva opción al crear un trabajo en el editor de scripts AWS Glue Studio Ray tal como podemos ver en la Figura 8.

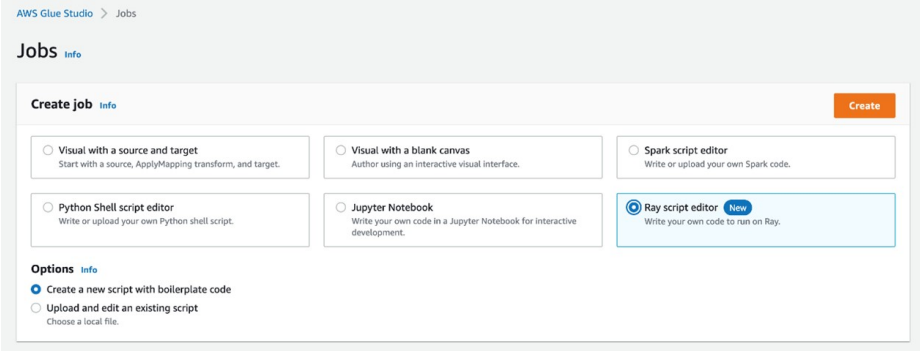



Figura 8. Crear un trabajo en AWS Glue Studio. Fuente: elaboración propia.

Por último, con **AWS Glue para Python Shell,** se puede utilizar un trabajo de Python Shell para ejecutar *scripts* de Python en AWS Glue. Mediante estos trabajos, es posible escribir trabajos de análisis y de integración de datos complejos en Python. Los trabajos de AWS Glue para Python Shell ahora ofrecen bibliotecas de análisis comunes listas para usar, incluidas Pandas, NumPy y Amazon SageMaker Data Wrangler. Puede usar la funcionalidad integrada para conectarse a una gran variedad de bases de datos, almacenamientos de datos y servicios de AWS.

Un **trabajo de *shell*** de Python puede usarse para ejecutar *scripts* de Python como un *shell* en AWS Glue. Con un trabajo de *shell* de Python puede ejecutar *scripts* compatibles con Python 2.7, Python 3.6 o Python 3.9.

Existen dos **alternativas** para utilizar AWS Glue Python para *shell*: con AWS Glue Studio o mediante el CLI de AWS. A continuación, se explica el funcionamiento de ambas.

Al definir su **trabajo de *shell* de Python en AWS Glue Studio,** debe proporcionar algunas de las siguientes propiedades:

- **Rol de IAM.**

- **Especificación del rol de AWS IAM,** que se usa para dar una autorización a los recursos que se utilizan para ejecutar el flujo de trabajo y obtener acceso a los almacenes de datos. Para obtener más información acerca de los permisos para ejecutar trabajos en AWS Glue, puedes consultar la administración de identidades y accesos para AWS Glue.

- **Tipo.** Elija Python *shell* (*shell* de Python) para ejecutar un *script* de Python con el comando de trabajo llamado pythonshell .

- **Versión de Python.** Elija la versión de Python. El valor por defecto es Python 3.9. Las versiones válidas son Python 3.6 y Python 3.9.

- **Cargar bibliotecas de análisis comunes** (recomendado). Elija esta opción para incluir bibliotecas comunes para Python 3.9 en el *shell* de Python. Si sus bibliotecas son personalizadas o entran en conflicto con las preinstaladas, puede optar por no instalar bibliotecas comunes. Sin embargo, puede instalar bibliotecas adicionales además de las bibliotecas comunes. Si selecciona esta opción, la opción «library-set» se establece en «analytics». Al anular la selección de esta opción, la opción «library-set» se establece en «none».

- **Nombre de archivo de *script* y ruta de *script***. El código del *script* define la lógica de procedimiento del trabajo. Proporcione el nombre del *script* y la ubicación en Amazon S3. Compruebe que no haya un archivo con el mismo nombre que el directorio de *script* en la ruta. Para obtener más información acerca de cómo usar *scripts*, puedes consultar la guía de programación de AWS Glue.

- ***Script.*** El código del *script* define la lógica de procedimiento del trabajo. Puede codificar el *script* en Python 3.6 o Python 3.9. Puedes editar *scripts* en AWS Glue Studio.

- **Unidades de procesamiento de datos.** Es el número máximo de unidades de procesamiento de datos (DPU) de AWS Glue que se pueden asignar cuando se ejecute este trabajo. Una DPU es una medida relativa de la potencia de

procesamiento que consta de 4 vCPU de capacidad de cómputo y 16 GB de memoria —para más información, puedes consultar los precios de AWS Glue—. Puede establecer el valor en 0,0625 o 1. El valor predeterminado es 0,0625. En cualquier caso, el disco local para la instancia será de 20 GB.

