Data warehouse na AWS

Março de 2016

© 2016, Amazon Web Services, Inc. ou suas afiliadas. Todos os direitos reservados.

Avisos

Este documento é fornecido apenas para fins informativos. Ele relaciona as atuais ofertas de produtos e práticas da AWS a contar da 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 "na forma como se encontra", sem garantia de qualquer tipo, expressa ou implícita. Este documento não cria quaisquer garantias, representações, compromissos contratuais, condições ou seguros da AWS, suas afiliadas, fornecedores ou licenciadores. As responsabilidades e as obrigações da AWS com os seus clientes são controladas por contratos da AWS, e este documento não é parte, nem modifica, qualquer contrato entre a AWS e seus clientes.

Sumário

Resumo

Engenheiros de dados, analistas de dados e desenvolvedores que trabalham em empresas de todo o mundo vêm buscando migrar o data warehouse para a nuvem, de forma a aumentar o desempenho e diminuir os custos. Este whitepaper abre a discussão acerca da abordagem moderna à analítica e à arquitetura de data warehouse, apresenta os serviços disponíveis na Amazon Web Services (AWS) para implementar tal arquitetura e mostra padrões de design comuns para criar soluções de data warehouse usando esses serviços.

Introdução

No mundo atual, dados e analítica são indispensáveis aos negócios. Quase todas as grandes empresas construíram data warehouses para fins de relatórios e analítica usando os dados de diversas fontes, inclusive seus próprios sistemas de processamento de transação e outros bancos de dados.

Mas construir e executar um data warehouse – um repositório central de informações vindas de uma ou mais fontes de dados – sempre foi complicado e caro. A maioria dos sistemas de data warehouse é de montagem complexa, custa milhões de dólares em despesas imediatas com software e hardware e pode levar meses nos processos de planejamento, aquisição, implementação e aplicação. Depois de fazer os investimentos iniciais e configurar o data warehouse, é preciso contratar uma equipe de administradores de banco de dados para que suas consultas continuem a ser rápidas e para proteger-se contra perda de dados.

A escalabilidade dos data warehouses tradicionais também é difícil. Quando os volumes de dados aumentarem ou você quiser disponibilizar analítica e relatórios a mais usuários, é preciso escolher entre aceitar lentidão nas consultas ou investimento de tempo e esforço em um caro processo de upgrade. Na verdade, algumas equipes de TI não incentivam aumentar os dados nem adicionar consultas, de forma a proteger os acordos de nível de serviço existentes. Várias empresas batalham para manter um relacionamento saudável com os fornecedores tradicionais de bancos de dados. Elas muitas vezes se veem forçadas a fazer o upgrade de hardware para um sistema gerenciado ou entrar em um ciclo de negociação prolongado por uma licença com prazo expirado. Ao alcançarem o limite de escalabilidade em um mecanismo do data warehouse, elas são forçadas a migrar para outro mecanismo do mesmo fornecedor com semântica de SQL diferente.

O Amazon Redshift mudou a forma como as empresas pensam em data warehouse ao diminuir drasticamente o custo e o esforço associados à implementação de sistemas de data warehouse sem comprometer recursos nem desempenho. O Amazon Redshift é uma solução de data warehouse rápida e totalmente gerenciada que deixa mais simples e acessível a análise de grandes volumes de dados usando as ferramentas existentes de Business Intelligence (BI). Com o Amazon Redshift, você pode ter o desempenho de mecanismos colunares de data warehouse que executam a arquitetura de processamento Massively Parallel Processing (MPP) a um décimo do custo. Você pode começar aos poucos, por US$ 0,25 por hora e sem compromisso e escalar até petabytes, a um custo de US$ 1.000 por terabyte por ano.

Desde o lançamento, em fevereiro de 2013, o Amazon Redshift é um dos serviços da AWS que mais cresce, com milhares de clientes de todos os setores e portes. Empresas como NTT DOCOMO, FINRA, Johnson & Johnson, Hearst, Amgen e NASDAQ já migraram para o Amazon Redshift. Por consequência, o Amazon Redshift foi classificado como líder no relatório *Forrester Wave: Enterprise Data Warehouse, Q4 2015*.[1]

Neste whitepaper, nós lhe apresentamos as informações de que você precisa para aproveitar a mudança estratégica que acontece no espaço de data warehouse – de on-premises para nuvem:

- Arquitetura de analítica moderna
- Opções de tecnologia de data warehouse disponíveis dentro dessa arquitetura
- Mergulho profundo no Amazon Redshift e em suas características únicas
- Um guia para construir um sistema completo de data warehouse na AWS com o Amazon Redshift e outros serviços
- Dicas práticas para migrar de outras soluções de data warehouse e aproveitar nosso ecossistema de parceiros

Arquitetura moderna de analítica e data warehouse

Mais uma vez, o *data warehouse* é um repositório central de informações vindas de uma ou mais fontes de dados. Os dados costumam fluir para dentro de um data warehouse a partir de sistemas transacionais e outros bancos de dados relacionais, tipicamente incluindo dados estruturados, semiestruturados e desestruturados. Os dados são processados, transformados e ingeridos em cadência regular. Os usuários, entre eles cientistas de dados, analistas de negócios e tomadores de decisão, acessam os dados por ferramentas de BI, clientes SQL e planilhas.

