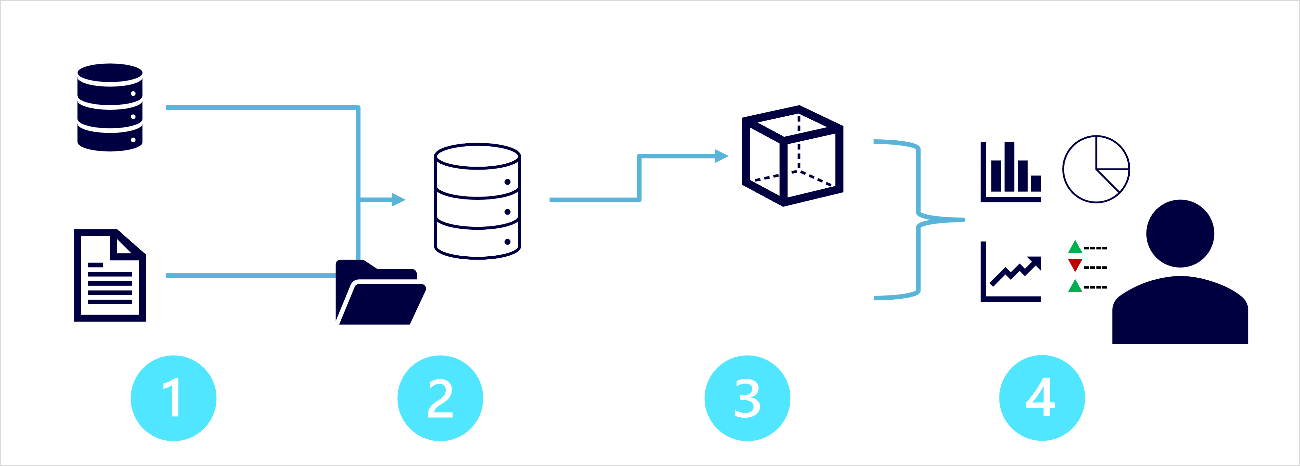
**Descripción del almacenamiento de datos moderno**

Completado100 XP

* 3 minutos

La arquitectura moderna de almacenamiento de datos puede variar, al igual que las tecnologías específicas que se usan para implementarla; de todos modos, en general, se incluyen los siguientes elementos:



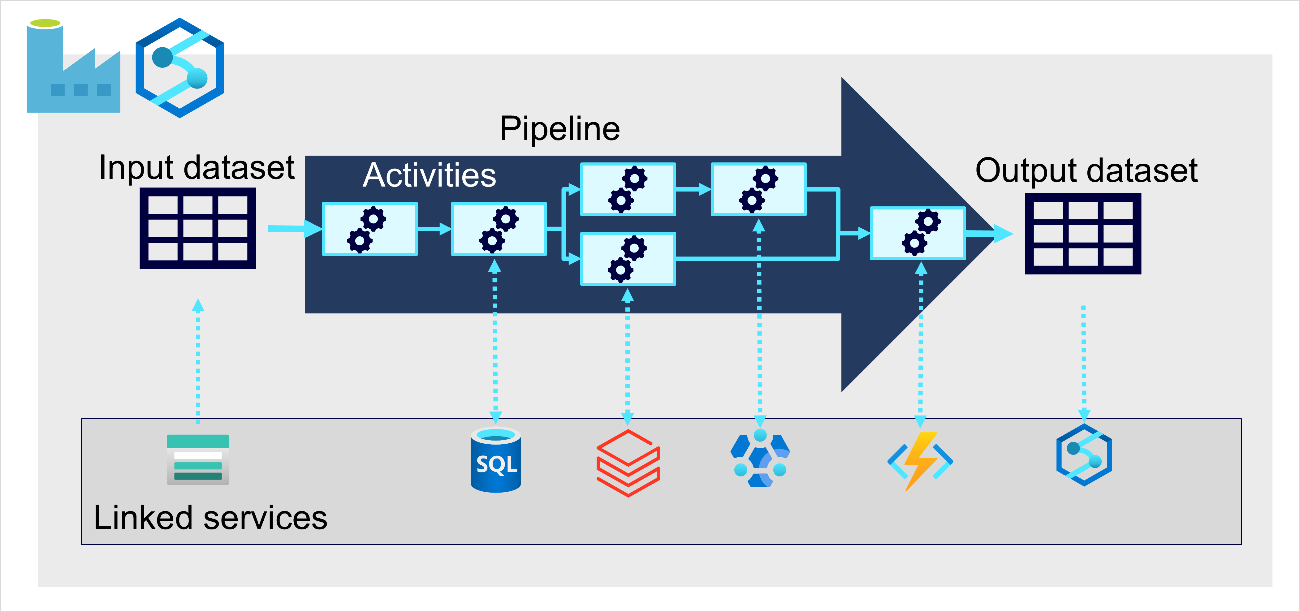
1. **Ingesta y procesamiento de datos**: los datos de uno o varios almacenes de datos transaccionales, archivos, flujos en tiempo real u otros orígenes se cargan en un lago de datos o en un almacenamiento de datos relacional. Normalmente, la operación de carga implica un proceso de *extracción, transformación y carga* (ETL) o de *extracción, carga y transformación* (ELT) en el que los datos se limpian, filtran y reestructuran para su análisis. En los procesos de ETL, los datos se transforman antes de cargarse en un almacén analítico, mientras que en un proceso de ELT los datos se copian en el almacén y, posteriormente, se transforman. En cualquier caso, la estructura de datos resultante está optimizada para las consultas analíticas. El procesamiento de datos suele realizarse mediante sistemas distribuidos que pueden procesar grandes volúmenes de datos en paralelo mediante clústeres de varios nodos. La ingesta de datos incluye el procesamiento por lotes de datos estáticos y el procesamiento en tiempo real de los datos de streaming.
2. **Almacén de datos analíticos**: los almacenes de datos para análisis a gran escala incluyen *almacenamientos de datos* relacionales, *lagos de datos* basados en sistema de archivos y arquitecturas híbridas que combinan características de almacenes de datos y lagos de datos (a veces bajo la denominación de *lagos de almacenamiento de datos* o *bases de datos de lago*). Los trataremos con más detalle más adelante.
3. **Modelo de datos analíticos**: aunque los analistas de datos y los científicos de datos pueden trabajar con los datos directamente en el almacén de datos analíticos, es habitual crear uno o varios modelos de datos que agreguen previamente los datos para facilitar la generación de informes, paneles y visualizaciones interactivas. A menudo, estos modelos de datos se describen como *cubos*, en los que los valores de datos numéricos se agregan en una o varias dimensiones (por ejemplo, para determinar las ventas totales por producto y región). El modelo encapsula las relaciones entre los valores de datos y las entidades dimensionales para admitir el análisis de tipo "rastrear agrupando datos/explorar en profundidad".
4. **Visualización de datos**: los analistas de datos consumen datos de modelos analíticos y directamente de almacenes analíticos para crear informes, paneles y otras visualizaciones. Además, los usuarios de una organización, que pueden no ser profesionales de la tecnología, pueden realizar informes y análisis de datos de autoservicio. Las visualizaciones de los datos muestran tendencias, comparaciones e indicadores clave de rendimiento (KPI) para una empresa u otra organización, y pueden tomar la forma de informes impresos, diagramas y gráficos en documentos o presentaciones de PowerPoint, paneles basados en web y entornos interactivos en los que los usuarios pueden explorar los datos visualmente.