Respecto al **uso de CLI en AWS Glue para Python Shell,** también se puede crear un trabajo de *shell* de Python con la AWS CLI, como en el siguiente ejemplo:

```
aws glue create-job --name python-job-cli --role Glue_DefaultRole
--command '{"Name" : "pythonshell", "PythonVersion": "3.9",
"ScriptLocation" : "s3://DOC-EXAMPLE-BUCKET/scriptname.py"}'
 --max-capacity 0.0625
```

**Orquestación en AWS Glue**

En AWS Glue, puedes iniciar trabajos (*jobs*) y *crawlers* mediante desencadenadores. Los **desencadenadores** son eventos que activan la ejecución de un trabajo o un *crawler* en respuesta a ciertas condiciones predefinidas. A continuación, tienes una descripción del funcionamiento para iniciar trabajos y *crawlers* en AWS Glue mediante desencadenadores:

- **Desencadenadores de AWS Glue:**

- **Desencadenadores de trabajo** (*job triggers*). Se utilizan para iniciar la ejecución de un trabajo de AWS Glue en respuesta a un evento específico. Puedes configurar desencadenadores para que se activen en horarios programados, en función de cambios en los datos, o en respuesta a eventos externos.

- **Desencadenadores de *crawler*** (*crawler triggers*). Se utilizan para iniciar la ejecución de un *crawler* de AWS Glue en respuesta a un evento específico. Puedes configurar desencadenadores para que se activen en horarios programados, en función de cambios en los metadatos de los datos, o en respuesta a eventos externos.

- **Configuración de desencadenadores:**

  - **Horarios programados.** Puedes configurar desencadenadores para que se activen en horarios programados, como cada día a una hora específica o cada hora. Esto te permite automatizar la ejecución de trabajos y *crawlers* en intervalos regulares.

  - **Eventos de cambio de datos.** Los desencadenadores también se pueden configurar para que se activen en respuesta a cambios en los datos. Por ejemplo, puedes configurar un desencadenador para que inicie un trabajo cada vez que se agreguen nuevos datos a un *bucket* de Amazon S3.

  - **Eventos externos.** Además, puedes configurar desencadenadores para que se activen en respuesta a eventos externos, como notificaciones de Amazon CloudWatch, cambios en un repositorio de código fuente o mensajes en una cola de Amazon SQS.

- **Proceso de ejecución.** Una vez que se activa un desencadenador, AWS Glue inicia la ejecución del trabajo o *crawler* asociado. Durante la ejecución, AWS Glue provisiona automáticamente la infraestructura necesaria, como instancias de procesamiento y almacenamiento, para realizar las tareas especificadas en el trabajo o *crawler*.

- **Monitoreo y gestión.** AWS Glue proporciona herramientas integradas para monitorear y gestionar la ejecución de trabajos y *crawlers*. Puedes ver el estado de ejecución en tiempo real, monitorear el progreso, detectar y solucionar problemas, y optimizar el rendimiento según sea necesario.

- **Gestión de versiones y auditoría.** AWS Glue también proporciona capacidades para gestionar versiones de desencadenadores y realizar auditorías de ejecuciones anteriores. Puedes ver un historial de ejecuciones, revisar los registros de actividad y realizar un seguimiento de los cambios en la configuración de los desencadenadores a lo largo del tiempo.

En resumen, los **desencadenadores** en AWS Glue te permiten automatizar la ejecución de trabajos y *crawlers* en respuesta a eventos específicos, lo que simplifica la administración y la ejecución de tareas de procesamiento y preparación de datos en la nube.

## Uso de flujos de trabajo en AWS Glue

En AWS Glue se pueden utilizar **flujos de trabajo** para crear y visualizar actividades de ETL complejas que implican varios rastreadores, trabajos y desencadenadores. Cada flujo de trabajo administra la ejecución y el monitoreo de todos sus trabajos y rastreadores. Un flujo de trabajo registra el progreso de ejecución y el estado, ya que ejecuta cada componente. Esto le proporciona información general de la tarea de mayor envergadura y los detalles de cada paso. La consola de AWS Glue ofrece una representación visual de un flujo de trabajo en forma de gráfico.

### Propiedades de ejecución del flujo de trabajo

Para compartir y administrar el estado en toda la ejecución de flujo de trabajo, se pueden definir **propiedades de ejecución** de flujo de trabajo predeterminadas. Estas propiedades, que son pares nombre-valor, están disponibles para todos los trabajos en el flujo de trabajo. Puedes utilizar la AWS Glue API para recuperar las propiedades de ejecución de flujo de trabajo y es necesario modificarlas para los trabajos que vengan después en el flujo de trabajo.