Por que se dar o trabalho de construir um data warehouse? Não seria melhor simplesmente rodar as consultas de analítica em um banco de dados OLTP (Online Transaction Processing), onde as transações são registradas? Para responder a questão, vamos analisar as diferenças entre data warehouse e bancos de dados OLTP. Os data warehouses são otimizados para as operações de gravação em lote e leitura de grandes volumes de dados, enquanto os bancos de dados OLTP são otimizados para operações de gravação contínua e elevados volumes de operações pequenas de leitura. No geral, os data warehouses empregam esquemas desnormalizados, como os esquemas estrela e floco de neve, por conta dos elevados requisitos de transferência de dados, enquanto os bancos de dados OLTP empregam esquemas altamente normalizados, mais adequados para requisitos de altas transferências de transação. O esquema estrela é formado por algumas grandes tabelas de fatos que fazem referência a diversas tabelas de dimensão. O esquema floco de neve, uma extensão do esquema estrela, é formado por tabelas de dimensão ainda mais normalizadas.

Para obter os benefícios do uso de um data warehouse gerenciado como datastore independente com seu sistema de OLTP de origem ou outro, recomendamos que você crie um pipeline eficiente de dados. Esse pipeline extrai os dados do sistema de origem, os converte em um esquema adequado para data warehouse e os carrega no data warehouse. Na próxima seção, debateremos os blocos de construção de um pipeline de analítica e os diferentes serviços da AWS que você pode usar para arquitetar ao pipeline.

Arquitetura de analítica

Os pipelines de analítica são concebidos de forma a suportar grandes volumes de fluxos de dados de entrada de fontes heterogêneas como bancos de dados, aplicações e dispositivos.

Um pipeline de analítica típico tem os seguintes estágios:

1. Coletar dados.
2. Armazenar os dados.
3. Processar os dados.
4. Analisar e visualizar os dados.

Veja a ilustração na Figura 1 a seguir.

Figura 1: Pipeline de analítica

Coleta de dados

No estágio de coleta de dados, leve em consideração que você provavelmente tem tipos diferentes de dados, como dados transacionais, dados de log, dados de streaming e dados de Internet das Coisas (IoT). A AWS fornece soluções para armazenamento de cada um desses tipos de dados.

Dados transacionais

Os dados transacionais, como de transações de compras por comércio eletrônico e transações financeiras, são tipicamente armazenados em sistemas de gerenciamento de banco de dados relacional (RDBMS, Relational Database Management Systems) ou sistemas de bancos de dados NoSQL. A escolha da solução de banco de dados depende do caso de uso e das características da aplicação. O banco de dados NoSQL é adequado quando os dados não estiverem bem estruturados para se encaixarem em um esquema definido ou quando o esquema mudar com muita frequência. A solução RDBMS, por outro lado, é adequada quando as transações acontecerem em várias linhas de tabela e as consultas exigirem uniões complexas. O Amazon DynamoDB é um serviço de banco de dados NoSQL totalmente gerenciado que pode ser usado como armazenamento OLTP para seus aplicativos. O Amazon RDS lhe permite implementar uma solução de banco de dados relacional baseada em SQL para seu aplicativo.

Dados do log

A captura confiável de logs gerados pelo sistema o ajudará a resolver problemas, conduzir auditorias e executar analítica usando as informações armazenadas nos logs. O Amazon Simple Storage Service (Amazon S3) é uma conhecida solução de armazenamento para dados não transacionais, como dados de log, que são usados para analítica. Por fornecer 11 "9s" de durabilidade (ou seja, 99,999999999% de durabilidade), o Amazon S3 também é uma popular solução de arquivamento.

Dados de streaming

Aplicações web, dispositivos móveis e vários aplicativos e serviços de software podem gerar uma quantidade inacreditável de dados de streaming – chegando por vezes a terabytes por hora – que precisam ser coletados, armazenados e processados continuamente.[2] Usando os serviços do Amazon Kinesis, você pode fazer isso com simplicidade e a um baixo custo.

Dados de IoT

Dispositivos e sensores em todo o mundo enviam mensagens continuamente. As empresas veem uma necessidade cada vez maior hoje em dia de capturar esses dados e obter BI deles. Usando o AWS IoT, os dispositivos conectados interagem com facilidade e segurança com a nuvem AWS. O AWS IoT facilita o uso de serviços da AWS como AWS Lambda, Amazon Kinesis, Amazon S3, Amazon Machine Learning e Amazon DynamoDB para criar aplicações que coletam, processam, analisam e agem com base nos dados de IoT, sem ter que gerenciar nenhuma infraestrutura.

Processamento de dados

O processo de coleta fornece dados que potencialmente têm informações úteis. Você pode analisar as informações extraídas para conferir BI que ajudará seus negócios a prosperarem. Essa BI pode, por exemplo, lhe mostrar o comportamento do usuário e a relativa popularidade dos produtos. As melhores práticas para coletar esse tipo de BI consistem em carregar seus dados brutos em um data warehouse para executar uma análise mais profunda.

