



Frameworks de Big Data

UNIDADE 02

Arquitetura de Software em Big Data: Desvendando os Pilares para Extrair Insights Valiosos

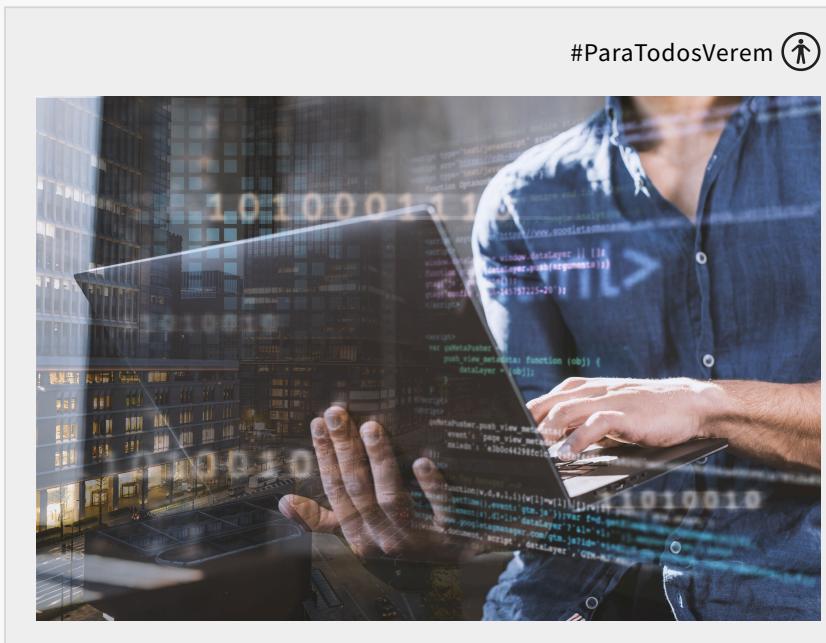
Nesta semana, exploraremos a interseção entre tecnologia e estratégia por meio de um estudo de caso envolvendo o Walmart, o maior varejista do mundo. Investigaremos o impacto do processamento de dados em sua estratégia de negócios, destacando como a arquitetura de software em Big Data desempenha um papel fundamental na extração de insights valiosos.

Durante a aula, vamos desvendar os pilares da arquitetura de software em Big Data, fornecendo uma compreensão abrangente de como as tecnologias são implementadas para lidar com grandes volumes de dados de forma eficiente.

Como complemento, na videoaula, exploraremos o *Sqoop* em um ambiente clusterizado *Hadoop*, uma ferramenta essencial para a integração de dados entre sistemas de armazenamento, como o Hadoop, e bancos de dados relacionais.

Ao conectar os conceitos aprendidos na aula com o estudo de caso do Walmart, analisaremos como a empresa utiliza a arquitetura de software em Big Data para impulsionar seu sucesso. Destacaremos como essas tecnologias permitem ao Walmart tomar decisões estratégicas baseadas em insights gerados a partir de grandes conjuntos de dados.

Em suma, esta semana nos proporcionará uma visão abrangente de como a tecnologia, estratégia e processamento de dados são elementos essenciais para o sucesso empresarial, utilizando o Walmart como estudo de caso principal e explorando os fundamentos da arquitetura de software em Big Data, juntamente com a aplicação prática do Sqoop em ambientes clusterizados Hadoop.



Fonte: Banco de imagens Freepik.

O mundo está inundado por dados. Empresas geram terabytes de informações diariamente, desde transações de clientes até registros de sensores e mídias sociais. Essa avalanche de dados, conhecida como Big Data, oferece um potencial enorme para insights valiosos, mas também impõe desafios à sua análise e manipulação. É aí que entra a **Arquitetura de Software em Big Data**, a base para construir sistemas escaláveis, confiáveis e eficientes para lidar com essa

vastidão de informações.



