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

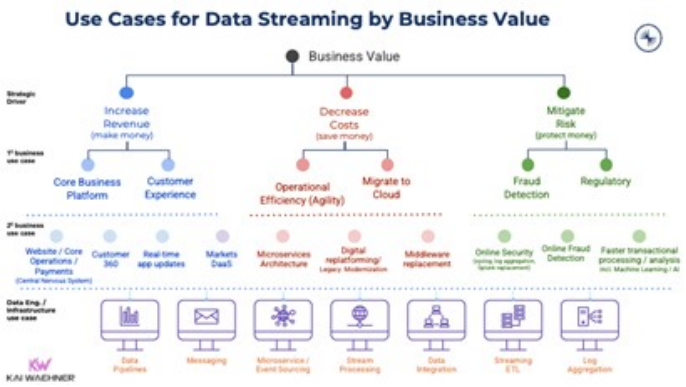


Figura 12. *Overview* de casos de uso de *streaming*. Fuente: Waehner, 2022.

¿Cuál es la diferencia entre los datos de lotes y los datos de *streaming*?

E l **procesamiento por lotes** es el método que utilizan las computadoras para completar trabajos de datos repetitivos y de gran volumen de forma periódica. Puede ser utilizado para calcular consultas arbitrarias en diferentes conjuntos de datos. Por lo general, deriva los resultados informáticos de todos los datos que abarca y permite un análisis profundo de conjuntos de macrodatos. Los sistemas basados en MapReduce, como Amazon EMR, son ejemplos de plataformas compatibles con los trabajos por lotes.

Por el contrario, el **procesamiento de secuencias** requiere capturar una secuencia de datos y actualizar de forma gradual las métricas, los informes y las estadísticas de resumen en respuesta a cada registro de datos que llega. Es más adecuado para las funciones de respuesta y análisis en tiempo real.

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

AWS Kinesis

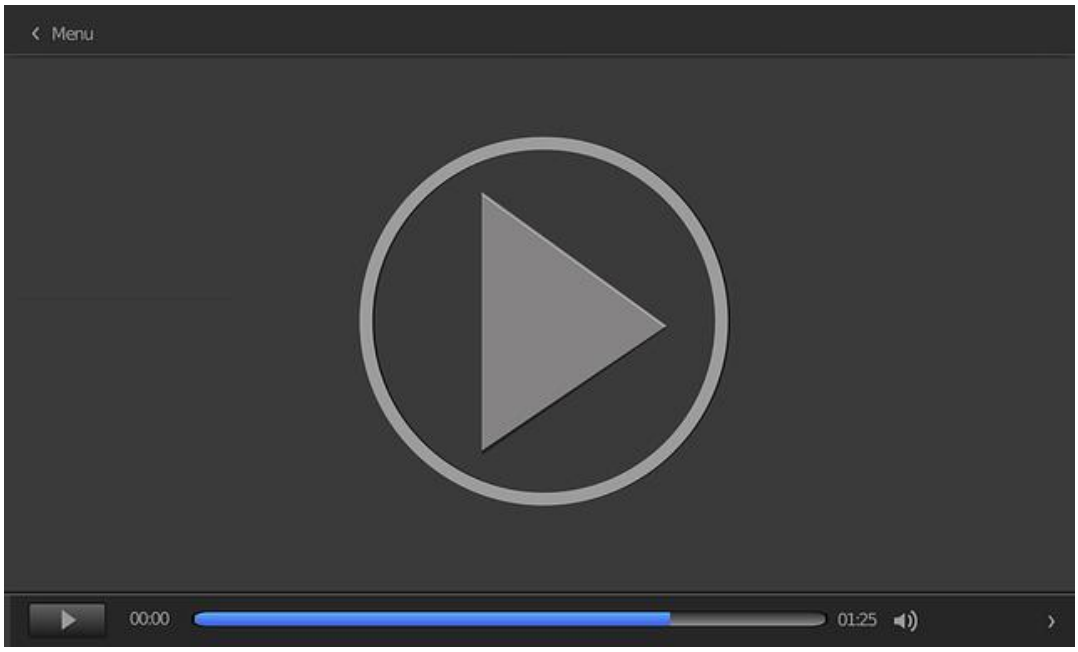
AWS Kinesis es un servicio de *streaming* de datos completamente administrado que permite que las empresas procesen y analicen grandes volúmenes de datos en tiempo real de manera eficiente y escalable. Diseñado para aplicaciones que requieren el procesamiento de datos en tiempo real, Kinesis facilita la ingesta, el almacenamiento, el procesamiento y la entrega de datos en *streaming* de forma rápida y confiable.

A continuación, tienes una descripción más detallada de los **componentes y características** clave de AWS Kinesis:

- **Kinesis Data Streams.** Kinesis Data Streams es el componente principal de AWS Kinesis. Permite a los usuarios ingresas datos en tiempo real a través de productores (como aplicaciones, dispositivos IoT, servidores de registros, etc.) y luego procesar esos datos utilizando consumidores (aplicaciones, análisis en tiempo real, sistemas de almacenamiento, etc.). Los datos se dividen en fragmentos (*shards*) para permitir la escalabilidad horizontal y la distribución de la carga.

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

A continuación, se muestra el vídeo *Amazon Kinesis Data Streams Fundamentals* (Amazon Web Services, 2020), que es una introducción sobre AWS Kinesis Data Streams.



Amazon Kinesis Data Streams Fundamentals.

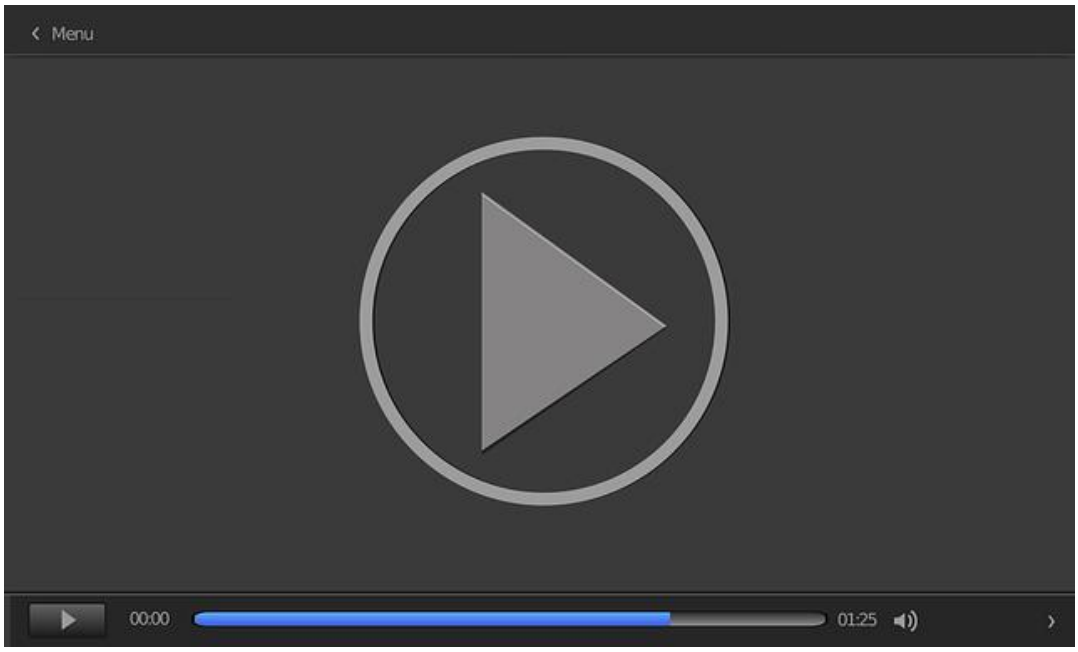
Accede al vídeo:

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

- **Kinesis Data Firehose.** Kinesis Data Firehose es un servicio que permite la carga directa y continua de datos en Amazon S3, Amazon Redshift, Amazon Elasticsearch Service y Splunk para su posterior análisis. Elimina la complejidad de administrar infraestructura y reduce el esfuerzo requerido para procesar y almacenar datos en tiempo real.

