Opciones de análisis de big data en AWS

Enero de 2016

Avisos

Este documento se suministra únicamente con fines informativos. Representa la oferta actual de productos y prácticas de AWS a partir de la fecha de publicación de este documento. Dichas prácticas y productos pueden modificarse sin previo aviso. Los clientes son responsables de realizar sus propias evaluaciones independientes de la información contenida en este documento y de cualquier uso de los productos o servicios de AWS, cada uno de los cuales se ofrece “tal cual”, sin garantía de ningún tipo, ya sea explícita o implícita. Este documento no genera ninguna garantía, declaración, compromiso contractual, condición ni certeza por parte de AWS, sus filiales, proveedores o licenciantes. Las responsabilidades y obligaciones de AWS con respecto a sus clientes se controlan mediante los acuerdos de AWS y este documento no forma parte ni modifica ningún acuerdo entre AWS y sus clientes.

Contenido

Resumen

Este documento técnico ayuda a los arquitectos, científicos de datos y desarrolladores a conocer las opciones de análisis de big data disponibles en la nube de AWS. Proporciona una descripción general de los servicios disponibles, que incluye la información siguiente:

- Patrones de uso ideales
- Modelo de costos
- Desempeño
- Durabilidad y disponibilidad
- Escalabilidad y elasticidad
- Interfaces
- Patrones de uso no recomendados

Este documento termina con escenarios representativos de las opciones de análisis en uso e incluye recursos adicionales para empezar a trabajar con análisis de big data en AWS.

Introducción

A medida que nos convertimos en una sociedad más digital, la cantidad de datos que se crean y se recopilan aumenta más rápido. El análisis de estos datos en continuo crecimiento puede suponer todo un reto si se utilizan las herramientas tradicionales. Para salvar la distancia existente entre los datos que se generan y los datos que pueden analizarse eficazmente es necesario innovar.

Las herramientas y tecnologías de big data ofrecen oportunidades y plantean retos a la hora de analizar eficazmente los datos para conocer mejor las preferencias de los clientes, obtener una ventaja competitiva en el mercado y hacer crecer su negocio. Las arquitecturas de administración de datos han evolucionado desde el modelo de almacenamiento tradicional hacia otras más complejas que dan respuesta a requisitos adicionales, como el procesamiento en tiempo real y por lotes, datos estructurados y no estructurados y transacciones a alta velocidad, entre otros.

Amazon Web Services (AWS) proporciona una amplia plataforma de servicios administrados con la que podrá crear, asegurar y escalar aplicaciones integrales de big data de forma rápida y sencilla. Tanto si necesita transmisiones en tiempo real para sus aplicaciones como si quiere procesar los datos por lotes, AWS tiene la infraestructura y las herramientas necesarias para iniciar su siguiente proyecto de big data. No tiene que adquirir hardware ni mantener o escalar infraestructuras: dispondrá de todo lo necesario para recopilar, almacenar, procesar y analizar big data. AWS cuenta con un ecosistema de soluciones de análisis diseñado específicamente para administrar este creciente volumen de datos y ofrecerle información valiosa sobre su negocio.

La ventaja de AWS en el análisis de big data

El análisis de grandes conjuntos de datos requiere una gran capacidad de cómputo, cuyo tamaño puede variar en función de la cantidad de datos de entrada y del tipo de análisis. Esta característica de las cargas de trabajo de big data encaja perfectamente con el modelo de informática en la nube de pago por uso, en el que las aplicaciones se pueden ampliar y reducir fácilmente en función de la demanda. Si los requisitos cambian, puede adaptar fácilmente el entorno (tanto horizontal como verticalmente) en AWS para satisfacer sus necesidades, sin esperar a disponer de hardware adicional ni invertir más de lo necesario para conseguir suficiente capacidad.

En las aplicaciones críticas de infraestructuras más tradicionales, los diseñadores de sistemas no tienen otra opción que aprovisionar recursos de sobra, ya que el sistema debe poder administrar un volumen creciente de datos si aumentan las necesidades del negocio. En cambio, con AWS se puede obtener más capacidad y recursos de computación en cuestión de minutos, lo que significa que puede ampliar y reducir sus aplicaciones de big data en función de la demanda, y tener un rendimiento óptimo del sistema.

Además, tendrá una capacidad informática flexible en una infraestructura global con acceso a las numerosas regiones geográficas[1] que ofrece AWS y la posibilidad de utilizar otros servicios escalables con los que podrá crear sofisticadas aplicaciones de big data. Entre estos servicios adicionales se encuentran Amazon Simple Storage Service (Amazon S3)[2] para almacenar datos y AWS Data Pipeline[3] para organizar tareas que los muevan y transformen fácilmente. AWS IoT,[4] permite que los dispositivos conectados interactúen con las aplicaciones en la nube y con otros dispositivos.

Además, AWS tiene muchas opciones para ayudar a obtener datos en la nube, incluidos la protección de los dispositivos como AWS Import/Export Snowball[5] para acelerar transferencias de datos a escala de petabytes, Amazon Kinesis Firehose[6] para cargar datos de streaming y conexiones privadas escalables a través de AWS Direct Connect[7]. Puesto que el uso de los móviles continúa creciendo rápidamente, puede utilizar el conjunto de servicios dentro de AWS Mobile Hub[8] y recopilar y medir los datos y el uso de la aplicación o exportar estos datos a otro servicio para análisis personalizados adicionales.

Por todas estas capacidades, la plataforma AWS es ideal para resolver los problemas que plantea big data. Son muchos los clientes que ya han implementado cargas de trabajo de análisis en ella satisfactoriamente. Para obtener más información sobre casos prácticos, consulte Big Data y HPC. Con la tecnología de la nube de AWS.[9]

Estos servicios se describen siguiendo el orden del proceso de recopilación, procesamiento, almacenamiento y análisis de big data:

- Amazon Kinesis Streams
- AWS Lambda
- Amazon Elastic MapReduce
- Amazon Machine Learning
- Amazon DynamoDB
- Amazon Redshift
- Amazon Elasticsearch Service
- Amazon QuickSight

Las instancias Amazon EC2 están disponibles para las aplicaciones de big data autoadministradas.

Amazon Kinesis Streams

Amazon Kinesis Streams[10] le permite crear aplicaciones personalizadas para procesar o analizar datos en streaming en tiempo real. Amazon Kinesis Streams puede capturar de forma continua terabytes de datos por hora provenientes de cientos de miles de orígenes, como secuencias de clics en sitios web, transacciones financieras, fuentes de redes sociales, registros de TI y eventos de seguimiento de ubicación.

Con Amazon Kinesis Client Library (KCL) puede crear aplicaciones de Amazon Kinesis y usar datos en streaming para alimentar paneles en tiempo real, generar alertas y aplicar precios y publicidad dinámicos. También puede emitir datos desde Amazon Kinesis Streams a otros servicios de AWS, como Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon Elastic MapReduce (Amazon EMR) y AWS Lambda.

Aprovisione el nivel de entrada y salida que requiere su flujo de datos, en bloques de 1 megabyte por segundo (MB/s), con la consola de administración, la API[11] o los SDK[12] de AWS. El tamaño del flujo de datos se puede ampliar o reducir en cualquier momento sin reiniciar el flujo y sin que los orígenes de datos resulten afectados. En cuestión de segundos, los datos introducidos en el flujo estarán disponibles para analizarlos.

El flujo de datos se almacena en varias zonas de disponibilidad de una región durante 24 horas. Durante ese periodo, los datos están disponibles para lectura, relectura, reposición y análisis, o para su traslado a un almacén a largo plazo (como Amazon S3 o Amazon Redshift). Con KCL los desarrolladores podrán centrarse en crear aplicaciones de negocio, al mismo tiempo que desaparecen las tareas pesadas que no aportan valor asociadas al balanceador de carga del streaming de datos, la coordinación de los servicios distribuidos y el procesamiento de datos tolerante a errores.

Patrones de uso ideales

Amazon Kinesis Streams es útil cuando se necesita mover rápidamente datos de donde se han producido (los orígenes de datos) y procesarlos de forma continua. El procesamiento de los datos puede consistir en transformarlos antes de emitirlos hacia otro almacenamiento, generar métricas y análisis en tiempo real, derivar y agregar múltiples flujos para formar otros más complejos, o para procesamiento en etapas posteriores. Estos son algunos usos típicos de Amazon Kinesis Streams para la realización de análisis.

- **Análisis de datos en tiempo real:** Amazon Kinesis Streams permite efectuar análisis en tiempo real de datos en streaming, por ejemplo secuencias de clic en sitios web y análisis de conexión con los clientes.

- **Introducción y procesamiento de logs y datos:** con Amazon Kinesis Streams, puede hacer que los generadores de datos envíen datos directamente en un flujo de Amazon Kinesis. Por ejemplo, puede enviar registros del sistema y de aplicaciones a Amazon Kinesis Streams y tener acceso a ese flujo para procesarlos en cuestión de segundos. Esto impide la pérdida de datos del registro si se produce un error en el front-end o en el servidor de la aplicación y, al mismo tiempo, reduce el almacenamiento local del registro en el origen. Amazon Kinesis Streams permite la entrada acelerada de los datos, ya que no es necesario acumular lotes de datos en los servidores antes de enviarlos.
- **Métricas e informes en tiempo real:** puede utilizar los datos introducidos en Amazon Kinesis Streams para extraer las métricas y generar indicadores clave de rendimiento y, de este modo, impulsar informes y paneles en tiempo real. De este modo, la lógica de la aplicación puede procesar los datos mientras se transmiten de forma continua, en lugar de esperar a que lleguen lotes completos.

Modelo de costos

Amazon Kinesis Streams tiene un sencillo modelo de pago por uso, sin costos por anticipado y sin cuotas mínimas: solo pagará por los recursos que consuma. Un flujo de Amazon Kinesis se compone de uno o varios fragmentos; cada fragmento ofrece una capacidad de cinco transacciones de lectura por segundo, hasta un máximo de 2 MB de datos leídos por segundo. Cada fragmento puede admitir hasta 1 000 transacciones de lectura por segundo y un total máximo de 1 MB de datos escritos por segundo.

La capacidad de los datos del flujo es una función del número de fragmentos que especifica para el flujo. La capacidad total del flujo es la suma de las capacidades de sus fragmentos. Solo hay dos componentes de precio: un cargo por hora por fragmento y un cargo por cada millón de transacciones PUT. Para obtener más información, consulte Precios de Amazon Kinesis Streams.[13] Las aplicaciones que se ejecutan en Amazon EC2 y que procesan flujos de Amazon Kinesis también incurren en los costos estándar de Amazon EC2.

Desempeño

Amazon Kinesis Streams permite elegir la capacidad de rendimiento necesaria a partir del concepto de fragmentos. Cada fragmento de un flujo de Amazon Kinesis permite capturar hasta 1 megabyte de datos por segundo con una

velocidad de 1 000 transacciones de escritura por segundo. Sus aplicaciones de Amazon Kinesis pueden leer datos de cada fragmento con una velocidad de hasta 2 megabytes por segundo. Puede emplear tantos fragmentos como necesite para conseguir la capacidad de rendimiento que desee; por ejemplo, un flujo de datos de 1 gigabyte por segundo requeriría 1 024 fragmentos.

Durabilidad y disponibilidad

Amazon Kinesis Streams replica los datos de forma sincrónica en tres zonas de disponibilidad de una región de AWS, lo que proporciona un alto nivel de disponibilidad y durabilidad de los datos. Asimismo, puede almacenar un cursor en DynamoDB para realizar un seguimiento duradero de lo que se ha leído de un flujo de Amazon Kinesis. En caso de que la aplicación falle mientras está leyendo datos del flujo, puede reiniciarla y usar el cursor para continuar en el punto exacto donde se produjo el error.

Escalabilidad y elasticidad

Puede aumentar o reducir la capacidad del flujo en cualquier momento según las necesidades operativas o del negocio y sin interrumpir el procesamiento en curso. Mediante el uso de las llamadas a la API o las herramientas para desarrolladores, puede automatizar el escalado de su entorno de Amazon Kinesis Streams para satisfacer la demanda y garantizar que solo paga por lo que necesita.

Interfaces

Amazon Kinesis Streams cuenta con dos interfaces: la de entrada, que usan los productores de datos para introducir datos en Amazon Kinesis Streams, y la de salida, que se usa para procesar y analizar los datos entrantes. Los productores pueden escribir los datos mediante la API PUT de Amazon Kinesis, un Kit de desarrollo de software (SDK) o abstracción de kit de herramientas de AWS[14], Amazon Kinesis Producer Library (KPL)[15] o el agente de Amazon Kinesis.[16]

Para el procesamiento de los datos que ya se han introducido en un flujo de Amazon Kinesis existen bibliotecas cliente que permiten crear y ejecutar aplicaciones de procesamiento de datos en streaming en tiempo real. KCL[17] actúa como intermediario entre Amazon Kinesis Streams y las aplicaciones de negocio que contienen la lógica de procesamiento específica. También puede integrarse la lectura desde un flujo de Amazon Kinesis en Apache Storm mediante Amazon Kinesis Storm Spout[18].