### Gráfico del flujo de trabajo

La Figura 9 muestra el gráfico de un flujo de trabajo básico en la consola de AWS Glue. Su flujo de trabajo podría tener decenas de componentes.

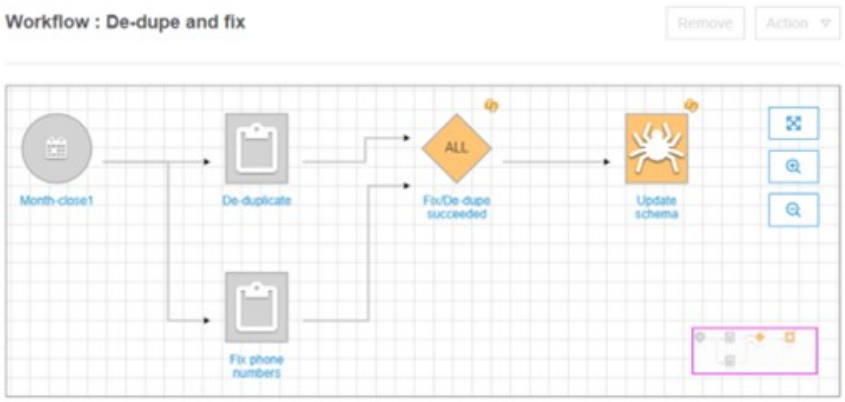



Figura 9. Flujo de trabajo. Fuente: Amazon Web Service, s. f.-b.

Este **flujo de trabajo** se inicia mediante un desencadenador de programación, Month-close1 , que inicia dos trabajos, De-duplicate y Fix phone numbers . Tras la realización completa de ambos trabajos, un desencadenador de eventos, Fix/De-dupe succeeded , inicia un rastreador, Update schema .

**Configurar e implementar un flujo de trabajo en Glue**

Se puede usar la consola de AWS Glue para crear y construir un flujo de trabajo y un nodo a la vez y de forma manual. Un flujo de trabajo contiene trabajos, rastreadores y desencadenadores. Antes de crear un **flujo de trabajo** de forma manual, hay que crear los trabajos y los rastreadores que este vaya a incluir. Es mejor especificar los rastreadores de ejecución bajo demanda para los flujos de trabajo. Se pueden crear nuevos desencadenadores mientras se desarrolla su flujo de trabajo o puede clonar desencadenadores existentes en el flujo de trabajo. Al clonar un **desencadenador,** se agregan al flujo de trabajo todos los objetos del catálogo asociados a este (los trabajos o los rastreadores que lo activan y los trabajos o los rastreadores que lo inician).

Se puede diseñar un propio flujo de trabajo añadiendo desencadenadores al gráfico de flujo de trabajo y definiendo los eventos vistos y las acciones para cada desencadenador. Puedes comenzar con un **desencadenador de arranque,** que puede ser un desencadenador bajo demanda o programado, y completar el gráfico añadiendo desencadenadores de eventos (condicional). Los pasos se desarrollarán a continuación.

**Paso 1. Crear el flujo de trabajo**

- Inicia sesión en la AWS Management Console y abre la consola de AWS Glue en https://console.aws.amazon.com/glue/. En el panel de navegación, en ETL, elige «Workflows» (‘flujos de trabajo’).
- Selecciona «Add workflow» (‘añadir flujo de trabajo’) y completa el formulario «Add a new ETL workflow» (‘añadir un nuevo flujo de trabajo de ETL’). Las propiedades de ejecución predeterminadas opcionales que añadas estarán disponibles como argumentos para todos los trabajos en el flujo de trabajo. Para obtener más información, consulta obtención y configuración de propiedades de ejecución de flujo de trabajo en AWS Glue.
- Selecciona «Add workflow».El nuevo flujo de trabajo aparecerá en la lista en la página «Workflows».