Para tal, existem dois tipos de fluxos de processamento: em lote e em tempo real. As formas mais comuns de processamento (Online Analytic Processing [OLAP] e Online Transaction Processing [OLTP]) usam esses tipos. O processamento Online Analytic Processing (OLAP) no geral é baseado em lotes. Em contraste, os sistemas OLTP são orientados ao processamento em tempo real e não costumam ser bem adequados para processamento baseado em lotes. Se você separar processamento de dados do sistema OLTP, evita que o processamento de dados afete a carga de trabalho de OLTP.

Primeiro vamos analisar o que está envolvido no processamento em lotes.

Extract Transform Load (ETL)

ETL é o processo de extrair dados de várias fontes para carregá-lo em sistemas de data warehouse. O ETL normalmente é um processo contínuo e permanente com um fluxo de trabalho bem-definido. Durante esse processo, os dados são inicialmente extraídos de uma ou mais fontes. Os dados extraídos são limpos, aprimorados, transformados e carregados em um data warehouse. Ferramentas de estrutura Hadoop, como Apache Pig e Apache Hive, são comumente usadas em um pipeline ETL para executar transformações em grandes volumes de dados.

Extract Load Transform (ELT)

O ELT é uma variante do ETL na qual os dados extraídos são carregados primeiro no sistema de destino. As transformações são executadas depois que os dados são carregados no data warehouse. O ELT costuma funcionar bem quando o sistema de destino é poderoso o suficiente para lidar com transformações. O Amazon Redshift costuma ser usado nos pipelines ELT por ser altamente eficiente em executar transformações.

Online Analytical Processing (OLAP)

Os sistemas OLAP armazenam dados históricos agregados em esquemas multidimensionais. Amplamente usados em data mining, os sistemas OLAP lhe permitem extrair dados e identificar tendências em várias dimensões. Por ser otimizado para uniões rápidas, o Amazon Redshift é muito usado para criar sistemas OLAP.

Agora vamos analisar o que está envolvido no processamento de dados em tempo real.

Processamento em tempo real

Nós já falamos sobre streaming de dados antes e mencionamos o Amazon Kinesis como solução para capturar e armazenar dados de streaming. Você pode processar esses dados de forma sequencial e incremental, registro a registro ou sobre janelas de tempo deslizantes, e usar os dados processados para uma grande variedade de possibilidades analíticas, incluindo correlações, agregações, filtragem e amostragem. Esse tipo de processamento é chamado processamento em tempo real. Informações derivadas de processamento em tempo real oferecem às empresas visibilidade em vários aspectos de negócios e atividade de clientes – como uso do serviço (para medição ou faturamento), atividade do servidor, cliques no site e geolocalização de dispositivos, pessoas e produtos físicos – e lhes permite responder imediatamente às situações que surgirem. O processamento em tempo real exige uma camada de processamento altamente concomitante e escalável.

Para processar dados de streaming em tempo real, você pode usar o AWS Lambda. O Lambda pode processar os dados diretamente pelo AWS IoT ou pelo Amazon Kinesis Streams. O Lambda lhe permite executar código sem provisionar ou gerenciar servidores.

O Amazon Kinesis Client Library (KCL) é outra forma de processar dados do Amazon Kinesis Streams. O KCL lhe oferece mais flexibilidade que o AWS Lambda para acomodar os dados de entrada em lote e processá-los melhor. Você também pode usar o KCL para aplicar extensas transformações e personalizações à sua lógica de processamento.

O Amazon Kinesis Firehose é a forma mais fácil de carregar dados de streaming no AWS. Ele pode capturar os dados de streaming e carregá-los automaticamente no Amazon Redshift, permitindo analítica quase em tempo real com os painéis e as ferramentas de BI existentes que você já usa atualmente. É possível definir suas regras de lote com o Firehose, pois depois ele cuida de agrupar com confiança os dados e entregá-lo ao Amazon Redshift.

Armazenamento de dados

Você pode armazenar seus dados em um data warehouse ou em um data mart, como apresentado a seguir.

Data warehouse

Como já dissemos, o *data warehouse* é um repositório central de informações vindas de uma ou mais fontes de dados. Com o uso de data warehouse, você pode executar analítica rápida em grandes volumes de dados e revelar padrões ocultos nos dados ao aproveitar ferramentas de BI. Os cientistas de dados consultam um data warehouse para executar analítica offline e identificar tendências. Usuários de toda a organização consomem os dados usando consultas SQL ad hoc, relatórios periódicos e painéis para tomar decisões de negócios críticas.

Data mart

O *data mart* é uma forma simples de data warehouse focada em um assunto ou em uma área funcional específica. Por exemplo: você pode ter data marts específicos para cada divisão da sua organização ou data marts segmentados para cada região. Você pode construir data marts com base em um grande data warehouse, armazenamentos operacionais ou um híbrido dos dois. Os data marts são simples de projetar, construir e administrar. No entanto, como os data marts são concentrados em áreas funcionais específicas, fazer consultar através de áreas funcionais pode ser uma tarefa complexa por conta da distribuição.

Você pode usar o Amazon Redshift para construir data marts, além de data warehouses.