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

A continuación, se muestra el vídeo *Introduction to Kinesis Data Firehose | Amazon Web Services* (Amazon Web Services, 2023b), que es una introducción sobre AWS Kinesis Firehose.



Introduction to Kinesis Data Firehose | Amazon Web Services.

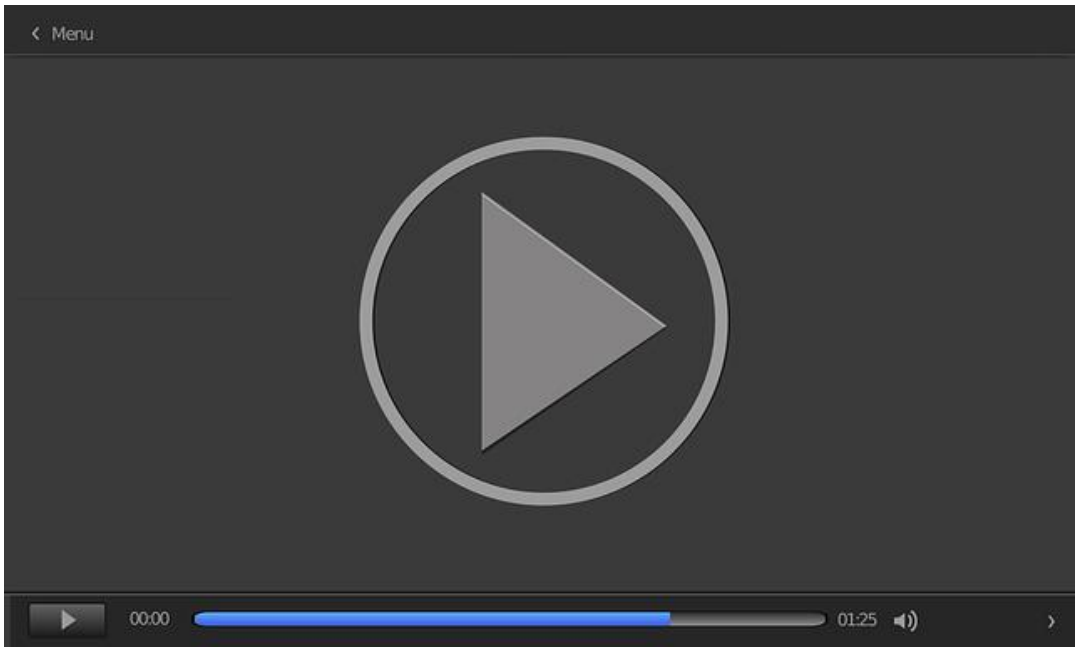
Accede al vídeo:

https://www.youtube.com/embed/qRoyF9dEqgw

- **AWS Flink Administrador** (antiguo Kinesis Data Analitycs). Kinesis Data Analytics es un servicio que les permite a los usuarios realizar análisis en tiempo real de datos en *streaming* utilizando SQL estándar, Java o Scala a través del poderoso motor de procesamiento de Flink.

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

A continuación, se muestra el vídeo *Introduction to Amazon Managed Service for Apache Flink | Amazon Web Services* (Amazon Web Services, 2023c), es una introducción sobre Amazon Flink Adminnistrador.



Introduction to Amazon Managed Service for Apache Flink | Amazon Web Services.

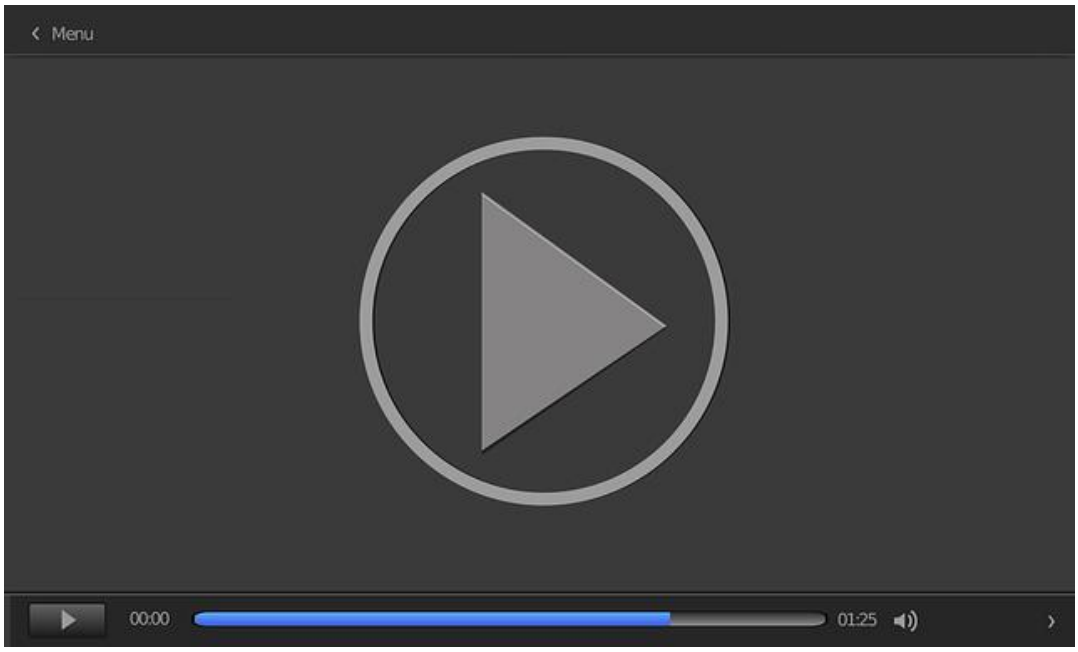
Accede al vídeo:

https://www.youtube.com/embed/vI1GiMSHuxM

- **Kinesis Vídeo Streams.** Kinesis Vídeo Streams es un servicio para la captura, procesamiento y almacenamiento de secuencias de vídeo en tiempo real. Permite a los usuarios transmitir secuencias de vídeo a escala desde dispositivos como cámaras IP, drones, cámaras de vigilancia y dispositivos móviles, y, luego, procesar y analizar esas secuencias utilizando otros servicios de AWS.

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

A continuación, se muestra el vídeo *AWS re:Invent 2017 - Introducing Amazon Kinesis Vídeo Streams* (Amazon Web Services, 2023d), que es una introducción sobre AWS Kinesis Vídeo Streams.



AWS re:Invent 2017 - Introducing Amazon Kinesis Vídeo Streams.

Accede al vídeo:

https://www.youtube.com/embed/STEMa3t5NOQ

Las **características** clave de AWS Kinesis:

- **Escalabilidad y durabilidad.** Kinesis Data Streams es altamente escalable y duradero, lo que permite manejar grandes volúmenes de datos en tiempo real y proporcionar una alta disponibilidad y tolerancia a fallos.
- **Bajo latencia.** Kinesis Data Streams ofrece baja latencia para la ingesta y la entrega de datos, lo que les permite a los usuarios procesar datos en tiempo real y tomar decisiones instantáneas.
- **Integración con servicios de AWS.** Kinesis se integra estrechamente con otros servicios de AWS, como Amazon S3, Amazon Redshift, Amazon Elasticsearch

Service y AWS Lambda, lo que facilita el procesamiento y análisis de datos en tiempo real utilizando la infraestructura de AWS.

