Opções de análise de big data na AWS

Janeiro de 2016

Avisos

Este documento é fornecido apenas para fins informativos. Ele relaciona as atuais ofertas de produtos e práticas da AWS na data de emissão deste documento, que estão sujeitas a alterações sem aviso prévio. Os clientes são responsáveis por fazer sua própria avaliação independente das informações neste documento e de qualquer uso dos produtos ou serviços da AWS, cada um dos quais é fornecido “como está”, sem garantia de qualquer tipo, expressa ou implícita. Este documento não cria quaisquer garantias, representações, compromissos contratuais, condições ou promessas da AWS, suas afiliadas, fornecedores ou licenciadores. As responsabilidades e obrigações da AWS para com seus clientes são controladas por contratos da AWS, e este documento não modifica nem faz parte de qualquer contrato entre a AWS e seus clientes.

Sumário

Resumo

Este whitepaper ajuda arquitetos, cientistas de dados e desenvolvedores a entender as opções de análise de big data disponíveis na Nuvem AWS, fornecendo uma visão geral dos serviços, com as seguintes informações:

- Padrões de uso ideal
- Modelo de custo
- Desempenho
- Durabilidade e disponibilidade
- Escalabilidade e elasticidade
- Interfaces
- Antipadrões

Este documento finaliza com cenários que demonstram as opções de análise em uso, bem como fornece recursos adicionais para começar a usar a análise de big data na AWS.

Introdução

À medida que nos tornamos uma sociedade mais digital, a quantidade de dados criada e coletada aumenta e acelera de forma significativa. A análise desses dados cada vez maiores se torna um desafio com o uso de ferramentas de análise tradicionais. É necessário ter inovação para preencher a lacuna entre os dados que estão sendo gerados e os que podem ser efetivamente analisados.

As ferramentas e tecnologias de big data oferecem oportunidades e desafios de analisar os dados com eficiência para entender melhor as preferências do cliente, obter uma vantagem competitiva no mercado e usá-los para expandir sua empresa. As arquiteturas de gerenciamento de dados evoluíram do modelo tradicional de data warehouse para arquiteturas mais complexas que abordam mais requisitos, como processamento em tempo real e em lotes, dados estruturados e não estruturados, transações de alta velocidade, entre outros.

A Amazon Web Services (AWS) fornece uma ampla plataforma de serviços gerenciados para ajudar você a criar, proteger e dimensionar aplicativos de big data de forma perfeita e completa, com rapidez e facilidade. Não importa se seus

aplicativos exigem streaming em tempo real ou processamento de dados em lotes, a AWS fornece a infraestrutura e as ferramentas para lidar com seu próximo projeto de big data. Sem necessidade de comprar hardware e de manter e escalar uma infraestrutura, somente o que você precisa para coletar, armazenar, processar e analisar big data. A AWS tem um ecossistema de soluções de análise concebido especificamente para gerenciar essa crescente quantidade de dados e fornecer informações sobre seu negócio.

A vantagem da AWS na análise de big data

A análise de grandes conjuntos de dados requer uma capacidade de computação significativa, cujo tamanho pode variar de acordo com a quantidade de dados de entrada e tipo de análise. Essa característica de cargas de trabalho de big data é ideal para o modelo de computação em nuvem de pagamento conforme o uso, no qual as aplicações podem aumentar ou diminuir facilmente com base na demanda. À medida que os requisitos mudam, é possível redimensionar facilmente seu ambiente (horizontal ou verticalmente) na AWS para atender às suas necessidades, sem precisar aguardar por hardware adicional ou fazer mais investimentos para provisionar uma maior capacidade.

Para aplicativos críticos em uma infraestrutura mais tradicional, a única alternativa dos projetistas de sistemas é o provisionamento excessivo, pois é preciso que o sistema tenha capacidade de gerenciar um grande crescimento dos dados adicionais devido a um aumento nas necessidades de negócios. Em contrapartida, na AWS é possível provisionar mais capacidade e computação em questão de minutos. Desse modo, seus aplicativos de big data aumentam e diminuem conforme a demanda, e seu sistema é executado o mais próximo possível da eficiência ideal.

Além disso, você obtém computação flexível em uma infraestrutura global com acesso a muitas regiões geográficas[1] diferentes oferecidas pela AWS, juntamente com a capacidade de usar outros serviços escaláveis que são aumentados para criar aplicativos de big data sofisticados. Esses outros serviços incluem o Amazon Simple Storage Service (Amazon S3)[2] para armazenamento de dados, e o AWS Data Pipeline[3] para organização de trabalhos para mover e transformar tais dados facilmente. O AWS IoT,[4] que permite a interação de dispositivos com aplicativos de nuvem e outros dispositivos conectados.

Além disso, a AWS tem muitas opções para ajudar a obter dados na nuvem, incluindo dispositivos seguros, como AWS Import/Export Snowball[5] para acelerar transferências de dados em escala de petabytes, Amazon Kinesis Firehose[6] para carregar dados de streaming e conexões privadas escaláveis por meio do AWS Direct Connect.[7] Como o celular continua a crescer rapidamente em termos de uso, é possível usar o conjunto de serviços no Hub do AWS Mobile[8] para coletar e medir o uso e os dados do aplicativo ou exportar esses dados para outro serviço para análise adicional personalizada.

Essas capacidades da plataforma da AWS as tornam ideais para a resolução de problemas de big data, e muitos clientes já implementaram cargas de trabalho de análise de big data bem-sucedidas na AWS. Para mais informações sobre estudos de caso, consulte Big Data e HPC. Desenvolvido pela Nuvem AWS.[9]

Os serviços a seguir estão descritos na ordem de coleta, processamento, armazenamento e análise de big data:

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

Além disso, as instâncias EC2 da Amazon estão disponíveis para aplicativos de big data autogerenciados.

Amazon Kinesis Streams

O Amazon Kinesis Streams[10] permite que você crie aplicativos personalizados que processam ou analisam os dados de streaming em tempo real. O Amazon Kinesis Streams pode capturar e armazenar continuamente terabytes de dados por hora de centenas de milhares de origens, como clickstreams de sites, transações financeiras, feeds de mídia social, logs de TI e eventos de rastreamento de localização.

Com a Amazon Kinesis Client Library (KCL), você pode criar aplicativos do Amazon Kinesis e usar dados de streaming para alimentar painéis em tempo real, gerar alertas e implementar definições de preço e publicidade dinâmicas. Você também pode emitir dados do Amazon Kinesis Streams para outros serviços da AWS, como Amazon Simple Storage Service (Amazon S3), Amazon Redshift, Amazon Elastic MapReduce (Amazon EMR) e AWS Lambda.

Forneça o nível de entrada e saída necessário para seu fluxo de dados, em blocos de 1 megabyte por segundo (MB/s), usando o Console de Gerenciamento da AWS, API,[11] ou SDK.[12] O tamanho do seu fluxo pode ser ajustado para cima ou para baixo a qualquer momento sem necessidade de reiniciar, e sem qualquer impacto nas fontes de dados que estão enviando dados para o fluxo. Dentro de segundos, os dados enviados a um stream ficam disponíveis para análise.

Os dados do stream são armazenados em várias zonas de disponibilidade de uma região por 24 horas. Durante esse período, os dados ficam disponíveis para serem lidos, lidos novamente, preenchidos e analisados, ou movidos para o armazenamento de longo prazo (como o Amazon S3 ou Amazon Redshift). A KCL permite que os desenvolvedores se concentrem na criação de seus aplicativos empresariais enquanto remove o trabalho pesado não diferenciado associado a balanceamento de carga de dados de streaming, coordenação de serviços distribuídos e processamento de dados tolerante a falhas.

Padrões de uso ideal

O Amazon Kinesis Streams é útil onde há necessidade de retirar dados rapidamente dos produtores (por exemplo, fontes de dados) e processar continuamente. Esse processamento pode transformar os dados antes de emiti-los para outro armazenamento de dados, impulsionar métricas e análises em tempo real ou gerar e agregar vários streams a streams mais complexos ou a processamentos de downstream. Veja a seguir cenários típicos de utilização do Amazon Kinesis Streams para análise.

- **Análise de dados em tempo real** – O Amazon Kinesis Streams possibilita a análise de dados em streaming em tempo real, como a análise de dados da sequência de cliques em sites e análise de envolvimento do cliente.

- **Entrada e processamento de feeds de dados e logs** – Com o Amazon Kinesis Streams, você pode solicitar que os produtores gerem dados diretamente em um stream do Amazon Kinesis. Por exemplo, é possível enviar logs do sistema e do aplicativo para o Amazon Kinesis Streams e acessar o stream para processamento em segundos. Isso evita que os dados de logs sejam perdidos se o front-end ou o servidor do aplicativo falhar e reduz o armazenamento de logs local na fonte. O Amazon Kinesis Streams fornece uma entrada acelerada de dados, pois você não organiza os dados em lotes nos servidores antes de enviá-los para entrada.
- **Métricas e relatórios em tempo real** – Você pode usar dados incorporados ao Amazon Kinesis Streams para extrair métricas e gerar KPIs para sustentar relatórios e painéis em tempo real. Isso permite que a lógica do aplicativo de processamento de dados processe os dados à medida que o streaming acontece continuamente, em vez de esperar que os lotes de dados cheguem.

Modelo de custo

O Amazon Kinesis Streams possui uma definição de preço simples de pagamento conforme o uso, sem custos iniciais e taxas mínimas, e você só paga pelos recursos que consumir. Um stream do Amazon Kinesis é composto por um ou mais fragmentos. Cada fragmento fornece uma capacidade de 5 transações de leitura por segundo, até um total máximo de 2 MB de dados lidos por segundo. Cada fragmento pode oferecer suporte a até 1.000 transações de gravação por segundo e até um total máximo de 1 MB de dados gravados por segundo.

A capacidade de dados do seu stream é uma função do número de estilhaços que você especifica para o stream. A capacidade total do stream é a soma da capacidade de cada estilhaço. Há apenas dois componentes de definição de preço: uma cobrança por hora por estilhaço e outra a cada 1 milhão de transações PUT. Para obter mais informações, consulte Definição de preço do Amazon Kinesis Streams.[13] Aplicativos que são executados no Amazon EC2 e processam streams do Amazon Kinesis também incorrem custos padrão do Amazon EC2.

Desempenho

O Amazon Kinesis Streams permite que você escolha a capacidade da taxa de transferência de que precisa em termos de estilhaço. Com cada estilhaço em um stream do Amazon Kinesis, é possível capturar até 1 megabyte por segundo de

dados em 1.000 transações de gravação por segundo. Seus aplicativos do Amazon Kinesis pode ler dados de cada estilhaço em até 2 megabytes por segundo. Você poderá provisionar quantos estilhaços precisar para obter a capacidade de taxa de transferência desejada; por exemplo, um stream de dados de 1 gigabyte por segundo requer 1.024 estilhaços.

Durabilidade e disponibilidade

O Amazon Kinesis Streams replica os dados simultaneamente em três zonas de disponibilidade de uma região da AWS, proporcionando alta disponibilidade e durabilidade dos dados. Além disso, é possível armazenar um cursor no DynamoDB para acompanhar de modo durável o que foi lido de um stream do Amazon Kinesis. Em caso de falha em seu aplicativo que estava no meio de uma leitura de dados do stream, é possível reiniciar seu aplicativo e usar o cursor para retomar do ponto exato em que o aplicativo com falha parou.

Escalabilidade e elasticidade

Você pode aumentar ou diminuir a capacidade do stream a qualquer momento de acordo com suas necessidades operacionais ou de negócios sem interrupções no processamento contínuo do stream. Ao utilizar chamadas de API ou ferramentas de desenvolvimento, você pode automatizar a escalabilidade do ambiente do Amazon Kinesis Streams para atender à demanda e pagar somente pelo que precisa.