Patrones de uso no recomendados

No se recomienda el uso de Amazon Kinesis Streams en los siguientes escenarios:

- **Rendimiento regular a pequeña escala:** aunque Amazon Kinesis Streams funciona con datos en streaming a una velocidad de 200 KB/s o inferior, está diseñado y optimizado para velocidades mayores.
- **Almacenamiento y análisis de datos a largo plazo:** Amazon Kinesis Streams no es adecuado para el almacenamiento de datos a largo plazo. De forma predeterminada, los datos se conservan durante 24 horas, plazo que puede extenderse hasta los 7 días. Los datos que deban almacenarse durante más de 7 días pueden trasladarse a otro servicio de almacenamiento de larga duración, como Amazon S3, Amazon Glacier, Amazon Redshift o DynamoDB.

AWS Lambda

AWS Lambda[19] le permite ejecutar código sin aprovisionar ni administrar servidores. Solo paga el tiempo de procesamiento que consume, sin ningún cargo mientras su código no se ejecuta. Con Lambda puede ejecutar código para prácticamente cualquier tipo de aplicación o servicio back-end, sin ningún esfuerzo de administración. Solo tiene que cargar su código y Lambda se ocupará de todo lo necesario para ejecutarlo y escalarlo con alta disponibilidad. Puede configurar el código para que se ejecute automáticamente como respuesta a otros servicios de AWS o llamarlo directamente desde cualquier web o aplicación móvil.

Patrones de uso ideales

Lambda le permite ejecutar código en respuesta a activaciones como cambios de datos, modificaciones del estado del sistema o acciones de los usuarios. Lambda puede activarse directamente desde servicios de AWS como Amazon S3, DynamoDB, Amazon Kinesis Streams, Amazon Simple Notification Service (Amazon SNS) y Amazon CloudWatch, lo que le permite crear diversos sistemas de procesamiento de datos en tiempo real.

- **Procesamiento de archivos en tiempo real:** puede activar Lambda para invocar un proceso cuando se haya cargado o modificado un archivo en Amazon S3. Por ejemplo, puede cambiar una imagen de color a escala de grises una vez cargada en Amazon S3.

- **Procesamiento de flujo en tiempo real:** puede usar Amazon Kinesis Streams y Lambda para procesar datos de streaming para análisis de secuencias de clics, filtrado de logs y análisis de redes sociales.
- **Extracción, transformación y carga:** puede utilizar Lambda para ejecutar trabajos que transforman los datos y los cargan de un repositorio de datos a otro.
- **Alternativa a cron:** utilice expresiones de programación para ejecutar una función de Lambda a intervalos regulares como una solución más económica y disponible que ejecutar cron en una instancia EC2.
- **Procesar eventos de AWS:** muchos otros servicios, como, por ejemplo, AWS CloudTrail, pueden actuar como orígenes de eventos tan solo con iniciar sesión en Amazon S3 y con las notificaciones de bucket de S3 para activar las funciones de Lambda.

Modelo de costos

Con Lambda paga únicamente por lo que utiliza. El cargo se basa en el número de solicitudes de las funciones y el tiempo de ejecución del código. La capa gratuita de Lambda incluye 1 millón de solicitudes gratis al mes y 400 000 GB-segundos de tiempo de computación al mes. A partir de ese número, el cargo es de 0,20 USD por millón de solicitudes (0,0000002 USD por solicitud). Adicionalmente, la duración de la ejecución del código se cobra en función de la memoria asignada. Se le cobra 0,00001667 USD por cada GB por segundo utilizado. Para obtener más información, consulte Precios de AWS Lambda.

Desempeño

Una vez que su código se haya transferido a Lambda, normalmente las funciones estarán disponibles a los pocos segundos de haberlo cargado. Lambda está diseñado para procesar los eventos en cuestión de milisegundos. La latencia será mayor inmediatamente después de haber creado o actualizado una función en Lambda, o cuando no se haya usado recientemente.

Durabilidad y disponibilidad

Lambda está diseñado para el uso de replicación y redundancia con el fin de conseguir una alta disponibilidad, tanto del servicio en sí como de las funciones de Lambda que utiliza. No hay intervalos de mantenimiento ni de inactividad programada para ninguno de ellos. En caso de error, las funciones de Lambda

invocadas sincrónicamente responderán con una excepción. Las funciones de Lambda invocadas de forma asincrónica vuelven a intentarse al menos tres veces, al cabo de las cuales el evento puede rechazarse.

Escalabilidad y elasticidad

No hay límite en el número de funciones de Lambda que pueden ejecutarse. Sin embargo, Lambda aplica una limitación de seguridad predeterminada de 100 ejecuciones simultáneas por cuenta y por región. Los miembros del equipo de soporte de AWS pueden aumentar este límite.

Lambda está diseñado para encargarse de adaptar la capacidad automáticamente. No hay ningún límite estricto para escalar una función. Lambda asigna capacidad dinámicamente para afrontar la velocidad de entrada de eventos.

Interfaces

Las funciones de Lambda pueden administrarse de distintas formas. Puede enumerar, eliminar, actualizar y supervisar fácilmente sus funciones de Lambda en el panel de la consola de Lambda. También puede usar AWS CLI y AWS SDK para administrar las funciones de Lambda.

Las funciones de Lambda pueden activarse por un evento de AWS, como las notificaciones bucket de Amazon S3, los flujos de DynamoDB, los registros de CloudWatch, Amazon SES, Amazon Kinesis Streams, Amazon SNS o Amazon Cognito, entre otros. Cualquier llamada API de cualquier servicio compatible con CloudTrail pude procesarse como un evento en Lambda respondiendo a los registros de auditoría de CloudTrail. Para obtener más información acerca de orígenes de eventos, consulte Componentes básicos: Función AWS Lambda y Orígenes de eventos admitidos.[20]

Lambda es compatible con distintos lenguajes de programación, como Java, Node.js y Python. Su código puede incluir bibliotecas ya existentes, incluso las nativas. Las funciones de Lambda pueden iniciar procesos utilizando los lenguajes compatibles con Amazon Linux AMI[21]. como Bash, Go y Ruby. Para obtener más información, consulte la documentación de Node.js[22], Python[23] y Java[24].

Patrones de uso no recomendados

No se recomienda el uso de Lambda en los siguientes escenarios:

- **Aplicaciones de ejecución prolongada**: cada función de Lambda debe completarse antes de 300 segundos. En el caso de aplicaciones prolongadas que puedan requerir que las tareas se ejecuten durante más de cinco minutos, se recomienda usar Amazon EC2 o crear una cadena de funciones Lambda en la que la función 1 llama a la función 2, que a su vez llama a la función 3 y así sucesivamente hasta completar el proceso.
- **Sitios web dinámicos**: si bien es posible ejecutar un sitio web estático con Lambda, la ejecución de un sitio web muy dinámico y de gran volumen puede tener un desempeño prohibitivo. En esta situación sería recomendable el uso de Amazon EC2 y Amazon CloudFront.
- **Aplicaciones con estado**: el código de Lambda debe escribirse en un estilo "sin estado"; es decir, debería asumir que no hay ninguna afinidad hacia la infraestructura de computación subyacente. El acceso al sistema de archivos local, los procesos secundarios y otros elementos similares no pueden prolongarse más allá del tiempo de vida de la solicitud, y cualquier estado persistente debe almacenarse en Amazon S3, DynamoDB u otro servicio de almacenamiento disponible en Internet.

Amazon EMR

Amazon EMR[25] es un entorno informático altamente distribuido para procesar y almacenar datos de forma sencilla, rápida y económica. Amazon EMR utiliza Apache Hadoop, una infraestructura de código abierto, para distribuir los datos y el procesamiento en un clúster de tamaño variable de instancias Amazon EC2 y le permite usar las herramientas de Hadoop más comunes, como Hive, Pig y Spark, entre otras. Hadoop proporciona una plataforma para ejecutar el procesamiento y los análisis de big data. Amazon EMR realiza todas las tareas pesadas asociadas con el aprovisionamiento, administración y mantenimiento de la infraestructura y el software de un clúster Hadoop.

Patrones de uso ideales

La plataforma flexible de Amazon EMR reduce los problemas del procesamiento a gran escala al convertir los conjuntos de datos en trabajos más pequeños

y distribuirlos entre muchos nodos de cómputo de un clúster Hadoop. Esta capacidad resulta adecuada para muchos patrones de uso en el análisis de big data. A continuación se ofrecen algunos ejemplos:

- Procesamiento y análisis de registros
- Extracción, transformación y carga (ETL) de grandes volúmenes de datos y movimiento de datos
- Modelos de riesgos y análisis de amenazas
- Segmentación de anuncios y análisis de secuencias de clics
- Genómica
- Análisis predictivos
- Minería y análisis de datos ad hoc

Para obtener más información, consulte el documento técnico acerca de las prácticas recomendadas para Amazon EMR.[26]

Modelo de costos

Con Amazon EMR puede lanzar un clúster persistente que permanezca indefinidamente o un clúster temporal que finalice cuando se complete el análisis. En cualquiera de estos dos escenarios, solo paga las horas en que el clúster está activo.

Amazon EMR admite una serie de tipos de instancia Amazon EC2 (estándar, alta CPU, alta memoria, uso intensivo de E/S, etc.) y todas las opciones de precio de instancias Amazon EC2 (bajo demanda, reservado y puntual). Cuando lance un clúster de Amazon EMR (denominado también “flujo de trabajo”), elegirá cuántas instancias Amazon EC2 y de qué tipo se deben aprovisionar. El precio de Amazon EMR se suma al precio de Amazon EC2. Para obtener más información, consulte Precios de Amazon EMR.[27]

Desempeño

El desempeño de Amazon EMR viene determinado por el tipo de las instancias EC2 en las que elija ejecutar el clúster y por cuántas decida usar para efectuar el análisis. Debe elegir un tipo de instancia adecuado para sus requisitos de procesamiento, con suficiente memoria, almacenamiento y capacidad de procesamiento. Para obtener más información sobre las especificaciones de las instancias EC2, consulte Tipos de instancias de Amazon EC2.[28]

Durabilidad y disponibilidad

De forma predeterminada, Amazon EMR es tolerante a los errores de los nodos principales, y continúa la ejecución si deja de funcionar un nodo esclavo. Actualmente, Amazon EMR no aprovisiona automáticamente otro nodo para sustituir a los nodos esclavos que dejan de funcionar, pero los clientes pueden supervisar el estado de los nodos y reemplazar los que hayan fallado usando CloudWatch.

Para ayudar a abordar la improbable circunstancia de un error en un nodo principal, es recomendable que realice un backup de los datos en un almacén persistente como Amazon S3. También puede elegir ejecutar Amazon EMR con la distribución MapR[29], que proporciona una arquitectura distinta a NameNode que puede tolerar varios errores simultáneamente con restauración y conmutación por error automáticos. Los metadatos se distribuyen y replican de la misma forma que los datos. Con una arquitectura sin NameNode, no existe ningún límite práctico en cuanto a la cantidad de archivos que pueden almacenarse, ni tampoco hay ningún tipo de dependencia en los sistemas de almacenamiento externos conectados en red.

Escalabilidad y elasticidad

Con Amazon EMR, es sencillo redimensionar un clúster en ejecución[30]. Puede añadir nodos principales que contengan Hadoop Distributed File System (HDFS) en cualquier momento para aumentar la capacidad de procesamiento y almacenamiento de HDFS (además de la velocidad). Además, puede usar Amazon S3 de forma nativa o con EMFS de forma complementaria o alternativa a HDFS local, lo que le permite desacoplar la memoria y ejecutar desde su almacenamiento, mejorando así la flexibilidad y la economía.

También puede añadir y quitar en cualquier momento los nodos que pueden procesar tareas de Hadoops pero que no mantienen HDFS. Algunos clientes añaden cientos de instancias a sus clústeres cuando se produce el procesamiento por lotes y eliminan las instancias adicionales cuando termina el procesamiento. Por ejemplo, es posible que no sepa cuántos datos tendrán que administrar los clústeres en seis meses, o puede que haya picos en las necesidades de procesamiento. Con Amazon EMR no necesita adivinar sus requisitos futuros ni aprovisionar para un aumento de la demanda, ya que puede añadir o quitar capacidad fácilmente en cualquier momento.

Asimismo, puede añadir nuevos clústeres de distintos tamaños y retirarlos en cualquier momento con solo unos clics en la consola o a través de una llamada API[31] mediante programación.

Interfaces

Amazon EMR admite muchas herramientas basadas en Hadoop que se pueden usar para el análisis de big data, y cada una tiene su propia interfaz. A continuación se incluye un breve resumen de las opciones más populares:

Hive

Hive es un almacén de datos y paquete de análisis de código abierto que se ejecuta sobre Hadoop. Hive funciona con Hive QL, un lenguaje basado en SQL que permite a los usuarios estructurar, resumir y consultar datos. Hive QL no solo utiliza el lenguaje SQL estándar, sino que incorpora una compatibilidad excelente con funciones map/reduce y tipos de datos complejos y ampliables definidos por el usuario, como JSON y Thrift. Esta capacidad permite procesar orígenes de datos complejos y no estructurados, como documentos de texto y archivos de registro.