- **Facilidad de uso.** Kinesis proporciona una interfaz de usuario intuitiva y API para facilitar la configuración, administración y supervisión de flujos de datos en tiempo real.

- **Seguridad y cumplimiento normativo.** Kinesis ofrece opciones de seguridad avanzadas, como cifrado de datos en tránsito y en reposo, control de acceso basado en roles (IAM) y cumplimiento de normativas como HIPAA y GDPR.

En resumen, **AWS Kinesis** es una solución completa y escalable para el procesamiento y análisis de datos en tiempo real. Con sus componentes y características clave, les permite a las empresas capturar, procesar y analizar datos en *streaming* de manera eficiente y confiable, lo que les permite tomar decisiones más informadas y rápidas en un mundo cada vez más conectado y dinámico.

Arquitectura de AWS Kinesis

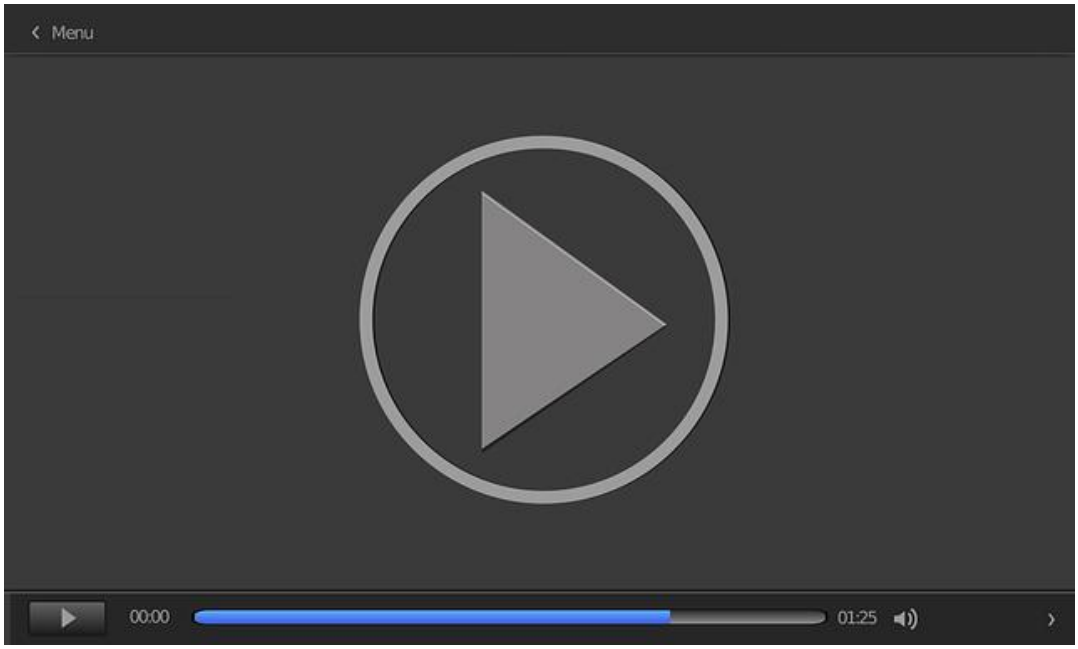
La **arquitectura de AWS Kinesis** está diseñada para facilitar la ingesta, el procesamiento y el análisis de datos en tiempo real a escala. La arquitectura de Kinesis se compone de varios componentes que trabajan juntos para permitir el flujo continuo de datos en *streaming* y su posterior procesamiento. A continuación, tienes una descripción de la arquitectura típica de AWS Kinesis:

- **Kinesis Data Streams.** Es el componente central de la arquitectura de Kinesis. Permite a los usuarios capturar y procesar datos en tiempo real mediante la ingestión de datos en fragmentos (*shards*) distribuidos. Cada fragmento es una unidad de capacidad de procesamiento que puede manejar una cantidad específica de datos. Los datos ingresados en un *stream* se almacenan temporalmente en los fragmentos y se pueden procesar utilizando consumidores de datos.

- **Productores y consumidores de datos.** Los productores son las fuentes de datos que ingresan datos en un *stream* de Kinesis. Pueden ser aplicaciones, dispositivos IoT, servidores de registros, sensores, etc. Los productores envían datos al *stream* de Kinesis utilizando la API de Kinesis.

Los consumidores son las **aplicaciones** o los **sistemas** que procesan y analizan los datos en *streaming*. Pueden ser aplicaciones de análisis en tiempo real, sistemas de almacenamiento, servicios de procesamiento de datos, etc. Los consumidores obtienen acceso a los datos del *stream* de Kinesis utilizando la API de Kinesis o integrándose directamente con otros servicios de AWS.

A continuación, en el vídeo *Data Consumer Options for Amazon Kinesis Data Streams | Amazon Web Services* (Amazon Web Services, 2023e), se habla sobre las posibilidades de consumidores de datos dentro de Kinesis Data Streams.



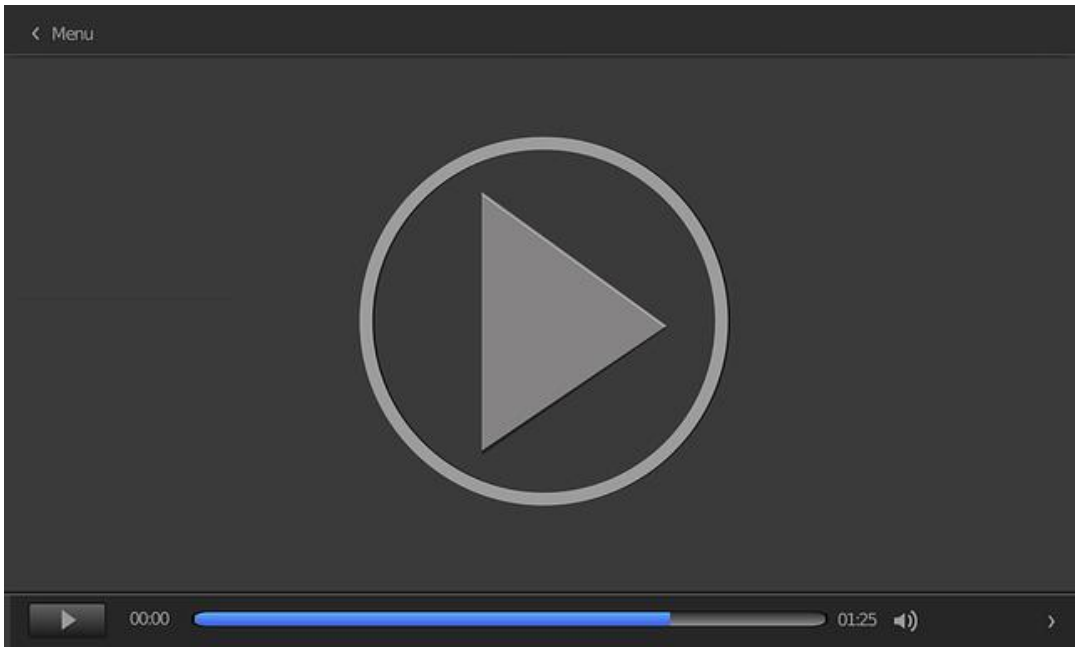
Data Consumer Options for Amazon Kinesis Data Streams | Amazon Web Services.