Análise e visualização

Depois de processar os dados e disponibilizá-los para posterior análise, é preciso ter as ferramentas certas para analisar e visualizar os dados processados.

Em vários casos, você pode executar análise de dados com as mesmas ferramentas usadas para processar dados. Você pode usar ferramentas como o SQL Workbench para analisar os dados no Amazon Redshift com ANSI SQL. O Amazon Redshift também funciona bem com soluções famosas de BI de terceiros que estão disponíveis no mercado.

O Amazon QuickSight é um serviço de 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 Amazon QuickSight é integrado ao Amazon Redshift e atualmente está em pré-estreia, com disponibilidade geral planejada ainda para 2016.

Se você estiver usando o Amazon S3 como armazenamento principal, uma forma popular de fazer análise e visualização é executar notebooks Apache Spark no Amazon Elastic MapReduce (Amazon EMR). Usando esse processo, você tem a flexibilidade de executar SQL ou código personalizado escrito em linguagens como Python e Scala.

Para outra abordagem de visualização, o Apache Zeppelin é uma solução de BI de código aberto que você pode executar no Amazon EMR para visualizar dados no Amazon S3 usando Spark SQL. Você também pode usar o Apache Zeppelin para visualizar dados no Amazon Redshift.

Pipeline de analítica com os serviços da AWS

A AWS oferece uma ampla variedade de serviços para implementar uma plataforma completa de analítica. A Figura 2 mostra os serviços debatidos anteriormente e onde eles se encaixam dentro do pipeline de analítica.

Figura 2: Pipeline de analítica com os serviços da AWS

Opções de tecnologia de data warehouse

Nesta seção, nós debatemos as opções disponíveis para construir um data warehouse: bancos de dados orientados a linhas, bancos de dados orientados a colunas e arquiteturas de Massively Parallel Processing.

Bancos de dados orientados a linhas

Os bancos de dados orientados a linhas geralmente armazenam linhas inteiras em um bloco físico. O alto desempenho das operações de leitura é conquistado por meio de índices secundários. Bancos de dados como Oracle Database Server, Microsoft SQL Server, MySQL e PostgreSQL são sistemas de bancos de dados orientados a linhas. Esses sistemas vêm sendo tradicionalmente usados para data warehouse, mas são mais adequados para processamento de transações (OLTP) que para analítica.

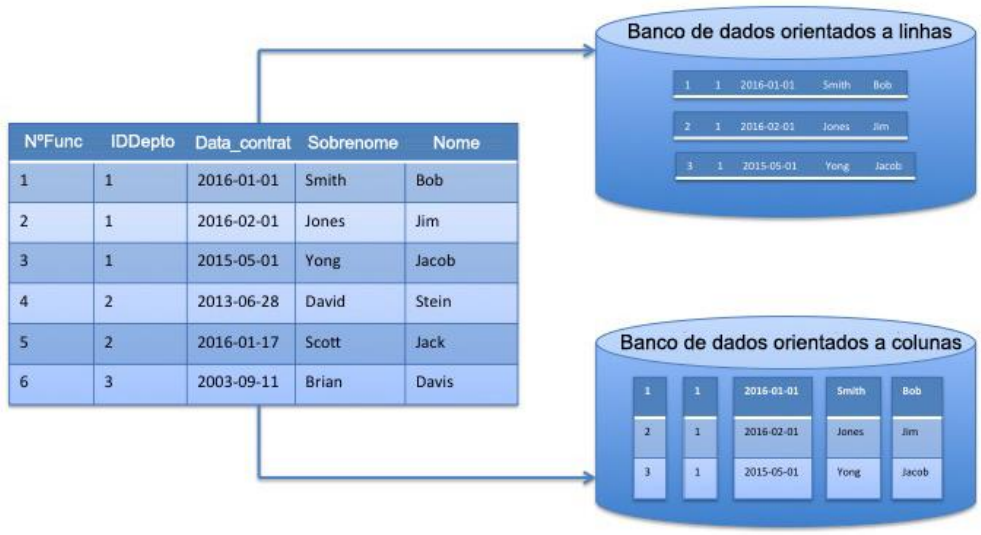
Para otimizar o desempenho de um sistema baseado em linhas usado como data warehouse, os desenvolvedores usam diversas técnicas, inclusive construir visões materializadas, criar tabelas de sumarização pré-agregadas, construir índices em todas as combinações possíveis de predicados, implementar particionamento de dados para aproveitar o corte da partição por otimizador de consulta e executar uniões baseadas no índice.

Armazenamentos de dados tradicionais baseados em linhas são limitados pelos recursos disponíveis em uma única máquina. Os data marts aliviam o problema até certo ponto, usando fragmentação funcional. Você pode separar o data warehouse em vários data marts, cada um deles atendendo a uma área funcional específica. Mas quando os data marts crescem com o tempo, o processamento de dados fica mais lento.

Em um armazenamento de dados baseado em linhas, todas as consultas precisam ler todas as colunas de todas as linhas dos blocos que atendam ao predicado da consulta, inclusive as colunas que você não selecionou. Essa abordagem cria um gargalo significativo no desempenho dos data warehouses, nos quais suas tabelas têm mais colunas, mas suas consultas usam somente algumas.