Hive permite extensiones de usuario mediante funciones definidas por el usuario, escritas en Java. Amazon EMR ha realizado numerosas mejoras en Hive, incluida la integración directa con DynamoDB y Amazon S3. Por ejemplo, con Amazon EMR puede cargar particiones de tablas automáticamente desde Amazon S3, escribir datos en tablas en Amazon S3 sin usar archivos temporales y obtener acceso a recursos de Amazon S3, como scripts de operaciones map y reduce personalizadas y bibliotecas adicionales. Para obtener más información, consulte Apache Hive[32] en *Amazon EMR Release Guide*.

Pig

Pig es un paquete de análisis de código abierto que se ejecuta sobre Hadoop. Pig funciona con Pig Latin, un lenguaje similar a SQL que permite a los usuarios estructurar, resumir y consultar datos. Además de las operaciones de estilo SQL, Pig Latin también incluye una compatibilidad excepcional con funciones map y reduce y tipos de datos complejos y ampliables definidos por el usuario. Esta capacidad permite procesar orígenes de datos complejos y no estructurados, como documentos de texto y archivos de registro.

Pig permite extensiones de usuario mediante funciones definidas por el usuario escritas en Java. Amazon EMR ha realizado numerosas mejoras en Pig, como la

capacidad de usar varios sistemas de archivos (normalmente, Pig solo puede obtener acceso a un sistema de archivos remoto), la posibilidad de cargar archivos JAR y scripts del cliente desde Amazon S3 (por ejemplo, "REGISTER s3://my-bucket/piggybank.jar") y funcionalidad adicional para el procesamiento de String y DateTime. Para obtener más información, consulte Apache Pig[33] en *Amazon EMR Release Guide*.

Spark

Spark es un motor de análisis de datos de código abierto que funciona sobre Hadoop con los fundamentos de MapReduce en memoria. Spark proporciona velocidad adicional para determinados análisis y es la base de otras herramientas, como Shark (almacenamiento de datos por medio de SQL), Spark Streaming (aplicaciones de streaming), GraphX (sistemas gráficos) y MLlib (aprendizaje automático). Para obtener más información, consulte la entrada de blog Installing Apache Spark on an Amazon EMR Cluster.[34]

HBase

HBase es una base de datos distribuida no relacional de código abierto que sigue el modelo de BigTable de Google. Se desarrolló como parte del proyecto Hadoop de Apache Software Foundation y se ejecuta sobre el sistema de archivos distribuido Hadoop (HDFS) para proporcionar una funcionalidad como la de BigTable. HBase proporciona una forma eficiente y tolerante a errores de almacenar grandes cantidades de datos dispersos mediante la compresión y almacenamiento basados en columnas. Además, HBase proporciona una búsqueda rápida de los datos, ya que los datos están almacenados en memoria en lugar de en disco.

HBase se ha optimizado para las operaciones de escritura secuenciales, y es muy eficiente para las inserciones, actualizaciones y eliminaciones por lotes. HBase funciona perfectamente con Hadoop, compartiendo su sistema de archivos y sirviendo de entrada y salida directa para los trabajos de Hadoop. HBase también se integra con Apache Hive, lo que permite ejecutar consultas de tipo SQL de tablas HBase, uniones con tablas basadas en Hive y compatibilidad con la conectividad de bases de datos Java (JDBC). Con Amazon EMR, puede realizar backups de HBase en Amazon S3 (completos o incrementales y manuales o automáticos) y puede restaurar los datos de un backup creado previamente. Para obtener más información, consulte HBase y EMR[35] en la *Guía del desarrollador de Amazon EMR*.

Impala

Impala es una herramienta de código abierto del ecosistema de Hadoop para la consulta ad hoc interactiva usando la sintaxis SQL. En lugar de usar MapReduce, emplea un motor de procesamiento en paralelo de forma masiva (MPP) similar al que se encuentra en los sistemas de administración de bases de datos relacionales (RDBMS) tradicionales. Con esta arquitectura, puede consultar sus datos en tablas HDFS o HBase muy rápidamente y aprovechar la capacidad de Hadoop de procesar tipos de datos diversos y proporcionar el esquema en tiempo de ejecución. Esto convierte a Impala en una excelente herramienta para realizar análisis interactivos de baja latencia.

Impala tiene también funciones definidas por el usuario en Java y C++, y puede conectarse a herramientas de inteligencia de negocio (BI) a través de controladores ODBC y JDBC. Impala utiliza el metaalmacén de Hive para mantener información sobre los datos de entrada, incluidos los nombres de particiones y los tipos de datos. Para obtener más información, consulte Impala y EMR[36] en la *Guía del desarrollador de Amazon EMR*.

Hunk

Hunk fue desarrollado por Splunk para que los datos almacenados sean accesibles, utilizables y evaluables para cualquiera. Con Hunk puede explorar, analizar y visualizar de manera interactiva los datos almacenados en Amazon EMR y Amazon S3, aprovechando los análisis de Splunk sobre Hadoop. Para obtener más información, consulte la sección Amazon EMR con Hunk: Splunk Analytics para Hadoop y NoSQL.[37]

Presto

Presto es un motor de consultas SQL distribuido y de código abierto optimizado para el análisis ad-hoc de datos con baja latencia. Es conforme al estándar ANSI SQL, incluyendo consultas complejas, agregaciones, uniones y funciones de ventana. Presto puede procesar datos de varios orígenes, incluido el sistema de archivos distribuido de Hadoop (HDFS) y Amazon S3.

Otras herramientas de terceros

Amazon EMR admite también una gran variedad de otras aplicaciones y herramientas comunes en el ecosistema de Hadoop, como R (estadísticas), Mahout (aprendizaje automático), Ganglia (supervisión), Accumulo (base de datos NoSQL segura), Hue (interfaz de usuario para analizar datos de Hadoop), Sqoop (conector de bases de datos relacionales), HCatalog (administración de tablas y almacenamiento), etc.

Asimismo, puede instalar su propio software sobre Amazon EMR para ayudar a abordar las necesidades de su negocio. AWS ofrece la posibilidad de mover rápidamente grandes cantidades de datos de Amazon S3 a HDFS, de HDFS a Amazon S3 y entre buckets de Amazon S3 mediante S3DistCp[38] de Amazon EMR, una extensión de la herramienta de código abierto DistCp que usa MapReduce para mover eficazmente grandes cantidades de datos.

Opcionalmente puede usar el sistema de archivos de EMR (EMRFS), que es una implementación de HDFS que permite a los clústeres de Amazon EMR almacenar datos en Amazon S3. Puede activar el cifrado en el servidor y en el cliente de Amazon S3 y también una vista común para EMRFS. Cuando se utiliza EMRFS, se crea un almacén de metadatos en DynamoDB de forma transparente para ayudar a administrar las interacciones con Amazon S3 y permitirle tener varios clústeres de EMR que utilicen fácilmente en Amazon S3 los mismos metadatos y almacenamiento de EMRFS.

Patrones de uso no recomendados

No se recomienda el uso de Amazon EMR en los siguientes escenarios:

- **Conjuntos de datos pequeños:** Amazon EMR se ha diseñado para el procesamiento en paralelo de forma masiva; si su conjunto de datos es lo suficientemente pequeño para ejecutarse rápidamente en una sola máquina y un solo subproceso, la sobrecarga añadida para los trabajos map y reduce puede no merecer la pena para pequeños conjuntos de datos que puedan procesarse fácilmente en memoria en un solo sistema.
- **Requisitos de transacción ACID**: a pesar de que hay formas de lograr ACID (atomicidad, coherencia, aislamiento y durabilidad, por sus siglas en inglés) o un nivel limitado de ACID en Hadoop, cuando la carga de trabajo plantea requisitos más estrictos puede ser recomendable emplear otra base de datos, como Amazon RDS o una base de datos relacional que ejecute Amazon EC2.

Amazon Machine Learning

Amazon ML[39] es un servicio que facilita el uso de análisis predictivos y tecnología de aprendizaje automático. Amazon ML ofrece asistentes y herramientas de visualización que le guían por el proceso de creación de modelos de aprendizaje automático (ML) sin tener que aprender algoritmos y tecnologías ML complejos. Una vez elaborados los modelos, Amazon ML permite obtener fácilmente

predicciones para una aplicación mediante operaciones de API y sin tener que escribir código de generación de predicciones personalizado ni administrar ninguna infraestructura.

Amazon ML puede crear modelos de ML basados en datos almacenados en Amazon S3, Amazon Redshift o Amazon RDS. Los asistentes integrados le guiarán por las etapas de exploración interactiva de los datos, entrenamiento del modelo ML, evaluación de la calidad del modelo y ajuste de las salidas para orientarlas a los objetivos de negocio. Una vez preparado un modelo, puede solicitar predicciones por lotes o usando la API en tiempo real de baja latencia.

Patrones de uso ideales

Amazon ML es ideal para descubrir patrones en sus datos y usar esos patrones crear modelos de ML que generen predicciones sobre nuevos puntos de datos aún no vistos. Por ejemplo, puede hacer lo siguiente:

- **Capacitar las aplicaciones para señalar transacciones sospechosas:** crear un modelo de ML que prediga si una transacción nueva es legítima o fraudulenta.
- **Prever la demanda del producto:** introducir la información de pedidos histórica para predecir cantidades futuras de pedidos.
- **Personalizar el contenido de las aplicaciones:** predecir qué artículos interesarán más al usuario y obtener esas predicciones desde la aplicación en tiempo real.
- **Predecir la actividad de los usuarios:** analizar el comportamiento del usuario para personalizar su sitio web y proporcionar una mejor experiencia de usuario.
- **Escuchar las redes sociales:** recibir y analizar las fuentes de redes sociales que puedan afectar a las decisiones de negocio.

Modelo de costos

Con Amazon ML solo paga por lo que usa. No se requieren pagos mínimos ni compromisos iniciales. Amazon ML factura una tasa por hora de ejecución para la creación de modelos predictivos y un pago adicional por el número de predicciones generadas para la aplicación. Para las predicciones en tiempo real también se paga un cargo por hora según la capacidad reservada, basándose en la cantidad de memoria requerida para ejecutar el modelo.

El cargo por análisis de datos, entrenamiento del modelo y evaluación se basa en el número de horas de ejecución requerido para llevarlos a cabo, y depende del volumen de datos de entrada, el número de atributos que contienen y el número y tipos de transformaciones aplicadas. La tasa por análisis de datos y creación de modelos es de 0,42 USD por hora. Las tasas por predicción se dividen en las categorías por lotes y en tiempo real. El cargo de las predicciones por lotes es de 0,10 USD por cada 1 000 predicciones, redondeado al alza hasta el siguiente millar completo, mientras que para las predicciones en tiempo real es de 0,0001 USD por predicción, redondeado al alza hasta el centavo siguiente. En el caso de las predicciones en tiempo real, también existe un cargo de capacidad reservada de 0,001 USD por hora para cada 10 MB de memoria aprovisionado para su modelo.

Durante la creación de los modelos debe especificar el tamaño de memoria máximo de cada modelo para poder gestionar los costos y controlar el desempeño predictivo. El cargo por capacidad reservada solo se paga mientras el modelo está habilitado para predicciones en tiempo real. Los cargos por almacenamiento de datos en Amazon S3, Amazon RDS o Amazon Redshift se facturan por separado. Para obtener más información, consulte Precios de Amazon Machine Learning.[40]

Desempeño

El tiempo dedicado a crear modelos o solicitar predicciones por lotes desde ellos depende del número de registros de los datos de entrada, los tipos y distribución de los atributos de esos registros y de la complejidad de la "receta" de procesamiento de datos que especifique.

La mayoría de las solicitudes de predicción en tiempo real tienen un tiempo de respuesta inferior a 100 ms, lo bastante rápido para aplicaciones interactivas en web, móviles o de escritorio. El tiempo exacto que requiere la API de tiempo real para generar una predicción depende del tamaño del registro de entrada y la complejidad de la "receta"[41] de procesamiento de datos asociada al modelo de ML que genera las predicciones. De forma predeterminada, cada modelo de ML habilitado para predicciones en tiempo real puede usarse para solicitar hasta 200 transacciones por segundo, y este número puede incrementarse consultándolo con el soporte al cliente. Para supervisar el número de predicciones que solicitan sus modelos de ML puede usar las métricas de CloudWatch.

Durabilidad y disponibilidad

Amazon ML está diseñado para una alta disponibilidad. No hay intervalos de mantenimiento ni de inactividad programada. El servicio se ejecuta en los centros de datos de Amazon, cuya alta disponibilidad está demostrada, y con replicación de la pila de servicios en tres centros situados en cada región de AWS para conseguir tolerancia a errores en caso de un problema en un servidor o una caída del servicio de una zona de disponibilidad.

Escalabilidad y elasticidad

Puede procesar conjuntos de datos de hasta 100 GB para crear modelos de ML o para solicitar predicciones por lotes. En caso de grandes volúmenes de predicciones por lotes, puede dividir los registros de datos de entrada en paquetes individuales para hacer posible su procesamiento.

De forma predeterminada pueden ejecutarse hasta cinco tareas simultáneas, número que puede ampliarse poniéndose en contacto con el servicio de atención al cliente. Amazon ML es servicio administrado, de modo que no hay necesidad de aprovisionar servidores y por lo tanto al ir creciendo la aplicación puede adaptar la capacidad sin tener que reservar capacidad de más ni pagar por recursos que no se utilizan.

