Até agora neste curso nós olhamos

streaming de dados como construir pipelines de streaming resilientes.

Nós olhamos como criar um volume variável de ingestão.

Analisamos como processar dados que podem atrasar ou

desordenada usando o fluxo de dados.

E então nós olhamos como fazer consultas e

dados, mesmo que estejam sendo transmitidos usando o BigQuery.

E exibindo esses dados com o Data Studio.

Mas o que ainda não vimos é outras opções, no que diz respeito à sincronização.

Então, o BigQuery é uma solução de propósito geral muito boa.

Algo que funcionaria na maioria dos casos que você está preocupado.

Mas de vez em quando, você se depara com uma situação em que

a latência do BigQuery será problemática.

No BigQuery, os dados que estão sendo transmitidos estão disponíveis

em questão de segundos, e às vezes você vai querer uma latência menor do que isso.

Você quer que suas informações estejam disponíveis em questão de milissegundos para

exemplo ou microssegundos.

Você também pode ter problemas com a taxa de transferência do BigQuery,

que é cerca de 100.000 registros por segundo pode não ser suficiente, e

você pode querer lidar com um maior rendimento.

E assim, o que veremos neste capítulo final é como lidar com tais

requisitos de taxa de transferência e latência quando o BigQuery não é suficiente, para onde você vai?

Então, falaremos sobre o Cloud Spanner e falaremos sobre o Bigtable.

Estas serão duas das nossas opções que poderíamos considerar.

E depois passamos muito tempo olhando para o Bigtable.

Vamos ver como projetar para o Bigtable,

especificamente como projetar esquemas, como projetar a chave de linha do Bigtable.

Veremos como inserir dados no Bigtable.

Nós vamos fazer um laboratório que essencialmente leva pipeline de fluxo de dados e

que está sendo transmitido para o BigQuery e o modifica.

Para que ele esteja transmitindo as velocidades médias para o BigQuery, mas

está transmitindo as condições atuais, que é 30 vezes mais dados

as condições atuais, vamos transmiti-lo para o Bigtable.

E finalmente, examinaremos algumas considerações de desempenho.

Então, se você está tentando escolher onde devo armazenar dados no GCP,

este é um conjunto de perguntas que você poderia considerar.

Então, por exemplo, a primeira pergunta que você pode querer considerar é:

são os dados que você está tentando armazenar, são dados estruturados ou

são dados não estruturados?

Então, se é dados estruturados, então a próxima pergunta que

Você quer responder: as transações são importantes para você?

Ou sua carga de trabalho é basicamente somente leitura,

é sua carga de trabalho principalmente em torno da análise de dados?

Se você está pensando em cargas de trabalho transacional, então

3:00

a próxima pergunta é, você quer aqueles carregamentos de trabalho transacional

você está querendo ser via SQL?

3:10

Ou você está bem com a consulta no SQL?

Então, em outras palavras, seus dados são relacionais?

Nesse caso, você quer fazer consultas SQL em seus dados.

Ou os seus dados não são relacionais? Nesse caso, você quer fazer armazenamentos de objeto para

exemplo.

Então agora, por exemplo, se você tem dados estruturados, precisa de transações e

você quer poder consultá-lo com SQL, então você tem duas opções.

3:40

A opção mais comum seria ir em frente e usar o Cloud SQL.

3:45

O Cloud SQL é se um banco de dados for suficiente.

Mas se um banco de dados não for suficiente, se você precisar de vários bancos de dados,

você precisa de escalabilidade horizontal, então o Cloud Spanner é uma boa solução.

O Cloud Spanner e o Cloud SQL proporcionam uma latência de milissegundos.

4:05

Mas se a sua carga de trabalho não estiver em torno de transações, mas

se sua carga de trabalho estiver relacionada à análise de dados, se sua carga de trabalho estiver relacionada à análise de dados,

então toda essa ideia de transações e

bloqueio, etc., é apenas ouvir que você não quer pagar.

E nesse ponto, sua pergunta agora se torna, você precisa de atualizações?

Você precisa de baixa latência?

Se você não precisa de nenhum deles, se uma latência de segundos for suficiente,

Em seguida, o BigQuery é a solução mais econômica, por isso, use o BigQuery.

É um banco de dados SQL.

Ele fornece latência razoável, mas não latência muito baixa.

Mas se, por outro lado, você quiser latência de milissegundos, mas você é basicamente

preocupado com as cargas de trabalho de análise, o Bigtable é uma boa solução.

4:58

Agora vamos ver a segunda parte disso.

Se são dados não estruturados com os quais você está essencialmente lidando,

em seguida, os dados não estruturados vão idealmente para o armazenamento em nuvem.