REFLEXÃO

Durante uma crise natural, como o furacão Sandy, redes varejistas como a Walmart enfrentam dificuldades logísticas cruciais para antecipar e atender às demandas dos consumidores. A análise de Big Data se torna essencial, fornecendo insights valiosos que orientam decisões operacionais e estratégicas. Para superar esses desafios, a rede varejista pode usar ferramentas de Big Data para prever a demanda por produtos essenciais e sazonais, garantindo o sucesso do negócio. É crucial estabelecer uma arquitetura de hardware/software adequada para suportar essas demandas emergenciais.

Então, qual seria a arquitetura ideal de Big Data para atender à rede varejista nesse momento de crise? Ou, em outras palavras, o que caracteriza uma arquitetura ideal de Big Data?

Nesta aula, iremos trilhar uma jornada para desvendar os pilares da Arquitetura de Software em Big Data. Exploraremos conceitos, modelos e ferramentas essenciais para construir sistemas robustos e flexíveis, capazes de transformar o Big Data em insights acionáveis e gerar valor para as empresas.

| Fundamentos da Arquitetura de Software em Big Data

As arquiteturas de soluções para Big Data foram concebidas para lidar com a *ingestão, processamento e análise de grandes e complexos conjuntos de dados* que não poderiam ser suportados caso fossem utilizados por meio de sistemas de banco de dados convencionais. O ponto de entrada das organizações no domínio do Big Data varia, dependendo das capacidades dos usuários e das ferramentas disponíveis. Para alguns, isso pode implicar centenas de gigabytes de dados, enquanto para outros, envolve centenas de terabytes. À medida que as ferramentas para lidar com conjuntos de Big Data se desenvolvem, o conceito de Big Data também evolui. Cada vez mais, o termo está associado ao valor que pode ser extraído dos dados por meio de análises avançadas, ao invés de se preocupar apenas com o tamanho dos dados, embora, em muitos casos, eles sejam consideravelmente grandes.

O que é Arquitetura de Software em Big Data?

Um dos primeiros passos essenciais para lidar com ações de Big Data é o planejamento adequado. Nesse sentido, é crucial considerar a frequência e as condições em que os dados são encontrados. Geralmente, nesta fase, é comum encontrar as operações de *Extração*, *Transformação* e *Carga*. Uma vez concluída essa etapa, os dados ficam disponíveis, dando início às etapas subsequentes de processamento e exibição das informações.

Mas afinal, o que significa essa sigla e qual é a sua função em um processo de ETL? Em resumo, ETL é a abreviação de *Extrair*, *Transformar* e *Carregar*. Esse processo desempenha um papel fundamental na integração e armazenamento de dados, pois possibilita a coleta de informações de diversas fontes. Posteriormente, os dados são transformados para um formato coerente e, por fim, são carregados em um banco de dados ou Data Warehouse de destino. A Figura 1 representa visualmente esse processo.