Interfaces

Crear un origen de datos es tan sencillo como introducir los datos en Amazon S3. También puede obtener los datos directamente de Amazon Redshift o bases de datos MySQL administradas por Amazon RDS. Una vez definido el origen de datos, puede interactuar con Amazon ML a través de la consola. El acceso por programa a Amazon ML se consigue a través de los SDK de AWS y la API de Amazon ML.[42] También puede crear y administrar entidades de Amazon ML con AWS CLI, disponible para sistemas con Windows, Mac y Linux/UNIX.

Patrones de uso no recomendados

No se recomienda el uso de Amazon ML en los siguientes escenarios:

- **Conjuntos de datos muy grandes:** aunque Amazon ML admite hasta 100 GB de datos, actualmente no contempla la entrada de datos en la escala de terabytes. Para estos casos es habitual usar Amazon EMR para ejecutar la biblioteca de aprendizaje automático (MLlib) de Spark.

- **Tareas de aprendizaje no contempladas:** Amazon ML se puede usar para crear modelos de ML que realicen clasificación binaria (elegir una de dos opciones y proporcionar una medida de confianza), clasificación de varias opciones (ampliar más allá de dos opciones) o regresión numérica (predecir un número directamente). Las tareas de ML no contempladas, como predicción de secuencias o uso de clústeres sin supervisión pueden afrontarse usando Amazon EMR para ejecutar Spark y MLlib.

Amazon DynamoDB

Amazon DynamoDB[43] es un servicio de base de datos NoSQL rápido y totalmente administrado que permite almacenar y recuperar de manera sencilla y económica cualquier cantidad de datos, así como atender cualquier nivel de tráfico de solicitudes. DynamoDB contribuye a reducir la carga administrativa que supone tener que utilizar y escalar un clúster de base de datos distribuido de alta disponibilidad. Esta alternativa de almacenamiento satisface los requisitos de latencia y desempeño de las aplicaciones más exigentes al proporcionar una latencia inferior a diez milisegundos y un desempeño extraordinariamente rápido y predecible con escalabilidad perfecta del rendimiento y el almacenamiento.

DynamoDB almacena datos estructurados en tablas, indexados por clave principal, y permite obtener un acceso de lectura y escritura de baja latencia a los elementos cuyo tamaño va de 1 byte a 400 KB. DynamoDB permite tres tipos de datos (número, cadena y binario), tanto en forma escalar como en conjuntos multivalor. Admite almacenes de documentos como JSON, XML o HTML en estos tipos de datos. Las tablas no presentan un esquema fijo, por lo que cada elemento puede tener un número de atributos distinto, La clave principal puede ser una clave hash de un solo atributo o una clave compuesta de rango hash.

DynamoDB ofrece índices secundarios tanto globales como locales para conseguir mayor flexibilidad al consultar atributos que no sean la clave principal. DynamoDB ofrece lecturas con coherencia final (de forma predeterminada) y lecturas de coherencia alta (opcional), así como transacciones implícitas de nivel de elemento para operaciones put, update, delete y condicionales de elementos, además para operaciones de incremento y decremento.

DynamoDB se integra con otros servicios, como Amazon EMR, Amazon Redshift, AWS Data Pipeline y Amazon S3, lo que permite realizar tareas de análisis,

almacenamiento de datos, importación y exportación de datos, backup y archivado.

Patrones de uso ideales

DynamoDB es ideal para las aplicaciones existentes o nuevas que necesitan una base de datos NoSQL flexible con bajas latencias de lectura y escritura, y la posibilidad de aumentar o reducir el almacenamiento y el rendimiento según sea necesario sin hacer cambios en el código y sin tiempos de inactividad.

Entre los casos de uso comunes, se incluyen los siguientes:

- Aplicaciones móviles
- Juegos
- Publicación de anuncios online
- Votaciones en directo
- Interacción con el público en eventos en directo
- Redes de sensores
- Adquisición de registros
- Control de acceso a contenido basado en web
- Almacenamiento de metadatos para objetos de Amazon S3
- Carros de compra de comercio electrónico
- Administración de sesiones web

Muchos de estos casos de uso requieren una base de datos de alta disponibilidad y escalabilidad, ya que el tiempo de inactividad o la reducción del desempeño tiene un impacto negativo inmediato sobre el negocio de una organización.

Modelo de costos

Con DynamoDB solo paga por lo que usa y no hay ninguna cuota mínima. DynamoDB tiene tres componentes de precio: capacidad de desempeño aprovisionada (por hora), almacenamiento de datos indexados (por GB al mes) y transferencia de datos entrante o saliente (por GB mensuales). Los nuevos clientes pueden empezar a utilizar DynamoDB de forma gratuita como parte de la capa de uso gratuita de AWS.[44] Para obtener más información, consulte Precios de Amazon DynamoDB.[45]

Desempeño

El uso de SSD y el hecho de limitar la indexación de atributos dan como resultado un alto rendimiento y una baja latencia[46], y reducen drásticamente el costo de las operaciones de lectura y escritura. A medida que crecen los conjuntos de datos, se necesita un desempeño predecible para poder mantener la baja latencia de las cargas de trabajo. Para conseguir este desempeño predecible, hay que definir la capacidad de rendimiento aprovisionada necesaria para una tabla dada.

De forma transparente, el servicio administra el suministro de recursos para conseguir la tasa de rendimiento solicitada, por lo que no tiene que preocuparse por instancias, hardware, memoria y otros factores que podrían influir en la tasa de rendimiento de una aplicación. Las reservas de la capacidad de rendimiento aprovisionada son elásticas y se pueden aumentar o disminuir bajo demanda.

Durabilidad y disponibilidad

DynamoDB integra tolerancia a errores, que replica los datos de manera automática y sincrónica entre tres centros de datos de una región para ofrecer alta disponibilidad y para ayudar a proteger los datos frente a errores de las máquinas individuales o incluso de la instalación. DynamoDB Streams[47] captura toda la actividad de los datos de la tabla y permite configurar una replicación regional de una región geográfica a otra para así conseguir una disponibilidad aún mayor.

Escalabilidad y elasticidad

DynamoDB ofrece alta escalabilidad y elasticidad. No existe ningún límite en cuanto a la cantidad de datos que se pueden almacenar en una tabla de DynamoDB. Además, el servicio asigna automáticamente más almacenamiento a medida que se almacenan más datos con las llamadas API de escritura de DynamoDB. Se crean automáticamente particiones de datos y se reconfiguran las existentes según sea necesario, mientras que el uso de SSD ofrece tiempos de respuesta predecibles de baja latencia a cualquier escala. Además, el servicio es elástico en el sentido de que basta con aumentar[48] o reducir[49] la capacidad de lectura y escritura de una tabla a medida que cambien sus necesidades.

Interfaces

DynamoDB proporciona una API REST de bajo nivel, así como SDK de nivel más alto para Java, .NET y PHP que incluyen la API de REST de bajo nivel y proporcionan algunas funciones de mapeo relacional de objeto (ORM, por sus siglas en inglés). Estas API ofrecen interfaces de administración y de datos para DynamoDB. La API proporciona actualmente trece operaciones que permiten administrar tablas (creación, enumeración, eliminación y obtención de metadatos) y trabajar con atributos (obtención, escritura y eliminación de atributos, consulta mediante un índice y análisis completo).

Aunque el lenguaje SQL estándar no está disponible, puede utilizar la operación Select de DynamoDB para crear consultas de tipo SQL que recuperen un conjunto de atributos en función de los criterios que establezca. También puede trabajar con DynamoDB utilizando la consola.

Patrones de uso no recomendados

No se recomienda el uso de DynamoDB en los siguientes escenarios:

- **Aplicación existente previamente enlazada a una base de datos relacional tradicional:** si se intenta portar una aplicación existente a la nube de AWS y se necesita seguir utilizando una base de datos relacional, se puede optar por utilizar Amazon RDS (Amazon Aurora, MySQL, PostgreSQL, Oracle o SQL Server) o una de las numerosas AMI de base de datos preconfiguradas de Amazon EC2. También puede instalar el software de base de datos de su elección en una instancia EC2 que administre.
- **Uniones o transacciones complejas:** aunque muchas soluciones son capaces de aprovechar DynamoDB para dar soporte a sus usuarios, es posible que su aplicación pueda requerir uniones, transacciones complejas y otras infraestructuras relacionales proporcionadas por plataformas de bases de datos tradicionales. En este caso, puede ser conveniente considerar el uso de Amazon Redshift, Amazon RDS o Amazon EC2 con una base de datos autoadministrada.
- **Datos de objetos binarios grandes (BLOB):** si piensa almacenar datos BLOB grandes (más de 400 KB), como, por ejemplo, vídeo digital, imágenes o música, le convendrá usar Amazon S3. Sin embargo, DynamoDB aún tiene un importante papel que desempeñar en este escenario, ya que realiza un seguimiento de los metadatos (por ejemplo, nombre del elemento, tamaño, fecha de creación, propietario, ubicación, etc.) de los objetos binarios.

- **Datos grandes con una tasa baja de E/S:** DynamoDB utiliza unidades SSD y está optimizado para cargas de trabajo con gran velocidad de E/S por GB almacenado. Si piensa almacenar cantidades de datos muy grandes a las que no se tiene acceso con frecuencia, puede que haya otras opciones de almacenamiento más adecuadas, como Amazon S3.

Amazon Redshift

Amazon Redshift[50] es un servicio rápido y totalmente administrado de almacenamiento de datos a escala de petabytes que permite analizar eficazmente todos los datos de forma sencilla y rentable empleando las herramientas de inteligencia de negocio de las que ya dispone. Está optimizado para conjuntos de datos desde algunos cientos de gigabytes hasta un petabyte o más, y está diseñado para costar menos de la décima parte que la mayoría de las soluciones de almacenamiento de datos tradicionales.

Amazon Redshift ofrece alta velocidad de consulta y E/S para prácticamente cualquier tamaño del conjunto de datos mediante el uso de tecnología almacenamiento en columnas y la paralelización y distribución de las consultas por varios nodos. Automatiza la mayoría de las tareas administrativas comunes asociadas con el aprovisionamiento, configuración, supervisión, backup y protección de un almacén de datos, por lo que es muy fácil y económico de administrar y mantener. Esta automatización le permite crear almacenes de datos a escala de petabytes en solo unos minutos, en lugar de las semanas o meses que requieren las implementaciones locales tradicionales.

Patrones de uso ideales

Amazon Redshift es ideal para el procesamiento analítico online (OLAP) empleando las herramientas de inteligencia de negocio de las que ya dispone. Las organizaciones emplean Amazon Redshift para hacer lo siguiente:

- Analizar datos de ventas globales de varios productos
- Almacenar datos bursátiles históricos
- Analizar clics e impresiones de anuncios
- Acumular datos de juegos
- Analizar tendencias sociales
- Medir la calidad asistencial, la eficacia de las operaciones y el desempeño financiero en el área de atención de salud

Modelo de costos

Para disponer de un clúster de almacén de datos Amazon Redshift no es necesario afrontar gastos anticipados ni asumir compromisos a largo plazo. Esto le libera del gasto de capital y la complejidad de la planificación y la compra de capacidad para el almacén de datos antes de que surja la necesidad. Los cargos se basan en el tamaño y el número de nodos del clúster.

No se aplica ningún cargo adicional por el almacenamiento de backups de hasta el 100% del almacenamiento aprovisionado. Por ejemplo, si dispone de un clúster activo con dos nodos XL para un total de 4 TB de almacenamiento, AWS le proporciona hasta 4 TB de almacenamiento de backup en Amazon S3 sin ningún costo adicional. El almacenamiento de backup que supere el tamaño de almacenamiento aprovisionado y los backups almacenados después de terminar el clúster se facturan según las tarifas estándar de Amazon S3.[51] No existen cargos por transferencia de datos para las comunicaciones entre Amazon S3 y Amazon Redshift. Para obtener más información, consulte Precios de Amazon Redshift.[52]

Desempeño

Amazon Redshift utiliza una serie de innovaciones para obtener un desempeño muy alto en los conjuntos de datos con una capacidad que oscila entre cientos de gigabytes y un petabyte o incluso más. Utiliza un almacenamiento en columnas, compresión de datos y asignaciones de zona para reducir la cantidad de operaciones de E/S necesarias para realizar consultas.

Amazon Redshift cuenta con una arquitectura de procesamiento paralelo de forma masiva (MPP), que paraleliza y distribuye operaciones SQL para que pueda beneficiarse de todos los recursos disponibles. El hardware subyacente está diseñado para un procesamiento de datos de alto desempeño, para lo que utiliza almacenamiento conectado local que maximiza el desempeño entre los CPU y las unidades, y una red de malla 10 GigE que maximiza el desempeño entre los nodos. El desempeño se puede ajustar según sus necesidades almacenamiento de datos: AWS ofrece computación densa (DC) con unidades SSD, así como opciones de almacenamiento denso (DS).

Durabilidad y disponibilidad

Amazon Redshift detecta y reemplaza automáticamente un nodo defectuoso del clúster de almacén de datos. El clúster del almacén de datos pasa a modo de solo lectura hasta que se aprovisione y añada un nodo de sustitución a la base de datos, lo que normalmente solo lleva unos minutos. Amazon Redshift habilita el nodo de sustitución de inmediato y transmite primero los datos a los que se obtiene acceso con más frecuencia desde Amazon S3 primero para permitirle reanudar las consultas de los datos lo antes posible.