**Paso 2. Añadir un desencadenador de arranque**

- En la página «Workflows», selecciona el nuevo flujo de trabajo. A continuación, en la parte inferior de la página, asegúrate de seleccionar la pestaña «Graph» (‘gráfico’).
- Selecciona «Add trigger» (‘añadir desencadenador’) y en el cuadro de diálogo «Add trigger», realiza uno de los procedimientos detallados en la Tabla 2.

| Procedimientos disponibles | | |
|---|---|---|
| A | Elige «Clone existing» (‘clonación existente’) y seleccione un desencadenador a clonar. A continuación, elige «Add» (‘añadir’). | El desencadenador aparece en el gráfico, junto con los trabajos y los rastreadores que ve e inicia.<br>Si has seleccionado por error el desencadenador erróneo, selecciona el desencadenador en el gráfico y, a continuación, elige «Remove» (‘eliminar’). |
| B | Elige «Add new» (‘añadir nuevo’) y completa el formulario «Add trigger». | Para «Trigger type» (‘tipo de desencadenador’), selecciona «Schedule» (‘programación’), «On demand» (‘bajo demanda’) o «EventBridge event» (‘evento de EventBridge’).<br>▸ Para el tipo de desencadenador «Schedule», elige una de las opciones en «Frequency» (‘frecuencia’). Selecciona «Custom» (‘personalizado’) e ingresa una expresión cron.<br>▸ Para el tipo de desencadenador «EventBridge», ingresa «Number of events» (‘número de eventos’) —tamaño del lote—, y, opcionalmente, ingresa «Time delay» (‘tiempo de retraso’) —ventana por lotes—. Si omites «Time delay», la ventana por lotes tiene un valor predeterminado de quince minutos. |
| C | Elige «Add». El desencadenador aparece en el gráfico junto con un nodo de marcador de posición con la etiqueta «Add node» (‘añadir nodo’). En el ejemplo de a continuación, el desencadenador de inicio es un desencadenador de programación denominado `Month-close1`. En este momento, el desencadenador no está guardado aún. | --- |

Tabla 2. Procedimientos. Fuente: elaboración propia.

> Un gráfico con dos nodos rectangulares: un desencadenador y un nodo de marcador de posición. Una flecha señala desde el nodo del desencadenador hasta el nodo de marcador de posición.

Si has añadido un nuevo desencadenador, realiza una de las **acciones** siguientes:

- Elige el nodo del marcador de posición «Add node».

- Asegúrate de que el desencadenador de arranque esté seleccionado y en el menú «Action» (‘acción’) de arriba del gráfico, elige «Add jobs/crawlers to trigger» (‘añadir trabajos/rastreadores al desencadenador’).

- En el cuadro de diálogo «Add jobs(s) and crawler(s) to trigger» (‘añadir trabajos y rastreadores al desencadenador’), selecciona uno o más trabajos, o desencadenadores y, a continuación, elige «Add».

E l **desencadenador** se guarda y los trabajos y los rastreadores seleccionados aparecen en el gráfico con los conectores del desencadenador. Si has añadido por error los trabajos y los rastreadores erróneos, puedes seleccionar el desencadenador o un conector y elegir «Remove».

**Paso 3. Agregar más desencadenadores**

Continúa construyendo su flujo de trabajo al agregar más **desencadenadores** de tipo «Event» (‘evento’). Para ampliar o reducir la imagen, o para agrandar el lienzo del gráfico, utiliza los íconos a la derecha del gráfico. Para agregar los desencadenadores, realice los pasos que se menciones a continuación. En primer lugar, realice una de las acciones siguientes:

- Para clonar un desencadenador existente, asegúrese de que no se selecciona ningún nodo en el gráfico y, en el menú «Action» elige «Add trigger».

- Para añadir un nuevo desencadenador que vea un trabajo o rastreador determinado en el gráfico, selecciona el nodo de trabajo o rastreador y, a continuación, elige el nodo de marcador de posición «Add trigger».

---

Puedes añadir más trabajos o rastreadores para ver este desencadenador en un paso posterior.

---

En el cuadro de diálogo «Add trigger», realiza una de las siguientes acciones:

- Elige «Add new» y completa el formulario «Add trigger». A continuación, elige «Add».
  - El desencadenador aparecerá en el gráfico. Se completará el desencadenador en un paso posterior.
- Elige «Clone existing» (‘clonación existente’) y selecciona un desencadenador a clonar. A continuación, elige «Add».
  - El desencadenador aparecerá en el gráfico, junto con los trabajos y los rastreadores que ve e inicia.
- Si has elegido por error el desencadenador erróneo, selecciona el desencadenador en el gráfico y, a continuación, elige «Remove».