Accede al vídeo:

https://www.youtube.com/embed/STEMa3t5NOQ

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

A continuación, en el vídeo *Data Producer Options for Amazon Kinesis Data Streams | Amazon Web Services* (Amazon Web Services, 2023f), se habla sobre las posibilidades de productores de datos dentro de Kinesis Data Streams.



Data Producer Options for Amazon Kinesis Data Streams | Amazon Web Services.

Accede al vídeo:

https://www.youtube.com/embed/tmhCRGV0XOM

- **Kinesis Data Firehose.** Es un servicio opcional que simplifica la carga directa y continua de datos en tiempo real desde Kinesis Data Streams a otros servicios de AWS, como Amazon S3, Amazon Redshift, Amazon Elasticsearch Service y Splunk. Elimina la necesidad de administrar infraestructura adicional y proporciona una forma fácil de procesar y almacenar datos en tiempo real para análisis posteriores.

- **Kinesis Data Analytics.** Es otro servicio opcional que permite realizar análisis en tiempo real de datos en *streaming* utilizando SQL estándar. Facilita la ejecución de consultas SQL sobre datos en movimiento para extraer información valiosa y tomar decisiones instantáneas. Kinesis Data Analytics puede integrarse directamente con Kinesis Data Streams para procesar datos en tiempo real y generar resultados analíticos en tiempo real.

- **Integración con servicios de AWS.** Kinesis se integra estrechamente con otros servicios de AWS, como Amazon S3, Amazon Redshift, Amazon Elasticsearch Service, AWS Lambda y más. Esta integración permite que los usuarios utilicen la infraestructura de AWS para procesar y analizar datos en *streaming* de manera eficiente y escalable.

- **Seguridad y gestión de acceso.** Kinesis ofrece opciones avanzadas de seguridad y control de acceso, como cifrado de datos en tránsito y en reposo, control de acceso basado en roles (IAM), y políticas de acceso granular. Esto permite que los usuarios protejan los datos en *streaming* y garanticen el cumplimiento normativo.

En resumen, la **arquitectura de AWS Kinesis** está diseñada para facilitar la ingestión, el procesamiento y el análisis de datos en tiempo real a escala. Con sus componentes y características clave, Kinesis permite que las empresas capturen, procesen y analicen datos en *streaming* de manera eficiente y confiable, lo que les permite tomar decisiones más informadas y rápidas en un mundo cada vez más conectado y dinámico.

Figura 13. Amazon Kinesis Data Streams. Fuente: Amazon Web Services, s. f.-c.

Kinesis Data Streams

Kinesis Data Streams es un servicio de *streaming* de datos completamente administrado de AWS diseñado para facilitar la ingesta, el procesamiento y el análisis de grandes volúmenes de datos en tiempo real. Como parte de la plataforma de AWS, Kinesis Data Streams les permite a las empresas capturar datos en *streamin*g desde diversas fuentes, como aplicaciones, dispositivos IoT, servidores de registros y más, y, luego, procesar y analizar esos datos de manera eficiente y escalable.

Con Kinesis Data Streams, las empresas pueden construir **aplicaciones de análisis en tiempo real,** que les permitan tomar decisiones basadas en datos en tiempo real. El servicio está diseñado para ser altamente escalable, duradero y confiable, lo que lo hace adecuado para una amplia gama de casos de uso, como análisis de clic en sitios web, procesamiento de registros de servidor, monitoreo de flujos de sensores, análisis de registros de aplicaciones y mucho más.

Al aprovechar Kinesis Data Streams, las empresas pueden procesar y analizar grandes **volúmenes de datos** en tiempo real utilizando herramientas y servicios familiares de AWS, como Amazon S3, Amazon Redshift, Amazon Elasticsearch Service y más. Además, el servicio ofrece opciones avanzadas de seguridad y control de acceso, como cifrado de datos en tránsito y en reposo, control de acceso

basado en roles (IAM) y políticas de acceso granular, lo que garantiza la protección de los datos en *streaming* y el cumplimiento normativo.

En resumen, **Kinesis Data Streams** es una solución poderosa y versátil para la ingestión, procesamiento y análisis de datos en tiempo real en la nube de AWS. Con su capacidad para capturar y procesar grandes volúmenes de datos en tiempo real, Kinesis Data Streams les permite a las empresas obtener información valiosa y tomar decisiones informadas de manera más rápida y eficiente en un mundo cada vez más conectado y dinámico.

Características de Kinesis Data Streams

Amazon Kinesis Data Streams ofrece una serie de características que lo convierten en una opción poderosa y versátil para la ingestión, procesamiento y análisis de datos en tiempo real. A continuación, tienes algunas de las características clave de AWS Kinesis Data Streams:

- **Escalabilidad horizontal.** Kinesis Data Streams puede manejar grandes volúmenes de datos y cargas de trabajo escalando horizontalmente a través de la adición o eliminación de *shards* según sea necesario.
- **Baja latencia.** Ofrece baja latencia para la ingestión y la recuperación de datos, lo que permite el procesamiento y el análisis de datos en tiempo real con tiempos de respuesta rápidos.
- **Durabilidad.** Los datos ingresados en un *stream* de Kinesis se almacenan de forma duradera en el servicio, lo que garantiza que no se pierdan incluso en caso de fallos o interrupciones.
- **Procesamiento paralelo.** Los datos ingresados en un *stream* se dividen en fragmentos (*shards*), lo que permite el procesamiento paralelo de datos y una mayor capacidad de procesamiento.

- **Integración con servicios de AWS.** Kinesis Data Streams se integra estrechamente con otros servicios de AWS, como Amazon S3, Amazon Redshift, Amazon Elasticsearch Service y más, lo que facilita el procesamiento y el análisis de datos en tiempo real utilizando la infraestructura de AWS.

- **Control de acceso y seguridad.** Ofrece opciones avanzadas de seguridad y control de acceso, como cifrado de datos en tránsito y en reposo, control de acceso basado en roles (IAM) y políticas de acceso granular, lo que garantiza la protección de los datos en *streaming* y el cumplimiento normativo.

- **Monitoreo y gestión.** Kinesis Data Streams proporciona herramientas y métricas para monitorear y gestionar el rendimiento y la salud de los *streams*, lo que facilita la detección y la resolución de problemas de manera proactiva.

- **Coste eficiente.** Con una estructura de precios basada en el número de *shards* y el volumen de datos procesados, Kinesis Data Streams ofrece una opción costo-eficiente para la ingestión y el procesamiento de datos en tiempo real.

Estas **características** hacen de Amazon Kinesis Data Streams una opción popular para una amplia gama de aplicaciones, desde análisis de clic en tiempo real hasta procesamiento de registros de servidor y monitoreo de flujos de sensores. Permite a las empresas capturar, procesar y analizar datos en tiempo real de manera eficiente y escalable, lo que les permite tomar decisiones más informadas y rápidas en un mundo cada vez más conectado y dinámico.

Componentes principales de Kinesis Data Streams

Los principales **componentes** de Amazon Kinesis Data Streams son:

- ***Stream.*** Un *stream* es el componente central de Kinesis Data Streams. Representa un flujo continuo de datos en tiempo real y actúa como el punto de entrada para los datos que ingresan al sistema. Los datos se dividen en fragmentos (*shards*) dentro del *stream* para permitir la escalabilidad horizontal y la distribución de la carga.