Además, el clúster del almacén de datos permanece disponible en caso de error de una unidad, ya que Amazon Redshift refleja los datos de todo el clúster y usa los datos de otro nodo para reconstruir la unidad fallida. Los clústeres de Amazon Redshift residen en una sola zona de disponibilidad.[53] No obstante, si desea disponer de una configuración con varias zonas, puede implementar una imagen reflejada y administrar la replicación y la conmutación por error.

Escalabilidad y elasticidad

Con tan solo unos clics en la consola o una llamada a la API[54] puede cambiar fácilmente el número y el tipo de los nodos de su almacén de datos a medida que cambien sus necesidades de desempeño o capacidad. Amazon Redshift le permite comenzar con solo un nodo de 160 GB y ampliarlo hasta un petabyte o más de datos de usuario comprimidos utilizando varios nodos. Para obtener más información, consulte la sección Clústeres y nodos[55] en el tema Clústeres de Amazon Redshift, de la *guía de administración de Amazon Redshift*.

Cuando se cambia el tamaño, Amazon Redshift coloca el clúster existente en modo de solo lectura, aprovisiona un nuevo clúster del tamaño que desee y realiza una copia en paralelo de los datos del clúster anterior en el nuevo clúster. Durante este proceso solo paga por el clúster de Amazon Redshift activo. Puede continuar realizando consultas en el clúster anterior mientras se aprovisiona el nuevo. Una vez copiados los datos al nuevo clúster, Amazon Redshift redirige las consultas automáticamente a ese clúster y elimina el anterior.

Interfaces

Amazon Redshift cuenta con controladores JDBC y ODBC personalizados que se pueden descargar de la pestaña Connect Client de la consola, lo que le permite usar una amplia gama de clientes SQL que ya conozca. También

puede usar unidades PostgreSQL JDBC y ODBC estándar. Para obtener más información sobre los controladores de Amazon Redshift, consulte Amazon Redshift y PostgreSQL.[56]

Existen numerosos ejemplos de integraciones validadas con muchos proveedores de BI y ETL conocidos.[57] Las cargas y descargas se inician en paralelo en cada nodo de cómputo para maximizar la velocidad de la adquisición de datos por el clúster del almacén de datos y de las transferencias entre Amazon S3 y DynamoDB. Puede cargar fácilmente datos en streaming en Amazon Redshift utilizando Amazon Kinesis Firehose, lo que hace posible análisis casi en tiempo real con las herramientas de inteligencia de negocio y los paneles que ya utiliza hoy. A través de la consola o mediante operaciones de API CloudWatch dispone gratuitamente de métricas del uso de capacidad de computación, memoria, almacenamiento y tráfico de lectura/escritura del clúster de almacenamiento de datos de Amazon Redshift.

Patrones de uso no recomendados

No se recomienda el uso de Amazon Redshift en los siguientes escenarios:

- **Conjuntos de datos pequeños:** Amazon Redshift está concebido para el procesamiento en paralelo en un clúster, de modo que si el conjunto de datos es inferior a 100 gigabytes, no conseguirá todos los beneficios que Amazon Redshift puede ofrecer. En este caso Amazon RDS puede ser una solución más adecuada.
- **Procesamiento de transacciones online (OLTP):** Amazon Redshift se ha diseñado para las cargas de trabajo del almacén de datos que producen capacidades analíticas extraordinariamente rápidas y económicas. Si lo que necesita es un sistema de transacciones rápido, puede optar por una base de datos relacional tradicional basada en Amazon RDS o una base de datos NoSQL como DynamoDB.
- **Datos no estructurados:** los datos en Amazon Redshift se deben estructurar con un esquema definido, en lugar de soportar una estructura de esquema arbitraria para cada fila. Si sus datos no están estructurados, puede realizar la extracción, transformación y carga (ETL) en Amazon Elastic MapReduce (Amazon EMR) para prepararlos y después volver a cargarlos en Amazon Redshift.
- **Datos de objetos binarios grandes (BLOB):** si piensa almacenar archivos binarios grandes (como vídeo digital, imágenes o música),

quizás le convenga almacenar los datos en Amazon S3 y hacer referencia a su ubicación en Amazon Redshift. En este escenario, Amazon Redshift realiza un seguimiento de los metadatos (como el nombre del elemento, el tamaño, la fecha de creación, el propietario o la ubicación) de los objetos binarios, pero los objetos grandes en sí se almacenarían en Amazon S3.

Amazon Elasticsearch Service

Amazon ES[58] es un servicio administrado que facilita las tareas de implementación, operación y escalado de Elasticsearch en la nube de AWS. Elasticsearch es un motor distribuido de búsqueda y análisis en tiempo real. Permite explorar los datos a una velocidad y a una escala como nunca había sido posible. Se usa para búsqueda de texto completo, búsqueda estructurada, análisis o los tres combinados.

Con la consola puede definir y configurar un clúster de Amazon ES en cuestión de minutos. Amazon ES administra el trabajo necesario para configurar un dominio, desde el aprovisionamiento de capacidad de infraestructura necesaria hasta la instalación del software de Elasticsearch.

Cuando el dominio ya está activo, Amazon ES automatiza las tareas administrativas más comunes, como los backups, la supervisión de las instancias y la aplicación de correcciones del software e que se basa la instancia de Amazon ES. Detecta y sustituye automáticamente los nodos de Elasticsearch con errores, reduciendo los costos asociados a las infraestructuras autoadministradas y al software de Elasticsearch. El servicio permite aumentar o reducir el tamaño del clúster con una sola llamada API o con unos pocos clics en la consola.

Con Amazon ES obtiene acceso directo a la API de código abierto Elasticsearch, de modo que el código y las aplicaciones que ya utiliza en sus entornos de Elasticsearch existentes funcionarán perfectamente. La posibilidad de integración con Logstash, una canalización de datos de código abierto, le ayudará a procesar registros y otros datos de eventos. También se incluye soporte integrado para Kibana, una plataforma de análisis y visualización de código abierto que le permitirá a entender mejor los datos.

Patrones de uso ideales

Amazon ES es ideal para consultas y búsquedas en grandes cantidades de datos.

Las organizaciones pueden usar Amazon ES para lo siguiente:

- Analizar registros de actividad, como los de las aplicaciones o sitios web de cara al cliente
- Analizar registros de CloudWatch con Elasticsearch
- Analizar datos de uso de productos provenientes de distintos servicios y sistemas
- Analizar inclinaciones en las redes sociales y datos CRM y detectar tendencias sobre marcas y productos
- Analizar actualizaciones de flujos de datos de otros servicios de AWS, como Amazon Kinesis Streams y DynamoDB
- Proporcionar a los clientes una completa experiencia de búsqueda y navegación
- Supervisar el uso de las aplicaciones móviles

Modelo de costos

Con Amazon ES solo paga los recursos de computación y almacenamiento que utiliza. No se requieren pagos mínimos ni compromisos iniciales. El cargo se hace por horas de instancia de Amazon ES, almacenamiento de Amazon EBS (si elige esta opción) y tarifas de transferencia de datos estándar.[59]

Si utiliza volúmenes de EBS para el almacenamiento, Amazon ES le permite elegir el tipo de volumen. Si elige almacenamiento en IOPS provisionadas (SSD),[60] se le cobrará el almacenamiento y la velocidad aprovisionada. Sin embargo, no se le cobrará la E/S consumida. También tiene la opción de pagar almacenamiento adicional en función del tamaño acumulado de volúmenes de EBS conectados a los nodos de datos de su dominio.

Amazon ES proporciona sin costo espacio de almacenamiento para instantáneas automatizadas de cada dominio de Amazon ES. Las instantáneas manuales se cobran conforme a las tarifas de almacenamiento de Amazon S3. Para obtener más información, consulte Precios de Amazon Elasticsearch Service.[61]

Desempeño

El desempeño de Amazon ES depende de múltiples factores, como el tipo de instancia, la carga de trabajo, el índice, el número de fragmentos empleado, las réplicas de lectura y las configuraciones de almacenamiento (de la instancia o de EBS, como una unidad SSD de propósito general). Los índices se componen de fragmentos de datos que pueden distribuirse por instancias distintas en varias zonas de disponibilidad.

Amazon ES mantiene las réplicas de lectura de los fragmentos en una zona de disponibilidad distinta cuando se comprueba el reconocimiento de la zona. Amazon ES puede usar el almacenamiento rápido de la instancia en SSD para guardar índices, o bien múltiples volúmenes de EBS. Los motores de búsqueda requieren un uso intenso de los dispositivos de almacenamiento, por lo que disponer de discos más rápidos mejora el rendimiento de consulta y búsqueda.

Durabilidad y disponibilidad

Puede configurar sus dominios de Amazon ES para alta disponibilidad habilitando la opción de reconocimiento de zona, ya sea en el momento de crear el dominio o modificando un dominio activo. Cuando está habilitado el reconocimiento de zona, Amazon ES distribuye las instancias en las que se basa el dominio entre dos zonas de disponibilidad distintas. De este modo, si se habilitan réplicas en Elasticsearch, las instancias de distribuyen automáticamente para ofrecer una replicación entre zonas.

Para conseguir durabilidad de los datos en su dominio de Amazon ES, puede servirse de instantáneas automatizadas y manuales. Las instantáneas le permiten recuperar su dominio con datos precargados, o crear un nuevo dominio con ellos. Las instantáneas se almacenan en Amazon S3, que es un sistema de almacenamiento de objetos seguro, duradero y altamente escalable. De forma predeterminada, Amazon ES crea automáticamente una instantánea de cada dominio cada día. También puede usar las API de instantánea de Amazon ES para crear instantáneas manuales adicionales. Las instantáneas manuales se almacenan en Amazon S3. Las instantáneas manuales pueden utilizarse para recuperación de desastres entre regiones y para conseguir una durabilidad adicional.

Escalabilidad y elasticidad

Puede añadir o quitar instancias y modificar volúmenes de Amazon EBS fácilmente para adaptarse al crecimiento de los datos. Basta con escribir unas líneas de código que supervisen el estado del dominio mediante las métricas de CloudWatch y llamen a la API de Amazon ES para aumentar o reducir la capacidad del dominio en función de los umbrales que defina. El servicio ejecuta la adaptación sin ningún tiempo de inactividad.

Amazon ES admite un volumen de EBS (tamaño máximo 512 GB) por cada instancia asociada a un clúster. Con un máximo de 10 instancias permitidas por cada clúster de Amazon ES, los clientes pueden asignar aproximadamente 5 TB de almacenamiento a un mismo dominio de Amazon ES.

Interfaces

Amazon ES admite la API de Elasticsearch,[62] de modo que el código, las aplicaciones y las conocidas herramientas que ya utiliza en sus entornos de Elasticsearch existentes funcionarán perfectamente. Los SDK de AWS admiten todas las operaciones de API de Amazon ES, facilitando la administración e interacción con los dominios desde la tecnología que prefiera. AWS CLI o la consola también permiten crear y administrar los dominios.

Amazon ES permite la integración de diversos servicios de AWS, como los datos en streaming desde Amazon S3, Amazon Kinesis Streams y DynamoDB Streams. Estas integraciones usan una función de Lambda como controlador de eventos en la nube que responde a los nuevos datos procesándolos y enviándolos por streaming al dominio de Amazon ES. Amazon ES también se integra con CloudWatch para supervisar las métricas de los dominios y con CloudTrail para las llamadas de API de configuración de auditoría de los dominios.

Amazon ES incluye integración nativa con Kibana, una plataforma de análisis y visualización de código abierto, y permite la integración con Logstash, que es una canalización de datos de código abierto que le ayuda a procesar registros y otros datos de eventos. Puede configurar un dominio de Amazon ES como almacén back-end para todos los registros provenientes de su implementación de Logtash y así adquirir fácilmente datos estructurados y no estructurados provenientes de diversos orígenes.

Patrones de uso no recomendados

No se recomienda el uso de Amazon ES en los siguientes escenarios:

- **Procesamiento de transacciones online (OLTP):** Amazon ES es un motor distribuido de búsqueda y análisis en tiempo real. No es compatible con transacciones ni con el procesamiento basado en operaciones de datos. Si lo que necesita es un sistema de transacciones rápido, un sistema de base de datos relacional tradicional basado en Amazon RDS o una funcionalidad de base de datos de tipo NoSQL como DynamoDB son una opción mejor.
- **Almacenamiento petabyte:** con un máximo de 10 instancias permitidas por cada clúster de Amazon ES, puede asignar aproximadamente 5 TB de almacenamiento a un mismo dominio de Amazon ES. Para cargas de trabajo mayores debe considerarse el uso de Elasticsearch autoadministrado sobre Amazon EC2.

Amazon QuickSight

En octubre de 2015, AWS presentó la versión preliminar de Amazon QuickSight, un servicio de inteligencia de negocio (BI) rápido y basado en la nube que permite crear fácilmente visualizaciones, realizar análisis ad hoc y obtener rápidamente perspectivas de negocio a partir de los datos.