**Exploración de canalizaciones de ingesta de datos**

Completado100 XP

* 5 minutos

Ahora que comprende un poco la arquitectura de una solución de almacenamiento de datos moderna y algunas de las tecnologías de procesamiento distribuido que se pueden usar para controlar grandes volúmenes de datos, es el momento de explorar cómo se ingieren los datos en un almacén de datos analíticos de uno o varios orígenes.



En Azure, la ingesta de datos a gran escala se implementa mejor mediante la creación de *canalizaciones* que organicen procesos de ETL. Puede crear y ejecutar canalizaciones mediante [Azure Data Factory](https://azure.microsoft.com/services/data-factory), o puede usar el mismo motor de canalización en [Azure Synapse Analytics](https://azure.microsoft.com/services/synapse-analytics) si quiere administrar todos los componentes de la solución de almacenamiento de datos en un área de trabajo unificada.

En cualquier caso, las canalizaciones constan de una o varias *actividades* que operan en los datos. Un conjunto de datos de entrada proporciona los datos de origen y las actividades se pueden definir como un flujo de datos que manipula incrementalmente los datos hasta que se genera un conjunto de datos de salida. Las canalizaciones utilizan *servicios vinculados* para cargar y procesar datos, y esto le permite usar la tecnología adecuada para cada paso del flujo de trabajo. Por ejemplo, puede usar un servicio vinculado de Azure Blob Store para ingerir el conjunto de datos de entrada y, posteriormente, usar servicios como Azure SQL Database para ejecutar un procedimiento almacenado que busque valores de datos relacionados, antes de ejecutar una tarea de procesamiento de datos en Azure Databricks o Azure HDInsight, o aplicar lógica personalizada mediante una función de Azure. Por último, puede guardar el conjunto de datos de salida en un servicio vinculado, como Azure Synapse Analytics. Las canalizaciones también pueden incluir algunas actividades integradas, que no requieren un servicio vinculado.

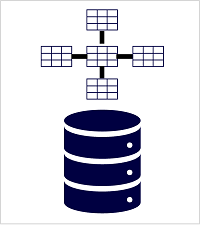
# Exploración de almacenes de datos analíticos

Completado100 XP

* 8 minutos

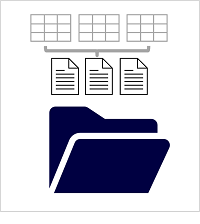
Hay dos tipos comunes de almacén de datos analíticos.

## Almacenamientos de datos



Un almacenamiento de datos es una base de datos relacional en la que los datos se almacenan en un esquema optimizado para el análisis de datos en lugar de en cargas de trabajo transaccionales. Normalmente, los datos de un almacén transaccional se organizan en un esquema en el que los valores numéricos se almacenan en tablas de hechos centrales, que están relacionadas con una o varias tablas de dimensiones que representan entidades por las que se pueden agregar los datos. Por ejemplo, una tabla de hechos podría contener datos de pedidos de ventas, que se pueden agregar por las dimensiones de cliente, producto, tienda y tiempo (lo que le permite, por ejemplo, encontrar fácilmente los ingresos totales mensuales de ventas por producto para cada tienda). Este tipo de esquema de tabla de hechos y dimensiones se denomina esquema de estrella; aunque a menudo se extiende a un esquema de copo de nieve mediante la adición de tablas adicionales relacionadas con las tablas de dimensiones para representar jerarquías dimensionales (por ejemplo, el producto puede estar relacionado con categorías de productos). Un almacenamiento de datos es una excelente opción si tiene datos transaccionales que se pueden organizar en un esquema estructurado de tablas y quiere usar SQL para consultarlos.

## Lagos de datos



Un lago de datos es un almacén de archivos, normalmente en un sistema de archivos distribuido para el acceso a datos de alto rendimiento. A menudo se usan tecnologías como Spark o Hadoop para procesar consultas en los archivos almacenados y devolver datos para informes y análisis. Estos sistemas suelen aplicar un enfoque de esquema en lectura para definir esquemas tabulares en archivos de datos semiestructurados en el punto donde se leen los datos para su análisis, sin aplicar restricciones cuando se almacenan. Los lagos de datos son excelentes para admitir una combinación de datos estructurados, semiestructurados e incluso no estructurados que quiere analizar sin necesidad de aplicar el esquema cuando los datos se escriben en el almacén.

### Enfoques híbridos

Puede usar un enfoque híbrido que combine características de lagos de datos y almacenamientos de datos en una base de datos de lago o un lago de almacenamiento de datos. Los datos sin procesar se almacenan como archivos en un lago de datos y una capa de almacenamiento relacional abstrae los archivos subyacentes y los expone como tablas, que se pueden consultar mediante SQL. Los grupos de SQL de Azure Synapse Analytics incluyen PolyBase, que permite definir tablas externas basadas en archivos de un lago de datos (y otros orígenes) y consultarlas mediante SQL. Synapse Analytics también admite un enfoque de base de datos de lago en el que puede usar plantillas de base de datos para definir el esquema relacional del almacenamiento de datos, al tiempo que almacena los datos subyacentes en un almacenamiento de lago de datos, separando el almacenamiento y el proceso de la solución de almacenamiento de datos. Los lagos de almacenamiento de datos son un enfoque relativamente nuevo en los sistemas basados en Spark y se habilitan mediante tecnologías como Delta Lake, que agrega funcionalidades de almacenamiento relacional a Spark, por lo que se pueden definir tablas que exijan esquemas y coherencia transaccional, admitan orígenes de datos de streaming y cargados por lotes y proporcionen una API de SQL para realizar consultas.

## Servicios de Azure para almacenes analíticos

En Azure, hay tres servicios principales que puede usar para implementar un almacén analítico a gran escala

[**Azure Synapse Analytics**](https://azure.microsoft.com/services/synapse-analytics) es una solución de un extremo a otro unificada para el análisis de datos a gran escala. Reúne varias tecnologías y funcionalidades, y esto permite combinar la integridad y la confiabilidad de los datos de un almacenamiento de datos relacional basado en SQL Server escalable y de alto rendimiento con la flexibilidad de una solución Apache Spark de código abierto y lago de datos. También incluye compatibilidad nativa para el análisis de registros y telemetría con grupos del Explorador de datos de Azure Synapse, así como canalizaciones de datos integradas para la ingesta y la transformación de datos. Todos los servicios de Azure Synapse Analytics se pueden administrar a través de una única interfaz de usuario interactiva denominada Azure Synapse Studio, que incluye la capacidad de crear cuadernos interactivos en los que se pueden combinar código de Spark y contenido de Markdown. Synapse Analytics es una excelente opción cuando se quiere crear una única solución de análisis unificada en Azure.