- ***Shard.*** Un *shard* es una unidad de capacidad de procesamiento en un *stream* de Kinesis. Cada *shard* tiene una capacidad de ingestión y recuperación de datos específica y puede manejar un volumen determinado de datos por segundo. Los datos ingresados en un *stream* se distribuyen entre los *shards* disponibles.

- **Productores.** Los productores son las fuentes de datos que ingresan datos en un *stream* de Kinesis. Pueden ser aplicaciones, dispositivos IoT, servidores de registros, sensores, etc. Los productores envían datos al *stream* de Kinesis utilizando la API de Kinesis.

- **Consumidores.** Los consumidores son las aplicaciones o los sistemas que procesan y analizan los datos en *streaming*. Pueden ser aplicaciones de análisis en tiempo real, sistemas de almacenamiento, servicios de procesamiento de datos, etc. Los consumidores obtienen acceso a los datos del *stream* de Kinesis utilizando la API de Kinesis o integrándose directamente con otros servicios de AWS.

- **Particiones.** Dentro de cada *shard*, los datos se dividen en particiones. Estas particiones permiten el procesamiento paralelo de datos y ayudan a garantizar que los datos se procesen de manera eficiente y escalable.

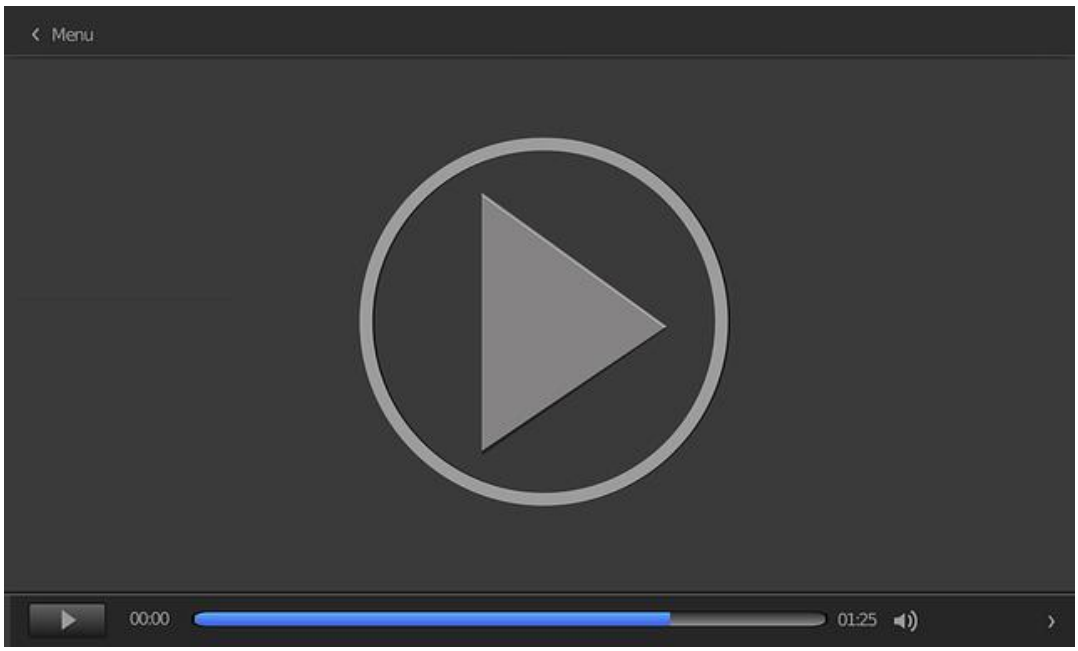
- **AWS Management Console.** La consola de administración de AWS proporciona una interfaz gráfica para configurar, monitorear y administrar *streams* de Kinesis. Permite a los usuarios crear, modificar y eliminar *streams*, así como supervisar el rendimiento y el estado de los *streams* existentes.

- **AWS SDK.** AWS proporciona SDK (*software development kits*) para varios lenguajes de programación que facilitan la integración de aplicaciones con Kinesis Data Streams. Estos SDK permiten que los desarrolladores interactúen con los *streams* de Kinesis y envíen y reciban datos de manera programática.

Estos **componentes** trabajan juntos para permitir la ingestión, el procesamiento y el análisis de datos en tiempo real en AWS utilizando Kinesis Data Streams. Los productores envían datos al *stream*, que se dividen en *shards* para su procesamiento

paralelo. Los consumidores acceden a los datos del *stream* para realizar análisis en tiempo real, almacenamiento o procesamiento adicional.

En el vídeo *Getting Started with Kinesis Data Streams | Amazon Web Services* (Amazon Web Services, 2023g) aprenderás a utilizar esta herramienta.



Getting Started with Kinesis Data Streams | Amazon Web Services.

Accede al vídeo:

https://www.youtube.com/embed/1I1DcJvmd4w

AWS Data Firehose (anterior Kinesis Data Firehose)

Amazon Kinesis Data Firehose es un servicio completamente administrado de AWS que facilita la carga directa y continua de datos en tiempo real desde múltiples fuentes a servicios de almacenamiento y análisis en la nube de AWS. Diseñado para simplificar el proceso de ingestión de datos en *streaming* y reducir la complejidad operativa, Kinesis Data Firehose automatiza gran parte del trabajo necesario para cargar, transformar y almacenar datos en tiempo real, permitiendo a las empresas enfocarse en el análisis y la generación de información valiosa a partir de sus datos.

A continuación, tendrás una descripción profunda de sus características y componentes principales.

Características principales

- **Carga automática de datos.** Kinesis Data Firehose permite cargar automáticamente datos en tiempo real desde múltiples fuentes, como *streams* de Kinesis, servicios de mensajería, como Amazon SQS, flujos de *log,* como Amazon CloudWatch Logs, y eventos de servicios, como AWS Lambda.

- **Transformación de datos.** El servicio ofrece opciones para transformar los datos en tiempo real antes de cargarlos en el destino final. Esto incluye transformaciones simples, como el cambio de formato, así como transformaciones más complejas utilizando AWS Lambda para filtrar, enriquecer o procesar los datos antes de cargarlos.

- **Integración con servicios de AWS.** Kinesis Data Firehose se integra estrechamente con otros servicios de AWS, como Amazon S3, Amazon Redshift, Amazon Elasticsearch Service y Splunk. Esto permite cargar datos en tiempo real en servicios de almacenamiento y análisis en la nube de AWS de manera eficiente y escalable.

- **Gestión automatizada de recursos.** Kinesis Data Firehose gestiona automáticamente los recursos subyacentes necesarios para la carga y el procesamiento de datos en tiempo real, lo que simplifica la administración y reduce la sobrecarga operativa.

- **Escalabilidad y durabilidad.** El servicio es altamente escalable y duradero, lo que permite manejar grandes volúmenes de datos y cargas de trabajo sin problemas. Los datos se almacenan de forma duradera en el destino final, garantizando la integridad y la disponibilidad de los datos.

- **Monitorización y métricas.** Kinesis Data Firehose proporciona métricas y registros detallados para monitorear el rendimiento y la salud de los flujos de datos en tiempo real, lo que facilita la detección y resolución de problemas.

Componentes principales

- ***Delivery stream.*** Un *delivery stream* es el componente central de Kinesis Data Firehose. Representa el flujo continuo de datos en tiempo real que se carga en un destino específico, como Amazon S3 o Amazon Redshift.

- **Productores y fuentes de datos.** Los productores son las fuentes de datos que envían datos al *delivery stream* de Kinesis Data Firehose. Pueden ser *streams* de Kinesis, servicios de mensajería, flujos de log, eventos de servicios, etc.