QuickSight usa un motor de cálculo nuevo, ultrarrápido, en paralelo y en memoria (SPICE) para realizar cálculos avanzados y obtener visualizaciones velozmente. QuickSight se integra automáticamente con los servicios de datos de AWS, permite a las organizaciones escalar hasta cientos de miles de usuarios y ofrece una respuesta rápida a las consultas gracias a su motor de consultas SPICE. Por la décima parte del costo de las soluciones tradicionales, QuickSight permite ofrecer funcionalidad de BI muy económica a todos los miembros de su organización. Para obtener más información e inscribirse para la sesión preliminar, consulte QuickSight.[63]

Amazon EC2

Amazon EC2[64], con instancias que actúan como máquinas virtuales de AWS, ofrece una plataforma ideal para utilizar sus propias aplicaciones autoadministradas para el análisis de big data sobre la infraestructura de AWS.

Prácticamente cualquier software que pueda instalar en entornos virtuales Linux o Windows puede ejecutarse en Amazon EC2, aplicando el modelo de precios basado en el uso. Lo que no se incluye son los servicios administrados a nivel de aplicación que vienen con los otros servicios mencionados en este documento técnico. Existen numerosas opciones para el análisis autoadministrado de big data. Los siguientes son algunos ejemplos:

- Un producto NoSQL, como MongoDB
- Un almacén de datos o almacén de columnas como Vertica
- Un clúster de Hadoop
- Un clúster de Apache Storm
- Entorno Apache Kafka

Patrones de uso ideales

- **Entorno especializado:** si va a ejecutar una aplicación personalizada, una variante de un conjunto estándar de Hadoop o una aplicación no cubierta por ninguno de nuestros servicios, Amazon EC2 le proporciona la flexibilidad y escalabilidad necesarias para satisfacer sus necesidades computacionales.
- **Requisitos de conformidad:** algunos requisitos de conformidad puede que necesiten que ejecute usted mismo aplicaciones en Amazon EC2 en lugar de hacerlo un servicio administrado.

Modelo de costos

Amazon EC2 posee diversos tipos de instancias en varias familias de instancias (estándar, alta CPU, alta memoria, uso intensivo de E/S, etc.) y diferentes opciones de precio (bajo demanda, reservado y puntual). Según los requisitos de la aplicación, es posible que desee usar servicios adicionales junto con Amazon EC2, como Amazon Elastic Block Store (Amazon EBS) para el almacenamiento persistente conectado directamente o Amazon S3 como almacén de objetos duradero; cada uno de estos servicios tiene su propio modelo de precios. Si ejecuta su aplicación de big data en Amazon EC2, tendrá que hacerse cargo de las cuotas de licencias, como haría en su propio centro de datos. AWS Marketplace[65] ofrece muchos paquetes de software de big data de otros fabricantes preconfigurados para que se lancen con solo pulsar un botón.

Desempeño

El desempeño de Amazon EC2 se controla mediante el tipo de instancia que elige para su plataforma de big data. Cada tipo de instancia tiene una cantidad diferente de CPU, RAM, almacenamiento, IOPS y capacidad de red, para que pueda elegir el nivel de desempeño adecuado para los requisitos de su aplicación.

Durabilidad y disponibilidad

Las aplicaciones críticas deben ejecutarse en un clúster en varias zonas de disponibilidad dentro de una región de AWS, para que un error en una instancia o centro de datos no afecte a los usuarios de la aplicación. Para las aplicaciones en las que el tiempo de actividad no sea crítico, puede realizar un backup en Amazon S3 y restaurarla en cualquier zona de disponibilidad de la región si se produce un error en una instancia o zona. Existen otras opciones que dependen de las aplicaciones que vaya a ejecutar y de los requisitos, como la creación de un reflejo de su aplicación.

Escalabilidad y elasticidad

Auto Scaling[66] es un servicio que permite escalar automáticamente la capacidad de Amazon EC2, para aumentarla o reducirla, de acuerdo con las condiciones que defina. Con Auto Scaling, puede asegurarse de que el número de instancias EC2 que utiliza aumente sin interrupciones durante los picos de demanda, a fin de mantener el desempeño y de que se reduzca automáticamente durante los períodos de menor demanda para minimizar los costos. Auto Scaling resulta especialmente adecuado para aquellas aplicaciones que muestran variaciones de uso según la hora, el día o la semana. Auto Scaling está disponible a través de CloudWatch y está a su disposición sin ningún pago adicional aparte de las tarifas de CloudWatch.

Interfaces

A Amazon EC2 se puede obtener acceso mediante programación a través de API, con el SDK o mediante la consola. A través de la consola o mediante operaciones de API CloudWatch, dispone gratuitamente de métricas del uso de capacidad de computación, memoria y almacenamiento, del consumo de red y del tráfico de lectura/escritura de las instancias.

Las interfaces del software de análisis de big data que ejecute sobre Amazon EC2 variarán en función de las características del software que elija.

Patrones de uso no recomendados

No se recomienda el uso de Amazon EC2 en los siguientes escenarios:

- **Servicio administrado:** si su requisito es un servicio administrado en el que se abstraiga la capa de infraestructura y administración del análisis de big data, este modelo en el que usted mismo debe administrar su propio software de análisis en Amazon EC2 podría no ser la opción correcta para su caso.
- **Falta de conocimientos o recursos:** si su organización no tiene o no quiere gastar los recursos o conocimientos para instalar y administrar una instalación de alta disponibilidad para el sistema mencionado, quizá le convenga usar el equivalente de AWS, como Amazon EMR, DynamoDB, Amazon Kinesis Streams o Amazon Redshift.

Solución de problemas de big data en AWS

En este documento técnico se han examinado algunas de las herramientas de AWS que tiene a su disposición para los análisis de big data. Esto constituye un excelente punto de referencia al iniciar el diseño de las aplicaciones de big data. No obstante, hay otros aspectos que deben tenerse en cuenta para seleccionar las herramientas apropiadas para un caso de uso específico. En general, cada carga de trabajo de análisis tendrá determinadas características y requisitos que determinarán la herramienta que debe utilizar. A continuación se citan algunos factores:

- ¿Con qué rapidez necesita los resultados de los análisis? ¿Cuál sería la franja de tiempo adecuada: en tiempo real, en solo unos segundos o en una hora?
- ¿Cuánto valor proporcionarán estos análisis a su organización y cuáles son las limitaciones presupuestarias?
- ¿Cómo de grandes son los datos y a qué velocidad aumentan?
- ¿Cómo están estructurados los datos?
- ¿Qué capacidades de integración tienen los productores y consumidores?
- ¿Cuánta latencia es aceptable entre los productores y los consumidores?
- ¿Cuál es el costo del tiempo de inactividad o qué disponibilidad y durabilidad debe tener la solución?
- ¿Es la carga de trabajo de análisis constante o elástica?

Cada una de estas características o requisitos le ayuda a determinar cuál es la

herramienta correcta que debe utilizar. En algunos casos, simplemente bastará con emparejar la carga de trabajo de análisis de big data con uno de los servicios en función de los requisitos. Sin embargo, en las cargas de trabajo de análisis de big data reales, hay muchas características y requisitos diferentes y a veces en conflicto para el mismo conjunto de datos.

Por ejemplo, algunos conjuntos de resultados podrían tener que producirse en tiempo real cuando un usuario interactúa con el sistema, mientras que otros análisis podrían procesarse por lotes y ejecutarse diariamente. Estos requisitos diferentes para el mismo conjunto de datos deben desligarse y resolverse con más de una herramienta. Si intenta resolver los dos ejemplos anteriores en el mismo conjunto de herramientas terminará, o bien aprovisionando de más y, por lo tanto, pagando por un tiempo de respuesta innecesario, o bien con una solución que no es lo suficientemente rápida como para responder a sus usuarios en tiempo real. Si encuentra la herramienta más adecuada para cada conjunto de problemas de análisis distintos, conseguirá usar sus recursos de computación y almacenamiento de la forma más rentable posible.

Big data no necesariamente equivale a “grandes costos”. Por ello, al diseñar las aplicaciones es importante asegurarse de que el diseño sea rentable. Si no lo es en comparación con las otras alternativas, probablemente no sea el diseño correcto. Otra idea errónea ampliamente extendida es que disponer de varios conjuntos de herramientas para solucionar un problema de big data es más caro o más difícil de administrar que disponer de una sola herramienta grande. Si se toma el mismo ejemplo de dos requisitos diferentes para el mismo conjunto de datos, la solicitud en tiempo real puede ser lenta en cuanto a CPU, pero alta en cuanto a E/S, mientras que la solicitud de procesamiento más lenta puede requerir un uso intensivo de recursos de computación. Desdoblar los requisitos puede acabar siendo mucho más barato y más fácil de administrar, ya que puede desarrollar cada herramienta para las especificaciones exactas sin tener que aprovisionar de más. Con el modelo de infraestructura como servicio de AWS, en el que paga por lo que usa y exclusivamente por lo que usa, se logra un valor muy superior, porque se puede llevar a cabo el análisis por lotes en solo una hora y, por consiguiente, solo abonar los recursos de computación de esa hora. Además, tal vez descubra que este enfoque es más fácil de administrar que usar un solo sistema que intente satisfacer todos los requisitos. Abordar diferentes requisitos con una sola herramienta es como intentar encajar una pieza cuadrada (tales como las solicitudes en tiempo real) en un espacio circular (por ejemplo, un gran almacén de datos).
La plataforma de AWS le permite desacoplar fácilmente su arquitectura porque

dispone de diferentes herramientas para analizar el mismo conjunto de datos. Los servicios de AWS están integrados de serie, por lo que es muy fácil y muy rápido mover un subconjunto de datos de una herramienta a otra mediante paralelización. Vamos a poner todo esto en práctica estudiando unos ejemplos de escenarios de análisis de big data del mundo real; a continuación, explicaremos paso a paso la solución arquitectónica de AWS que permitiría resolver cada problema.

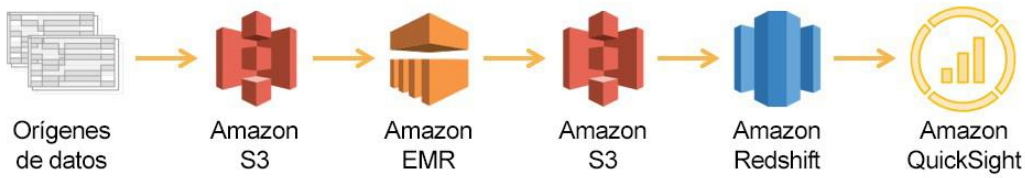
Ejemplo 1: Almacén de datos empresariales

Una firma de confección multinacional tiene más de mil establecimientos comerciales, vende algunas líneas a través de grandes almacenes y tiendas de descuentos y tiene presencia en Internet. Desde una perspectiva técnica, actualmente estos tres canales operan de forma independiente. Poseen sistemas de punto de venta y de administración y departamentos contables diferentes. No hay ningún sistema que combine todos estos conjuntos de datos para que el director general disponga de información sobre el negocio. El director general quiere tener una perspectiva de los canales de toda la empresa y estar habilitado para realizar análisis ad-hoc cuando sea necesario. Algunos ejemplos de análisis que desea realizar la firma son:

- ¿Qué tendencias existen en los canales?
- ¿Qué regiones geográficas funcionan mejor con los distintos canales?
- ¿En qué medida son eficaces sus anuncios y cupones?
- ¿Qué tendencias existen en cada línea de ropa?
- ¿Qué factores externos podrían afectar a las ventas (por ejemplo, la tasa de desempleo o las condiciones meteorológicas)?
- ¿De qué manera afectan a las ventas las características del establecimiento? Por ejemplo, la antigüedad de los empleados o de la dirección, si es centro comercial abierto o en un edificio, la ubicación de la mercancía en el establecimiento, los expositores de fin de pasillo, los folletos de ventas, el mobiliario para exhibición, etc.

Un almacén de datos empresariales es un modo excelente de resolver este problema. Este almacén de datos tendrá que recopilar datos de cada uno de los sistemas de los tres canales y de registros públicos para obtener datos meteorológicos y económicos. Cada origen de datos enviará sus datos diariamente para que los use el almacén de datos. Debido a que cada origen de datos puede

estar estructurado de forma diferente, se realiza un proceso de extracción, transformación y carga (ETL) para reformatear los datos en una estructura común. Luego, podrán realizarse análisis de los datos de todos los orígenes simultáneamente. Para ello, se usa la siguiente arquitectura de flujo de datos:



Almacén de datos empresariales

1. El primer paso de este proceso es obtener los datos de muchos orígenes diferentes en Amazon S3. Se ha elegido Amazon S3 porque es una plataforma de almacenamiento de larga duración, económica y escalable en la que se puede escribir en paralelo desde muchos orígenes diferentes a un costo muy bajo.
2. Se usa Amazon EMR para transformar y limpiar los datos del formato de origen a fin de enviarlos al formato de destino. Amazon EMR está integrado con Amazon S3 de serie para permitir subprocesos de rendimiento paralelos desde cada nodo del clúster a y desde Amazon S3. Normalmente, los almacenes de datos reciben los datos nuevos durante la noche, desde sus numerosos orígenes. Debido a que no se necesitan estos análisis en mitad de la noche, el único requisito de este proceso de transformación es que finalice por la mañana, cuando el director y otros usuarios profesionales necesitan tener los resultados. En consecuencia, puede utilizar el Mercado de subastas de Amazon EC2[67] para reducir aún más el costo de la transformación. Una buena estrategia de subasta podría ser empezar a pujar a un precio muy bajo a media noche e ir aumentando el precio con el tiempo hasta que se consiga la capacidad. Cuando se aproxime la fecha límite, si las pujas no han tenido éxito, puede recurrir a los precios bajo demanda para asegurarse de que todavía satisface sus requisitos de plazo de finalización. Cada origen puede tener un proceso de transformación diferente en Amazon EMR. Sin embargo, mediante el modelo de pago por uso de AWS, puede crear un clúster de Amazon EMR separado para cada

transformación y ajustarlo de modo que tenga la capacidad adecuada, y así completar todos los trabajos de transformación al precio más bajo posible sin competir con los recursos de otros trabajos.