Bancos de dados orientados a colunas

Bancos de dados orientados a colunas organizam cada uma delas em seu conjunto de blocos físicos, em vez de colocar todas as linhas em um bloco. Essa funcionalidade lhes permite ter mais eficiência em E/S para consultas de somente leitura, pois só é preciso ler as colunas acessadas por uma consulta no disco (ou na memória). Essa abordagem faz de bancos de dados orientados a colunas uma escolha melhor que bancos de dados orientados a linhas para data warehouse.



NºFunc	IDDepto	Data_contrat	Sobrenome	Nome
1	1	2016-01-01	Smith	Bob
2	1	2016-02-01	Jones	Jim
3	1	2015-05-01	Yong	Jacob
4	2	2013-06-28	David	Stein
5	2	2016-01-17	Scott	Jack
6	3	2003-09-11	Brian	Davis

Figura 3: Bancos de dados orientados a linhas vs. orientados a colunas

A Figura 3, anterior, ilustra a principal diferença entre bancos de dados orientados a linhas e orientados a colunas. As linhas são embaladas com seus próprios blocos em um banco de dados orientado a linhas, e as colunas são embaladas em seus próprios blocos em um banco de dados orientados a colunas.

Após E/S mais rápidas, o próximo maior benefício do uso de um banco de dados orientado a colunas é a melhora da compactação. Como todas as colunas são embaladas em um conjunto próprio de blocos, todo bloco físico contém o mesmo tipo de dados. Quando todos os dados são do mesmo tipo, o banco de dados pode usar algoritmos de compactação extremamente eficientes. Como consequência, você precisa de menos armazenamento em comparação com um banco de dados orientado a linha. Essa abordagem também resulta em E/S significativamente menor, pois os mesmos dados são armazenados em menos blocos.

Alguns bancos de dados orientados a colunas usados para data warehouse incluem Amazon Redshift, Vertica, Teradata Aster e Druid.

Arquiteturas Massively Parallel Processing

Uma arquitetura MPP lhe permite usar todos os recursos disponíveis no cluster para processamento de dados, aumentando drasticamente o desempenho de data warehouses de escala de petabytes. Os data warehouses com arquitetura MPP lhe permitem melhorar o desempenho ao simplesmente adicionar mais nós ao cluster. Amazon Redshift, Druid, Vertica, GreenPlum e Teradata Aster são alguns dos data warehouses construídos sobre uma arquitetura MPP. Estruturas de código aberto, como Hadoop e Spark, também são compatíveis com MPP.

Mergulho profundo com Amazon Redshift

Como conta com a tecnologia MPP colunar, o Amazon Redshift oferece os principais benefícios de um data warehouse com bom desempenho e boa relação custo/benefício, E/S reduzida e menos requisitos de armazenamento. Ele se baseia em ANSI SQL, para que você possa executar consultas existentes com pouca ou nenhuma modificação. Como consequência, ele se tornou uma opção popular para data warehouses e data marts corporativos atualmente. Nesta seção, nós mergulhamos mais fundo no Amazon Redshift e falamos mais sobre seus recursos.

O Amazon Redshift proporciona consulta rápida e desempenho de E/S para conjuntos de dados de praticamente qualquer tamanho de dados por usar armazenamento colunar, paralelização e distribuição de consultas entre vários nós. Ele automatiza a maioria das tarefas administrativas comuns ligadas a provisionamento, configuração, monitoramento, backup e segurança de warehouse de dados, o que torna o gerenciamento fácil e barato. Usando essa automação, você pode construir data warehouse de escala petabytes em minutos, em vez de semanas ou meses levados por implementações tradicionais on-premises.

Desempenho

O Amazon Redshift usa armazenamento colunar, compactação de dados e mapas de zona para reduzir a quantidade de E/S necessária para realizar consultas. A classificação intercalada permite desempenho rápido sem as despesas extraordinárias de manutenção de índices ou projeções.

O Amazon Redshift emprega uma arquitetura MPP para aproveitar todos os recursos disponíveis ao paralelizar e distribuir operações SQL. 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 opção de SSD (solid-state drive) e DS (Dense Storage). A implementação contínua de upgrades de software oferece melhorias contínuas no desempenho sem nenhuma intervenção do usuário.

Durabilidade e disponibilidade

Para oferecer a maior durabilidade e disponibilidade de dados possível, o Amazon Redshift detecta e substitui automaticamente qualquer nó falho no cluster do data warehouse. Ele disponibiliza imediatamente seu nó de substituição e carrega os dados acessados com mais frequência primeiro, de forma que você consiga retomar a consulta dos seus dados o mais rápido possível. Como o Amazon Redshift espelha os dados através do cluster, ele os usa de outro nó para reconstruir o nó falho. O cluster estará em modo somente leitura até que um nó de substituição seja provisionado e adicionado ao cluster, o que costuma levar somente alguns minutos.

Os clusters do Amazon Redshift ficam dentro de uma zona de disponibilidade.[3] Se você quiser ter uma configuração multi-AZ para o Amazon Redshift, pode criar um espelho e fazer o autogerenciamento da replicação e do failover.