Interfaces

Há duas interfaces para o Amazon Kinesis Streams: entrada, que é usada por produtores de dados para colocar os dados no Amazon Kinesis Streams; e saída, que é usada para processar e analisar os dados que entram. Os produtores podem gravar dados usando a API do PUT do Amazon Kinesis, uma abstração do kit de desenvolvimento de software (SDK) ou toolkit da AWS,[14] a Amazon Kinesis Producer Library (KPL),[15] ou o Amazon Kinesis Agent.[16]

Para processar dados que já foram colocados em um stream do Amazon Kinesis, são fornecidas bibliotecas cliente para criar e operar aplicativos de processamento de dados de streaming em tempo real. A KCL[17] age como um intermediário entre o Amazon Kinesis Streams e seus aplicativos comerciais que contêm a lógica de processamento específica. Há também uma integração que realiza a leitura de um stream do Amazon Kinesis no Apache Storm via Amazon Kinesis Storm Spout.[18]

Antipadrões

O Amazon Kinesis Streams tem os seguintes antipadrões:

- **Taxa de transferência consistente em pequena escala** – Mesmo que o Amazon Kinesis Streams funcione para streaming de dados a 200 KB/s ou menos, ele é projetado e otimizado para transferências de dados maiores.
- **Armazenamento físico de dados em longo prazo e análise** – O Amazon Kinesis Streams não é adequado para armazenamento de dados em longo prazo. Por padrão, os dados são retidos por 24 horas, sendo possível estender o período de retenção para até 7 dias. Você pode mover quaisquer dados que precisem ser armazenados por mais de 7 dias para outro serviço de armazenamento durável, como Amazon S3, Amazon Glacier, Amazon Redshift ou DynamoDB.

AWS Lambda

O AWS Lambda[19] permite executar códigos sem provisionar ou gerenciar servidores. Você paga somente pelo tempo de computação utilizado – não há cobrança quando seu código não está em execução. Com o Lambda, você pode executar códigos para qualquer tipo de aplicativo ou serviço de back-end praticamente, todos sem administração. Faça o upload do seu código e o Lambda cuidará de tudo que for necessário para executar e fazer o ajuste de escala de seu código com alta disponibilidade. Você pode configurar seu código para ser acionado automaticamente de outros serviços da AWS ou usá-lo diretamente em qualquer aplicativo da web ou móvel.

Padrões de uso ideal

O Lambda permite que você execute o código em resposta aos acionadores como alterações nos dados, mudanças de estado do sistema ou ações executadas por usuários. O Lambda pode ser acionado diretamente por serviços da AWS, como o Amazon S3, DynamoDB, Amazon Kinesis Streams, Amazon Simple Notification Service (Amazon SNS) e o Amazon CloudWatch, permitindo criar uma variedade de sistemas de processamento de dados em tempo real.

- **Processamento de arquivos em tempo real** – Você pode acionar o Lambda para invocar um processo no qual um arquivo foi carregado no Amazon S3 ou foi modificado. Por exemplo, alterar uma imagem em cores para a escala de cinzas após ter sido carregado no Amazon S3.

- **Processamento de stream em tempo real** – Você pode usar o Amazon Kinesis Streams e o Lambda para processar streaming de dados para análise de clickstream, filtragem de registros e análise de mídia social.
- **Extração, transformação e carregamento** – Você pode usar o Lambda para executar trabalhos que transformam os dados e os carregam de um repositório de dados a outro.
- **Substituição de cron** – Utilize expressões de agendamento para executar uma função do Lambda em intervalos regulares como uma solução mais econômica e disponível do que a execução do cron em uma instância do EC2.
- **Processamento de eventos da AWS** – Muitos outros serviços, como AWS CloudTrail, podem agir como fontes de eventos, basta fazer o login no Amazon S3 e usar as notificações de bucket do S3 para acionar as funções do Lambda.

Modelo de custo

Com o Lambda, você paga somente por aquilo que usa. Você é cobrado com base na quantidade de solicitações para suas funções e pelo tempo que seus códigos são executados. O nível gratuito do Lambda inclui 1 milhão de solicitações por mês gratuitas e 400.000 GB-segundos de tempo de computação por mês. Depois disso, é cobrado 0,20 USD por 1 milhão de solicitações (0,0000002 USD por solicitação). Além disso, a duração da execução de seu código é cobrada em relação à memória alocada. É cobrado 0,00001667 USD por cada GB-segundo usado. Para obter mais informações, consulte Definição de preço do AWS Lambda.

Desempenho

Depois de implantar seu código no Lambda pela primeira vez, suas funções tipicamente estão prontas para serem usadas dentro de segundos de upload. O Lambda é projetado para processar eventos dentro de milissegundos. A latência será maior imediatamente depois que uma função do Lambda for criada, atualizada ou se não tiver sido usada recentemente.

Durabilidade e disponibilidade

O Lambda é projetado para usar replicação e redundância para fornecer alta disponibilidade tanto para o serviço em si quanto para as funções do Lambda que opera. Não há janelas de manutenção ou tempos de inatividade agendados para nenhum dos processos. Quando há falha, as funções do Lambda que estão

sendo invocadas de modo síncrono responderão com uma exceção. Funções do Lambda sendo chamadas de forma assíncrona são repetidas pelo menos 3 vezes, após as quais o evento pode ser rejeitado.

Escalabilidade e elasticidade

Não há limite em relação à quantidade de funções do Lambda que você pode executar. Entretanto, o Lambda tem uma limitação de segurança padrão de 100 execuções concomitantes por conta por região. Um membro da equipe de suporte da AWS pode aumentar esse limite.

O Lambda é projetado para escalar de modo automático em seu nome. Não há limites fundamentais para a escalabilidade de uma função. O Lambda aloca a capacidade de modo dinâmico para corresponder à taxa de eventos de entrada.

Interfaces

As funções do Lambda podem ser gerenciadas de várias maneiras. Você pode listar, excluir, atualizar e monitorar facilmente suas funções do Lambda usando o painel no console do Lambda. Também é possível usar a CLI e o SDK da AWS para gerenciar suas funções do Lambda.

Você pode acionar uma função do Lambda de um evento da AWS, como notificações de bucket do Amazon S3, DynamoDB Streams, CloudWatch Logs, Amazon SES, Amazon Kinesis Streams, Amazon SNS, Amazon Cognito e muito mais. Qualquer chamada de API em qualquer serviço que ofereça suporte para o CloudTrail pode ser processada como um evento no Lambda por meio da resposta aos logs de auditoria do CloudTrail. Para obter mais informações sobre fontes de eventos, consulte Componentes principais: Função e origens de eventos do AWS Lambda.[20]

O Lambda oferece suporte para linguagens de programação como Java, Node.js e Python. Seu código pode incluir bibliotecas existentes, incluindo as nativas. As funções do Lambda podem iniciar facilmente processos usando linguagens suportadas pelo Amazon Linux AMI,[21] incluindo Bash, Go e Ruby. Para obter mais informações, consulte documentação de Node.js,[22] Python,[23] e Java.[24]

Antipadrões

O Lambda tem os seguintes antipadrões:

- **Aplicativos de execução prolongada** – Cada função do Lambda precisa ser concluída em 300 segundos. Para aplicativos de execução prolongada que possam exigir a execução de trabalhos por mais de cinco minutos, é recomendável o Amazon EC2 ou é possível criar uma cadeia de funções do Lambda em que a função 1 chama a função 2, que chama a função 3 e assim por diante até que o processo seja concluído.
- **Sites dinâmicos** – Enquanto é possível executar um site estático com o Lambda, a execução de um site altamente dinâmico e com grande volume pode ser proibitiva em relação ao desempenho. A utilização do Amazon EC2 e do Amazon CloudFront seria o caso de uso recomendado.
- **Aplicativos stateful** – O código do Lambda deve ser escrito em um estilo "stateless", ou seja, deve supor que não há afinidade com a infraestrutura de computação subjacente. Acesso ao sistema de arquivos local, processos secundários e artefatos similares podem não se estender além do tempo de vida da solicitação e qualquer estado persistente deve ser armazenado no Amazon S3, DynamoDB ou em outro serviço de armazenamento disponível na internet.

Amazon EMR

O Amazon EMR[25] é uma estrutura de computação altamente distribuída para processar dados facilmente e armazená-los de maneira rápida com um bom custo-benefício. O Amazon EMR usa o Apache Hadoop, uma estrutura de código aberto para distribuir os dados e processá-los em um cluster redimensionável de instâncias do Amazon EC2, além de permitir o uso das ferramentas mais comuns do Hadoop, como Hive, Pig, Spark, entre outras. O Hadoop fornece uma estrutura para executar o processamento e a análise de big data. O Amazon EMR faz todo o trabalho pesado envolvido no provisionamento, no gerenciamento e na manutenção da infraestrutura e do software de um cluster Hadoop.

Padrões de uso ideal

A estrutura flexível do Amazon EMR reduz os grandes problemas de processamento e conjuntos de dados em tarefas menores e os distribui entre muitos nós de computação em um cluster Hadoop. Esse recurso é empregado em muitos padrões de uso com análise de big data. Veja a seguir alguns exemplos:

- Processamento e análise de logs
- Extração, transformação e carregamento (ETL) e movimento de dados grandes
- Modelos de risco e análise de ameaças
- Segmentação de anúncios e análise de streams de cliques
- Genômica
- Análise preditiva
- Extração de dados e análise ad-hoc

Para obter mais informações, consulte o whitepaper Melhores práticas para o Amazon EMR[26].

Modelo de custo

Com o Amazon EMR, você pode iniciar um cluster persistente que permanece ativo indefinidamente ou um cluster temporário que é encerrado após a conclusão da análise. Em ambos os cenários, você paga somente pelas horas em que o cluster permanece ativo.

O Amazon EMR oferece suporte a diversos tipos de instância do Amazon EC2 (padrão, CPU de alta performance, alto uso de memória, alta taxa de E/S, entre outros) e a todas as opções de definição de preço de instância do Amazon EC2 (sob demanda, reservado e spot). Ao iniciar um cluster do Amazon EMR (também chamado de "fluxo de trabalho"), você escolhe a quantidade e o tipo de instâncias do Amazon EC2 a serem provisionadas. O preço do Amazon EMR é somado ao preço do Amazon EC2. Para obter mais informações, consulte a Definição de preço do Amazon EMR.[27]

Desempenho

O desempenho do Amazon EMR é impulsionado pelo tipo de instâncias do EC2 nas quais você opta por executar o cluster e na quantidade que deseja executar sua análise. Escolha um tipo de instância adequado aos seus requisitos de processamento, com nível suficiente de memória, armazenamento e capacidade de processamento. Para obter mais informações sobre as especificações de instância do EC2, consulte Tipos de instâncias do Amazon EC2.[28]

Durabilidade e disponibilidade

Por padrão, o Amazon EMR é tolerante a falhas de nó core e prossegue com a execução da tarefa se um nó escravo é desativado. No momento, o Amazon EMR não provisiona automaticamente outro nó para assumir o controle de escravos com falha, mas os clientes podem monitorar a integridade dos nós e substituir aqueles que apresentam falhas com o CloudWatch.

Para ajudar a enfrentar o evento improvável de falha de um nó principal, recomendamos que você faça backup dos seus dados em um armazenamento persistente, como o Amazon S3. Você também pode optar por executar o Amazon EMR com a distribuição de MapR,[29] que fornece uma arquitetura sem NameNode capaz de tolerar várias falhas simultâneas com failover e fallback automáticos. Os metadados são distribuídos e replicados da mesma maneira que os dados. Com uma arquitetura sem NameNode, não há limite prático para quantos arquivos podem ser armazenados, e também nenhuma dependência de armazenamento externo conectado à rede.

Escalabilidade e elasticidade

Com o Amazon EMR, é fácil redimensionar um cluster em execução.[30] Você pode adicionar nós core que mantêm o Hadoop Distributed File System (HDFS) a qualquer momento para aumentar seu poder de processamento e a capacidade de armazenamento (e taxa de transferência) do HDFS. Além disso, você pode usar o Amazon S3 no modo nativo ou utilizar o EMFS junto com o HDFS local ou no lugar dele, o que permite dissociar sua memória e computação do armazenamento, proporcionando maior flexibilidade e economia.

Também é possível adicionar e remover a qualquer momento nós de tarefas que podem processar trabalhos do Hadoop, mas não manter o HDFS. Alguns clientes adicionam centenas de instâncias aos seus clusters quando ocorre o processamento em lotes e removem as instâncias adicionais quando o processamento é concluído. Por exemplo, você pode não saber quantos dados seus clusters gerenciarão em seis meses ou ter diversas necessidades de processamento. Com o Amazon EMR, você não precisa adivinhar quais serão seus requisitos futuros nem obter provisionamentos para picos de demanda, pois pode adicionar ou remover capacidade facilmente a qualquer momento.

Além disso, você pode adicionar clusters totalmente novos de vários tamanhos e removê-los a qualquer momento com alguns cliques no console ou por meio de uma chamada de API programática[31].