3. Cada trabajo de transformación pone entonces los datos formateados y limpios en Amazon S3. Volvemos a utilizar Amazon S3 porque Amazon Redshift puede consumir estos datos en varios subprocesos en paralelo desde cada nodo. Esta ubicación en Amazon S3 sirve también como registro histórico y es el origen de confianza formateado entre sistemas. Otras herramientas pueden usar los datos ubicados en Amazon S3 para análisis en caso de que surjan nuevos requisitos con el tiempo.
4. Amazon Redshift carga, ordena, distribuye y comprime los datos en sus tablas para que puedan ejecutarse consultas analíticas eficazmente y en paralelo. Amazon Redshift se ha diseñado para las cargas de trabajo del almacén de datos y puede ampliarse fácilmente añadiendo otro nodo cuando aumente el tamaño de los datos con el paso del tiempo y se expanda el negocio.
5. Para visualizar los análisis, puede usar Amazon QuickSight o una de las muchas plataformas de visualización de los socios que se conectan con Amazon Redshift mediante el uso de conexiones OBDC o JDBC. Aquí es donde el director general y su equipo podrán ver los informes y los gráficos. Ahora, los directivos pueden usar los datos para tomar decisiones más acertadas sobre los recursos de la compañía, lo que en última instancia aumenta los ingresos y el valor para los accionistas.

Esta arquitectura es muy flexible y se puede ampliar fácilmente si se expande el negocio, se importan más orígenes de datos, se abren nuevos canales o se lanza una aplicación móvil con datos específicos de los clientes. En cualquier momento se pueden integrar herramientas adicionales y el almacén se puede ampliar con unos pocos clics aumentando el número de nodos del clúster de Amazon Redshift.

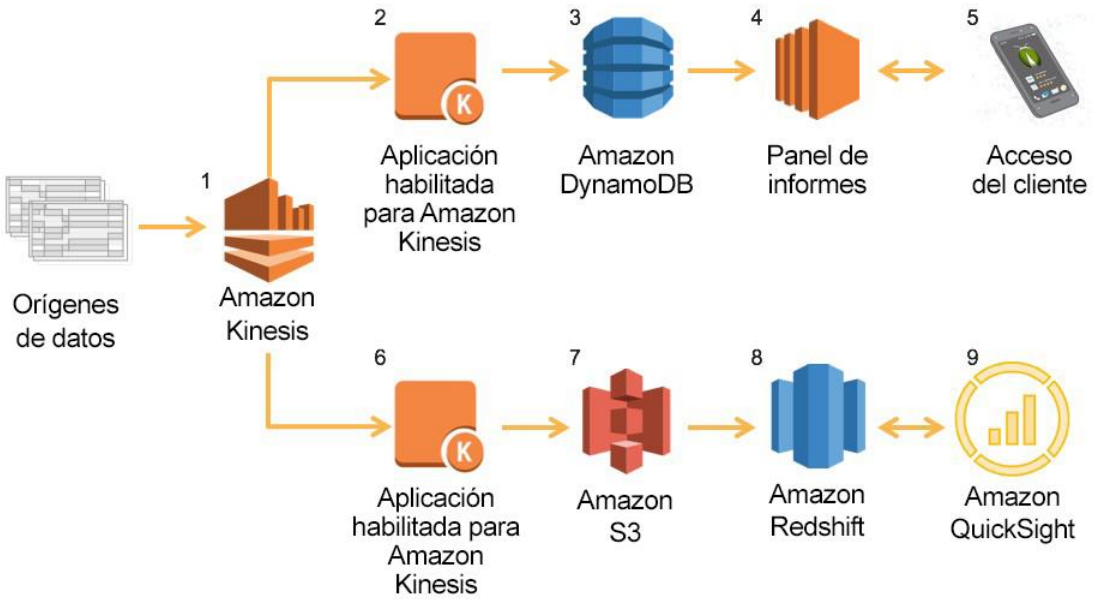
Ejemplo 2: Captura y análisis de datos de sensores

Un fabricante internacional de aparatos de aire acondicionado tiene muchos aparatos grandes de aire acondicionado que vende a varias compañías comerciales e industriales. Pero no se limitan tan solo a vender unidades de aire acondicionado. Además, la compañía ofrece servicios complementarios, para que los clientes puedan ver paneles en tiempo real en una aplicación móvil o navegador web. Cada unidad envía información de su sensor para su procesamiento y análisis. Los clientes y el fabricante utilizan estos datos. Gracias a esta prestación, el fabricante puede ver los conjuntos de datos e identificar tendencias.

Actualmente, el fabricante tiene algunos miles de unidades reservadas con esta capacidad. Tiene previsto entregar estas unidades a los clientes en los próximos meses y espera que, con el tiempo, miles de unidades de todo el mundo usen esta plataforma. Si tiene éxito, la compañía desearía ampliar este servicio a su línea de consumo, con un volumen mucho mayor y una cuota de mercado más grande. La solución tendrá que administrar cantidades masivas de datos y ampliarse sin interrupción a medida que se desarrolle el negocio. ¿Cómo se diseña un sistema como este? En primer lugar, lo dividimos en dos flujos de trabajo, ambos procedentes de los mismos datos:

- La información actual de las unidades de aire acondicionado, con requisitos casi en tiempo real y un gran número de clientes que usan esta información.
- Toda la información histórica de las unidades de aire acondicionado, con el fin de efectuar tendencias y análisis para uso interno.

A continuación se muestra la arquitectura de flujo de datos que permite solucionar este problema de big data:



Captura y análisis de datos de sensores

1. El proceso comienza cuando cada unidad de aire acondicionado proporciona un flujo de datos constante a Amazon Kinesis Streams. Este servicio proporciona una interfaz elástica y duradera con la que las unidades se comunican y que se podrá escalar fácilmente cuando se vendan más unidades de aire acondicionado y se activen online.
2. Usando las herramientas proporcionadas por Amazon Kinesis Streams, como Kinesis Client Library o el SDK de Kinesis, se crea una aplicación sencilla en Amazon EC2. Esta lee los datos conforme llegan a Amazon Kinesis Streams, los analiza y determina si estos datos justifican que se actualice el panel en tiempo real. Busca cambios en el funcionamiento del sistema, fluctuaciones de temperatura y cualquier error que pueda tener lugar en las unidades.
3. Este flujo de datos debe producirse casi en tiempo real para que los clientes y los equipos de mantenimiento sepan lo antes posible si hay un problema con la unidad. La información del panel ofrece algunos datos de tendencias acumulados, pero sobre todo refleja el estado actual, así como los errores del sistema. Por tanto, la cantidad de datos necesarios para rellenar el panel es relativamente pequeña. Además, se producirán numerosos accesos a estos datos desde los orígenes siguientes:
    - Los clientes que comprueban su sistema con un dispositivo móvil o navegador.

- Los equipos de mantenimiento que comprueban el estado de la flota.
- Los algoritmos de datos e inteligencia y los análisis de la plataforma de informes para identificar tendencias que puedan enviarse como alertas (por ejemplo, cuando el ventilador de una unidad de aire acondicionado ha estado funcionando más tiempo de lo normal, pero la temperatura del edificio no ha descendido).

Se ha elegido DynamoDB para almacenar este conjunto de datos casi en tiempo real, porque ofrece una alta disponibilidad y escalabilidad. El caudal de datos debe poder ampliarse y reducirse fácilmente para satisfacer las necesidades de sus consumidores cuando se adopte la plataforma y aumente el uso.

4. El panel de informes es una aplicación web personalizada basada en este conjunto de datos y ejecutada en Amazon EC2. Proporciona contenido basándose en el estado del sistema y en las tendencias; además, alerta a los clientes y a los profesionales de mantenimiento si hay cualquier problema con la unidad.
5. El cliente obtiene acceso a los datos desde un dispositivo móvil o un navegador web para conocer el estado actual del sistema y ver tendencias históricas.

El flujo de datos (pasos 2 al 5) que hemos descrito se ha creado para informar en tiempo real a los consumidores humanos. Se ha creado y diseñado para ofrecer una baja latencia y puede ampliarse muy rápidamente para satisfacer la demanda. El flujo de datos (pasos 6 al 9) que se ilustra en la parte inferior del diagrama no tiene esos requisitos de velocidad y latencia tan estrictos. Esto permite que el arquitecto diseñe una pila de soluciones diferente que pueda contener cantidades mayores de datos a un costo por byte de información mucho menor y elegir recursos de computación y almacenamiento más económicos.

6. Para leer el flujo de Amazon Kinesis, existe una aplicación independiente compatible con Amazon Kinesis que probablemente se ejecuta en una instancia EC2 menor con un escalado más lento. Aunque esta aplicación va a analizar el mismo conjunto de datos que el flujo de datos superior, la finalidad última es almacenar estos datos para mantener un registro a largo plazo y alojar el conjunto de datos en un almacén de datos. Este conjunto de datos acabará siendo el conjunto de datos completo enviado desde los sistemas y permitirá efectuar una serie de análisis mucho más amplios sin los requisitos de que se generen casi en tiempo real.

7. La aplicación compatible con Amazon Kinesis transformará los datos en un formato adecuado para el almacenamiento a largo plazo, así como para su carga en el almacén de datos y su almacenamiento en Amazon S3. Los datos ubicados en Amazon S3 no solo sirven como un punto de recepción paralelo para Amazon Redshift, sino que estarán en un almacén duradero que contendrá todos los datos que se vayan a ejecutar alguna vez a través de este sistema. Por ello, puede ser un origen único de datos confiables. Se puede usar para cargar otras herramientas de análisis si surgen requisitos adicionales Amazon S3 incluye también integración nativa con Amazon Glacier por si los datos necesitaran pasar por un almacenamiento en frío de larga duración y bajo costo.
8. Se usará de nuevo Amazon Redshift como almacén de datos para el conjunto de datos de mayor tamaño. Puede ampliarse fácilmente cuando aumente el conjunto de datos con solo añadir otro nodo al clúster.
9. Para visualizar los análisis, se podría usar alguna de las muchas plataformas de visualización de los socios a través de la conexión OBDC/JDBC a Amazon Redshift. Ahí es donde se pueden realizar informes, gráficos y análisis a partir del conjunto de datos para encontrar determinadas variables y tendencias que podrían provocar un desempeño insuficiente o averías en las unidades de aire acondicionado.

Esta arquitectura puede empezar con un tamaño reducido e ir creciendo según sea necesario. Asimismo, al desvincular los dos flujos de trabajo diferentes uno de otro, pueden crecer a su propio ritmo en función de los requisitos y sin compromisos iniciales. De este modo, el fabricante puede evaluar el éxito o el fracaso del nuevo servicio sin tener que efectuar una inversión importante. Resulta fácil imaginar nuevas incorporaciones, tales como Amazon ML que permite prever con exactitud cuánto durará una unidad de aire acondicionado y enviar equipos de mantenimiento preventivo en función de los algoritmos de predicción, con el fin de prestar al cliente el mejor servicio y experiencia posibles. Este nivel de servicio sería un factor diferenciador ante la competencia y produciría un aumento de las ventas futuras.

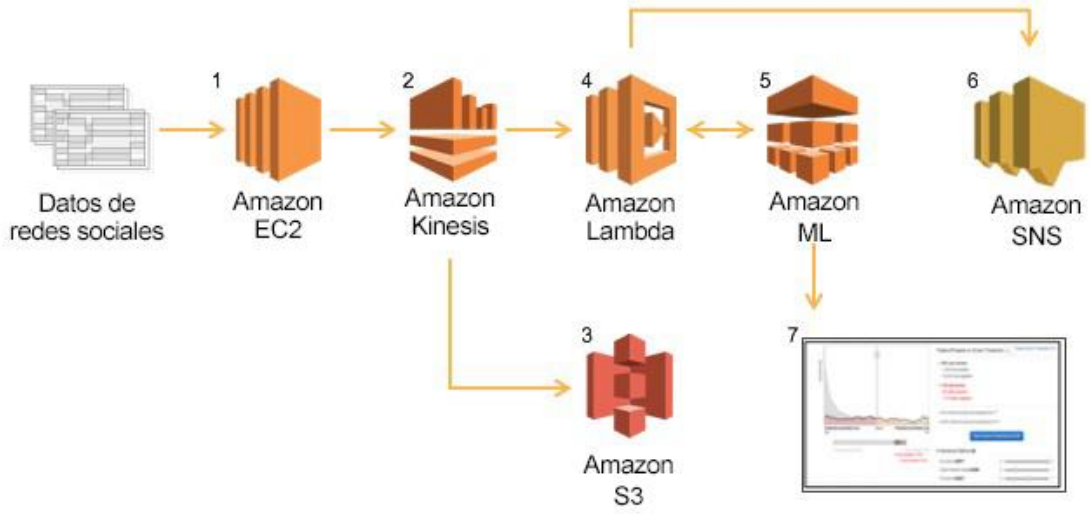
Ejemplo 3: Análisis de opinión en los medios sociales

Un importante fabricante de juguetes ha experimentado un rápido crecimiento y está expandiendo su línea de productos. Tras el lanzamiento de cada juguete, la compañía desea saber si los clientes disfrutan de sus productos y los utilizan.