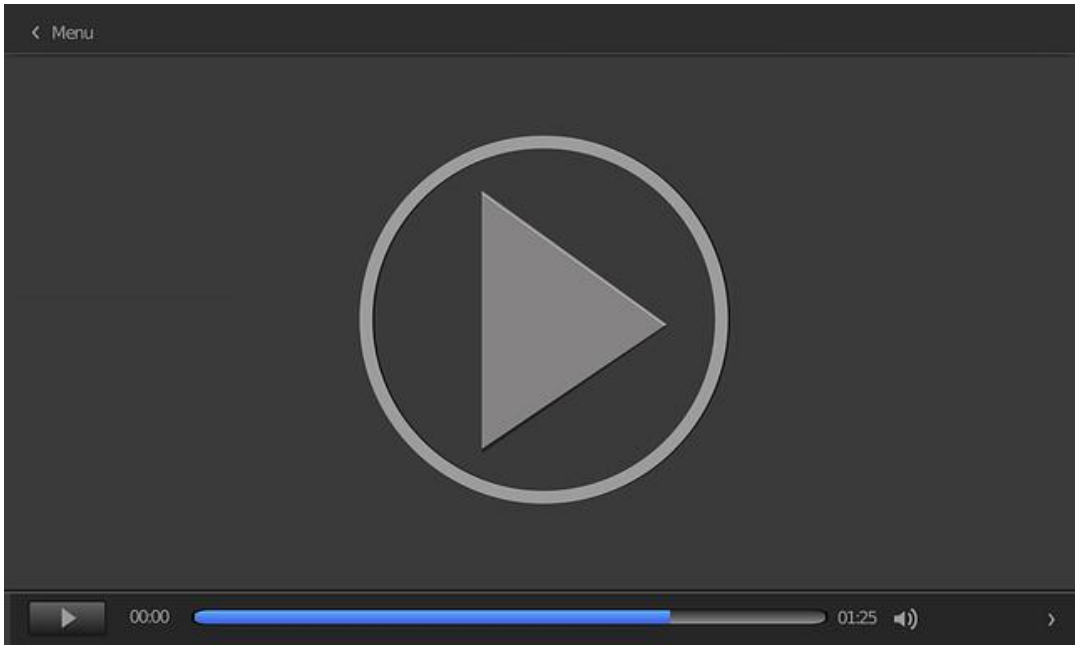
Si ha añadido un nuevo desencadenador, complete estos pasos:

- **Seleccione el nuevo desencadenador.** Se selecciona el desencadenador «De-dupe/fix succeeded», y los nodos de marcador de posición aparecen para eventos para ver (número uno) y acciones (número dos). Resulta en un gráfico con muchos nodos, dos de los cuales son nodos de marcador de posición, que se denominan números uno y dos.
- Este paso es **opcional** si el desencadenador ya ve un evento y desea añadir más trabajos o rastreadores que ver. Elige el nodo de marcador de posición de eventos que ver y, en el cuadro de diálogo «Add job(s) and crawler(s) to watch», selecciona uno o más trabajos o rastreadores. Elige un evento que ver («SUCCEEDED» —‘correcto’—, «FAILED» —‘erróneo’—, etc.) y haz clic en «Add».
  - Asegúrate de que se seleccione el desencadenador y elige el nodo de marcador de posición de acciones.

- En el cuadro de diálogo «Add job(s) and crawler(s) to watch», selecciona uno o más trabajos o desencadenadores y, a continuación, elige «Add».

- Los trabajos y los rastreadores seleccionados aparecerán en el gráfico con los conectores del desencadenador.

A continuación, en el vídeo *Automated ETL Workflow Orchestration with AWS Glue, Athena, Lambda, EventBridge, and Step Functions* (Wall Street Mindset, 2024), se muestra un ejemplo de orquestación con AWS Glue, más otros servicios de AWS.




Automated ETL Workflow Orchestration with AWS Glue, Athena, Lambda, EventBridge, and Step Functions.

Accede al vídeo:

https://www.youtube.com/embed/Olt0XekklhU

**Integración con otros servicios de AWS**

Glue se integra estrechamente con otros servicios de AWS, como S3, Redshift y Athena, lo que facilita la carga de datos en almacenes de datos y la realización de análisis avanzados. Esta integración permite que los usuarios aprovechen una amplia gama de herramientas y servicios de AWS para el **análisis de datos,** lo que les permite obtener *insights* valiosos de sus datos de manera eficiente y escalable.

En resumen, estos son los principales **servicios de AWS Glue** que proporcionan una plataforma completa y escalable para la preparación y el procesamiento de datos en la nube. Con sus capacidades de descubrimiento automático de esquemas, transformaciones ETL flexibles, programación de trabajos automatizada y sólida integración con otros servicios de AWS, Glue les permite a las organizaciones acelerar el tiempo de obtención de información y obtener *insights* valiosos de sus datos.

# Tema 2. Ingesta de datos para big data e inteligencia artificial en cloud

## 2.3. Ingesta de datos en streaming

La **ingesta de datos** en *streaming* es el proceso de capturar, procesar y almacenar datos en tiempo real a medida que se generan. A diferencia de la ingesta de datos tradicional, que puede implicar la recopilación y el procesamiento de grandes volúmenes de datos estáticos, la ingesta de datos en *streaming* se centra en el flujo continuo de datos que proviene de diversas fuentes, como sensores, dispositivos IoT, aplicaciones web, redes sociales y sistemas de monitoreo, como podemos ver en la Figura 10.