Mas se você precisar de SDKs para dispositivos móveis,

então o armazenamento baseado em fogo seria uma boa solução.

Da mesma forma, se você não tiver dados SQL, se precisar de dispositivos móveis, coloque-os no firewall.

Se você não precisa de celular,

se for principalmente aplicativos da Web, coloque-os no armazenamento de dados.

Mas o que estamos vendo neste curso é então Cloud Spanner,

Cloud SQL, Bigtable e BigQuery.

E a maneira como você pensa sobre essas coisas é que o BigQuery é o mais comum.

5:38

Muito rentável, latência de segundos.

5:43

Cloud SQL, muito comum novamente.

Dados relacionais, dados transacionais, respaldados por um mySQL

banco de dados ou banco de dados PostgreSQL.

E então você tem duas outras soluções.

E essas são as duas soluções que analisamos neste capítulo.

Se você tiver dados SQL transacionais e

sua carga de trabalho é muito maior do que você pode ajustar em um único banco de dados.

Se você precisar de escalabilidade horizontal em outras palavras, vá para Cloud Spanner.

6:18

Se você está fazendo análise de dados, mas suas necessidades são mais

do que o BigQuery pode suportar, vá para o Bigtable.

Então, novamente, tanto o Spanner quanto o Bigtable são quando o BigQuery e

O Coud SQL não será suficiente para as suas necessidades.

Então, vamos dar uma olhada no Bigtable e no Spanner,

começando com Spanner. >> O que é o Cloud Spanner?

O Cloud Spanner é o primeiro banco de dados globalmente consistente e escalável horizontalmente.

É proprietário, não é open source.

Considere o que significa ter um banco de dados relacional consistente, mas

também distribuído e global.

Pense no que pode estar envolvido na coordenação de transações em

componentes do banco de dados relacional localizados em todo o mundo.

Parece um problema muito difícil de resolver.

7:10

O Cloud Spanner não é para todos os aplicativos.

Há momentos em que você vai querer usar o Cloud SQL e outros serviços.

O Cloud Spanner é adequado para aplicativos que exigem suporte a banco de dados relacional,

consistência forte, transações e escalabilidade horizontal.

Os casos de uso natural incluem aplicativos financeiros e

aplicações de inventário tradicionalmente servidas pela tecnologia de banco de dados relacional.

Veja alguns exemplos, casos de uso de missão crítica.

Alimentar a autenticação e provisionamento de clientes para empresas multinacionais.

Criação de sistemas consistentes para transações e gerenciamento de estoque e

os serviços financeiros em indústrias de varejo.

Suportando sistemas de alto volume que requerem baixa latência e

alto rendimento nas indústrias de publicidade e mídia.

7:59

O Cloud SQL é bom se você conseguir um único banco de dados.

Mas se suas necessidades são tais que você precisa de vários bancos de dados,

Cloud Spanner é uma ótima escolha.

O gráfico mostrado ilustra esse ponto.

O Cloud SQL atinge uma parede com cerca de 8.000 consultas por segundo.

Se você olhar para o 99º percentil de latência,

É claro que o desempenho degrada além de 5.000 consultas por segundo.

A escala horizontal através do processamento distribuído é complicada e

difícil para a maioria dos sistemas de banco de dados relacionais, como o MySQL.

No entanto, a Spanner distribui o trabalho com facilidade.

Ele distribui globalmente, se necessário, e fornece desempenho consistente.

Para suportar mais rendimento no Spanner, basta adicionar mais nós.

Agora as informações mostradas neste gráfico são de um blog.

Portanto, não é uma declaração oficial de nível de serviço ou

números de desempenho confiáveis.

É apenas uma ilustração.

Se você está lidando com um caso limítrofe, você vai querer executar o seu próprio teste e

Veja os números publicados e os contratos de nível de serviço atuais.

8:59

A arquitetura do Spanner permite alta disponibilidade e posicionamento global.

Os dados são replicados em várias zonas do Cloud, que podem estar em uma região ou

em várias regiões.

O posicionamento do banco de dados é configurável.

Você pode escolher a região para hospedar seu banco de dados.

Direitos são síncronos.

Os dados são sempre consistentes e

tem propriedades de ativos como qualquer outro banco de dados relacional.

Há muito a ser aprendido sobre o Spanner.

Existem white papers e recursos online.

Alguns pontos principais a considerar sobre Spanner, ele usa relacional familiar

semântica, para que os analistas de banco de dados tradicionais se adaptem facilmente a ele.

Os dados são divididos dentro da zona, fornecendo alto rendimento.

E fornece alta disponibilidade por design, então

não há intervenção manual necessária para lidar com uma falha de zona.