[**Azure Databricks**](https://azure.microsoft.com/services/databricks) es una implementación de Azure de la popular plataforma Databricks. Databricks es una completa solución de análisis de datos integrada en Apache Spark y ofrece funcionalidades nativas de SQL, así como clústeres de Spark optimizados para cargas de trabajo para el análisis de datos y la ciencia de datos. Databricks proporciona una interfaz de usuario interactiva a través de la cual se puede administrar el sistema y se pueden explorar los datos en cuadernos interactivos. Debido a su uso común en varias plataformas en la nube, es posible que quiera considerar el uso de Azure Databricks como almacén analítico si quiere usar la experiencia existente con la plataforma o si necesita operar en un entorno de varias nubes o admitir una solución portátil en la nube.

[**Azure HDInsight**](https://azure.microsoft.com/services/hdinsight) es un servicio de Azure que admite varios tipos de clústeres de análisis de datos de código abierto. Aunque no es tan fácil de usar como Azure Synapse Analytics y Azure Databricks, puede ser una opción adecuada si la solución de análisis se basa en varios marcos de código abierto o si necesita migrar una solución local existente basada en Hadoop a la nube.

**Nota**

Cada uno de estos servicios puede considerarse como un almacén de datos analíticos, en el sentido de que proporcionan un esquema y una interfaz a través de los cuales se pueden consultar los datos. Sin embargo, en muchos casos, los datos se almacenan realmente en un lago de datos y el servicio se usa para procesar los datos y ejecutar consultas. Algunas soluciones pueden incluso combinar el uso de estos servicios. Un proceso de ingesta de extracción, carga y transformación (ELT) puede copiar datos en el lago de datos y, posteriormente, usar uno de estos servicios para transformar los datos y otro para consultarlos. Por ejemplo, una canalización puede usar un trabajo de MapReduce que se ejecuta en HDInsight o un cuaderno que se ejecuta en Azure Databricks para procesar un gran volumen de datos en el lago de datos y, posteriormente, cargarlo en tablas de un grupo de SQL en Azure Synapse Analytics.

## Aprovisionar un área de trabajo de Azure Synapse Analytics

Para usar Azure Synapse Analytics, debe aprovisionar un recurso en el área de trabajo de Azure Synapse Analytics en la suscripción de Azure.

1. Abra Azure Portal en [https://portal.azure.com](https://portal.azure.com/) e inicie sesión con las credenciales asociadas con su suscripción de Azure.

**Sugerencia**

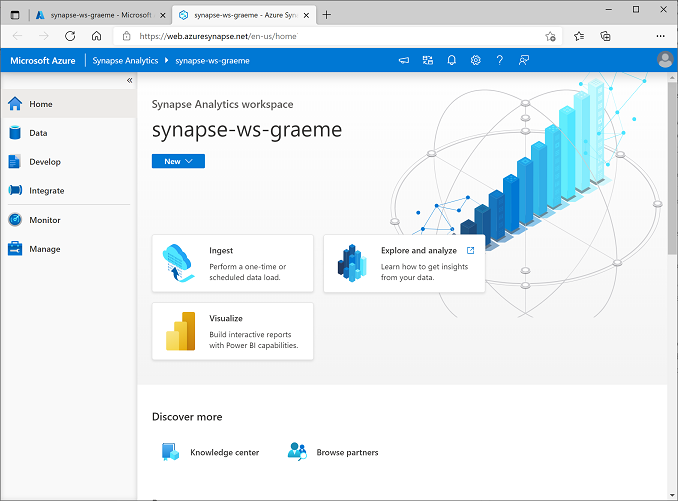
Asegúrese de que está trabajando en el directorio que contiene la suscripción, lo que se indica en la parte superior derecha, debajo del identificador de usuario. Si no es así, seleccione el icono de usuario y cambie el directorio. Tenga en cuenta que si anteriormente usó una suscripción de espacio aislado de Microsoft Learn, es posible que el portal haya usado de forma predeterminada el directorio Microsoft Learn Sandbox. Si es así, deberá cambiar a su propio directorio.

1. En Azure Portal, en la página **Inicio**, use el icono **+ Crear** para recurso para crear un nuevo recurso.
2. Busque Azure Synapse Analytics, y cree un recurso de **Azure Synapse Analytics** con la siguiente configuración:
   * **Suscripción**: suscripción de Azure
     + **Grupo de recursos**: cree un grupo de recursos con un nombre apropiado, como "synapse-rg".
     + **Grupo de recursos administrado**: escriba un nombre adecuado, por ejemplo, "synapse-managed-rg".
   * **Nombre del área de trabajo**: escriba un nombre único para el área de trabajo, por ejemplo, "synapse-ws-su\_nombre".
   * **Región**: seleccione cualquiera de las siguientes regiones:
     + Este de Australia
     + Centro de EE. UU.
     + Este de EE. UU. 2
     + Norte de Europa
     + Centro-sur de EE. UU.
     + Sudeste de Asia
     + Sur de Reino Unido
     + Oeste de Europa
     + Oeste de EE. UU.
     + WestUS 2
   * **Seleccionar Data Lake Storage Gen 2**: en la suscripción.
     + **Nombre de cuenta**: cree una cuenta con un nombre único, por ejemplo, "datalakeyour\_name".
     + **Nombre del sistema de archivos**: cree un sistema de archivos con un nombre único, por ejemplo, "fsyour\_name".

**Nota**

Un área de trabajo de Synapse Analytics requiere dos grupos de recursos en la suscripción de Azure; uno para los recursos creados explícitamente y otro para los recursos administrados utilizados por el servicio. También requiere una cuenta de almacenamiento de Data Lake en la que almacenar datos, scripts y otros artefactos.

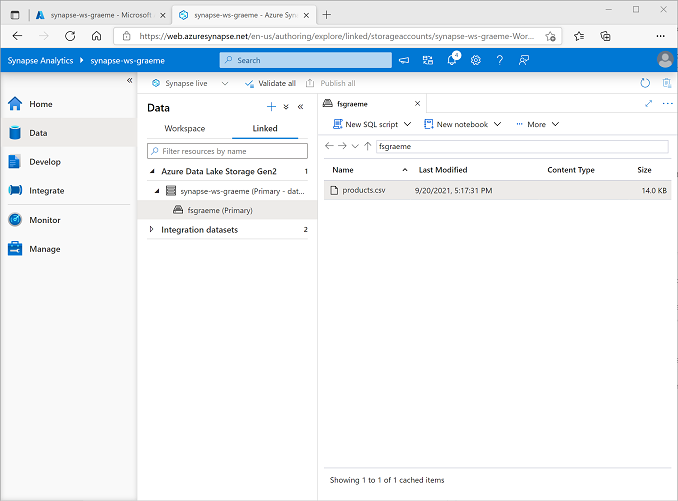
1. Cuando haya especificado estos detalles, seleccione **Revisar y crear** y, a continuación, seleccione **Crear** para crear el área de trabajo.
2. Espere a que se cree el área de trabajo; puede tardar unos cinco minutos.
3. Una vez completada la implementación, vaya al grupo de recursos que se creó y observe que contiene el área de trabajo de Synapse Analytics y una cuenta de almacenamiento de Data Lake.
4. Seleccione el área de trabajo de Synapse y, en su página **Información general**, en la tarjeta **Abrir Synapse Studio**, seleccione **Abrir** para abrir Synapse Studio en una nueva pestaña del explorador. Synapse Studio es una interfaz basada en web que puede usar para trabajar con el área de trabajo de Synapse Analytics.
5. En el lado izquierdo de Synapse Studio, use el icono **››** para expandir el menú; esto muestra las distintas páginas de Synapse Studio que usará para administrar recursos y realizar tareas de análisis de datos, como se muestra aquí:



## Ingerir datos

Una de las tareas clave que puede realizar con Azure Synapse Analytics es definir canalizaciones que transfieran (y, si es necesario, transformen) datos de una amplia variedad de orígenes al área de trabajo para su análisis.