Com apenas alguns cliques no Amazon Redshift Management Console, você pode configurar um ambiente robusto de recuperação de desastres (DR) com o Amazon Redshift. Você pode manter cópias dos seus backups em várias regiões da AWS. No caso de uma interrupção de serviço em uma região da AWS, você pode restaurar o cluster pelo backup em uma região da AWS diferente. Você pode ganhar acesso de leitura/gravação ao seu cluster em poucos minutos após iniciar a operação de restauro.

Escalabilidade e elasticidade

Com alguns cliques no console ou com uma simples chamada de API, você pode alterar facilmente o número e o tipo de nós no data warehouse de acordo com suas necessidades de mudanças no desempenho ou na capacidade.[4] 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 Sobre clusters e nós no *Guia de gerenciamento de clusters do Amazon Redshift Cluster*.[5]

Ao redimensionar, o Amazon Redshift coloca o cluster existente no modo somente leitura, provisiona um novo cluster do tamanho de sua preferência e copia os dados do cluster anterior 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.

Você pode usar as ações de API do Amazon Redshift para iniciar programativamente clusters, escalar clusters, criar backups, restaurar backups e muito mais. Usando essa abordagem, você pode integrar as ações de API à pilha de automação existente ou construir uma automação personalizada adequada às suas necessidades.

Interfaces

O Amazon Redshift tem drivers personalizados de Java Database Connectivity (JDBC) e Open Database Connectivity (ODBC) que você pode baixar pela guia **Connect Client** do console, o que significa que pode usar 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 Amazon Redshift, veja Amazon Redshift e PostgreSQL no *Guia do desenvolvedor de banco de dados Amazon Redshift*. [6]

Você também pode encontrar vários exemplos de integrações validadas com muitos fornecedores conhecidos de BI e ETL.[7] Nessas integrações, carregamento e descarregamento são executados paralelamente em cada nó computacional, de forma a maximizar a velocidade na qual você pode ingerir ou exportar dados de e para vários recursos, como Amazon S3, Amazon EMR e Amazon DynamoDB. Você pode facilmente carregar dados de streaming ao Amazon Redshift usando Amazon Kinesis Firehose, permitindo analítica em tempo quase real com ferramentas e painéis existentes de BI. Você pode localizar métrica para utilização computacional, de memória, de armazenamento e tráfego de leitura/gravação para seu cluster de data warehouse Amazon Redshift usando o console ou as operações de API do Amazon CloudWatch.

Segurança

Para ajudar a oferecer segurança de dados, você pode executar o Amazon Redshift dentro de uma nuvem privada virtual baseada no serviço Amazon Virtual Private Cloud (Amazon VPC). Você pode usar o modelo de rede definida por software do VPC para definir as regras do firewall que restringem o tráfego com base nas regras que você configura.[8] O Amazon Redshift tem suporte para conexões habilitadas por SSL entre o aplicativo cliente e seu cluster de warehouse de dados do Amazon Redshift, que permite que os dados sejam criptografados em trânsito.

Os nós computacionais do Amazon Redshift armazenam seus dados, mas os dados só podem ser acessados pelo nó líder do cluster. Esse isolamento oferece outra camada de segurança. O Amazon Redshift se integra ao AWS CloudTrail para permitir auditoria de todas as chamadas de API do Amazon Redshift.[9] Para ajudar a manter seus dados seguros em repouso, o Amazon Redshift criptografa cada bloco usando a criptografia AES-256 acelerada por hardware à medida que cada bloco é gravado no disco. Essa criptografia ocorre a um nível baixo no subsistema de E/S; o subsistema de E/S criptografa tudo o que está gravado no disco, incluindo resultados de consultas intermediárias. Os blocos são copiados da mesma forma, o que significa que os backups também são criptografados. Por padrão, o Amazon Redshift cuida do gerenciamento de chaves, mas você pode escolher gerenciar suas chaves usando seus próprios módulos de segurança de hardware (HSMs, hardware security modules) ou gerenciar as chaves com o AWS Key Management Service.[10],[11]

Modelo de custo

O Amazon Redshift não exige comprometimento de longo prazo nem custos imediatos. Com essa abordagem de definição de preço, 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 dois 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 acima do armazenamento provisionado e backups armazenados após seu cluster estar concluído são cobrados de acordo com as taxas padrão do Amazon S3.[12] Não existe cobrança de transferência de dados para comunicações entre o Amazon S3 e o Amazon Redshift. Para obter mais informações, consulte a definição de preços do Amazon Redshift.[13]

Padrões de uso ideal

O Amazon Redshift é ideal para OLAP (Online Analytical Processing) usando suas ferramentas existentes de BI. As organizações estão usando o Amazon Redshift para:

- Executar BI e relatórios corporativos
- 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

Antipadrões

O Amazon Redshift não é ideal para os seguintes padrões de uso:

- **Conjuntos de dados pequenos** – O Amazon Redshift foi desenvolvido para processamento paralelo em um cluster. Se seu 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.
- **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 Amazon DynamoDB.