Interfaces

O Amazon EMR oferece suporte a muitas ferramentas além do Hadoop que podem ser usadas para análise de big data, e cada uma delas possui suas próprias interfaces. Veja a seguir um breve resumo das opções mais populares:

Hive

O Hive é um data warehouse e um pacote de análises de código aberto executado além do Hadoop. O Hive é operado pela Hive QL, uma linguagem baseada em SQL que permite que os usuários estruturem, resumam e consultem dados. O Hive QL vai além do SQL padrão, adicionando suporte de primeira classe às funções mapear/reduzir e a tipos de dados complexos e extensíveis definidos pelo usuário, como JSON e Thrift. Esse recurso permite o processamento de fontes de dados complexas e não estruturadas, como documentos de texto e arquivos de log.

O Hive permite extensões de usuário por meio de funções definidas pelo usuário gravadas em Java. O Amazon EMR realizou diversas melhorias no Hive, incluindo a integração direta com o DynamoDB e o Amazon S3. Por exemplo, com o Amazon EMR é possível carregar partições de tabelas automaticamente do Amazon S3, gravar dados em tabelas no Amazon S3 sem usar arquivos temporários e acessar recursos no Amazon S3, como scripts para operações personalizadas de mapeamento e/ou redução e bibliotecas adicionais. Para obter mais informações, consulte Apache Hive[32] no *Guia de apresentação do EMR*.

Pig

O Pig é um pacote de análises de código aberto executado além do Hadoop. O Pig é operado pela Pig Latin, uma linguagem semelhante ao SQL que permite que os usuários estruturem, resumam e consultem dados. Assim como as operações semelhantes ao SQL, o Pig Latin também adiciona suporte de primeira classe para funções mapear e reduzir e tipos de dados complexos e extensíveis definidos pelo usuário. Esse recurso permite o processamento de fontes de dados complexas e não estruturadas, como documentos de texto e arquivos de log.

O Pig permite extensões de usuário por meio de funções definidas pelo usuário gravadas em Java. O Amazon EMR tem inúmeras melhorias para o Pig, incluindo a capacidade de usar vários sistemas de arquivos (normalmente, o Pig só pode acessar um sistema de arquivo remoto), a capacidade de carregar JARs e scripts de clientes do Amazon S3 (como "REGISTER s3://my-bucket/piggybank.jar") e funcionalidade adicional para o processamento de String e DateTime. Para obter mais informações, consulte Apache Pig[33] no *Guia de apresentação do EMR*.

Spark

O Spark é um mecanismo de análise de dados de código aberto desenvolvido no Hadoop com os princípios básicos do MapReduce na memória. O Spark proporciona mais velocidade a determinadas análises e é a base para outras ferramentas eficientes, como o Shark (data warehouse orientado por SQL), Spark Streaming (aplicativos de streaming), GraphX (sistemas de gráficos) e MLlib (aprendizagem de máquina). Para obter mais informações, consulte a postagem do blog Instalando o Apache Spark em um cluster do Amazon EMR.[34]

HBase

O HBase é um banco de dados distribuído, não relacional e de código aberto modelado de acordo com o BigTable do Google. Ele foi desenvolvido como parte do projeto Hadoop da Apache Software Foundation e é executado além do Hadoop Distributed File System (HDFS) para fornecer ao Hadoop recursos semelhantes aos do BigTable. O HBase fornece uma maneira eficiente e tolerante a falhas de armazenar grandes quantidades de dados esparsos usando compactação e armazenamento baseados em colunas. Além disso, o HBase proporciona consultas rápidas aos dados, pois eles são armazenados na memória, não em disco.

O HBase é otimizado para operações de gravação sequencial e altamente eficiente para inserções, atualizações e exclusões em lotes. O HBase funciona perfeitamente com o Hadoop, compartilhando seu sistema de arquivos e servindo como entrada e saída direta para tarefas do Hadoop. O HBase também se integra ao Apache Hive, possibilitando consultas tipo SQL em tabelas HBase, junções com tabelas baseadas no Hive e suporte para Java Database Connectivity (JDBC). Com o Amazon EMR, é possível fazer backup do HBase no Amazon S3 (completo ou incremental, manual ou automático) e restaurar a partir de um backup criado anteriormente. Para obter mais informações, consulte HBase e EMR[35] no *Guia do desenvolvedor do Amazon EMR*.

Impala

Impala é uma ferramenta de código aberto no ecossistema Hadoop para consulta ad hoc interativa que usa sintaxe SQL. Em vez de usar o MapReduce, ele utiliza um mecanismo de processamento maciçamente paralelo (MPP) semelhante ao encontrado em sistemas de gestão tradicional de banco de dados relacional (RDBMS). Com essa arquitetura, é possível consultar os dados nas tabelas HDFS ou HBase muito rapidamente e aproveitar a capacidade do Hadoop em processar tipos de dados diversos e fornecer o schema em tempo de execução. Isso faz da Impala uma ótima ferramenta para realizar análises interativas e de baixa latência.

A Impala também possui funções definidas pelo usuário em Java e C++ e pode se conectar a ferramentas de business intelligence (BI) por meio de drivers ODBC e JDBC. A Impala usa o metastore do Hive para reter informações sobre os dados de entrada, incluindo os nomes das partições e os tipos de dados. Para obter mais informações, consulte Impala e EMR[36] no *Guia do desenvolvedor do Amazon EMR*.

Hunk

O Hunk foi desenvolvido pela Splunk para tornar os dados automáticos acessíveis, utilizáveis e valiosos para todos. Com o Hunk, é possível explorar, analisar e visualizar dados armazenados no Amazon EMR e no Amazon S3 de maneira interativa, aproveitando a análise no Hadoop. Para obter mais informações, consulte Amazon EMR com Hunk: Análise do Splunk para Hadoop e NoSQL.[37]

Presto

Presto é um mecanismo de consulta de SQL distribuído de código aberto, otimizado para análise de dados ad-hoc de baixa latência. Ele oferece suporte para SQL padrão ANSI, incluindo consultas complexas, agregações, junções e funções de janela. O Presto pode processar dados de várias fontes de dados, incluindo o Hadoop Distributed File System (HDFS) e o Amazon S3.

Outras ferramentas de terceiros

O Amazon EMR também oferece suporte para uma variedade de outros aplicativos e ferramentas populares no ecossistema do Hadoop, como R (estatísticas), Mahout (aprendizagem de máquina), Ganglia (monitoramento), Accumulo (banco de dados NoSQL seguro), Hue (interface do usuário para analisar dados do Hadoop), Sqoop (conector de banco de dados relacional), HCatalog (gerenciamento de tabelas e armazenamento) e muito mais.

Além disso, você pode instalar seu próprio software além do Amazon EMR para atender às suas necessidades de negócios. A AWS permite mover rapidamente grandes quantidades de dados do Amazon S3 para o HDFS, do HDFS para o Amazon S3 e entre os buckets do Amazon S3 usando a S3DistCp,[38] do Amazon EMR, uma extensão da ferramenta de código aberto DistCp que usa o MapReduce para mover grandes quantidades de dados com eficiência.

Você pode opcionalmente usar o EMR File System (EMRFS), uma implementação do HDFS que permite que os clusters do Amazon EMR armazenem dados no Amazon S3. Você pode habilitar a criptografia no lado do servidor e do cliente do Amazon S3, assim como uma visualização consistente para o EMRFS. Ao utilizar o EMRFS, um armazenamento de metadados é construído de forma transparente no DynamoDB para ajudar a gerenciar as interações com o Amazon S3 e permitir que você tenha vários clusters do EMR que utilizem os mesmos metadados do EMRFS e armazenamento no Amazon S3.

Antipadrões

O Amazon EMR tem os seguintes antipadrões:

- **Conjuntos de dados pequenos** – O Amazon EMR foi desenvolvido para processamento paralelo maciço. Se o conjunto de dados é pequeno o suficiente para ser executado rapidamente em uma única máquina em um único thread, os custos indiretos adicionais para mapear e reduzir tarefas podem não valer a pena para conjuntos de dados pequenos que podem ser processados facilmente na memória em um único sistema.
- **Requisitos de transações ACID** – Embora haja formas de alcançar ACID (atomicidade, consistência, isolamento, durabilidade) ou ACID limitada no Hadoop, um outro banco de dados, como o Amazon RDS ou um banco de dados relacional sendo executado no Amazon EC2, poderá ser uma melhor opção para cargas de trabalho com requisitos rígidos.

Amazon Machine Learning

O Amazon ML[39] é um serviço que facilita o uso de análise preditiva e a tecnologia de aprendizado de máquina. O Amazon ML fornece ferramentas de visualização e assistentes que orientam você no processo de criação de modelos de aprendizado de máquina (ML) sem precisar aprender tecnologias e algoritmos ML complexos. Depois que seus modelos estiverem prontos, o Amazon ML gera

previsões para o seu aplicativo, usando operações de API, sem precisar implementar códigos de geração de previsão personalizados ou gerenciar qualquer tipo de infraestrutura.

O Amazon ML pode criar modelos de ML com base nos dados armazenados no Amazon S3, Amazon Redshift ou no Amazon RDS. Assistentes integrados orientam você ao longo de etapas, explorando interativamente seus dados para treinar o modelo de ML, avaliar a qualidade do modelo e ajustar os resultados a fim de se alinhar aos objetivos do negócio. Depois que um modelo estiver pronto, é possível solicitar previsões em lotes ou usando a API em tempo real de baixa latência.

Padrões de uso ideal

O Amazon ML é ideal para descobrir padrões em seus dados para criar modelos de ML que possam gerar previsões sobre pontos de dados novos e não vistos. Por exemplo, é possível:

- **Permitir que os aplicativos marquem transações suspeitas** – Construa um modelo de ML que prevê se uma nova transação é legítima ou fraudulenta.
- **Prever demanda de produto** – Insira informações do histórico de pedidos para prever quantidades futuras do pedido.
- **Personalizar conteúdo de aplicativos** – Preveja quais itens um usuário estará mais interessado e recupere essas previsões a partir de seu aplicativo em tempo real.
- **Prever atividades de usuários** – Analise o comportamento do usuário para personalizar seu site e fornece uma melhor experiência do usuário.
- **Ouvir mídias sociais** – Inclua e analise feeds de mídias sociais que tem o potencial de impactar as decisões comerciais.

Modelo de custo

Com o Amazon ML, pague somente pelo que você usar. Não há taxas mínimas nem compromissos antecipados. O Amazon ML cobra uma taxa por hora para o tempo de computação usado para construir os modelos preditivos e então você paga pelo número de previsões geradas para seu aplicativo. Para previsões em tempo real, você pode pagar uma tarifa de capacidade reservada por hora com base na quantidade de memória necessária para executar seu modelo.

A cobrança para a análise de dados, treinamento do modelo e avaliação é baseada no número de horas de computação necessárias para realizá-los, além de depender do tamanho dos dados de entrada, do número de atributos internos e da quantidade e tipos de transformações aplicadas. As taxas para a análise de dados e construção de modelos têm preço de 0,42 USD por hora. As taxas para previsão são categorizadas em lote e em tempo real. Previsões em lote custam 0,10 USD por 1.000 previsões, arredondadas para cima para as próximas 1.000, enquanto as previsões em tempo real custam 0,0001 USD por previsão, arredondadas para cima para o próximo centavo. Para previsões em tempo real, também existe uma tarifa de capacidade reservada de 0,001 USD por hora para cada 10 MB de memória provisionada para o modelo.

Durante a criação do modelo, você especifica o tamanho máximo da memória de cada modelo para gerenciar custos e controlar o desempenho preditivo. Você paga a tarifa de capacidade reservada apenas enquanto o modelo estiver habilitado para previsões em tempo real. As cobranças para dados armazenados no Amazon S3, Amazon RDS ou Amazon Redshift são faturadas separadamente. Para obter mais informações, consulte a Definição de preço do Amazon Machine Learning.[40]

Desempenho

O tempo gasto para a criação de modelos, ou para a orquestração de previsões em lote a partir desses modelos, depende do número de registros de dados de entrada, dos tipos e da distribuição dos atributos dentro desses registros e da complexidade do "recebimento" do processamento de dados que você especificar.