1. En la página **Inicio** de Synapse Studio, seleccione **Ingerir** y, después, haga clic en **Built-in copy task** (Tarea de copia integrada) para abrir la herramienta **Copiar datos**.
2. En la herramienta Copiar datos, en el paso **Propiedades**, asegúrese de que las opciones **Built-in copy task** (Tarea de copia integrada) y **Run once now** (Ejecutar una vez ahora) están seleccionadas y haga clic en **Siguiente**.
3. En el paso **Origen**, en el subpaso **Conjunto de datos**, seleccione la siguiente configuración:
   * **Tipo de origen**: HTTP
   * **Conexión**: cree una conexión con las propiedades siguientes:
     + **Nombre**: productos de AdventureWorks
     + **Descripción**: lista de productos a través de HTTP
     + **Conectar mediante Integration Runtime**: AutoResolveIntegrationRuntime
     + **Dirección URL base**:
     + **Validación del certificado de servidor**: habilitar
     + **Tipo de autenticación**: anónimo
4. Después de crear la conexión, en el subpaso **Origen/Conjunto de datos**, asegúrese de que está seleccionada la siguiente configuración y, a continuación, seleccione **Siguiente**:
   * **Dirección URL relativa**: dejar en blanco
   * **Request method** (Método de solicitud): GET
   * **Encabezados adicionales**: dejar en blanco
   * **Copia binaria**: sin seleccionar
   * **Tiempo de espera de solicitud**: dejar en blanco
   * **Número máximo de conexiones simultáneas**: dejar en blanco
5. En el paso **Origen**, en el subpaso **Configuración**, seleccione **Vista previa de los datos** para obtener una vista previa de los datos del producto que la canalización va a ingerir y, a continuación, cierre la vista previa.
6. Después de previsualizar los datos, en el paso **Origen/Configuración**, asegúrese de que está seleccionada la siguiente configuración y, a continuación, seleccione **Siguiente**:
   * **Formato de archivo**: DelimitedText
   * **Delimitador de columna**: coma (,)
   * **Delimitador de fila**: avance de línea (\n)
   * **Primera fila como encabezado**: seleccionada
   * **Tipo de compresión**: ninguno
7. En el paso **Destino**, en el subpaso **Conjunto de datos**, seleccione la siguiente configuración:
   * **Tipo de destino**: Azure Data Lake Storage Gen 2 = **Conexión**: cree una conexión con las siguientes propiedades:
     + **Nombre**: Productos
     + **Descripción**: lista de productos
     + **Conectar mediante Integration Runtime**: AutoResolveIntegrationRuntime
     + **Método de autenticación**: clave de cuenta
     + **Método de selección de cuenta**: desde la suscripción
       - **Suscripción de Azure**: seleccione su suscripción
       - **Nombre de la cuenta de almacenamiento**: seleccione su cuenta de almacenamiento
     + **Prueba de conexión**: al servicio vinculado
8. Después de crear la conexión, en el paso **Destino/Conjunto de datos**, asegúrese de que está seleccionada la siguiente configuración y, a continuación, seleccione **Siguiente**:
   * **Ruta de acceso de la carpeta**: vaya a la carpeta del sistema de archivos
   * **Nombre de archivo**: products.csv
   * **Comportamiento de copia**: ninguno
   * **Número máximo de conexiones simultáneas**: dejar en blanco
   * **Tamaño de bloque (MB)**: dejar en blanco
9. En el paso **Destino,** en el subpaso **Configuración**, asegúrese de que están seleccionadas las siguientes propiedades. Luego, seleccione **Siguiente**:
   * **Formato de archivo**: DelimitedText
   * **Delimitador de columna**: coma (,)
   * **Delimitador de fila**: avance de línea (\n)
   * **Agregar encabezado al archivo**: seleccionado
   * **Tipo de compresión**: ninguno
   * **Número máximo de filas por archivo**: dejar en blanco
   * **Prefijo de nombre de archivo**: dejar en blanco
10. En el paso **Configuración**, configure las opciones siguientes y, a continuación, haga clic en **Siguiente**:
    * **Nombre de tarea**: copiar productos
    * **Descripción de la tarea**: copia de datos de productos
    * **Tolerancia a errores**: dejar en blanco
    * **Habilitar registro**: sin seleccionar
    * **Habilitar almacenamiento provisional**: sin seleccionar
11. En el paso **Revisar y finalizar**, en el subpaso **Revisar**, lea el resumen y, a continuación, haga clic en **Siguiente**.
12. En el paso **Implementación**, espere a que se implemente la canalización y, a continuación, haga clic en **Finalizar**.
13. En Synapse Studio, seleccione la página **Supervisar** y, en la pestaña **Ejecuciones de canalizaciones**, espere a que la canalización **Copy products** se complete con el estado **Correcto** (puede usar el botón **↻ Actualizar** de la página Ejecuciones de canalizaciones para actualizar el estado).
14. En la página **Datos**, seleccione la pestaña **Vinculado** y expanda la jerarquía de **Azure Data Lake Storage Gen 2** hasta que vea el almacenamiento de archivos para el área de trabajo de Synapse. A continuación, seleccione el almacenamiento de archivos para comprobar que un archivo llamado **products.csv** se ha copiado en esta ubicación, como se muestra aquí:



## Usar un grupo de SQL para analizar datos

Ahora que ha ingerido algunos datos en el área de trabajo, puede usar Synapse Analytics para consultarlos y analizarlos. Una de las formas más comunes de consultar datos es usar SQL y, en Synapse Analytics, puede usar un grupo de SQL para ejecutar código SQL.

1. En Synapse Studio, haga clic con el botón derecho en el archivo **products.csv** del almacenamiento de archivos del área de trabajo de Synapse, seleccione **New SQL script** (Nuevo script SQL) y seleccione **Seleccionar las primeras 100 filas**.
2. En el panel **SQL Script 1** (Script SQL 1) que se abre, revise el código SQL que se ha generado, que debe ser similar al siguiente:

SQLCopiar

-- This is auto-generated code

SELECT

    TOP 100 \*

FROM

    OPENROWSET(

        BULK 'https://datalakexx.dfs.core.windows.net/fsxx/products.csv',

        FORMAT = 'CSV',

        PARSER\_VERSION='2.0'

) AS [result]

Este código abre un conjunto de filas del archivo de texto que importó y recupera las primeras 100 filas de datos.