- **Dados desestruturados** – Os dados no Amazon Redshift devem ser estruturados por um esquema definido. O Amazon Redshift não é compatível com uma estrutura de esquema arbitrária para cada linha. Se seus dados não são estruturados, você pode realizar processos de ETL (extração, transformação e carregamento) no Amazon EMR para preparar os dados para o carregamento no Amazon Redshift. Para dados JSON, você pode armazenar pares de valores-chave e usar as funções JSON nativas nas suas consultas.[14]
- **Dados BLOB** – Se você planeja armazenar arquivos BLOB (Binary Large Object), como vídeos digitais, imagens ou músicas, convém 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 si são armazenados no Amazon S3.

Migração para o Amazon Redshift

Se você decidir migrar de um data warehouse existente para o Amazon Redshift, a estratégia de migração que você escolher dependerá de vários fatores:

- O tamanho do banco de dados e suas tabelas
- Largura de banda da rede entre o servidor de origem e a AWS
- Se a migração e o switchover para a AWS serão feitos em uma única etapa ou em uma sequência de etapas ao longo do tempo
- A velocidade de alteração dos dados no sistema de origem
- Transformações durante a migração
- A ferramenta do parceiro que você planeja usar para migração e ETL

Migração em uma etapa

A migração em uma etapa é uma boa opção para bancos de dados pequenos que não exigem operação contínua. Os clientes podem extrair bancos de dados existentes como arquivos CSV (valores separados por vírgulas) e, em seguida, usar serviços como o AWS Import/Export Snowball para entregar conjuntos de dados ao Amazon S3 a serem carregados no Amazon Redshift. Os clientes então testam o banco de dados do Amazon Redshift de destino quanto à consistência de dados com a origem. Quando todas as validações tiverem recebido aprovação, o banco de dados será alterado para AWS.

Migração em duas etapas

A migração em duas etapas costuma ser usada para bancos de dados de qualquer tamanho:

1. **Migração de dados inicial:** Os dados são extraídos do banco de dados de origem, preferivelmente durante o uso fora do pico, para minimizar o impacto. Os dados, então, são migrados para o Amazon Redshift ao seguir a abordagem de migração de uma etapa descrita anteriormente.
2. **Migração de dados alterados:** Dados que mudaram no banco de dados de origem após a migração de dados inicial são propagados até o destino antes do switchover. Esta etapa sincroniza os bancos de dados de origem e de destino. Quando todos os dados alterados tiverem sido migrados, você poderá validar os dados no banco de dados de destino, fazer os testes necessários e, se todos os testes tiverem sido aprovados, alternar para o data warehouse do Amazon Redshift.

Ferramentas para migração do banco de dados

Existem várias ferramentas e tecnologias para migração de dados. Você pode usar algumas dessas ferramentas alternadamente ou usar outras ferramentas terceirizadas ou de código aberto disponíveis no mercado.

1. O Serviço de Migração de Banco de Dados da AWS é compatível com os processos de migração de uma e duas etapas descritos anteriormente.[15] Para acompanhar o processo de migração de duas etapas, você habilita o log suplementar para capturar alterações no sistema de origem. Você pode habilitar o log suplementar em nível de tabela ou banco de dados.

2. As ferramentas adicionais de parceiros de integração de dados são as seguintes:

   - Attunity
   - Informatica
   - SnapLogic
   - Talend
   - Bryte

Para obter mais informações sobre integração de dados e parceiros de consultoria, consulte Amazon Redshift Partners.[16]

Projetando fluxos de trabalho de data warehouse

Nas seções anteriores, falamos sobre os recursos do Amazon Redshift que fazem dele a escolha ideal para data warehouse. Para entender como projetar fluxos de trabalho de data warehouse com o Amazon Redshift, vamos analisar o padrão de design mais comum junto com um caso de uso de exemplo.

Vamos supor que uma empresa de confecção multinacional tem mais de mil lojas de varejo, vende determinadas linhas de roupas em lojas de departamento e outlets, e tem presença online. Do ponto de vista técnico, esses três canais atualmente têm operação independente. Eles têm gerenciamentos, sistemas de pontos de vendas e departamentos de contabilidade diferentes. Nenhum sistema único mescla todos os conjuntos de dados relacionados para fornecer à CEO uma visão de 360 graus do negócio inteiro.

Vamos supor também que a CEO quer ter uma visão global desses canais e conseguir fazer consultas analíticas ad hoc como as seguintes:

- Quais tendências existem entre os canais?
- Quais regiões geográficas se saem melhor entre os canais?
- Qual a eficiência dos anúncios e promoções da empresa?
- Quais tendências existem entre cada linha de roupa?
- Quais forças externas exercem impactos sobre as vendas da empresa – por exemplo, a taxa de desemprego e as condições climáticas?

- Como os atributos da loja afetam as vendas – por exemplo, a permanência dos funcionários/gerência, centro comercial versus shopping center, local do merchandise na loja, promoção, painéis de exposição no final dos corredores, panfletos de vendas e mostruários internos?