- **Destinos.** Los destinos son los servicios de almacenamiento y análisis en la nube de AWS donde se cargan los datos en tiempo real. Estos pueden incluir Amazon S3 para almacenamiento a largo plazo, Amazon Redshift para análisis de datos, Amazon Elasticsearch Service para búsqueda y análisis de registros, y Splunk para análisis de registros y seguridad.

Casos de uso

- **Análisis de datos en tiempo real.** Kinesis Data Firehose permite cargar datos en tiempo real en servicios de análisis como Amazon Redshift y Amazon Elasticsearch Service para realizar análisis y generación de informes en tiempo real.

- **Almacenamiento de datos en tiempo real.** El servicio facilita la carga de datos en tiempo real en Amazon S3 para almacenamiento a largo plazo y análisis posterior.

- **Procesamiento de *logs*.** Kinesis Data Firehose puede utilizarse para cargar flujos de log en tiempo real desde servicios como CloudWatch Logs y enviarlos a destinos como Amazon S3 o Splunk para análisis y monitoreo.

En resumen, **Amazon Kinesis Data Firehose** es una solución poderosa y completamente administrada para la carga y procesamiento de datos en tiempo real en la nube de AWS. Con su capacidad para automatizar gran parte del trabajo necesario para cargar y transformar datos en tiempo real, Kinesis Data Firehose permite a las empresas capturar, almacenar y analizar datos de manera eficiente y escalable, lo que les permite obtener información valiosa y tomar decisiones informadas en tiempo real.

Funcionamiento

Amazon Data Firehose proporciona la forma más sencilla de adquirir, transformar y entregar secuencias de datos en cuestión de segundos a lagos de datos, almacenamientos de datos y servicios de análisis. Para utilizar Amazon Data Firehose, debe configurar una secuencia con un origen, un destino y las transformaciones necesarias. Amazon Data Firehose procesa continuamente la secuencia, la escala automáticamente en función de la cantidad de datos disponibles y la envía en cuestión de segundos.

- **Origen.** Seleccione el origen de su secuencia de datos, como un tema en Amazon Managed Streaming para Kafka (MSK), una secuencia en Kinesis Data Streams o escriba datos mediante la API Firehose Direct PUT. Amazon Data Firehose está integrado en más de veinte servicios de AWS, por lo que puede configurar una secuencia de orígenes, como registros de Amazon CloudWatch, registros de ACL web de AWS WAF, AWS Network Firewall Logs, Amazon SNS o AWS IoT.

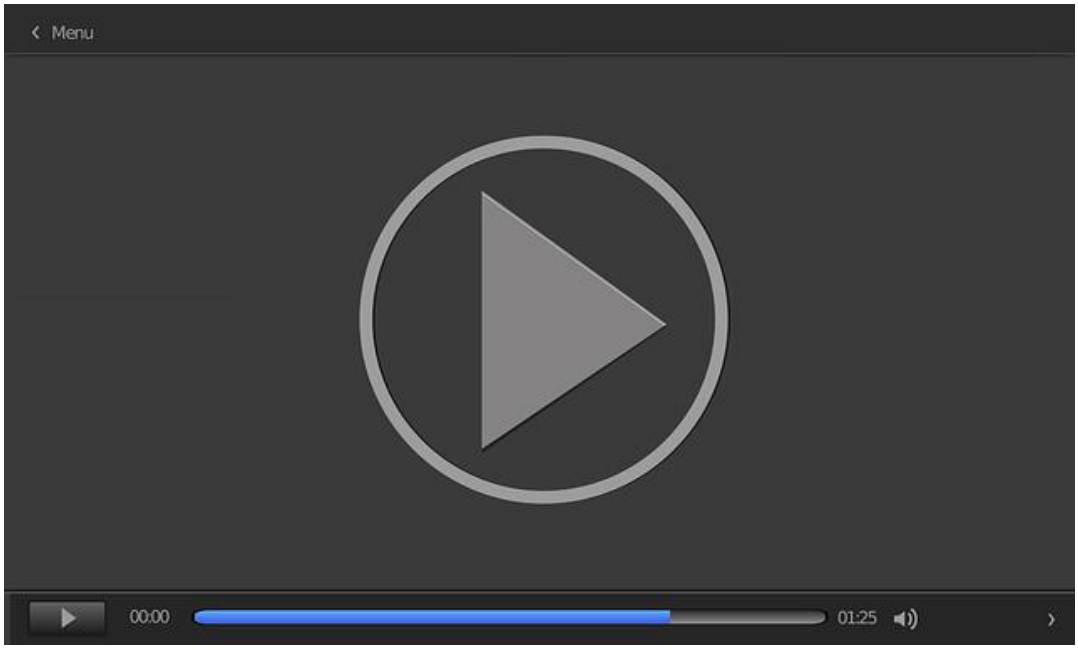
- **Transformación de datos (opcional).** Especifique si desea convertir la secuencia de datos en formatos como Parquet u ORC, descomprimir los datos, realizar transformaciones de datos personalizadas con la función de AWS Lambda o dividir dinámicamente los registros de entrada en función de los atributos para entregarlos en diferentes ubicaciones.

- **Destino.** Seleccione un destino para su secuencia, como Amazon S3, Amazon OpenSearch Service, Amazon Redshift, Splunk, Snowflake o un punto de conexión HTTP personalizado.

Figura 14. Amazon Data Firehose. Fuente: Amazon Web Service, s. f.-d.

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

En el vídeo *Introduction to Amazon Kinesis Firehose* (Amazon Web Services, 2016), se profundiza sobre el funcionamiento de Amazon Firehose.



Introduction to Amazon Kinesis Firehose.

Accede al vídeo:

https://www.youtube.com/embed/8L3lLSPPxpY

AWS Flink Administrado

Amazon Managed Service para Apache Flink es un servicio completamente administrado con el que puede procesar y analizar datos de transmisión mediante Java, Python, SQL o Scala. El servicio permite crear y ejecutar con rapidez código Java, SQL o Scala en orígenes de transmisión para realizar análisis de series temporales, alimentar paneles y crear métricas en tiempo real.

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

Managed Service para Apache Flink proporciona la **infraestructura** subyacente para sus aplicaciones de Apache Flink. Gestiona las capacidades principales, como el aprovisionamiento de recursos informáticos, la resiliencia de la conmutación por error de AZ, la computación paralela, el escalado automático y las copias de seguridad de las aplicaciones (implementadas como puntos de control e instantáneas).

Características de Apache Flink Administrado

- **Código abierto.** Amazon Managed Service para Apache Flink incluye bibliotecas de código abierto tales como Apache Flink, Apache Beam, Apache Zeppelin, SDK de AWS e integraciones de servicios de AWS. Apache Flink es un marco y motor que sirve para crear aplicaciones de *streaming* precisas y de alta disponibilidad. Apache Beam es un modelo unificado para definir aplicaciones de *streaming* y de procesamiento de datos por lotes que se ejecutan en varios motores de ejecución. El SDK de AWS elimina la complejidad de la codificación para muchos servicios de AWS al proporcionar API en su idioma preferido e incluye bibliotecas de AWS, ejemplos de código y documentación.

- **Apache Flink.** Apache Flink es un marco de procesamiento de datos de código abierto y distribuido diseñado para realizar análisis avanzados y procesamiento de datos en tiempo real y por lotes a gran escala. Es conocido por su capacidad para procesar datos de manera eficiente, con baja latencia y alta precisión, lo que lo hace adecuado para una amplia gama de casos de uso en entornos de *big data*. A continuación, tienes una descripción más detallada de sus características y componentes principales.