1. En la lista **Conectar a**, asegúrese de que **Integrado** está seleccionado: representa el grupo de SQL integrado que se creó con el área de trabajo.
2. En la barra de herramientas, use el botón **▷ Ejecutar** para ejecutar el código SQL y revise los resultados, que deben tener un aspecto similar al siguiente:

| **C1** | **c2** | **c3** | **c4** |
| --- | --- | --- | --- |
| ProductID | ProductName | Category | ListPrice |
| 771 | Mountain-100 Silver, 38 | Bicicletas de montaña | 3399.9900 |
| 772 | Mountain-100 Silver, 42 | Bicicletas de montaña | 3399.9900 |
| ... | ... | ... | ... |

1. Tenga en cuenta que los resultados constan de cuatro columnas denominadas C1, C2, C3 y C4, y que la primera fila de los resultados contiene los nombres de los campos de datos. Para corregir este problema, agregue un parámetro HEADER\_ROW = TRUE a la función OPENROWSET tal como se muestra aquí (reemplace datalakexx y fsxx por los nombres de la cuenta de almacenamiento del lago de datos y el sistema de archivos) y, luego, vuelva a ejecutar la consulta:

SQLCopiar

SELECT

    TOP 100 \*

FROM

    OPENROWSET(

        BULK 'https://datalakexx.dfs.core.windows.net/fsxx/products.csv',

        FORMAT = 'CSV',

        PARSER\_VERSION='2.0',

HEADER\_ROW = TRUE

) AS [result]

Ahora los resultados son similares a los siguientes:

| **ProductID** | **ProductName** | **Category** | **ListPrice** |
| --- | --- | --- | --- |
| 771 | Mountain-100 Silver, 38 | Bicicletas de montaña | 3399.9900 |
| 772 | Mountain-100 Silver, 42 | Bicicletas de montaña | 3399.9900 |
| ... | ... | ... | ... |

1. Modifique la consulta como se muestra a continuación (reemplazando datalakexx y fsxx por los nombres de la cuenta de almacenamiento de Data Lake y del sistema de archivos):

SQLCopiar

SELECT

    Category, COUNT(\*) AS ProductCount

FROM

    OPENROWSET(

        BULK 'https://datalakexx.dfs.core.windows.net/fsxx/products.csv',

        FORMAT = 'CSV',

        PARSER\_VERSION='2.0',

HEADER\_ROW = TRUE

) AS [result]

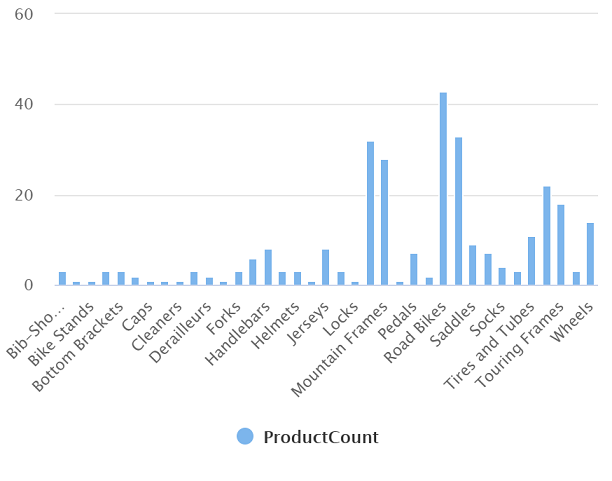
GROUP BY Category;

1. Ejecute la consulta modificada, que debe devolver un conjunto de resultados que contenga el número de productos de cada categoría, de la siguiente forma:

| **Category** | **ProductCount** |
| --- | --- |
| Culotes | 3 |
| Bastidores de bicicletas | 1 |
| ... | ... |

1. En el panel **Propiedades** de **SQL Script 1** (Script SQL 1), cambie el **Nombre** a **Count Products by Category** (Contar productos por categoría). A continuación, en la barra de herramientas, seleccione **Publicar** para guardar el script.
2. Cierre el panel de scripts **Count Products by Category** (Contar productos por categoría).
3. En Synapse Studio, seleccione la página **Desarrollar** y observe que el script SQL publicado **Count Products by Category** (Contar productos por categoría) se ha guardado allí.
4. Seleccione el script SQL **Count Products by Category** (Contar productos por categoría) para volver a abrirlo. A continuación, asegúrese de que el script está conectado al grupo de SQL **Integrado** y ejecútelo para obtener el recuento de productos.
5. En el panel **Resultados**, seleccione la vista **Gráfico** y, a continuación, seleccione la siguiente configuración para el gráfico:
   * **Tipo de gráfico**: columna
   * **Columna de categoría**: categoría
   * **Columnas de leyenda (serie)**: ProductCount
   * **Posición de la leyenda**: inferior central
   * **Etiqueta de leyenda (serie)**: dejar en blanco
   * **Valor mínimo de leyenda (serie)**: dejar en blanco
   * **Valor máximo de leyenda (serie)**: dejar en blanco
   * **Etiqueta de categoría**: dejar en blanco

El gráfico resultante debe ser similar al siguiente:



## Usar un grupo de Spark para analizar datos

Aunque SQL es un lenguaje común para consultar conjuntos de datos estructurados, muchos analistas de datos consideran que lenguajes como Python son útiles para explorar y preparar los datos para su análisis. En Azure Synapse Analytics, puede ejecutar código de Python (y otro) en un Grupo de Spark, que usa un motor de procesamiento de datos distribuido basado en Apache Spark.

1. En Synapse Studio, seleccione la página **Administrar**.
2. Seleccione la pestaña **Grupos de Apache Spark** y, a continuación, use el icono **+ Nuevo** para crear un nuevo grupo de Spark con la siguiente configuración:
   * **Nombre del grupo de Apache Spark**: spark
   * **Familia de tamaños de nodo**: optimizada para memoria
   * **Tamaño del nodo**: pequeño (4 núcleos virtuales/32 GB)
   * **Escalabilidad automática**: habilitada
   * **Número de nodos**: 3----3
3. Revise y cree el grupo de Spark y espere a que se implemente (puede tardar unos minutos).
4. Cuando se haya implementado el grupo de Spark, en Synapse Studio, en la página **Datos**, vaya al sistema de archivos del área de trabajo de Synapse. A continuación, haga clic con el botón derecho en **products.csv**, seleccione **Nuevo cuaderno** y seleccione **Cargar en DataFrame**.
5. En el panel **Notebook 1** (Cuaderno 1) que se abre, en la lista **Adjuntar a**, seleccione el grupo de Spark **spark** creado previamente y asegúrese de que la opción **Lenguaje** está establecida en **PySpark (Python)**.
6. Revise solo el código de la primera celda del cuaderno, que debe tener este aspecto:

PythonCopiar

%%pyspark

df = spark.read.load('abfss://fsxx@datalakexx.dfs.core.windows.net/products.csv', format='csv'

## If header exists uncomment line below

##, header=True

)

display(df.limit(10))

1. Use el icono **▷** situado a la izquierda de la celda de código para ejecutarlo y espere a obtener los resultados. La primera vez que ejecuta una celda en un cuaderno, se inicia el grupo de Spark, por lo que puede tardar más o menos un minuto en devolver los resultados.

**Nota**

Si se produce un error porque el kernel de Python aún no está disponible, vuelva a ejecutar la celda.