Figura 10. Fuentes. Fuente: Qlik, s. f.

# Tema 2. Ingesta de datos para big data e inteligencia artificial en cloud

**Componentes de la ingesta de datos en *streaming***

Los principales componentes de la ingesta de datos en *streaming* son los siguientes:

- **Fuente de datos.** La fuente de datos es el origen de los datos en *streaming*. Puede ser cualquier dispositivo o sistema que genere datos en tiempo real, como sensores, dispositivos IoT, aplicaciones móviles, redes sociales, sistemas de monitoreo de servidores, entre otros.

- ***Broker* de mensajes.** El *broker* de mensajes es un componente clave en la ingesta de datos en *streaming*. Actúa como intermediario entre las fuentes de datos y los consumidores de datos, recibiendo los datos entrantes y distribuyéndolos a los procesadores o los sistemas de almacenamiento correspondientes.

- **Procesamiento de datos.** Los datos en *streaming* suelen ser procesados en tiempo real para realizar diversas acciones, como filtrado, transformación, enriquecimiento, agregación o análisis. Esto puede involucrar el uso de herramientas y plataformas de procesamiento de datos en *streaming*, como Apache Kafka, Apache Flink, Apache Storm, AWS Kinesis, o Google Cloud Dataflow.

- **Almacenamiento de datos.** Los datos procesados en *streaming* pueden ser almacenados en sistemas de almacenamiento de datos en tiempo real o en almacenes de datos tradicionales. Esto permite que los datos estén disponibles para consultas y análisis posteriores. Algunas opciones comunes de almacenamiento incluyen Apache Cassandra, Amazon S3, Google Cloud Storage, y bases de datos en memoria, como Redis o Memcached.

- **Consumidores de datos.** Los consumidores de datos son los sistemas o las aplicaciones que utilizan los datos procesados en *streaming* para realizar acciones específicas, como generar informes, activar alertas, tomar decisiones en tiempo real o alimentar modelos de aprendizaje automático. Estos consumidores pueden ser aplicaciones web, servicios de análisis, sistemas de monitoreo o cualquier otra entidad que necesite acceso a los datos en tiempo real.

# Tema 2. Ingesta de datos para big data e inteligencia artificial en cloud

## Desafíos de la ingesta de datos en streaming

- **Latencia.** Reducir la latencia en la ingesta de datos es fundamental para garantizar que los datos estén disponibles para procesamiento y análisis en tiempo real.
- **Escalabilidad.** Los sistemas de ingesta de datos en *streaming* deben ser capaces de escalar para manejar grandes volúmenes de datos y picos de tráfico.
- **Tolerancia a fallos.** Es importante diseñar sistemas de ingesta de datos en *streaming* que sean robustos y tolerantes a fallos, para garantizar la disponibilidad y la integridad de los datos.
- **Integridad de los datos.** Mantener la integridad de los datos en un entorno de *streaming* puede ser un desafío debido a la naturaleza dinámica y continua de los datos.
- **Seguridad.** Garantizar la seguridad de los datos en un entorno de *streaming* es fundamental para proteger la privacidad y la confidencialidad de la información.

La Figura 11 muestra un conjunto de bloques de desafíos para tener en cuenta en un proceso en *streaming.*