Um data warehouse corporativo resolve esse problema. Ele coleta os dados de cada um dos vários sistemas dos três canais e também de dados disponíveis ao público, como condições climáticas e relatórios econômicos. Cada fonte de dados envia dados diariamente para consumo do 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 isso, usaremos a seguinte arquitetura de fluxo de dados:

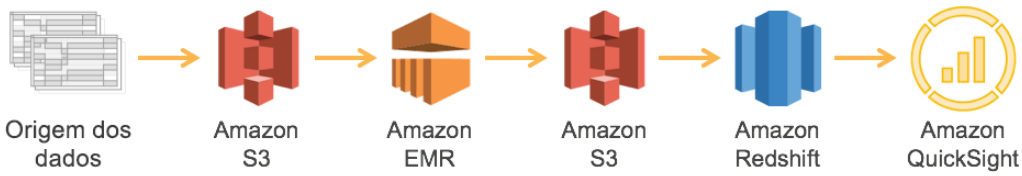


Figura 4: Fluxo de trabalho do data warehouse corporativo

1. A primeira etapa neste processo é obter os dados das várias fontes diferentes para o Amazon S3. O Amazon S3 oferece 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 formato de destino. O Amazon EMR tem integração incorporada com o Amazon S3, que permite threads em paralelo de taxa de transferência de cada nó em seu cluster do Amazon EMR de e para o Amazon S3.

Normalmente o data warehouse recebe dados novos todas as noites. Como não há necessidade de analítica no meio da noite, a única exigência acerca desse processo de transformação é que ele termine até de manhã, quando a CEO e outros usuários precisão acessar os relatórios e painéis. Assim, você pode usar o mercado Spot do Amazon EC2 para diminuir ainda mais o custo do ETL neste caso.[17] Uma boa estratégia de Spot é 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 origem pode ter um processo de transformação diferente no Amazon EMR, mas com o modelo de "pague conforme usa" da AWS, você pode criar um cluster independente do Amazon EMR para cada transformação e ajustá-lo exatamente para a capacidade certa para concluir todos os trabalhos de transformação de dados sem enfrentar os recursos dos outros trabalhos.

3. Cada job de transformação carrega dados formatados e limpos no Amazon S3. Usamos o Amazon S3 aqui novamente, pois o Amazon Redshift pode carregar os dados paralelamente do Amazon S3, usando vários threads de cada nó de cluster. O Amazon S3 também apresenta um registro histórico e serve de fonte formatada da 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. À medida que o tamanho dos dados aumenta com o tempo e os negócios se expandem, você pode facilmente aumentar a capacidade ao adicionar mais nós.
5. Para visualizar a analítica, você pode usar o Amazon QuickSight ou uma das várias plataformas de visualização de parceiros que se conectam ao Amazon Redshift usando ODBC ou JDBC. É nesse ponto que a CEO e a equipe veem relatórios, painéis e gráficos. Os executivos podem usar os dados para tomar decisões melhores sobre os recursos da empresa, o que poderia aumentar os ganhos e o valor para os acionistas.

Você pode expandir com facilidade essa arquitetura flexível conforme sua empresa se expandir, abrir novos canais, lançar aplicativos móveis específicos do cliente e trazer mais fontes de dados. São necessários apenas alguns cliques no Amazon Redshift Management Console ou algumas chamadas de API.

Conclusão

Vemos uma mudança estratégia no data warehouse à medida que as corporações migram seus bancos de dados analíticos e soluções de on-premises para a nuvem, de forma a aproveitar a simplicidade, o desempenho e a boa relação custo/benefício da nuvem. Este whitepaper oferece uma conta abrangente do estado atual do data warehouse na AWS. A AWS oferece uma série de serviços e um sólido ecossistema de parceiros que lhe permite construir e executar com facilidade um data warehouse corporativo na nuvem. O resultado é uma arquitetura de analítica de alto desempenho e boa relação custo/benefício capaz de escalar com seus negócios na infraestrutura global da AWS.

Colaboradores

As seguintes organizações e pessoas contribuíram para este documento:

- Babu Elumalai, arquiteto de soluções, Amazon Web Services
- Greg Khairallah, BDM principal, Amazon Web Services
- Pavan Pothukuchi, gerente de produto principal, Amazon Web Services
- Jim Gutenkauf, redator técnico sênior, Amazon Web Services
- Melanie Henry, editora técnica sênior, Amazon Web Services
- Chander Matrubhutam, marketing de produto, Amazon Web Services

Leitura complementar

Para obter mais ajuda, consulte estas fontes:

- Biblioteca de software do Apache Hadoop[18]
- Melhores práticas do Amazon Redshift[19]
- Arquitetura Lambda[20]

Notas

1 https://www.forrester.com/report/The+Forrester+Wave+Enterprise+Data+Warehouse+Q4+2015/-/E-RES124041

2 http://aws.amazon.com/streaming-data/

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

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

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

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

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

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

9 https://aws.amazon.com/cloudtrail/

10 http://docs.aws.amazon.com/redshift/latest/mgmt/working-with-HSM.html

11 https://aws.amazon.com/kms/

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

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

14 http://docs.aws.amazon.com/redshift/latest/dg/json-functions.html

15 https://aws.amazon.com/dms/

16 https://aws.amazon.com/redshift/partners/

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

18 https://hadoop.apache.org/

19 http://docs.aws.amazon.com/redshift/latest/dg/best-practices.html

20 https://en.wikipedia.org/wiki/Lambda_architecture