A maioria das solicitações de previsão em tempo real retornam uma reposta em 100 ms, o que as torna rápidas o suficiente para aplicativos interativos da web, móveis ou de desktop. O tempo exato gasto para que uma API em tempo real gere uma previsão varia de acordo com o tamanho do registro de dados de entrada e da complexidade do "recebimento"[41] do processamento de dados associado ao modelo de ML que está gerando as previsões. Por padrão, todos os modelos de ML que estão habilitados para previsões em tempo real podem ser usados para solicitar até 200 transações por segundo, o que pode ser aumentado bastando entrar em contato com o suporte ao cliente. Você pode monitorar o número de previsões solicitadas por seus modelos de ML usando as métricas do CloudWatch.

Durabilidade e disponibilidade

O Amazon ML é projetado para uma alta disponibilidade. Não há janelas de manutenção ou tempos de inatividade agendados. O serviço é executado nos centros de dados testados e de alta disponibilidade da Amazon, com replicação de pilha de serviço configurada em três locais em cada região da AWS para fornecer tolerância a falhas em caso de falha do servidor ou inativação da zona de disponibilidade.

Escalabilidade e elasticidade

Você pode processar conjuntos de dados com tamanho de até 100 GB para criar modelos de ML ou para solicitar previsões em lote. Para grandes volumes de previsões em lote, é possível dividir os registros de dados de entrada em blocos separados para permitir o processamento de um maior volume de dados de previsão.

Por padrão, é possível executar até cinco trabalhos simultâneos e, entrando em contato com o serviço de atendimento ao cliente, você pode aumentar esse limite. Como o Amazon ML é um serviço gerenciado, não há servidores para provisionar e, como resultado, é possível escalar o crescimento de seu aplicativo sem ter que provisionar excessivamente ou pagar por recursos não utilizados.

Interfaces

Criar uma fonte de dados é tão simples quanto adicionar seus dados no Amazon S3, ou você pode recuperar dados diretamente dos bancos de dados do Amazon Redshift ou MySQL gerenciados pelo Amazon RDS. Depois que sua fonte de dados for definida, é possível interagir com o Amazon ML usando o console. O acesso programático ao Amazon ML é ativado pelos AWS SDKs e Amazon ML API.[42] Você também pode criar e gerenciar entidades do Amazon ML usando o AWS CLI disponível nos sistemas Windows, Mac e Linux/UNIX.

Antipadrões

O Amazon ML tem os seguintes antipadrões:

- **Conjuntos de dados muito grandes** – Enquanto o Amazon ML pode oferecer suporte a 100 GB de dados, não há suporte para a inclusão de dados em escala de terabyte no momento. A utilização do Amazon EMR para executar a Machine Learning Library (MLlib) do Spark é uma ferramenta comum para tal caso de uso.

- **Tarefas de aprendizagem incompatíveis** – O Amazon ML pode ser usado para criar modelos de ML que realizam a classificação binária (escolhe um de duas escolhas e fornece uma medida de confiança), classificação multiclasse (estende a escolha além de duas opções) ou regressão numérica (prevê um número diretamente). Tarefas de ML incompatíveis como previsão de sequência ou clustering não supervisionado podem ser realizadas com o uso do Amazon EMR para executar o Spark e a MLlib.

Amazon DynamoDB

O Amazon DynamoDB[43] é um serviço de banco de dados NoSQL rápido e totalmente gerenciado que facilita e faz com que o armazenamento e a recuperação de qualquer quantidade de dados sejam econômicos, além de atender a qualquer nível de tráfego de solicitações. O DynamoDB ajuda a aliviar a carga administrativa de operar e escalar um cluster de banco de dados distribuído altamente disponível. Essa alternativa de armazenamento atende aos requisitos de latência e de taxa de transferência de aplicativos de alta demanda oferecendo latência inferior a 10 milissegundos e desempenho previsível com taxa de transferência total contínua e escalabilidade de armazenamento.

O DynamoDB armazena dados estruturados em tabelas, indexadas por chave primária, e permite acesso de gravação e leitura de baixa latência a itens que variam de 1 byte a 400 KB. O DynamoDB oferece suporte a três tipos de dados (número, string e binário), tanto em conjuntos escalares quanto em multivalores. Ele oferece suporte a armazenamentos de documentos como JSON, XML ou HTML nesses tipos de dados. Como as tabelas não têm um esquema fixo, cada item de dados pode ter um número diferente de atributos. A chave primária pode ser tanto uma chave de atributo hash único como uma chave de intervalo de hash composto.

O DynamoDB oferece índices secundários globais e locais, além de fornecer flexibilidade adicional para consulta de atributos além da chave primária. O DynamoDB oferece leituras consistentes eventuais (por padrão) e leituras consistentes fortes (opcionais), bem como transações de nível de item implícitas para operações de colocação, atualização, exclusão, condicionais e de incremento/decremento de item.

O DynamoDB está integrado a outros serviços, como Amazon EMR, Amazon Redshift, AWS Data Pipeline e Amazon S3, para análise, data warehouse, importação/exportação de dados, backup e arquivamento.

Padrões de uso ideal

O DynamoDB é ideal para aplicativos existentes ou novos que precisem de um banco de dados NoSQL flexível com baixas latências de leitura e gravação, além da capacidade de escalar o armazenamento e a taxa de transferência para mais ou para menos conforme necessário, sem alterações de código ou tempo de inatividade.

Entre os casos de uso comuns estão:

- Aplicativos móveis
- Jogos
- Veiculação de anúncios digital
- Votações dinâmicas
- Interação com o público em eventos ao vivo
- Redes de sensores
- Inclusão de log
- Controle de acesso para conteúdo com base na web
- Armazenamento de metadados para objetos do Amazon S3
- Carrinhos de compras de comércio eletrônico
- Gerenciamento de sessão de web

Muitos desses casos de uso exigem um banco de dados escalável e altamente disponível porque o tempo de inatividade ou a redução do desempenho tem um impacto negativo imediato sobre o negócio de uma organização.

Modelo de custo

Com o DynamoDB, você só paga por aquilo que usa e não existe taxa mínima. O DynamoDB possui três componentes de preços: capacidade de taxa de transferência provisionada (por hora), armazenamento físico de dados indexados (por GB por mês), entrada ou saída de dados (por GB por mês). Novos clientes podem começar a usar o DynamoDB gratuitamente como parte do nível de uso gratuito da AWS.[44] Para obter mais informações, consulte Definição de preço do Amazon DynamoDB.[45]

Desempenho

SSDs e indexação limitadora em atributos possibilitam uma alta taxa de transferência e baixa latência,[46] e reduz drasticamente o custo das operações de leitura e gravação. À medida que os conjuntos de dados crescem, o desempenho previsível é obrigatório de maneira que a baixa latência das cargas de trabalho possa ser mantida. O desempenho desejável pode ser obtido definindo-se o throughput provisionado exigido para uma determinada tabela.

Nos bastidores, o serviço processa o provisionamento dos recursos para atingir a taxa de transferência solicitada; não é preciso pensar em instâncias, hardware, memória e outros fatores que possam afetar a taxa de transferência de um aplicativo. As reservas de throughput provisionado são elásticas e podem ser aumentadas ou diminuídas sob demanda.

Durabilidade e disponibilidade

O DynamoDB tem um sistema de tolerância a falhas integrado que replica de maneira automática e síncrona os dados nos três datacenters de uma região tendo em vista a alta disponibilidade e para ajudar a proteger os dados contra falhas de máquinas individuais ou até mesmo de toda a instalação. O DynamoDB Streams[47] captura todas as atividades de dados que acontecem em sua mesa e permite a definição de replicação regional de uma região geográfica a outra para proporcionar ainda mais disponibilidade.

Escalabilidade e elasticidade

O DynamoDB é altamente escalável e elástico. Não há limite para o volume de dados que você pode armazenar em uma tabela do DynamoDB, e o serviço aloca automaticamente mais armazenamento à medida que você armazena mais dados usando as operações de API de gravação do DynamoDB. Os dados são particionados e reparticionados automaticamente conforme necessário, e o uso de SSDs oferece tempos de resposta de baixa latência previsíveis em qualquer escala. O serviço também é elástico, pois você pode simplesmente “aumentar”[48] ou “diminuir”[49] a capacidade de leitura e gravação de uma tabela de acordo com a mudança das suas necessidades.

Interfaces

O DynamoDB fornece uma API REST de baixo nível, bem como SDKs for Java de nível superior, .NET e PHP que envolvem a API REST de baixo nível e fornecem

algumas funções de mapeamento relacional de objeto (ORM). Essas APIs oferecem uma interface de gerenciamento e dados para o DynamoDB. Atualmente, a API oferece operações que permitem o gerenciamento de tabelas (criar, listar, excluir e obter metadados) e o trabalho com atributos (obter, gravar e excluir atributos; consulta por meio de um índice e varredura completa).

Embora o SQL padrão não esteja disponível, convém usar a operação de seleção do DynamoDB para criar consultas SQL similares que recuperam um conjunto de atributos com base em critérios fornecidos por você. Você também pode trabalhar com o DynamoDB usando o console.

Antipadrões

O DynamoDB tem os seguintes antipadrões:

- **Aplicativos pré-gravados vinculados a um banco de dados relacional tradicional** – Se você estiver tentando transportar um aplicativo existente para a Nuvem AWS e precisa continuar usando um banco de dados relacional, você pode optar por usar o Amazon RDS (Amazon Aurora, MySQL, PostgreSQL, Oracle ou SQL Server) ou um dos muitos AMIs de banco de dados do Amazon EC2 pré-configurados. Também é possível instalar um software de banco de dados de sua escolha em uma instância do EC2 gerenciada por você.
- **Associações e/ou transações complexas** – Embora muitas soluções sejam capazes de utilizar o DynamoDB para dar suporte a seus usuários, é possível que o aplicativo exija associações, transações complexas e outras infraestruturas relacionais fornecidas por plataformas de banco de dados tradicionais. Se for esse o caso, você pode explorar o Amazon Redshift, Amazon RDS ou Amazon EC2 com um banco de dados autogerenciado.
- **Dados de objetos binários grandes (BLOB)** – Se pretender armazenar dados BLOB grandes (maiores que 400 KB), como vídeos, imagens ou músicas digitais, você deverá levar em consideração o Amazon S3. No entanto, o DynamoDB ainda tem um papel a desempenhar nesse cenário para manter registro dos metadados (por exemplo, nome do item, tamanho, data da criação, proprietário e local, etc.) sobre os objetos binários.
- **Dados grandes com taxa de E/S baixa** – O DynamoDB usa unidades SSD e está otimizado para cargas de trabalho com uma taxa de E/S alta por GB armazenado. Caso você pretenda armazenar volumes muito grandes de

dados não acessados com frequência, outras opções podem ser uma escolha melhor, como o Amazon S3.

Amazon Redshift

O Amazon Redshift[50] é um serviço de data warehouse rápido, totalmente gerenciado e em escala de petabytes, que torna mais simples e acessível a análise eficiente de todos os seus dados usando as ferramentas de inteligência de negócios de que você dispõe. O serviço é otimizado para conjuntos de dados que variam desde algumas centenas de gigabytes a um petabyte ou mais e é projetado para custar um décimo do custo das soluções tradicionais de data warehouse.

O Amazon Redshift proporciona consulta rápida e desempenho de E/S para conjuntos de dados de praticamente qualquer tamanho por usar tecnologia de armazenamento colunar, enquanto paraleliza e distribui consultas entre vários nós. Ele automatiza a maioria das tarefas administrativas comuns ligadas a provisionamento, configuração, monitoramento, backup e segurança de data warehouse, o que deixa o gerenciamento e a manutenção mais fáceis e baratos. Essa automação permite a construção de data warehouses em escala de petabytes em minutos, em vez de semanas ou meses gastos pelas implementações tradicionais no local.

Padrões de uso ideal

O Amazon Redshift é ideal para online analytical processing (OLAP) usando suas ferramentas existentes de business intelligence. As organizações estão usando o Amazon Redshift para:

- Analisar dados globais de vendas de vários produtos
- Armazenar dados históricos sobre a comercialização de ações
- Analisar impressões e cliques em anúncios
- Agregar dados de jogos
- Analisar tendências sociais
- Avaliar a qualidade clínica, a eficiência das operações e o desempenho financeiro no setor de assistência médica

Modelo de custo

Um cluster de data warehouse do Amazon Redshift não requer compromissos de longo prazo nem custos antecipados. Assim, você fica livre do investimento e da

complexidade de planejar e comprar capacidade de data warehouse além das suas necessidades. As cobranças se baseiam no tamanho e no número de nós do seu cluster.