Características principales

- **Procesamiento de datos en tiempo real.** Flink es especialmente conocido por su capacidad para procesar datos en tiempo real con baja latencia. Permite a las empresas realizar análisis de datos en tiempo real y tomar decisiones basadas en eventos a medida que ocurren.

- **Procesamiento de datos por lotes.** Además del procesamiento de datos en tiempo real, Flink también admite el procesamiento de datos por lotes, lo que permite a las empresas realizar análisis retrospectivos y generar informes a partir de grandes volúmenes de datos históricos.

- **Modelo de programación rico.** Flink ofrece un modelo de programación rico y expresivo que permite a los desarrolladores escribir fácilmente aplicaciones complejas de procesamiento de datos. Soporta API en Java, Scala y Python.

- **Escalabilidad y tolerancia a fallos.** Flink está diseñado para ser altamente escalable y tolerante a fallos, lo que permite procesar grandes volúmenes de datos y manejar fallos de manera robusta. Puede ejecutarse en clústeres distribuidos de gran tamaño y escalar horizontalmente según sea necesario.

- **Soporte para *streams* y gráficos de datos.** Flink admite tanto el procesamiento de datos en *streaming* como el procesamiento de datos por lotes, lo que permite a las empresas analizar datos en tiempo real y procesar grandes volúmenes de datos históricos de manera eficiente.

- **Integración con ecosistema de *big data*.** Flink se integra con otros proyectos y tecnologías de *big data*, como Apache Hadoop, Apache Kafka, Apache HBase, Apache Cassandra y más, lo que facilita su incorporación en entornos existentes de *big data*.

Componentes principales

- **Flink Core.** El núcleo de Apache Flink proporciona las API y los motores de ejecución necesarios para procesar datos en tiempo real y por lotes. Incluye una amplia variedad de operadores y transformaciones para manipular y analizar datos de manera eficiente.

- **Flink Streaming.** Es una API de alto nivel que permite el procesamiento de datos en tiempo real utilizando flujos de datos continuos. Permite el procesamiento de datos en *streaming* con baja latencia y alta precisión.

- **Flink Batch.** Flink Batch es una API de alto nivel que permite el procesamiento de datos por lotes utilizando conjuntos de datos finitos. Permite el procesamiento de grandes volúmenes de datos históricos de manera eficiente y escalable.

- **Flink SQL.** Es una API que permite realizar consultas SQL sobre datos en *streaming* y por lotes. Facilita el análisis y la generación de informes utilizando lenguaje SQL estándar.

- **Flink ML.** Flink ML es una biblioteca de aprendizaje automático que permite realizar análisis predictivo y modelado de datos utilizando algoritmos de aprendizaje automático.

En resumen, **Apache Flink** es un marco de procesamiento de datos de código abierto y distribuido que ofrece capacidades avanzadas para realizar análisis en tiempo real y por lotes a gran escala. Con su rico modelo de programación, escalabilidad y tolerancia a fallos, Flink es ampliamente utilizado en entornos de *big data* para una variedad de casos de uso, desde análisis de clics en tiempo real hasta procesamiento de registros de servidor y análisis predictivo.

Apache Flink es un marco de procesamiento de datos de código abierto y distribuido que ha ganado una gran popularidad en el mundo de la **analítica de datos** en los últimos años. Se destaca por su capacidad para procesar datos tanto en tiempo real como en lotes, su alto rendimiento y su eficacia en el manejo de grandes volúmenes de datos.

Arquitectura de componentes

- **Cliente Flink (*Flink client*).** Es la interfaz de usuario y línea de comandos que permite a los usuarios enviar programas y comandos al clúster de Flink. A través del cliente, los desarrolladores pueden enviar y monitorear sus aplicaciones de Flink.

- **Clúster de Flink (*Flink cluster*).** El clúster de Flink es el entorno donde se ejecutan las aplicaciones de Flink. Está compuesto por un conjunto de nodos (o máquinas) que ejecutan el *software* de Flink y coordinan el procesamiento de datos.

- JobManager**.** El JobManager es el nodo maestro en el clúster de Flink y es responsable de coordinar y administrar las ejecuciones de los trabajos de Flink. Se encarga de recibir y programar los trabajos, coordinar la ejecución de las tareas y supervisar el progreso de los trabajos.

- TaskManager**.** Los TaskManagers son los nodos de trabajo en el clúster de Flink y son responsables de ejecutar las tareas individuales de los trabajos de Flink. Cada TaskManager ejecuta una o más instancias de tarea y se comunica con el JobManager para coordinar la ejecución de las tareas.

- **Tareas (*tasks*).** Las tareas son las unidades de procesamiento en Flink y representan las operaciones individuales que se aplican a los datos. Las tareas pueden ser operaciones de lectura, transformación o escritura, y se ejecutan de manera paralela en los TaskManagers .

- **Grafo de tareas (*task graph*).** El grafo de tareas es la representación interna de un programa de Flink y describe las dependencias entre las tareas. Está compuesto por nodos que representan las operaciones y los arcos que representan las comunicaciones entre las tareas.

- ***State backend.*** El *state backend* es el mecanismo que Flink utiliza para almacenar y administrar el estado de las aplicaciones de Flink. Puede ser configurado para utilizar diferentes sistemas de almacenamiento, como memoria, sistemas de archivos distribuidos (por ejemplo, HDFS) o bases de datos externas.

- ***Checkpointing* y *savepoints.*** Flink utiliza el *checkpointing* para garantizar la tolerancia a fallos y la recuperación de las aplicaciones en caso de fallo. Los *savepoints* son puntos de control manuales que permiten a los usuarios guardar el estado de una aplicación y reiniciarla desde ese punto en el futuro.

Flujo de ejecución en Flink

El proceso de ejecución de una aplicación de Flink sigue los siguientes pasos:

- **Despliegue de la aplicación.** El usuario despliega una aplicación de Flink utilizando el cliente de Flink, especificando el programa que desea ejecutar y las configuraciones de ejecución.

- **Inicio del** JobManager **y los** TaskManagers **.** Cuando se inicia la aplicación, se inicia un JobManager y uno o más TaskManagers en el clúster de Flink para ejecutar la aplicación.

- **Planificación y ejecución de tareas.** El JobManager recibe el programa de la aplicación y lo traduce en un grafo de tareas. Luego, planifica y coordina la ejecución de las tareas en los TaskManagers , asignando las tareas a los TaskManagers disponibles.

- **Ejecución de tareas.** Los TaskManagers ejecutan las tareas asignadas, procesando los datos de entrada, aplicando las operaciones definidas por el programa y enviando los resultados a las tareas subsiguientes.