Además, la compañía desea asegurarse de que los consumidores tengan una experiencia satisfactoria con sus productos. Ante el crecimiento del ecosistema de los juguetes, la compañía quiere asegurarse de que sus productos continúen siendo pertinentes para sus clientes y desea poder planificar los puntos de sus hojas de ruta en función de los comentarios de los clientes La compañía desea capturar la información siguiente en los medios sociales:

- Comprender cómo utilizan sus productos los clientes.
- Asegurarse de que los clientes estén satisfechos.
- Planear las hojas de ruta futuras.

Capturar los datos de distintas redes sociales resulta relativamente fácil, pero la dificultad reside en crear la inteligencia mediante programación. Una vez introducidos los datos, la compañía desea poder analizarlos y clasificarlos de manera rentable mediante programación. Para ello, se puede usar la siguiente arquitectura:



Análisis de opinión en los medios sociales

1. Lo primero que hay que hacer es decidir qué sitios de medios sociales se van a escuchar. Después, se crea una aplicación que sondee estos sitios mediante los API correspondientes y se ejecuta en Amazon EC2.
2. A continuación, se crea un flujo de Amazon Kinesis, ya que podríamos tener varios orígenes de datos: Twitter, Tumblr, etcétera. De este modo, se puede

crear un nuevo flujo cada vez que se agrega un origen de datos y se puede utilizar el código de aplicación y la arquitectura que ya existen. Además, en este ejemplo se crea un nuevo flujo de Amazon Kinesis para copiar los datos sin procesar en Amazon S3.

3. Con fines de archivo, análisis a largo plazo y consultas históricas, se almacenan los datos sin procesar en Amazon S3. Se pueden ejecutar modelos adicionales de procesamiento por lotes en Amazon ML a partir de los datos ubicados en Amazon S3, con objeto de efectuar análisis predictivos y realizar el seguimiento de las tendencias de compra de los consumidores.

4. Como se ilustra en el diagrama de la arquitectura, se utiliza Lambda para procesar y normalizar los datos, así como para solicitar predicciones de Amazon ML. Una vez que se devuelve la predicción de Amazon ML, la función Lambda puede adoptar medidas basándose en ella; por ejemplo, dirigir una publicación de un medio social al equipo de servicio al cliente para su estudio.

5. Amazon ML se utiliza para efectuar predicciones sobre los datos de entrada. Por ejemplo, se puede crear un modelo de ML para analizar los comentarios en los medios sociales y determinar si un cliente ha expresado opiniones negativas sobre el producto. Para obtener predicciones exactas con Amazon ML, es preciso comenzar con datos de entrenamiento y asegurarse de que los modelos de ML funcionen correctamente. Si esta es la primera vez que va a crear modelos de ML, consulte el Tutorial: Uso de Amazon ML para predecir respuestas a una oferta de marketing.[68] Como hemos indicado anteriormente, si se utilizan orígenes de datos de varias redes sociales, es conveniente utilizar un modelo de ML distinto para cada una de ellas con el fin de garantizar su exactitud.

6. Por último, los datos procesables se envían a Amazon SNS mediante Lambda y se distribuyen a los recursos apropiados en mensajes de texto o de correo electrónico para que los investiguen.

7. Para obtener resultados precisos al analizar opiniones, es fundamental crear un modelo de Amazon ML que se actualice con regularidad. Se pueden mostrar gráficamente en la consola otras métricas adicionales sobre un modelo concreto, tales como precisión, tasa de falsos positivos, exactitud o retirada de productos. Para obtener más información, consulte la sección Paso 4: Revisar el desempeño predictivo del modelo de ML y establecer un límite.[69]

Mediante una combinación de Amazon Kinesis Streams, Lambda, Amazon ML y Amazon SES hemos creado una plataforma escalable y fácil de configurar para escuchar los medios sociales. Cabe destacar que esta imagen no representa la creación de un modelo de ML. Esto habrá que llevarlo a cabo al menos una vez, pero suele realizarse regularmente para mantener actualizado el modelo. La frecuencia con que se crean modelos nuevos depende de la carga de trabajo y, en realidad, solo se hace para aportar precisión al modelo cuando cambian las circunstancias.

Conclusión

Cuanto mayor es el número de datos que se generan y recopilan, el análisis de datos requiere herramientas escalables, flexibles y de alto desempeño para proporcionar información útil y oportuna. Sin embargo, las organizaciones tienen ante sí un ecosistema de big data en constante expansión en el que aparecen y desaparecen nuevas herramientas a gran velocidad. Por consiguiente, puede resultar muy complicado mantenerse al día y elegir las herramientas apropiadas.

Este documento ofrece un punto de partida para ayudar a resolver este problema. Con el amplio conjunto de servicios administrados de recopilación, procesamiento y análisis de big data, la plataforma AWS facilita la creación, la implementación y el escalado de las aplicaciones de big data. Así, usted puede concentrarse en los problemas de negocio y no en actualizar y administrar estas herramientas.

AWS proporciona muchas soluciones para satisfacer los requisitos de los análisis de big data. La mayoría de las soluciones de arquitectura de big data utilizan varias herramientas de AWS para desarrollar una solución completa. Este enfoque puede ayudar a satisfacer los requisitos más estrictos del negocio de la manera más rentable, eficaz y robusta posible. El resultado es una arquitectura de big data flexible y capaz de escalarse a la par que su negocio en la infraestructura global de AWS.

Colaboradores

Las siguientes personas y organizaciones han participado en la redacción de este documento:

- Erik Swensson, director de arquitectura de soluciones, Amazon Web Services
- Erick Dame, arquitecto de soluciones, Amazon Web Services
- Shree Kenghe, arquitecto de soluciones, Amazon Web Services

Documentación adicional

Los recursos siguientes le pueden ayudar a empezar a ejecutar análisis de big data en AWS:

- Visite aws.amazon.com/big-data.[70]

  Consulte la cartera completa de servicios de big data, además de enlaces a otros recursos de AWS, como socios, tutoriales y artículos de big data, así como ofertas AWS Marketplace para soluciones de big data. Póngase en contacto con nosotros si necesita cualquier tipo de ayuda.

- Lea el blog de big data de AWS.[71]

  En el blog encontrará ejemplos del mundo real e ideas que le ayudarán a recopilar, almacenar, limpiar, procesar y visualizar sus big data.

- Pruebe una de las versiones de prueba de big data.[72]

  Explore el completo ecosistema de productos diseñados para abordar los desafíos de los big data mediante AWS. Las versiones de prueba han sido desarrolladas por los socios de consultoría y tecnología de AWS Partner Network (APN) y se proporcionan gratuitamente para fines educativos, de demostración y de evaluación.

- Reciba un curso de capacitación de AWS sobre big data.[73]

El curso Big Data on AWS le ofrece una introducción a las soluciones de big data basadas en la nube y Amazon EMR. Le mostraremos cómo usar Amazon EMR para procesar los datos mediante el amplio ecosistema de herramientas de Hadoop como Pig y Hive. También le enseñaremos a crear entornos de big data y a trabajar con Amazon DynamoDB y Amazon Redshift, conocerá los beneficios de Amazon Kinesis Streams y aprenderá a usar las prácticas recomendadas para diseñar entornos de big seguros y rentables.

- Consulte los casos prácticos de clientes relacionados con los big data.[74]

  Aprenda de la experiencia de otros clientes que han desarrollado plataformas de datos eficaces y satisfactorias en la nube de AWS.

Revisiones del documento

Enero de 2016	Revisión para agregar información sobre Amazon Machine Learning, AWS Lambda, Amazon Elasticsearch Service; actualización general.
Diciembre de 2014	Publicación inicial

Notas

1 http://aws.amazon.com/about-aws/globalinfrastructure/

2 http://aws.amazon.com/s3/

3 http://aws.amazon.com/datapipeline/

4 https://aws.amazon.com/iot/

5 https://aws.amazon.com/importexport/

6 http://aws.amazon.com/kinesis/firehose

7 https://aws.amazon.com/directconnect/

8 https://aws.amazon.com/mobile/

9 http://aws.amazon.com/solutions/case-studies/big-data/

10 https://aws.amazon.com/kinesis/streams

11 http://docs.aws.amazon.com/kinesis/latest/APIReference/Welcome.html

12 http://docs.aws.amazon.com/aws-sdk-php/v2/guide/service-kinesis.html

13 http://aws.amazon.com/kinesis/pricing/

14 http://aws.amazon.com/tools/

15 http://docs.aws.amazon.com/kinesis/latest/dev/developing-producers-with-kpl.html

16 http://docs.aws.amazon.com/kinesis/latest/dev/writing-with-agents.html

17 https://github.com/awslabs/amazon-kinesis-client

18 https://github.com/awslabs/kinesis-storm-spout

19 https://aws.amazon.com/lambda/

20 http://docs.aws.amazon.com/lambda/latest/dg/intro-core-components.html

21 https://aws.amazon.com/amazon-linux-ami/

22 http://docs.aws.amazon.com/lambda/latest/dg/nodejs-create-deployment-pkg.html

23 http://docs.aws.amazon.com/lambda/latest/dg/lambda-python-how-to-create-deployment-package.html

24 http://docs.aws.amazon.com/lambda/latest/dg/lambda-java-how-to-create-deployment-package.html

25 http://aws.amazon.com/elasticmapreduce/

26 https://media.amazonwebservices.com/AWS_Amazon_EMR_Best_Practices.pdf

27 http://aws.amazon.com/elasticmapreduce/pricing/

28 http://aws.amazon.com/ec2/instance-types/

29 http://aws.amazon.com/elasticmapreduce/mapr/

30 http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-manage-resize.html

31 http://docs.aws.amazon.com/ElasticMapReduce/latest/API/Welcome.html

32 http://docs.aws.amazon.com/ElasticMapReduce/latest/ReleaseGuide/emr-hive.html

33 http://docs.aws.amazon.com/ElasticMapReduce/latest/ReleaseGuide/emr-pig.html

34 http://blogs.aws.amazon.com/bigdata/post/Tx15AY5C50K70RV/Installing-Apache-Spark-on-an-Amazon-EMR-Cluster

35 http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-hbase.html

36 http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-impala.html

37 http://aws.amazon.com/elasticmapreduce/hunk/

38 http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/UsingEMR_s3distcp.html

39 https://aws.amazon.com/machine-learning/

40 https://aws.amazon.com/machine-learning/pricing/

41 http://docs.aws.amazon.com/machine-learning/latest/dg/suggested-recipes.html

42 http://docs.aws.amazon.com/machine-learning/latest/APIReference/Welcome.html

43 https://aws.amazon.com/dynamodb

44 http://aws.amazon.com/free/

45 http://aws.amazon.com/dynamodb/pricing/

46 Tiempos de respuesta medios de menos de diez milisegundos en el lado del servidor.

47 http://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Streams.html

48 DynamoDB permite cambiar su nivel de desempeño aprovisionado en hasta un 100 % mediante una única llamada de operación al API UpdateTable. Si desea aumentar el rendimiento en más del 100 %, llame de nuevo a UpdateTable.

49 Puede aumentar el rendimiento aprovisionado con la frecuencia que quiera; sin embargo, existe una limitación de dos reducciones al día.

50 https://aws.amazon.com/redshift/

51 http://aws.amazon.com/s3/pricing/

52 http://aws.amazon.com/redshift/pricing/

53 http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

54 http://docs.aws.amazon.com/redshift/latest/APIReference/Welcome.html

55 http://docs.aws.amazon.com/redshift/latest/mgmt/working-with-clusters.html#rs-about-clusters-and-nodes

56 http://docs.aws.amazon.com/redshift/latest/dg/c_redshift-and-postgres-sql.html

57 http://aws.amazon.com/redshift/partners/

58 https://aws.amazon.com/elasticsearch-service/

59 https://aws.amazon.com/ec2/pricing/

60 https://aws.amazon.com/ebs/details/

61 https://aws.amazon.com/elasticsearch-service/pricing/

62 https://aws.amazon.com/elasticsearch-service/faqs/

63 https://aws.amazon.com/quicksight

64 https://aws.amazon.com/ec2/

65 https://aws.amazon.com/marketplace

66 http://aws.amazon.com/autoscaling/

67 http://aws.amazon.com/ec2/spot/

68 http://docs.aws.amazon.com/machine-learning/latest/dg/tutorial.html

69 http://docs.aws.amazon.com/machine-learning/latest/dg/step-4-review-the-ml-model-predictive-performance-and-set-a-cut-off.html

70 http://aws.amazon.com/big-data

71 http://blogs.aws.amazon.com/bigdata/

72 https://aws.amazon.com/testdrive/bigdata/

73 http://aws.amazon.com/training/course-descriptions/bigdata/

74 http://aws.amazon.com/solutions/case-studies/big-data/