Não há custos adicionais para armazenamento de backup de até 100% do seu armazenamento provisionado. Por exemplo, se você tiver um cluster ativo com 2 nós XL para um total de 4 TB de armazenamento, a AWS fornecerá até 4 TB de armazenamento de backup para o Amazon S3 sem custos adicionais. O armazenamento de backup além do tamanho do armazenamento provisionado, e os backups armazenados após o encerramento do cluster, são faturados em taxas padrão do Amazon S3.[51] Não há taxa de transferência de dados para comunicação entre o Amazon S3 e o Amazon Redshift. Para obter mais informações, consulte a Definição de preço do Amazon Redshift.[52]

Desempenho

O Amazon Redshift usa diversas inovações para obter um desempenho muito elevado em conjuntos de dados que vão de centenas de gigabytes a um petabyte ou mais. Ele usa armazenamento colunar, compactação de dados e mapas de zona para reduzir a quantidade de E/S necessária para realizar consultas.

O Amazon Redshift tem uma arquitetura de processamento maciçamente paralelo (MPP), paralelizando e distribuindo operações SQL para aproveitar todos os recursos disponíveis. O hardware subjacente foi projetado para processamento de dados de alto desempenho, usando armazenamento vinculado local para maximizar as taxas de transferência entre as CPUs e as unidades, e uma rede de malha de 10 GigE para maximizar as taxas de transferência entre os nós. O desempenho pode ser ajustado dependendo das necessidades do data warehouse: a AWS oferece Dense Compute (DC) com opções de unidades SSD e Dense Storage (DS).

Durabilidade e disponibilidade

O Amazon Redshift detecta e substitui automaticamente um nó com falha em seu cluster de data warehouse. O cluster de data warehouse estará em modo somente leitura até que um nó de substituição seja provisionado e adicionado ao banco de dados, o que costuma levar somente alguns minutos. O Amazon Redshift disponibiliza imediatamente seu nó de substituição e executa seus dados acessados com mais frequência a partir do Amazon S3 para permitir que você continue consultando seus dados o mais rápido possível.

Além disso, seu cluster de data warehouse continuará disponível se houver uma falha na unidade, pois o Amazon Redshift espelha os dados através do cluster e os utiliza de outro nó para reconstruir as unidades com falha. Os clusters do Amazon Redshift residem dentro de uma Zona de disponibilidade,[53] mas se você quiser ter uma configuração multi-AZ para o Amazon Redshift, você pode configurar um espelho e, então, fazer o autogerenciamento da replicação e do failover.

Escalabilidade e elasticidade

Com alguns cliques no console ou com uma chamada de API,[54] você pode alterar facilmente o número, ou o tipo, de nós no data warehouse de acordo com suas necessidades de mudanças no desempenho ou na capacidade. O Amazon Redshift permite que você inicie com um único nó de 160 GB e escale até um petabyte ou mais de dados compactados de usuário usando vários nós. Para obter mais informações, consulte a seção Sobre clusters e nós[55], o tópico de Clusters do Amazon Redshift, no *Guia de gerenciamento do Amazon Redshift*.

Durante o redimensionamento, o Amazon Redshift coloca seu cluster existente no modo somente leitura, provisiona um novo cluster do tamanho que você escolher e copia os dados do cluster antigo para o novo em paralelo. Durante esse processo, você paga somente pelo cluster ativo do Amazon Redshift. Você poderá continuar executando consultas no seu antigo cluster enquanto o novo estiver sendo provisionado. Depois que seus dados forem copiados para o novo cluster, o Amazon Redshift redirecionará automaticamente as consultas para ele e removerá o cluster antigo.

Interfaces

O Amazon Redshift tem drivers personalizados de JDBC e ODBC que você pode baixar pela guia Connect Client do console, permitindo a utilização de uma ampla série de clientes SQL conhecidos. Você também pode usar os drivers PostgreSQL JDBC e ODBC padrão. Para obter mais informações sobre os drivers do Amazon Redshift, consulte o Amazon Redshift e PostgreSQL.[56]

Existem vários exemplos de integrações validadas com muito BI popular e fornecedores de ETL.[57] Cargas e descargas são tentadas em paralelo em cada nó de computação para maximizar a taxa na qual você pode ingerir dados em seu cluster de data warehouse, bem como para o Amazon S3 e DynamoDB. Você pode facilmente carregar dados de streaming ao Amazon Redshift usando

Amazon Kinesis Firehose, permitindo análise em tempo quase real com ferramentas e painéis existentes de Business Intelligence que você já utiliza no momento. Métricas da utilização de computação, de memória, de armazenamento e de tráfego de leitura/gravação para seu cluster de data warehouse do Amazon Redshift estão disponíveis gratuitamente via console ou as operações de API do CloudWatch.

Antipadrões

O Amazon Redshift tem os seguintes antipadrões:

- **Conjuntos de dados pequenos** – O Amazon Redshift é projetado para processamento em paralelo entre um cluster; se o conjunto de dados for inferior a 100 gigabytes, você não conseguirá aproveitar todos os benefícios que o Amazon Redshift tem a oferecer, e o Amazon RDS pode ser uma solução melhor.
- **Processamento de transações on-line (OLTP)** – O Amazon Redshift foi desenvolvido para cargas de trabalho de data warehouse que produzem recursos de análise extremamente rápidos e econômicos. Se você precisar de um sistema transacional rápido, talvez seja melhor escolher um sistema de banco de dados relacional tradicional construído sobre Amazon RDS ou um banco de dados NoSQL como o DynamoDB.
- **Dados não estruturados** – Os dados no Amazon Redshift devem ser estruturados por um esquema definido, não por uma estrutura de esquema arbitrário para cada linha. Se seus dados são não estruturados, você pode realizar processos de ETL (extração, transformação e carregamento) no Amazon EMR para preparar os dados para o carregamento no Amazon Redshift.
- **Dados BLOB** – Se planeja armazenar arquivos binários grandes (como vídeos, imagens ou músicas digitais), você pode armazenar os dados no Amazon S3 e fazer referência à localização deles no Amazon Redshift. Nesse cenário, o Amazon Redshift rastreia os metadados (como nome, tamanho, data da criação, proprietário, localização do item, etc.) sobre seus objetos binários, mas os grandes objetos em seriam armazenados no Amazon S3.

Amazon Elasticsearch Service

O Amazon ES[58] é um serviço gerenciado que facilita a implantação, operação e escalonamento do Elasticsearch na Nuvem AWS. O Elasticsearch é um mecanismo de busca e análise distribuído em tempo real. Ele permite que você explore seus dados em uma velocidade e escala nunca antes vistas. É usado para pesquisa de texto completo, pesquisa estruturada e análise, além de uma combinação dessas três categorias.

Você pode definir e configurar seu cluster do Amazon ES em questão de minutos usando o console. O Amazon ES gerencia o trabalho envolvido na configuração de um domínio, desde o provisionamento da capacidade de infraestrutura desejada até a instalação do software do Elasticsearch.

Após seu domínio estar em execução, o Amazon ES automatiza tarefas administrativas comuns, como a realização de backups, o monitoramento de instâncias e a aplicação de patches de software que alimenta sua Instância do Amazon ES. Ele automaticamente detecta e substitui os nós do Elasticsearch com falha, reduzindo os custos indiretos associados à infraestrutura autogerenciada e ao software do Elasticsearch. O serviço permite que você escale facilmente seu cluster com uma única chamada de API ou alguns cliques no console.

Com o Amazon ES, você obtém acesso direto a API de código aberto do Elasticsearch, de modo que o código e os aplicativos que você já usa com seus ambientes existentes funcionem perfeitamente. Ele suporta a integração com o Logstash, um pipeline de dados de código aberto que ajuda a processar logs e outros dados de eventos. Ele também inclui suporte integrado para o Kibana é uma plataforma de análise e visualização de código aberto que ajuda você a obter um melhor entendimento de seus dados.

Padrões de uso ideal

O Amazon ES é ideal para consulta e pesquisa de grandes quantidades de dados.

As organizações podem usar o Amazon ES para:

- Analisar logs de atividades, como logs para aplicativos ou sites voltados ao cliente
- Analisar logs do CloudWatch com o Elasticsearch
- Analisar dados de uso de produtos oriundos de vários serviços e sistemas

- Analisar sentimentos de mídias sociais e dados de CRM, e encontrar tendências para marcas e produtos
- Analise as atualizações do fluxo de dados de outros serviços da AWS, como o Amazon Kinesis Streams e o DynamoDB
- Fornece aos clientes uma experiência de pesquisa e navegação surpreendente
- Monitore o uso para aplicativos móveis

Modelo de custo

Com o Amazon ES, você paga somente pelos recursos de computação e armazenamento que utilizar. Não há tarifas mínimas nem compromissos antecipados. Você é cobrado pelas horas da instância do Amazon ES, pelo armazenamento do Amazon EBS (caso escolha essa opção) e pelas tarifas de transferência de dados padrão.[59]

Caso utilize volumes do EBS para armazenamento, o Amazon ES permite que você escolha o tipo de volume. Caso escolha o armazenamento com Provisioned IOPS (SSD),[60] você será cobrado pelo armazenamento e pela taxa de transferência provisionada. No entanto, você não é cobrado pela E/S que consumir. Você também tem a opção de pagar por armazenamento adicional com base no tamanho cumulativo dos volumes do EBS anexados aos nós de dados em seu domínio.

O Amazon ES fornece espaço de armazenamento para snapshots automatizados sem custo para cada domínio do Amazon ES. Snapshots manuais são cobrados de acordo com as taxas de armazenamento do Amazon S3. Para obter mais informações, consulte a Definição de preço do Amazon Elasticsearch Service.[61]

Desempenho

O desempenho do Amazon ES depende de vários fatores que incluem tipo de instância, carga de trabalho, índice, número de estilhaços usados, réplicas de leitura e configurações de armazenamento (armazenamento de instâncias ou do EBS, como SSD para finalidade geral). Os índices são compostos de estilhaços de dados que podem ser distribuídos em instâncias diferentes em diversas zonas de disponibilidade.

As réplicas de leitura dos estilhaços são mantidas pelo Amazon ES em uma zona de disponibilidade diferente caso o reconhecimento da zona seja verificado. O Amazon ES pode usar o armazenamento rápido de instâncias em SSD para

índices de armazenamento e também vários volumes do EBS. Um mecanismo de pesquisa utiliza intensamente os dispositivos de armazenamento. Portanto, ao tornar os discos mais rápidos, o resultado é um desempenho de consulta e pesquisa mais rápido.

Durabilidade e disponibilidade

Você pode configurar seus domínios do Amazon ES para alta disponibilidade, habilitando a opção de reconhecimento de zona no momento da criação do domínio ou modificando um domínio existente. Quando o reconhecimento de zona está habilitado, o Amazon ES distribui as instâncias que dão suporte ao domínio em duas diferentes zonas de disponibilidade. Então, caso as réplicas no Elasticsearch estejam habilitadas, as instâncias são automaticamente distribuídas de modo a entregar a replicação entre as zonas.

Você pode construir durabilidade de dados para o domínio do Amazon ES por meio de snapshots automatizados e manuais. É possível usar snapshots para recuperar seu domínio com dados pré-carregados ou para criar um novo domínio com dados pré-carregados. Os snapshots são armazenados no Amazon S3, que é um armazenamento de objetos seguro, durável e altamente escalável. Por padrão, o Amazon ES cria automaticamente snapshots diários de cada domínio. Além disso, você pode usar as APIs de snapshot do Amazon ES para criar snapshots manuais adicionais. Os snapshots manuais são armazenados no Amazon S3. Os snapshots manuais podem ser usados para recuperação de desastres entre regiões e para fornecer durabilidade adicional.

Escalabilidade e elasticidade

Você pode adicionar ou remover instâncias e modificar facilmente os volumes do Amazon EBS para acomodar o crescimento dos dados. Você pode escrever algumas linhas de código para monitorar o estado de seu domínio por meio das métricas do CloudWatch e chamar a API do Amazon ES para escalar seu domínio para cima ou para baixo com base nos limites definidos por você. O serviço executa a escalabilidade sem tempo de inatividade.