1. Finalmente, los resultados deben aparecer debajo de la celda y deben ser similares a estos:

| **c0** | **c1** | **c2** | **c3** |
| --- | --- | --- | --- |
| ProductID | ProductName | Category | ListPrice |
| 771 | Mountain-100 Silver, 38 | Bicicletas de montaña | 3399.9900 |
| 772 | Mountain-100 Silver, 42 | Bicicletas de montaña | 3399.9900 |
| ... | ... | ... | ... |

1. Quite la marca de comentario de la línea ,header=True (porque el archivo products.csv tiene los encabezados de columna en la primera línea), por lo que el código tiene el siguiente aspecto:

PythonCopiar

%%pyspark

df = spark.read.load('abfss://fsxx@datalakexx.dfs.core.windows.net/products.csv', format='csv'

## If header exists uncomment line below

, header=True

)

display(df.limit(10))

1. Vuelva a ejecutar la celda y compruebe que los resultados son similares a los siguientes:

| **ProductID** | **ProductName** | **Category** | **ListPrice** |
| --- | --- | --- | --- |
| 771 | Mountain-100 Silver, 38 | Bicicletas de montaña | 3399.9900 |
| 772 | Mountain-100 Silver, 42 | Bicicletas de montaña | 3399.9900 |
| ... | ... | ... | ... |

1. Tenga en cuenta que volver a ejecutar la celda tarda menos tiempo, porque el grupo de Spark ya se ha iniciado.
2. En los resultados, use el icono **＋Código** para agregar una nueva celda de código al cuaderno.
3. En la nueva celda de código vacía, agregue el código siguiente:

PythonCopiar

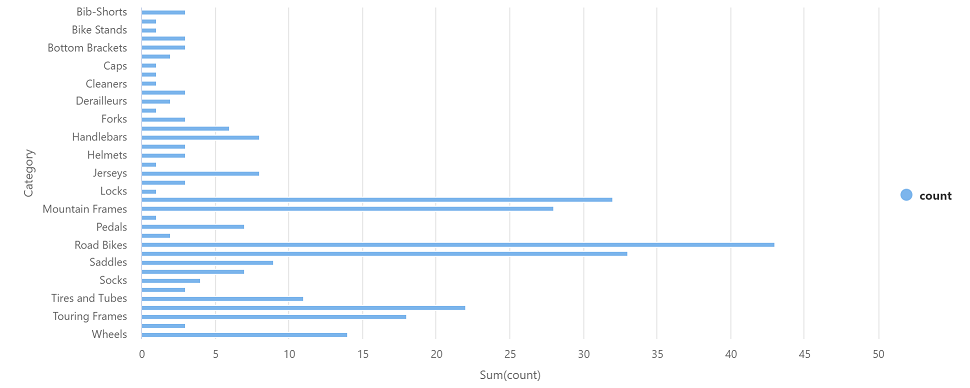
df\_counts = df.groupby(df.Category).count()

display(df\_counts)

1. Ejecute la nueva celda de código haciendo clic en su icono **▷** y revise los resultados, que deben tener un aspecto similar al siguiente:

| **Category** | **count** |
| --- | --- |
| Tubos de dirección | 3 |
| Ruedas | 14 |
| ... | ... |

1. En la salida de resultados de la celda, seleccione la vista **Gráfico**. El gráfico resultante debe ser similar al siguiente:



1. Cierre el panel **Notebook 1** (Cuaderno 1) y descarte los cambios.

## Eliminación de recursos de Azure

Si ha terminado de explorar Azure Synapse Analytics, debe eliminar los recursos que ha creado para evitar costos innecesarios de Azure.

1. Cierre la pestaña del explorador de Synapse Studio y vuelva a Azure Portal.
2. En Azure Portal, en la página **Inicio**, seleccione **Grupos de recursos**.
3. Seleccione el grupo de recursos del área de trabajo de Synapse Analytics (no el grupo de recursos administrado) y compruebe que contiene el área de trabajo de Synapse, la cuenta de almacenamiento y el grupo de Spark del área de trabajo.
4. En la parte superior de la página **Información general** del grupo de recursos, seleccione **Eliminar grupo de recursos**.
5. Escriba el nombre del grupo de recursos para confirmar que quiere eliminarlo y seleccione **Eliminar**.

Después de unos minutos, se eliminarán el área de trabajo de Azure Synapse y el área de trabajo administrada asociada a ella.

# Introducción

Completado100 XP

* 1 minuto

Un mayor uso de la tecnología por parte de personas, empresas y otras organizaciones, junto con la proliferación de dispositivos inteligentes y acceso a Internet, ha generado un crecimiento masivo del volumen de datos que se pueden generar, capturar y analizar. Gran parte de estos datos se pueden procesar en tiempo real (o al menos, casi en tiempo real) como un flujo perpetuo de datos, lo que permite la creación de sistemas que revelan conclusiones y tendencias instantáneas, o toman medidas inmediatas de respuesta a los eventos a medida que se producen.

## Objetivos de aprendizaje

En este módulo, obtendrá información sobre los conceptos básicos del procesamiento de flujos y los servicios de Microsoft Azure que puede usar para implementar soluciones de datos de flujos. En concreto, obtendrá información sobre cómo:

* Comparación del procesamiento por lotes y por flujos
* Descripción de elementos comunes de las soluciones de datos de flujos
* Descripción de las características y funciones de Azure Stream Analytics
* Descripción de las características y funciones de Spark Structured Streaming en Azure

# Comprensión del procesamiento de flujos y por lotes

Completado100 XP

* 9 minutos

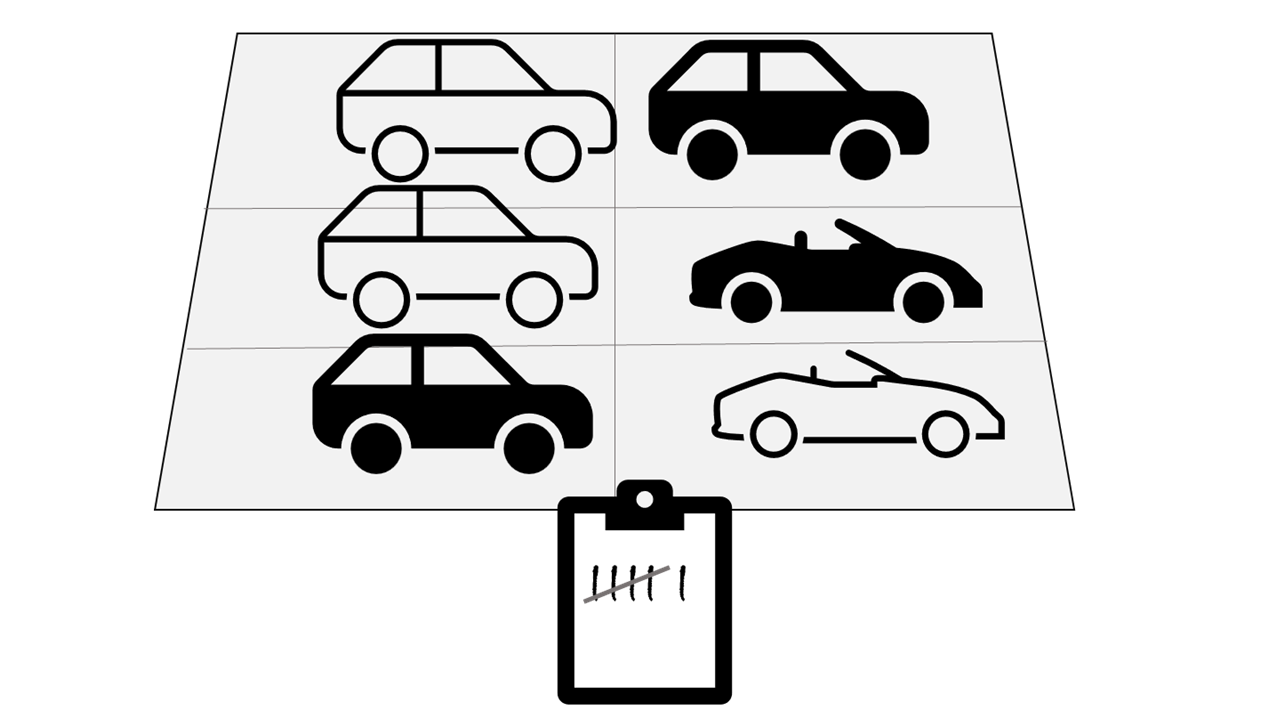
El procesamiento de datos es simplemente la conversión de datos sin procesar en información significativa a través de un proceso. Existen dos métodos generales para procesar los datos:

* Procesamiento por lotes, en el que se recopilan y almacenan varios registros de datos antes de procesarse juntos en una sola operación.
* Procesamiento de flujos, en el que un origen de datos se supervisa y procesa constantemente en tiempo real a medida que se producen nuevos eventos de datos.

## Procesamiento por lotes

En el procesamiento por lotes, los elementos de datos se recopilan en un grupo conforme van llegando. Después, todo el grupo se procesa más adelante como un lote. El momento en el que se procesa cada grupo se puede determinar de varias maneras. Por ejemplo, los datos se pueden procesar según un intervalo de tiempo programado (por ejemplo, cada hora), o bien el procesamiento puede desencadenarse cuando se alcance una determinada cantidad de datos o como resultado de algún otro evento.

Por ejemplo, supongamos que quiere analizar el tráfico de carreteras contando el número de automóviles en un tramo de carretera. Un enfoque de procesamiento por lotes requeriría recopilar los automóviles de un aparcamiento y, a continuación, contarlos en una sola operación mientras están en reposo.



Si la carretera está ocupada, con un gran número de automóviles que conducen a intervalos frecuentes, este enfoque puede ser poco práctico. Tenga en cuenta que no obtiene ningún resultado hasta que haya estacionado un lote de automóviles y los haya contado.

Un ejemplo real de procesamiento por lotes es la forma en que las empresas de tarjetas de crédito controlan la facturación. El cliente no recibe una factura por cada compra que hace con su tarjeta de crédito, sino una factura mensual para todas las compras de ese mes.

Entre las ventajas del procesamiento por lotes se incluyen las siguientes:

* Se pueden procesar grandes volúmenes de datos en un momento especificado.
* Se puede programar para ejecutarse mientras los equipos o sistemas están inactivos, como por la noche o en horas de poca actividad.

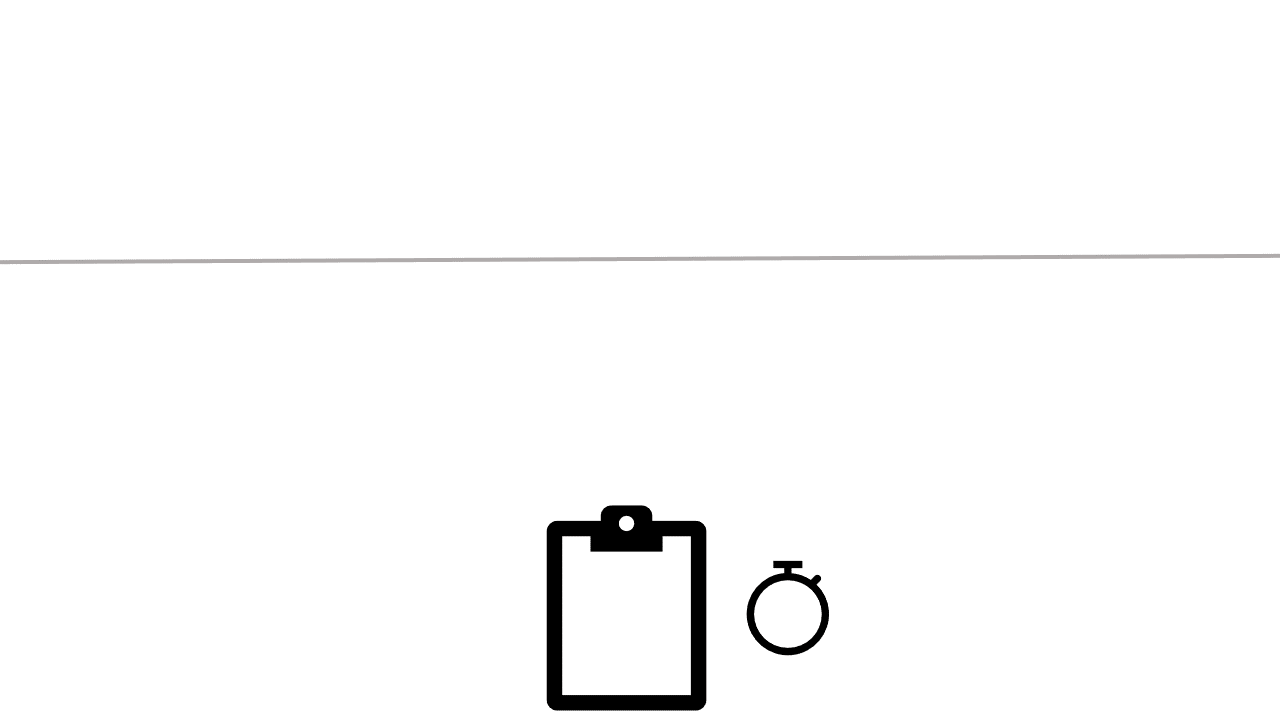
Entre las desventajas del procesamiento por lotes están las siguientes:

* El tiempo de retardo entre la ingesta de los datos y la obtención de los resultados.
* Todos los datos de entrada de un trabajo por lotes deben estar listos para poder procesar un lote. Esto significa que los datos deben comprobarse con cuidado. Los problemas con los datos, los errores y los bloqueos de los programas que se producen durante los trabajos por lotes provocan la detención de todo el proceso. Los datos de entrada deben comprobarse cuidadosamente antes de volver a ejecutar el trabajo. Incluso los errores de datos menores pueden impedir la ejecución de un trabajo por lotes.

## Información sobre el procesamiento de flujos

En el procesamiento en streaming, cada nuevo fragmento de datos se procesa cuando llega. A diferencia del procesamiento por lotes, no hay ningún tiempo de espera hasta el siguiente intervalo de procesamiento por lotes y los datos se procesan como unidades individuales en tiempo real en lugar de procesarse de lote en lote. El procesamiento de datos de streaming es beneficioso en la mayoría de los escenarios donde se generan datos dinámicos nuevos de forma continua.