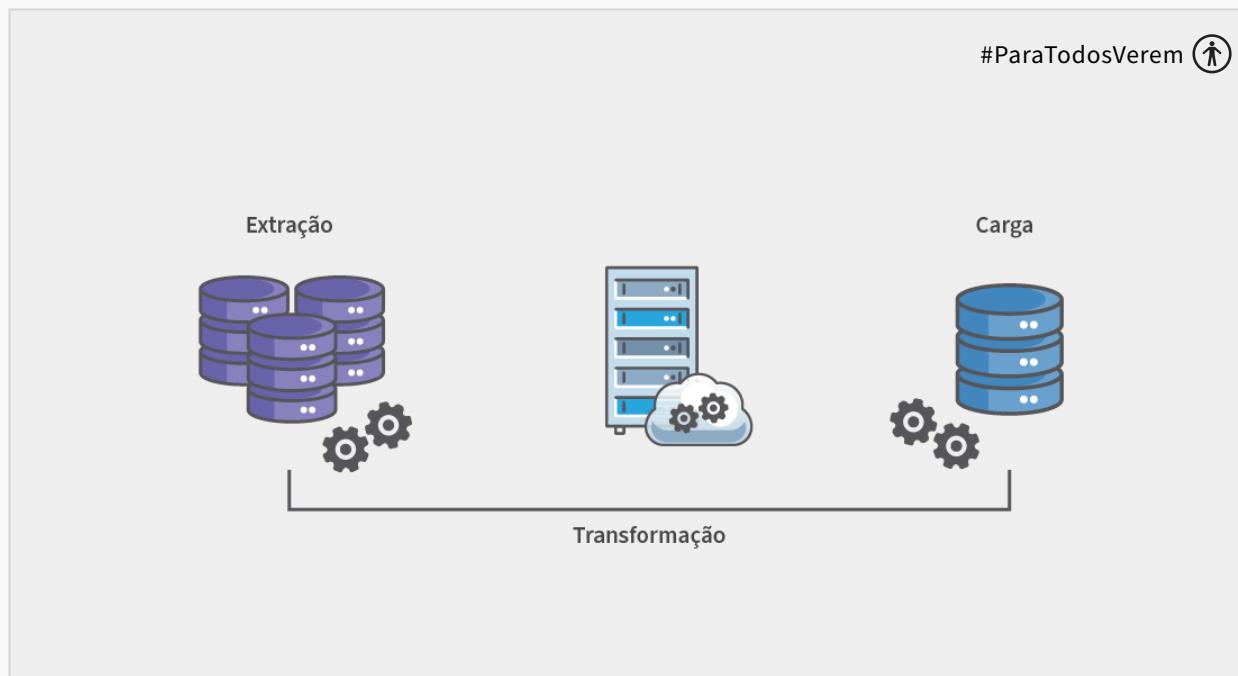


Figura 1: Processo de ETL. Fonte: O autor (2024).

A importância da computação em cluster

De acordo com ALECRIM (2013), a concepção de cluster refere-se a um sistema que conecta dois ou mais computadores para colaborarem na realização de uma tarefa. Nesse arranjo, as máquinas dividem as responsabilidades de processamento entre si, operando de maneira simultânea e eficaz. Portanto, a computação em cluster pode ser entendida como uma estratégia de reunir recursos de várias máquinas para gerenciar recursos computacionais de forma conjunta, visando a conclusão de tarefas.

Essa prática é justificada pela natureza e pelo volume dos dados, uma vez que os computadores individuais frequentemente não são capazes de lidar com as exigências de processamento em várias etapas. Assim, para atender às demandas computacionais significativas associadas ao armazenamento e processamento de dados, os clusters de computadores se mostram mais apropriados. Cada computador participante desse cluster é denominado como um nó, conforme ilustrado na *Figura 2*.