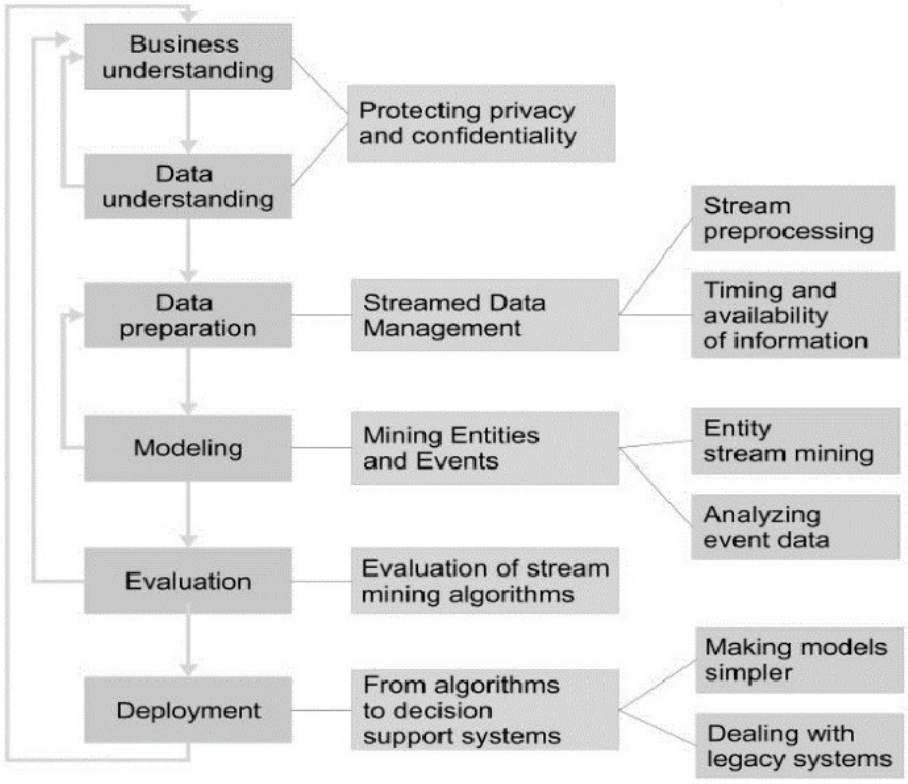



Figura 11. Conjunto de bloques de desafíos. Fuente: Saad et al., 2021.

En resumen, la **ingesta de datos en *streaming*** es un componente fundamental en los sistemas modernos de análisis de datos en tiempo real. Permite a las organizaciones capturar, procesar y actuar sobre los datos en tiempo real, lo que les permite tomar decisiones más informadas y rápidas en un mundo cada vez más conectado y dinámico.

## ¿Qué son los datos de transmisión?

Los **datos de *streaming*** son datos emitidos a un alto volumen de manera gradual y continua con el objetivo de lograr un procesamiento de baja latencia. Las organizaciones poseen cientos de orígenes de datos que suelen emitir simultáneamente mensajes, registros o datos, con un tamaño que va desde unos pocos bytes a varios megabytes (MB).

# Tema 2. Ingesta de datos para big data e inteligencia artificial en cloud

Los datos de *streaming* incluyen **datos de sensor, eventos y ubicación** que las empresas usan para lograr un análisis en tiempo real y la visibilidad de muchos aspectos de su negocio. Por ejemplo, las empresas pueden rastrear cambios en la opinión pública sobre sus marcas y productos al analizar de manera continua las secuencias de clic y las publicaciones de clientes en flujos de redes sociales y responder rápidamente según sea necesario.

**¿Cuáles son las características de los datos de *streaming*?**

Un flujo de datos posee las siguientes características específicas que lo definen:

- **Importante a nivel cronológico.** Los elementos individuales en una secuencia de datos contienen marcas de tiempo. La secuencia de datos en sí misma puede ser urgente, con una importancia disminuida después de un intervalo de tiempo específico. Por ejemplo, su aplicación realiza recomendaciones de restaurantes de acuerdo con la ubicación actual del usuario. Tiene que responder a los datos de geolocalización del usuario en tiempo real o los datos perderán importancia.

- **De flujo continuo.** Una secuencia de datos no tiene principio ni final. Recopila datos de manera continua y constante mientras sea necesario. Por ejemplo, los registros de actividad del servidor se acumulan mientras el servidor esté en ejecución.

- **Única.** Repetir la transmisión de una secuencia de datos es todo un desafío debido a la urgencia. Por lo tanto, el procesamiento de datos preciso y en tiempo real es fundamental. Lamentablemente, las provisiones para la retransmisión son limitadas en la mayoría de los orígenes de datos de *streaming*.

- **No homogénea.** Algunos orígenes pueden transmitir datos en varios formatos estructurados como JSON, Avro y valores separados por comas (CSV) con tipos de datos que incluyen cadenas, números, fechas y tipos binarios. Sus sistemas de procesamiento de secuencias deben tener las capacidades para manejar tales variaciones en los datos.

- **Imperfecta.** Los errores temporales en el origen pueden resultar en elementos dañados o faltantes en los datos transmitidos. Puede ser difícil garantizar la coherencia de datos debido a la naturaleza continua de la transmisión. Los sistemas de análisis y procesamiento de secuencias suelen incluir lógica para la validación de datos a fin de mitigar o minimizar esos errores.

**¿Por qué son importantes los datos de *streaming*?**

Los sistemas tradicionales de procesamiento de datos capturan datos en un almacenamiento de datos central y los procesan en grupos o lotes. Estos sistemas fueron creados para capturar y estructurar datos antes del análisis. Sin embargo, en los últimos años, la **naturaleza de los datos** empresariales y los **sistemas de procesamiento** de datos subyacentes han cambiado de manera significativa.