Por ejemplo, un enfoque mejor para nuestro hipotético problema de recuento de automóviles podría ser aplicar un enfoque de flujo de datos, contando los automóviles en tiempo real a medida que pasan:



En este enfoque, no es necesario esperar hasta que todos los automóviles hayan estacionado para comenzar a procesarlos, y puede sumar los datos a lo largo de intervalos de tiempo. Por ejemplo, contando el número de automóviles que pasan cada minuto.

Entre los ejemplos reales de datos de flujos se incluyen:

* Una institución financiera realiza un seguimiento de los cambios en el mercado de valores en tiempo real, calcula el valor en riesgo y reequilibra automáticamente las carteras en función de los movimientos de precio de las acciones.
* Una empresa de juegos en línea recopila datos en tiempo real sobre las interacciones de los jugadores con los juegos y los incorpora en su plataforma de juegos. Después, analiza los datos en tiempo real y ofrece incentivos y experiencias dinámicas para atraer a los jugadores.
* Un sitio web inmobiliario hace un seguimiento de un subconjunto de datos de dispositivos móviles y ofrece recomendaciones en tiempo real de las propiedades que pueden visitar los clientes en función de su ubicación geográfica.

El procesamiento en streaming es idóneo para las operaciones en las que la velocidad de ejecución es importante y que requieren una respuesta instantánea en tiempo real. Por ejemplo, un sistema que supervisa la presencia de humo y altas temperaturas en un edificio necesita activar alarmas y desbloquear puertas para permitir que los residentes puedan salir inmediatamente en caso de que se produzca un incendio.

## Diferencias entre los datos de streaming y por lotes

Además de las diferencias en la forma en que el procesamiento por lotes y en streaming controlan los datos, hay otras diferencias:

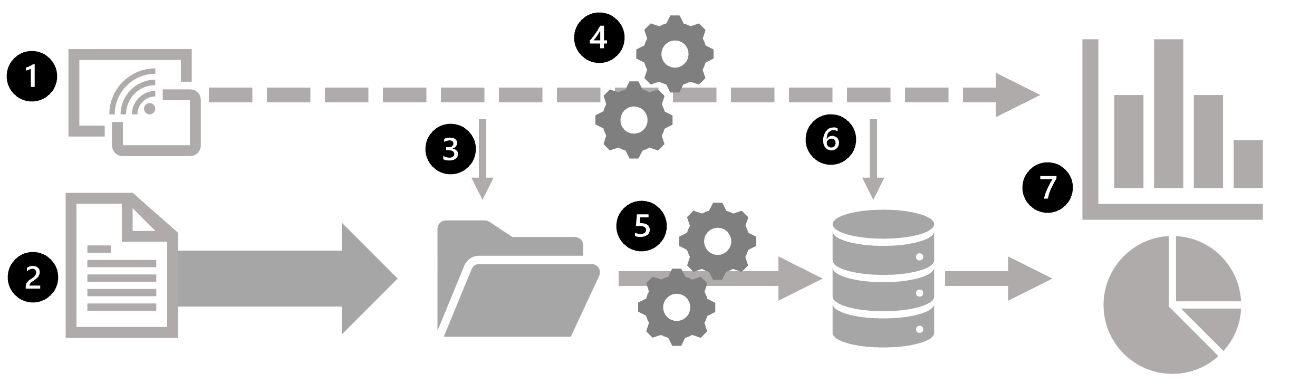
* Ámbito de los datos: el procesamiento por lotes puede procesar todos los datos del conjunto de datos. Normalmente, el procesamiento en streaming solo tiene acceso a los datos recibidos más recientemente recibidos o dentro de un período de tiempo cambiante (los últimos 30 segundos, por ejemplo).
* Tamaño de los datos: el procesamiento por lotes es adecuado para administrar grandes conjuntos de datos de forma eficaz. El procesamiento en streaming está diseñado para registros individuales o microlotes que constan de pocos registros.
* Rendimiento: la latencia del procesamiento por lotes suele ser de unas horas. Normalmente, el procesamiento en streaming se produce inmediatamente, con la latencia en segundos o milisegundos. La latencia es el tiempo que se tarda en recibir y procesar los datos.
* Análisis: normalmente se usa el procesamiento por lotes para realizar análisis complejos. El procesamiento en streaming se usa para funciones de respuesta simples, agregaciones o cálculos, como el cálculo de la media acumulada.

## Combinación del procesamiento por lotes y por flujos

Muchas soluciones de análisis a gran escala incluyen una combinación de procesamiento por lotes y de flujos, lo que permite el análisis de datos históricos y en tiempo real. Es habitual que las soluciones de procesamiento de flujos capturen datos en tiempo real, los procesen filtrándolos o sumándolos, y los presenten a través de paneles y visualizaciones en tiempo real (por ejemplo, mostrando el total de automóviles en funcionamiento que han pasado por una carretera durante la hora actual), al tiempo que se conservan los resultados procesados en un almacén de datos para el análisis histórico junto con los datos procesados por lotes (por ejemplo, para habilitar el análisis de volúmenes de tráfico durante el último año).

Incluso cuando no se requiere el análisis o la visualización en tiempo real de los datos, las tecnologías de flujos a menudo se usan para capturar datos en tiempo real y almacenarlos en un almacén de datos para su posterior procesamiento por lotes (esto equivale a redirigir todos los automóviles que viajan por una carretera a un aparcamiento antes de contarlos).

En el diagrama siguiente se muestran algunos métodos para combinar el procesamiento por lotes y de flujos en una arquitectura de análisis de datos a gran escala.



1. Los eventos de datos de un origen de datos de flujos se capturan en tiempo real.
2. Los datos de otros orígenes se ingieren en un almacén de datos (a menudo, un lago de datos) para el procesamiento por lotes.
3. Si no es necesario llevar a cabo análisis en tiempo real, los datos de flujos capturados se escriben en el almacén de datos para su posterior procesamiento por lotes.
4. Cuando se requiere un análisis en tiempo real, se usa una tecnología de procesamiento de flujos para preparar los datos de flujos para el análisis o visualización en tiempo real. A menudo, se filtran o suman los datos por periodos de tiempo.
5. Los datos que no son de flujos se procesan por lotes periódicamente para prepararlos para el análisis y los resultados se conservan en un almacén de datos analíticos (a menudo denominado almacén de datos) para el análisis histórico.
6. Los resultados del procesamiento de flujos también se pueden conservar en el almacén de datos analíticos para admitir el análisis histórico.
7. Las herramientas analíticas y de visualización se usan para presentar y explorar los datos históricos y en tiempo real.

**Nota**

Entre las arquitecturas de soluciones usadas con más frecuencia para un procesamiento de datos de flujos y por lotes de manera combinada, se encuentran arquitecturaslambda y delta. Los detalles de estas arquitecturas están fuera del ámbito de este curso, pero incorporan tecnologías tanto para el procesamiento de datos por lotes a gran escala como el procesamiento de flujos en tiempo real para crear una solución analítica de un extremo a otro.