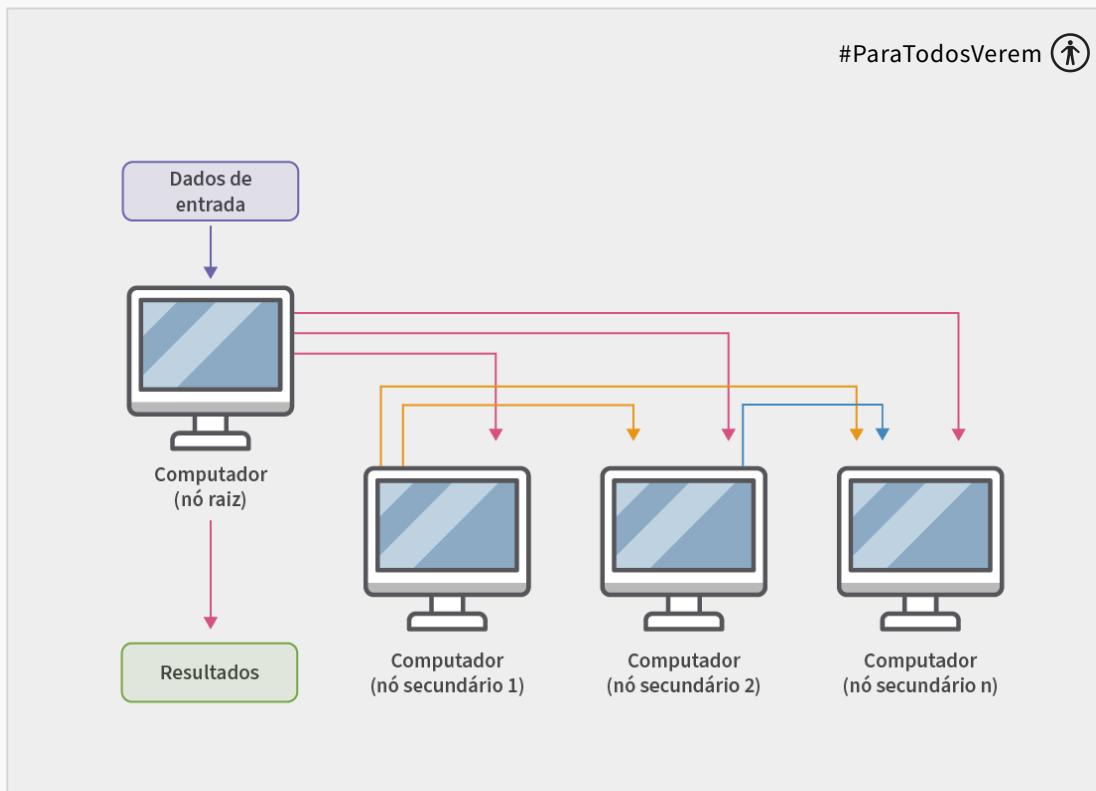


Figura 2: Representação de estrutura do modelo de computação em Cluster. Fonte: Adaptado de Geeksforgeeks.

Os benefícios ao utilizar a arquitetura em cluster, segundo PEREIRA (2020) são, em especial, referentes a:

+ Pooling de recursos

Agrupar o espaço de armazenamento disponível para dados é uma vantagem evidente, porém, a combinação de capacidade de CPU e memória também é de extrema importância. O processamento de conjuntos de dados volumosos demanda grandes quantidades desses três recursos.

+ Alta disponibilidade

Clusters podem garantir diferentes níveis de tolerância a falhas e garantias de disponibilidade para evitar que falhas de hardware ou software afetem o acesso aos dados e processamentos. Essa característica se torna cada vez mais crucial à medida que a análise em tempo real é enfatizada.

+ Escalabilidade simplificada

Clusters facilitam a escalabilidade horizontal adicionando máquinas ao grupo. Isso significa que o sistema pode responder às mudanças nos requisitos de recursos sem a necessidade de expandir os recursos físicos de uma única máquina.

O conceito de **arquitetura de software em Big Data**, portanto, traz a ideia de qual estrutura será necessária para suportar ou para criar sistemas que sejam capazes de lidar com grandes volumes de dados. Esse conceito se concentra em três pilares principais:

- **Armazenamento:** Como os dados serão armazenados e organizados para garantir eficiência e escalabilidade.
- **Processamento:** Como os dados serão processados e analisados para gerar insights valiosos.
- **Análise:** Como os insights serão visualizados e apresentados de forma clara e útil para os usuários.

Nesse cenário a ideia de estruturar uma arquitetura de software em Big Data de forma referencial é fundamental para superar os desafios dos 3 V's de Big Data, em relação a:

Volume: a quantidade de dados gerados cresce exponencialmente, exigindo soluções de armazenamento escaláveis.

Variedade: os dados vêm de diferentes fontes e formatos, necessitando de flexibilidade na análise.

Velocidade: os dados precisam ser processados em tempo real para gerar insights açãoáveis.

Dessa forma, lidar com soluções de Big Data requer planejamento cuidadoso devido à inadequação dos sistemas existentes para lidar com volumes massivos de informações. É essencial considerar o tipo, frequência e condições de armazenamento dos dados, além de planejar o processo de ETL para garantir insights valiosos com o uso de soluções de Big Data a partir de volumes massivos de dados heterogêneos que seriam impossíveis de serem tratados por métodos convencionais.

| Componentes Essenciais da Arquitetura

A Figura 3 representa uma estrutura típica de aplicações de Big Data, no entanto é importante notar que no caso de soluções individuais podem não englobar todos os componentes representados.