O Amazon ES oferece suporte a um volume do EBS (tamanho máximo de 512 GB) por instância associado a um cluster. Com um máximo de 10 instâncias permitidas por cluster do Amazon ES, os clientes podem alocar cerca de 5 TB de armazenamento para um único domínio do Amazon ES.

Interfaces

O Amazon ES oferece suporte à API do Elasticsearch[62], para que o código, os aplicativos e as ferramentas populares que você já usa com seus ambientes existentes do Elasticsearch funcionem perfeitamente. Como os SDKs da AWS são compatíveis com todas as operações de API do Amazon ES, é fácil gerenciar e interagir com seus domínios usando a tecnologia de sua preferência. A CLI da AWS ou o console também podem ser usados para criar e gerenciar seus domínios.

O Amazon ES oferece suporte para a integração com vários serviços da AWS, incluindo streaming de dados do Amazon S3, Amazon Kinesis Streams e DynamoDB Streams. As integrações utilizam uma função do Lambda como um manipulador de eventos na nuvem que responde a dados novos, processando-os e realizando o streaming dos dados para seu domínio do Amazon ES. O Amazon ES também se integra com o CloudWatch, para o monitoramento das métricas do domínio do Amazon ES, e com o CloudTrail, para auditoria de chamadas de API de configuração para domínios do Amazon ES.

O Amazon ES inclui integração com o Kibana, uma plataforma de análise e visualização de código aberto, e oferece suporte à integração com o Logstash, um pipeline de dados de código aberto que ajuda a processar logs e outros dados de eventos. Você pode configurar seu domínio do Amazon ES como armazenamento back-end para todos os logs oriundos da implementação do Logstash para incluir facilmente dados estruturados e não estruturados de uma variedade de fontes.

Antipadrões

O Amazon ES tem os seguintes antipadrões:

- **Processamento de transações on-line (OLTP)** – O Amazon ES é mecanismo de busca e análise distribuído em tempo real. Não há suporte para transações ou processamento de dados na manipulação dos dados. Se você precisa de um sistema transacional rápido, a melhor opção é um sistema tradicional de banco de dados relacional com base no Amazon RDS ou um banco de dados NoSQL que oferece funcionalidades como o DynamoDB.
- **Armazenamento em petabytes** – Com um máximo de 10 instâncias permitidas por cluster do Amazon ES, você pode alocar cerca de 5 TB de armazenamento para um único domínio do Amazon ES. Para cargas de

trabalho maiores que isso, considere o uso do Elasticsearch autogerenciado no Amazon EC2.

Amazon QuickSight

Em outubro de 2015, a AWS lançou uma demonstração do Amazon QuickSight, um serviço de Business Intelligence (BI) rápido e na nuvem que facilita a criação de visualizações, executa análise ad-hoc e obtém com rapidez insights de negócios com base nos seus dados.

O QuickSight utiliza um novo mecanismo de cálculo, super-rápido, paralelo e guardado na memória (SPICE) para realizar cálculos avançados e renderizar visualizações rapidamente. O QuickSight se integra automaticamente com os serviços de dados da AWS, permite que as organizações sejam escalonadas para centenas de milhares de usuários e proporciona desempenho de consulta rápido e responsivo por meio do mecanismo de consulta SPICE. Com um décimo do custo das soluções tradicionais, o QuickSight permite que você ofereça funcionalidade de BI a um custo econômico para qualquer pessoa em sua organização. Para saber mais e se cadastrar para a demonstração, consulte o QuickSight.[63]

Amazon EC2

O Amazon EC2,[64] com instâncias que atuam como máquinas virtuais da AWS, oferece uma plataforma ideal para a operação de seus próprios aplicativos de análise de big data autogerenciados na infraestrutura da AWS. Quase todos os softwares que você pode instalar em ambientes virtualizados Linux ou Windows podem ser executados no Amazon EC2 e é possível usar o modelo de preço pago pelo uso. O que você não obtém são os serviços gerenciados no nível do aplicativo que vêm com os outros serviços mencionados neste whitepaper. Há várias opções de análise de big data autogerenciada; aqui estão alguns exemplos:

- Oferta de NoSQL, como o MongoDB
- Data warehouse ou armazenamento colunar, como o Vertica
- Cluster do Hadoop
- Cluster do Apache Storm
- Ambiente do Apache Kafka

Padrões de uso ideal

- **Ambiente especializado** – Ao executar um aplicativo personalizado, uma variação de um conjunto do Hadoop padrão ou um aplicativo não coberto por uma de nossas outras ofertas, o Amazon EC2 fornecerá a flexibilidade e escalabilidade para atender às suas necessidades de computação.
- **Requisitos de conformidade** – Determinados requisitos de conformidade podem exigir que você mesmo execute os aplicativos no Amazon EC2, em vez de usar uma oferta de serviço gerenciado.

Modelo de custo

O Amazon EC2 tem diversos tipos de instância em várias famílias de instâncias (padrão, CPU de alta performance, alto uso de memória, alta taxa de E/S, etc.) e diferentes opções de preço (sob demanda, reservado e spot). Dependendo dos requisitos de seu aplicativo, você pode usar serviços adicionais juntamente com o Amazon EC2, como o Amazon Elastic Block Store (Amazon EBS), para armazenamento persistente vinculado diretamente, ou o Amazon S3, como um depósito de objetos duráveis, sendo que cada um deles tem seu próprio modelo de preços. Caso execute o aplicativo de big data no Amazon EC2, você será responsável pelas taxas de licença, como faria em seu próprio datacenter. O AWS Marketplace[65] oferece diversos pacotes de softwares de big data de terceiros pré-configurados para serem inicializados com o simples clique de um botão.

Desempenho

O desempenho no Amazon EC2 será determinado pelo tipo de instância que você escolher para a plataforma de big data. Cada tipo de instância tem uma quantidade diferente de operações de IOPs, CPU, RAM, armazenamento e recursos de rede, de modo que você pode escolher o nível de desempenho correto para os requisitos de seu aplicativo.

Durabilidade e disponibilidade

Aplicativos críticos devem ser executados em um cluster em várias zonas de disponibilidade em uma região da AWS para que nenhuma falha de instância ou do datacenter afete os usuários do aplicativo. Para aplicativos críticos sem tempo de atividade, você pode fazer backup do aplicativo no Amazon S3 e restaurá-lo para qualquer zona de disponibilidade na região se ocorrer falha de uma instância ou da zona de disponibilidade. Existem outras opções dependendo de qual aplicativo está sendo executado e de quais são seus requisitos, como o espelhamento do aplicativo.

Escalabilidade e elasticidade

O Auto Scaling[66] é um serviço que permite que você escale automaticamente a capacidade do Amazon EC2 para mais ou para menos de acordo com as condições definidas. Com o Auto Scaling, você pode garantir que o número de instâncias do EC2 que você está usando aumente facilmente durante os picos de demanda para manter o desempenho e diminua automaticamente durante quedas na demanda para minimizar custos. O Auto Scaling é ideal para aplicativos que experimentam variação de uso a cada hora, dia ou semana. O Auto Scaling é ativado pelo CloudWatch e é disponibilizado sem custo adicional além das taxas do CloudWatch.

Interfaces

O Amazon EC2 pode ser interligado de forma programática utilizando API, SDK ou o console. Métricas da utilização de computação, de memória, de armazenamento, de consumo de rede e de tráfego de leitura/gravação para suas instâncias estão disponíveis gratuitamente via console ou operações de API do CloudWatch.

As interfaces para seu software de análise de big data que é executado sobre o Amazon EC2 varia de acordo com as características do software que você escolher.

Antipadrões

O Amazon EC2 tem os seguintes antipadrões:

- **Serviço gerenciado** – Se a sua necessidade for de uma oferta de serviço gerenciado em que você separa a camada de infraestrutura e administração da análise de big data, então este modelo "faça você mesmo" de gerenciar seu próprio software de análise no Amazon EC2 pode não ser a escolha correta para seu caso de uso.
- **Falta de expertise ou recursos** – Se a sua organização não tem ou não pretende gastar recursos ou expertise para instalar e gerenciar um sistema de alta disponibilidade, você deve considerar utilizar um equivalente da AWS, como o Amazon EMR, o DynamoDB, o Amazon Kinesis Streams ou o Amazon Redshift.

Solução de problemas de big data na AWS

Neste whitepaper, examinamos algumas das ferramentas da AWS que estão à sua disposição para análise de big data. Aqui você encontra um bom ponto de referência para começar a projetar seus aplicativos de big data. Entretanto, existem aspectos adicionais que você deve considerar ao selecionar as ferramentas certas para seu caso de uso específico. Em geral, cada carga de trabalho de análise terá certas características e requisitos que determinam qual ferramenta usar, como:

- Com que rapidez você precisa dos resultados analíticos? Em tempo real, em segundos ou uma hora é o período de tempo mais apropriado?
- Que valor essas análises agregarão à sua organização e quais são as limitações orçamentárias?
- Qual é o tamanho dos dados e qual sua taxa de crescimento?
- Como os dados são estruturados?
- Quais recursos de integração os produtores e consumidores têm?
- Qual é o grau de latência aceitável entre os produtores e consumidores?
- Qual o custo do tempo de inatividade ou qual o nível de disponibilidade e resiliência que a solução precisa ter?
- A carga de trabalho analítica é consistente ou elástica?

Cada uma dessas características ou requisitos ajuda a orientá-lo sobre qual ferramenta utilizar. Em alguns casos, você pode mapear de forma simples e fácil suas cargas de trabalho de análise de big data para um dos serviços, com base em um conjunto de requisitos. Entretanto, na maioria das cargas de trabalho analíticas de big data do mundo real, há muitas características e requisitos diferentes, às vezes conflitantes, no mesmo conjunto de dados.

Por exemplo, alguns conjuntos de resultados podem ter requisitos em tempo real conforme um usuário interage com um sistema, enquanto outras análises poderiam ser agrupadas em lotes e serem executadas diariamente. Esses requisitos diferentes sobre o mesmo conjunto de dados devem ser dissociados e resolvidos usando mais de uma ferramenta. Se você tentar resolver ambos os exemplos acima com o mesmo conjunto de ferramentas, acabará gerando um provisionamento excessivo e consequentemente pagará mais por um tempo de

resposta desnecessário ou terá uma solução que não é rápida o suficiente para responder aos usuários em tempo real. Ao combinar a ferramenta mais adequada com cada conjunto individual de problemas analíticos, o resultado é o uso mais econômico de seus recursos de computação e armazenamento.

Big data não significa necessariamente "big costs". Portanto, ao projetar seus aplicativos, é importante certificar-se de que seu projeto tenha um bom custo-benefício. Caso não seja econômico, em relação às alternativa, então provavelmente o projeto não é o correto. Outra ideia errônea comum é ter vários conjuntos de ferramentas para resolver um problema de big data que traz um maior custo ou que é mais difícil de gerenciar do que ter uma grande ferramenta. Se você utilizar o mesmo exemplo de dois requisitos diferentes no mesmo conjunto de dados, a solicitação em tempo real poderá ser baixa na CPU, mas alta em E/S, enquanto a solicitação de processamento mais lento poderá apresentar um uso de computação muito intenso. O desacoplamento pode acabar sendo bem menos dispendioso e mais fácil de gerenciar porque você pode construir cada ferramenta para a especificação exata e evitar o provisionamento excessivo. O modelo de serviço de pagamento conforme o uso e apenas pela infraestrutura utilizada da AWS totaliza um valor muito melhor, pois você poderia executar as análises em lote em apenas uma hora e, portanto, teria que pagar apenas pelos recursos de computação dessa hora. Além disso, você pode achar essa abordagem mais fácil de gerenciar em vez de utilizar um sistema único que tenta atender a todos os requisitos. Solucionar requisitos diferentes com uma ferramenta é como tentar encaixar um pino quadrado (como solicitações em tempo real) em um furo redondo (como um grande data warehouse).

A plataforma AWS facilita o desacoplamento de sua arquitetura, permitindo que diferentes ferramentas analisem o mesmo conjunto de dados. Os serviços da AWS têm integração incorporada, o que torna fácil e rápido mover um subconjunto de dados de uma ferramenta para outra utilizando paralelização. Vamos colocar isso em prática explorando alguns cenários de problemas de análise de big data do mundo real e demonstrando detalhadamente uma de arquitetura da AWS.