- **Volumen de datos infinito.** Los volúmenes de datos generados de orígenes de secuencias pueden ser muy grandes, lo cual se traduce en un desafío para el análisis en tiempo real: la regulación de la integridad (validación), la estructura (evolución) o la velocidad (rendimiento y latencia) de los datos de *streaming*.

- **Sistemas de procesamiento de datos avanzados.** Al mismo tiempo, la infraestructura en la nube ha introducido la flexibilidad en la escala y el uso de recursos informáticos. Usa exactamente lo que necesita y paga solo por lo que utiliza. Tiene las opciones de agregación o filtros en tiempo real tanto antes como después de almacenar datos de *streaming*. La arquitectura de datos de *streaming* utiliza tecnologías en la nube para consumir, enriquecer, analizar y almacenar de forma permanente datos de *streaming* según sea necesario.

**¿Cuáles son los casos de uso para el *streaming* de datos?**

El sistema de procesamiento de secuencias resulta beneficioso en la mayoría de las situaciones en las que se generan **datos nuevos y dinámicos** de forma continua. Aplica a la mayoría de los sectores y casos de uso de macrodatos.

# Tema 2. Ingesta de datos para big data e inteligencia artificial en cloud

Por lo general, las empresas comienzan con aplicaciones sencillas, como la recopilación de registros del sistema y el procesamiento rudimentario, como la implementación de cálculos mínimos y máximos. Más adelante, estas **aplicaciones** evolucionan y se pasa al procesamiento más sofisticado casi en tiempo real.

Otros **ejemplos de datos** de *streaming* serían:

- **Análisis de datos.** Las aplicaciones procesan secuencias de datos para producir informes y realizar acciones como respuesta, por ejemplo, emitir alertas cuando las medidas clave superen ciertos umbrales. Las aplicaciones de procesamiento de secuencias más sofisticadas extraen información más profunda mediante la aplicación de algoritmos de ML a los datos de actividad de los clientes y la empresa.

- **Aplicaciones de IoT.** Los dispositivos de IoT son otro caso de uso para los datos de *streaming*. Los sensores en vehículos, equipo industrial y maquinaria agrícola envían datos a una aplicación de *streaming*. La aplicación supervisa el rendimiento, detecta posibles defectos de forma anticipada y envía el pedido de un recambio automáticamente, lo que evita el tiempo de inactividad del equipo.

- **Análisis financiero.** Las instituciones financieras usan datos de secuencias para controlar los cambios en la bolsa en tiempo real, procesar el valor en riesgo y modificar las carteras automáticamente en función de los cambios en los precios de las acciones. Otro caso de uso financiero es la detección de fraudes en transacciones con tarjetas de crédito mediante la inferencia en tiempo real frente a datos de transacción de *streaming*.

- **Recomendaciones en tiempo real.** Las aplicaciones de bienes raíces rastrean datos de geolocalización de los dispositivos móviles de los consumidores y realizan recomendaciones en tiempo real de propiedades para visitar. De manera similar, las aplicaciones de consumidores, ventas, alimentos y publicidades pueden integrar recomendaciones en tiempo real para brindar más valor a los clientes.

- **Garantías de servicio.** Puede implementar el procesamiento de secuencias de datos para rastrear y mantener niveles de servicios en las aplicaciones y el equipo. Por ejemplo, una compañía de energía solar debe mantener constante el suministro de electricidad a sus clientes, ya que si no lo hace se la sanciona. Implementa una aplicación de datos de *streaming* que supervise todos los paneles en el campo y programe el servicio en tiempo real. Por lo tanto, puede minimizar los períodos de bajo rendimiento de cada panel y los pagos de penalidades asociados.

- **Multimedia y videojuegos.** Los editores de medios transmiten miles de millones de registros de secuencias de clic de sus propiedades en línea, agregan y enriquecen los datos con información demográfica del usuario y optimizan la colocación de contenido. Esto ayuda a los editores a ofrecer al público una experiencia mejor y más relevante. De manera similar, las empresas de videojuegos en línea utilizan el procesamiento de secuencias de eventos para analizar las interacciones entre el jugador y el juego, y ofrecer experiencias dinámicas para involucrar a los jugadores.

- **Control de riesgos.** Las transmisiones en directo y las plataformas de redes sociales capturan datos del comportamiento del usuario en tiempo real para controlar riesgos en la actividad financiera de los usuarios, como las recargas, los reintegros y las recompensas. Visualizan paneles en tiempo real para ajustar las estrategias de riesgos de manera flexible.

En la Figura 12, podemos ver un *overview* de casos de uso de *streaming*.