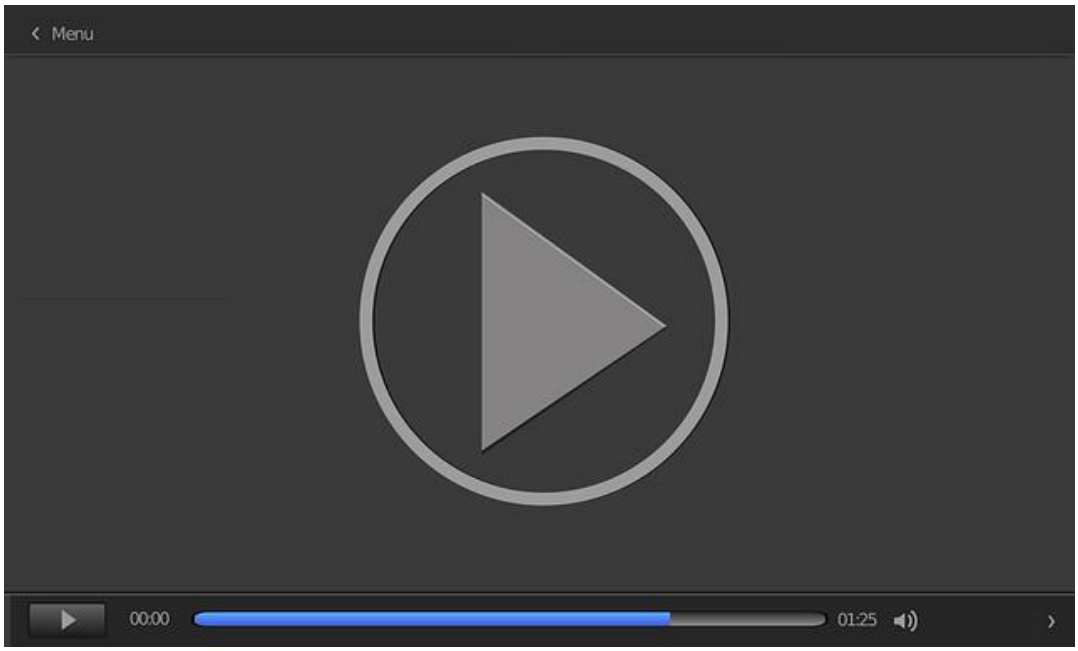
- ***Checkpointing* y tolerancia a fallos.** Durante la ejecución, Flink realiza *checkpointing* periódicamente para guardar el estado de la aplicación. En caso de fallo, Flink puede recuperar la aplicación desde el último punto de control para garantizar la tolerancia a fallos.

- **Finalización de la aplicación.** Una vez que se completan todas las tareas de la aplicación, Flink finaliza la ejecución y proporciona los resultados al usuario.

En resumen, **Apache Flink** ofrece una arquitectura robusta y escalable para el procesamiento de datos distribuidos en tiempo real y por lotes. Con su conjunto de componentes y su enfoque en la eficiencia y la tolerancia a fallos, Flink es ampliamente utilizado en una variedad de aplicaciones de *big data* y análisis de datos en tiempo real.

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

En el vídeo *One-Click Streaming with Amazon Managed Service For Apache Flink Blueprints | Amazon Web Services* (Amazon Web Services, 2023h), se explica como utiliza los *blueprints* de AWS Flink Administrado, que son la forma más fácil de ponerse en marcha y comenzar con una canalización de *streaming* completa de un extremo a otro. Con solo hacer clic en un botón, se puede poner en marcha una fuente de *streaming*, una aplicación de procesamiento de *streaming* y comenzar a procesar estos datos. Los proyectos contienen *scripts* especializados de Amazon CloudFormation para lanzar la infraestructura modular en su cuenta.



One-Click Streaming with Amazon Managed Service For Apache Flink Blueprints | Amazon Web Services.

Accede al vídeo:

https://www.youtube.com/embed/mWTnArI8xCI

Casos de uso

Apache Flink se utiliza en una amplia variedad de casos de uso en diferentes industrias. Algunos ejemplos comunes incluyen:

- **Análisis en tiempo real de eventos de transmisión,** como análisis de clics en sitios web, detección de fraudes, monitoreo de aplicaciones y sistemas, y más.
- **Procesamiento de lotes de grandes volúmenes de datos históricos,** como análisis de registros de servidor, procesamiento de registros de máquinas, análisis de datos de transacciones y más.
- **Procesamiento de datos en tiempo real para aplicaciones de IoT,** como monitoreo y análisis de sensores, control de procesos industriales, análisis de telemetría y más.

En resumen, **Apache Flink** es un marco de procesamiento de datos potente y versátil que ofrece capacidades avanzadas para analizar datos en tiempo real y por lotes a gran escala. Con su rico modelo de programación, escalabilidad y tolerancia a fallos, Flink es ampliamente utilizado en una variedad de casos de uso en diferentes industrias para obtener información valiosa a partir de grandes volúmenes de datos.

Las API de Flink

Amazon Managed Service para Apache Flink es compatible con las **API flexibles de Flink** en Java, Scala, Python y SQL especializadas para diferentes casos de uso, incluido el procesamiento de eventos con estado, ETL de *streaming* y análisis en tiempo real. Con operadores prediseñados y capacidades de análisis, puede crear una aplicación de *streaming* de Apache Flink en cuestión de horas en lugar de meses y las bibliotecas son ampliables, por lo que puede realizar el procesamiento en tiempo real para varios casos de uso.

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

Apache Beam

Apache Beam proporciona un modelo de programación unificado para el procesamiento de datos distribuidos. Permite a los desarrolladores escribir código una vez utilizando un conjunto de API coherentes y expresivas, y, luego, ejecutar ese código en múltiples motores de ejecución de procesamiento de datos, como Apache Flink, Apache Spark, Google Cloud Dataflow y más. Esto hace que sea fácil desarrollar aplicaciones de procesamiento de datos portátiles que puedan ejecutarse en una variedad de entornos de procesamiento de datos sin necesidad de modificar el código.

Características clave de Apache Beam

- **Modelo de programación unificado.** Apache Beam ofrece un modelo de programación unificado que les permite a los desarrolladores escribir código una vez y ejecutarlo en múltiples motores de ejecución de procesamiento de datos. Esto simplifica el desarrollo de aplicaciones de procesamiento de datos y facilita la portabilidad entre diferentes entornos de ejecución.

- **API expresivas y cohesivas.** Beam proporciona un conjunto de API coherentes y expresivas para la manipulación de datos, que incluyen API para la lectura, la transformación y la escritura de datos. Estas API están diseñadas para ser intuitivas y fáciles de usar, lo que facilita el desarrollo de aplicaciones de procesamiento de datos complejas.

- **Portabilidad y flexibilidad.** Una de las principales ventajas de Apache Beam es su capacidad para ejecutar código en múltiples motores de ejecución de procesamiento de datos. Esto permite a los desarrolladores escribir aplicaciones de procesamiento de datos una vez y ejecutarlas en una variedad de entornos de procesamiento de datos sin necesidad de modificar el código.

- **Optimización automática.** Beam realiza la optimización automática de los programas de procesamiento de datos, lo que permite mejorar el rendimiento y la eficiencia de las aplicaciones. Utiliza técnicas como la fusión de operaciones, la partición de datos y la planificación de consultas para optimizar la ejecución de los programas.

- **Integración con ecosistema de *big data*.** Apache Beam se integra con varios motores de ejecución de procesamiento de datos y herramientas de *big data*, como Apache Flink, Apache Spark, Google Cloud Dataflow y más. Esto facilita su incorporación en entornos existentes de *big data* y permite que los desarrolladores aprovechen las herramientas y las plataformas que ya están familiarizados.

Casos de uso de Apache Beam

Apache Beam se utiliza en una variedad de casos de uso en diferentes industrias. Algunos ejemplos comunes incluyen:

- Procesamiento de datos en tiempo real y por lotes.

- Análisis de datos de registros de servidor y telemetría.

- Procesamiento de eventos de transmisión, como análisis de clic en sitios web.

- Análisis de datos de aplicaciones de IoT.

- Procesamiento de datos en tiempo real para aplicaciones de análisis y monitoreo de sistemas.