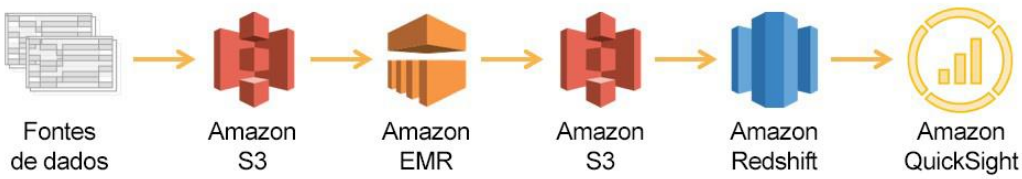
Exemplo 1: Enterprise Data Warehouse

Uma empresa de confecção multinacional tem mais de mil lojas de varejo, vende determinadas linhas através de lojas de departamento e de desconto, além de ter uma presença online. No momento, esses três canais atuam de forma

independente de um ponto de vista técnico. Eles têm gerenciamentos, sistemas de pontos de vendas e departamentos de contabilidade diferentes. Não há um sistema que unifique todos esses conjuntos de dados para permitir que o CEO tenha uma visão completa de todo o negócio. A CEO deseja ter uma visão global de seus canais e ser capaz de fazer análises ad-hoc quando necessário. Os exemplos de análise que as empresas desejam são:

- Quais tendências existem entre os canais?
- Quais regiões geográficas se saem melhor entre os canais?
- Qual a eficiência dos anúncios e cupons da empresa?
- Quais tendências existem entre cada linha de roupa?
- Quais fatores externos podem ter impacto nas vendas, por exemplo, taxa de desemprego ou clima?
- Como os atributos da loja afetam as vendas, por exemplo, estabilidade de funcionários/gerência, centro comercial versus shopping center, local de merchandise na loja, promoção, painéis de exposição no final dos corredores, panfletos de vendas, mostruários internos, etc.?

Um data warehouse corporativo é uma ótima maneira de resolver esse problema. O data warehouse precisa coletar dados de cada um dos vários sistemas dos três canais e de registros públicos sobre o clima e dados econômicos. Cada fonte de dados envia seus dados diariamente para serem utilizados pelo data warehouse. Como cada fonte de dados pode estar estruturada de forma diferente, será realizado um processo de extração, transformação e carregamento (ETL) para reformatar os dados em uma estrutura comum. Em seguida, pode ser feita uma análise dos dados de todas as fontes de forma simultânea. Para tanto, usamos a seguinte arquitetura de fluxo de dados:



Enterprise Data Warehouse

1. A primeira etapa neste processo é obter os dados das várias fontes diferentes para o Amazon S3. O Amazon S3 foi escolhido porque é uma plataforma de armazenamento resiliente, de baixo custo e escalável, na qual os dados podem ser gravados em paralelo a partir de várias fontes diferentes a um custo muito baixo.
2. O Amazon EMR será usado para transformar e depurar os dados do formato de origem para o destino e para um formato. O Amazon EMR tem integração incorporada com o Amazon S3 para permitir threads em paralelo de taxa de transferência de cada nó em seu cluster para, e do Amazon S3. Normalmente, os data warehouses recebem novos dados todas as noites de suas várias fontes diferentes. Como não há necessidade de análise no meio da noite, a única exigência acerca desse processo de transformação é que ele termine até de manhã, quando o CEO e outros usuários do negócio precisam dos resultados. Esse requisito significa que você pode utilizar o Amazon EC2 Spot Market[67] para reduzir ainda mais o custo de transformação. Uma boa estratégia de Spot poderia ser de iniciar as negociações a preços bem baixos à meia-noite, e continuar aumentando seu preço com o tempo até atingir a capacidade. Conforme se aproxima do prazo final, se as ofertas de Spot não foram bem-sucedidas, é possível recuar para os preços sob demanda para garantir que você ainda atenda aos seus requisitos de tempo de conclusão. Cada fonte pode ter um processo de transformação diferente no Amazon EMR, mas com o modelo de pagamento conforme o uso da AWS, você pode criar um cluster independente do Amazon EMR para cada transformação e ajustá-lo exatamente para a capacidade certa a fim de concluir todos os trabalhos de transformação de dados, pelo menor preço possível, sem entrar em conflito com os recursos dos outros trabalhos.
3. Em seguida, cada tarefa de transformação coloca os dados formatados e depurados no Amazon S3. O Amazon S3 é novamente usado aqui porque o Amazon Redshift pode consumir esses dados em vários threads em paralelo a partir de cada nó. Esse local no Amazon S3 também serve como um registro histórico e é a fonte formatada de verdade entre os sistemas. Os dados no Amazon S3 podem ser consumidos por outras ferramentas para análises, caso sejam introduzidos requisitos adicionais com o tempo.
4. O Amazon Redshift carrega, classifica, distribui e compacta os dados em suas tabelas para que as consultas analíticas possam ser executadas de forma eficiente e em paralelo. O Amazon Redshift foi desenvolvido para cargas de trabalho de data warehouse e pode ser facilmente ampliado adicionando-se

outro nó à medida que o tamanho dos dados aumenta com o tempo e os negócios se expandem.

5. Para visualizar as análises, o Amazon QuickSight pode ser usado, ou ainda uma das várias plataformas de visualização de parceiros por meio de conexão OBDC/JDBC do Amazon Redshift. Este é o ponto no qual os relatórios e gráficos podem ser visualizados pelo CEO e sua equipe. Esses dados agora podem ser usados pelos executivos para tomar decisões melhores sobre os recursos da empresa, o que poderia aumentar os ganhos e o valor para os acionistas.

Esta arquitetura é muito flexível e pode facilmente ser ampliada se os negócios forem expandidos, mais fontes de dados forem importadas, novos canais forem abertos ou se um aplicativo móvel for lançado com dados específicos para os clientes. Ferramentas adicionais podem ser integradas a qualquer momento, e o warehouse pode ser redimensionado em alguns cliques aumentando o número de nós no cluster do Amazon Redshift.

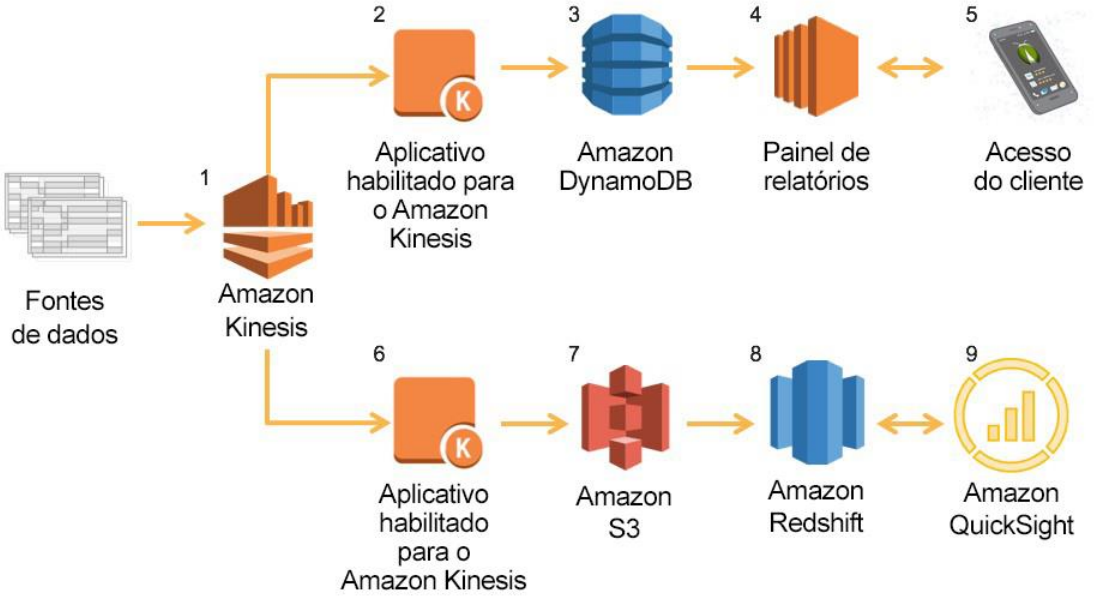
Exemplo 2: Captura e análise de dados do sensor

Uma fabricante internacional de aparelhos de ar condicionado vende muitos aparelhos grandes para várias empresas comerciais e industriais. Eles não vendem somente unidades de ar condicionado, mas, para se posicionarem melhor em relação à concorrência, também oferecem serviços complementares nos quais é possível visualizar painéis em tempo real em um aplicativo móvel ou navegador da web. Cada unidade envia suas informações de sensor para processamento e análise. Esses dados são usados tanto pelo fabricante quanto por seus clientes. Com essa capacidade, o fabricante pode visualizar o conjunto de dados e descobrir tendências.

No momento, a empresa tem alguns milhares de unidades pré-compradas com esse recurso. Ela planeja entregar essas unidades aos clientes nos próximos meses e espera que, em breve, milhares de unidades em todo o mundo estarão usando essa plataforma. Se obtiver êxito, a empresa gostaria de expandir essa oferta também à linha de consumidores com um volume bem maior e uma maior participação no mercado. A solução precisa ser capaz de lidar com grandes volumes de dados e escalar sem interrupções à medida que o negócio cresce. Como esse sistema deve ser projetado? Primeiro, divida-o em dois fluxos de trabalho, ambos com origem nos mesmos dados:

- As informações atuais das unidades de ar condicionado com requisitos quase em tempo real e um grande número de clientes consumindo essas informações.
- Todas as informações históricas sobre as unidades de ar condicionado para execução de operações de tendências e análises para uso interno.

A seguir, a arquitetura de fluxo de dados para resolver esse problema de big data:



Captura e análise de dados do sensor

1. O processo começa com cada unidade de ar condicionado fornecendo um fluxo de dados constante para o Amazon Kinesis Streams. Isso fornece uma interface elástica e resiliente com a qual as unidades podem ser comunicar e que pode ser redimensionada continuamente conforme cada vez mais unidades são vendidas e trazidas online.
2. Com as ferramentas do próprio Amazon Kinesis Streams, como o Kinesis Client Library ou SDK, um simples aplicativo é construído no Amazon EC2 para ler os dados conforme chegam ao Amazon Kinesis Streams e, em seguida, analisá-los e determinar se os dados justificam uma atualização para o painel de tempo real. O aplicativo procura por alterações na operação do sistema, em flutuações de temperatura e em qualquer erro que as unidades encontrarem.

3. Esse fluxo de dados precisa ocorrer quase em tempo real para que os clientes e as equipes de manutenção possam ser alertados o quanto antes, caso haja algum problema com a unidade. Os dados no painel até têm algumas informações de tendências agregadas, mas refletem principalmente o estado atual, além de qualquer erro no sistema. Portanto, os dados necessários para preencher o painel são relativamente pequenos. Além disso, haverá vários acessos potenciais a esses dados a partir das seguintes fontes:
   - Clientes consultando seus sistemas por meio de um dispositivo ou navegador móvel
   - Equipes de manutenção verificando o status de sua frota
   - Algoritmos e análises de dados e de inteligência na plataforma de relatórios reconhecem as tendências que podem ser então enviadas como alertas, por exemplo, quando a pá da hélice de um ar condicionado estiver funcionando de forma incomum por muito tempo e a temperatura do edifício não abaixa.

   O DynamoDB foi escolhido para armazenar esse conjunto de dados quase em tempo real, pois é altamente escalável e disponível; a taxa de transferência desses dados pode ser facilmente escada para cima ou para baixo para atender às necessidades de seus clientes à medida que a plataforma é adotada e a utilização aumenta.
4. O painel de relatórios é um aplicativo da web personalizado que é construído sobre esse conjunto de dados e é executado no Amazon EC2. Ele fornece conteúdo com base no status e nas tendências do sistema, bem como alerta os clientes e as equipes de manutenção sobre quaisquer problemas que possam surgir com a unidade.
5. O cliente acessa os dados através de um dispositivo móvel ou navegador da web para obter o status atual do sistema e visualizar as tendências históricas.

O fluxo de dados (etapas 2 a 5) que acabou de ser descrito é construído para gerar relatórios quase em tempo real com informações para clientes humanos. É construído e projetado para baixa latência e pode ser redimensionar rapidamente para atender à demanda. O fluxo de dados (etapas 6 a 9) que está ilustrado na parte inferior do diagrama não tem esses requisitos tão rigorosos de velocidade e latência. Isso permite ao arquiteto desenhar uma pilha de soluções diferente que pode suportar volumes de dados maiores a um custo bem menor por byte de informações e escolher recursos de computação e armazenamento menos dispendiosos.

6. Para ler o fluxo do Amazon Kinesis, existe um aplicativo separado ativado pelo Amazon Kinesis que provavelmente é executado em uma instância EC2 menor dimensionada em um ritmo mais lento. Embora esse aplicativo analise o mesmo conjunto de dados que o fluxo de dados superior, o objetivo final desses dados é armazenar para registro de longo prazo e hospedar em um data warehouse. Esse conjunto de dados acaba sendo todos os dados enviados dos sistemas e permite que um conjunto muito maior de análises seja realizado sem os requisitos quase em tempo real.
7. Os dados são transformados pelo aplicativo do Amazon Kinesis em um formato adequado para armazenamento em longo prazo, para carregamento em seu data warehouse, e para poder ser armazenado no Amazon S3. Os dados no Amazon S3 não só servem como um ponto de inserção paralelo para o Amazon Redshift, mas também como um armazenamento resiliente, que manterá todos os dados já executados por esse sistema e que poderão ser a única fonte de verdade. Isso pode ser usado para carregar outras ferramentas de análise se surgirem requisitos adicionais. O Amazon S3 também vem com integração nativa com o Amazon Glacier se qualquer dado precisar ser mantido em um armazenamento inativo de longo prazo e de baixo custo.
8. O Amazon Redshift é usado novamente como data warehouse para conjuntos de dados maiores. Ele pode ser facilmente escalado se o conjunto de dados ficar maior, bastando adicionar outro nó no cluster.
9. Para visualizar as análises, uma das várias plataformas de visualização parceiras pode ser usada por meio da conexão OBDC/JDBC com o Amazon Redshift. É aqui que os relatórios, gráficos e análises ad-hoc podem ser gerados no conjunto de dados para encontrar determinadas variáveis e tendências que possam indicar problemas de desempenho ou quebras com as unidades de ar condicionado.

Essa arquitetura pode começar pequena e crescer conforme a necessidade. Além disso, ao desacoplar dois streams de trabalho distintos, eles podem crescer em seu próprio ritmo de acordo com a necessidade, sem compromissos antecipados, permitindo que o fabricante avalie o sucesso ou falha dessa nova oferta sem ter que dispor de um grande investimento. Você poderia facilmente imaginar adições futuras, como o Amazon ML sendo capaz de prever exatamente a duração de uma unidade de ar condicionado para enviar de antemão as equipes de manutenção com base nos algoritmos de previsão a fim de fornecer aos clientes o melhor serviço e experiência

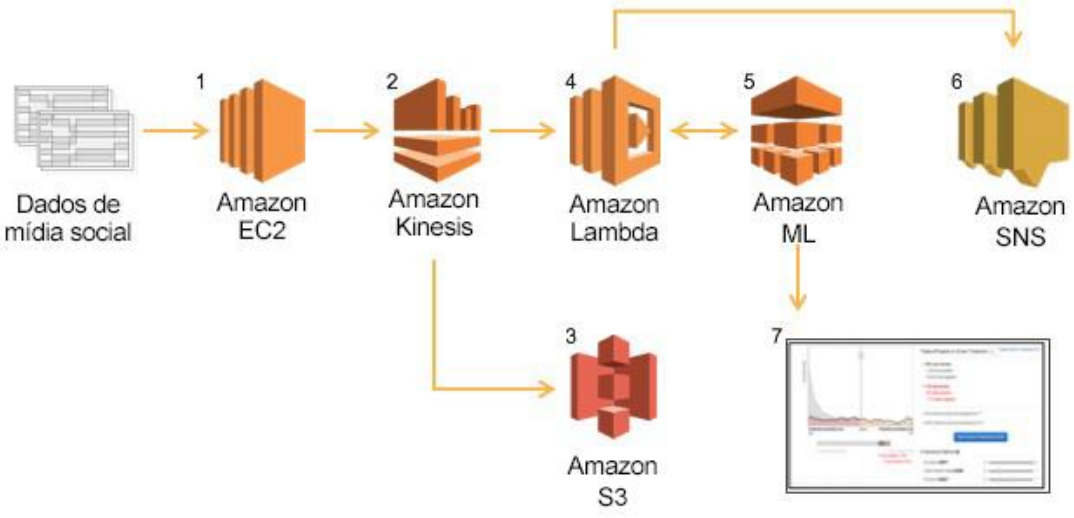
possíveis. Esse nível de serviço seria um diferencial em relação à concorrência e aumentaria as vendas futuras.

Exemplo 3: Análise do sentimento de mídias sociais

Um grande fabricante de brinquedos está crescendo rapidamente e expandindo sua linha de produtos. Depois de cada novo lançamento de brinquedo, a empresa quer entender como os clientes estão desfrutando e usando seus produtos. Além disso, a empresa deseja garantir que seus clientes estejam tendo uma boa experiência com seus produtos. À medida que o ecossistema de brinquedos cresce, a empresa deseja garantir que seus produtos ainda sejam relevantes aos clientes e que possam planejar futuros itens no roadmap com base no feedback dos clientes. A empresa quer capturar os seguintes itens das mídias sociais:

- Entender como os clientes estão usando seus produtos
- Garantir a satisfação do cliente
- Planejar roadmaps futuros

Capturar dados de várias redes sociais é relativamente fácil, mas o desafio é construir a inteligência de forma programática. Depois que os dados são absorvidos, a empresa quer ser capaz de analisá-los e classificá-los de uma forma econômica e programática. Para fazer isso, a arquitetura a seguir pode ser usada:



Análise do sentimento de mídias sociais

1. A primeira coisa a fazer é decidir quais sites de mídia social serão ouvidos. Então, crie um aplicativo que faça pesquisas nesses sites por meio das APIs correspondentes e execute-o no Amazon EC2.
2. Em seguida, um stream do Amazon Kinesis é criado, pois podemos ter várias fontes de dados: Twitter, Tumblr e assim por diante. Dessa forma, um novo stream pode ser criado toda vez que uma nova fonte de dados for adicionada, e você pode usar o código e a arquitetura existentes do aplicativo. Além disso, nesse exemplo, um novo stream do Amazon Kinesis é criado para copiar os dados brutos para o Amazon S3.
3. Para arquivamento, análise de longo prazo e referência histórica, os dados brutos são armazenados no Amazon S3. Modelos adicionais em lote do Amazon ML podem ser executados a partir de dados localizados no Amazon S3 para realizar a análise preditiva e acompanhar as tendências de compra dos clientes.
4. Conforme observado no diagrama da arquitetura, o Lambda é usado para o processamento e normalização dos dados e para a solicitação de previsões do Amazon ML. Depois de gerada a previsão do Amazon ML, a função do Lambda pode realizar uma ação com base nessa previsão, como rotear uma postagem de mídia social para a equipe de serviço de atendimento ao cliente para revisão posterior.
5. O Amazon ML é usado para fazer previsões sobre os dados de entrada. Por exemplo, um modelo do ML pode ser construído para analisar um comentário de mídia social a fim de determinar se o cliente expressou um sentimento negativo sobre um produto. Para obter previsões precisas com o Amazon ML, comece com dados de treinamento e certifique-se de que seus modelos do ML estejam funcionando corretamente. Se estiver criando modelos do ML pela primeira vez, consulte o Tutorial: Usando o Amazon ML para prever respostas para uma oferta de marketing.[68] As mencionado anteriormente, se várias fontes de dados de redes sociais forem usadas, um modelo de ML diferente para cada um é sugerido para garantir a precisão da previsão.
6. Por fim, dados acionáveis são enviados ao Amazon SNS usando o Lambda, os quais são entregue aos recursos apropriados por meio de texto ou e-mail para investigação posterior.
7. Como parte da análise de sentimento, é fundamental a criação de um modelo do Amazon ML atualizado regularmente para a obtenção de resultados precisos. Métricas adicionais sobre um modelo específico podem ser exibidas

graficamente por meio do console, como: precisão, taxa de falsos positivos, precisão e recuperação. Para obter mais informações, consulte a Etapa 4: Revise o desempenho preditivo do modelo do ML e defina um corte.[69]

Ao utilizar uma combinação de Amazon Kinesis Streams, Lambda, Amazon ML e Amazon SES, criamos uma plataforma de ouvidoria social escalável e que pode ser facilmente personalizada. É importante observar que esse cenário não ilustra a ação de criação de um modelo do ML. Isso seria feito pelo menos uma vez, mas em geral é feito regularmente para manter o modelo atualizado. A frequência de criação de um novo modelo depende da carga de trabalho e só é realmente realizada para tornar o modelo mais preciso quando as coisas mudam.

Conclusão

À medida que cada vez mais dados são gerados e coletados, sua análise requer ferramentas escaláveis, flexíveis e de alto desempenho para fornecer insights no momento oportuno. Entretanto, as organizações estão enfrentando um ecossistema de big data em crescimento, no qual novas ferramentas nascem e "morrem" de maneira muito rápida. Portanto, pode ser muito difícil acompanhar e escolher as ferramentas certas.

Este whitepaper oferece um primeiro passo para ajudá-lo a superar esse desafio. Com um amplo conjunto de serviços gerenciados que coletam, processam e analisam big data, a plataforma da AWS facilita a construção, implantação e escalação dos aplicativos de big data, permitindo que você se concentre nos problemas de seu negócio em vez de se preocupar com a atualização e gerenciamento dessas ferramentas.

A AWS fornece muitas soluções para atender aos requisitos de análise de big data. A maioria das soluções de arquitetura de big data usa várias ferramentas da AWS para criar uma solução completa: isso pode ajudar a atender aos rigorosos requisitos de negócios da maneira mais econômica, eficiente e flexível possível. O resultado é uma arquitetura de big data flexível e escalável conforme o crescimento da sua empresa na infraestrutura global da AWS.

Contribuidores

As pessoas e organizações a seguir contribuíram com este documento:

- Erik Swensson, gerente de arquitetura de soluções, Amazon Web Services
- Erick Dame, arquiteto de soluções, Amazon Web Services
- Shree Kenghe, arquiteto de soluções, Amazon Web Services

Outras fontes de leitura

Os recursos a seguir podem ajudá-lo a começar executar análises de big data na AWS:

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

  Veja o portfólio abrangente de serviços de Big Data, bem como links para outros recursos, como parceiros de big data da AWS, tutoriais, artigos e ofertas em soluções de big data do AWS Marketplace. Entre em contato conosco se precisar de ajuda.

- Leia o Blog de big data da AWS[71]

  O blog traz ideias e exemplos da vida real atualizados regularmente para ajudá-lo a coletar, armazenar, depurar, processar e visualizar big data.

- Experimente um dos Test drives de big data[72]

  Explore o rico ecossistema de produtos desenvolvidos para abordar os desafios de big data utilizando a AWS. Os test drives foram desenvolvidos pelos parceiros de consultoria e tecnologia da AWS Partner Network (APN) e são disponibilizados sem custo para fins de educação, demonstração e avaliação.

- Participe de um Curso de treinamento em big data da AWS[73]

  O curso de big data na AWS apresenta soluções de big data em nuvem e o Amazon EMR. Mostramos como utilizar o Amazon EMR para processar

dados utilizando o amplo ecossistema de ferramentas do Hadoop, como o Pig e o Hive. Também abordamos a criação de ambientes de big data, o trabalho com o DynamoDB e o Amazon Redshift, como entender os benefícios do Amazon Kinesis Streams e utilizar as melhores práticas para projetar ambientes de big data seguros e econômicos.

- Consulte os Estudos de casos de clientes de big data[74]

  Aprenda com a experiência de outros clientes que construíram plataformas de big data robustas e eficientes na nuvem AWS.

Revisões do documento

Janeiro de 2016	Revisado para incluir informações sobre o Amazon Machine Learning, AWS Lambda, Amazon Elasticsearch Service; atualização geral
Dezembro de 2014	Primeira publicação

Observações

[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] Milissegundos de um único dígito comuns para tempos de resposta médios do lado do servidor

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

[48] O DynamoDB permite que você altere seu nível de taxa de transferência provisionada em até 100% com uma única chamada de operação de API UpdateTable. Para aumentar a taxa de transferência em mais de 100%, é possível simplesmente chamar UpdateTable novamente.

[49] Você pode aumentar sua taxa de transferência provisionada sempre que desejar, no entanto, há um limite de duas reduções por dia.

[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/