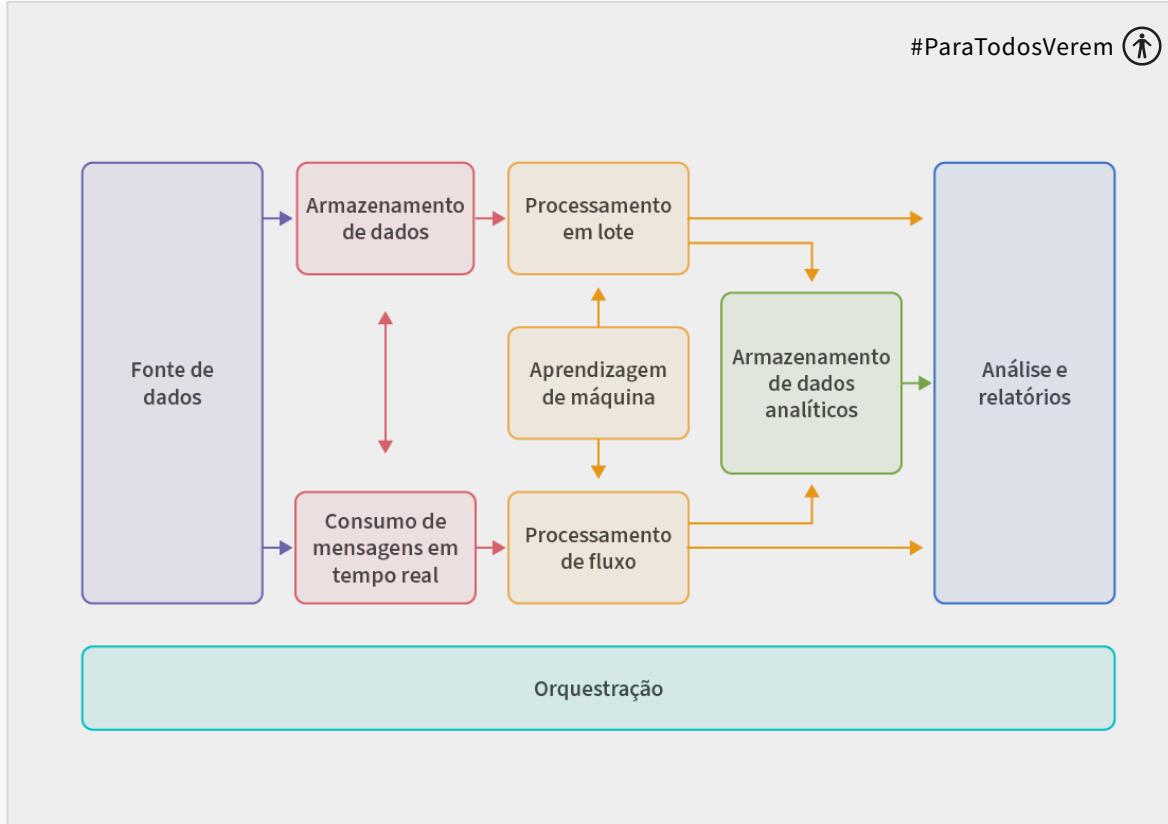


Figura 3: Estrutura típica de aplicações de Big Data. Fonte: Adaptado de Microsoft.

Segundo a Microsoft, os componentes de a maioria das arquiteturas de Big Data incorpora uma seleção dos seguintes componentes:

+ **Fontes de dados**

São os pontos de origem dos dados. Podem incluir armazenamentos de dados de aplicativos, arquivos estáticos produzidos por aplicativos e fontes de dados em tempo real, como dispositivos IoT.

+ Armazenamento de dados

Os dados processados em lotes são comumente armazenados em um repositório distribuído, conhecido como *Data lake*, que pode conter grandes volumes de arquivos em diversos formatos.

+ Processamento em lotes

Trabalhos em lotes são usados para filtrar, agregar e preparar os dados para análise. Isso geralmente envolve a leitura de arquivos de origem, processamento e gravação da saída em novos arquivos.

+ Ingestão de mensagens em tempo real

Para fontes em tempo real, é necessário capturar e armazenar mensagens para processamento de fluxo. Isso pode envolver um armazenamento simples de mensagens ou um repositório de ingestão de mensagens.

+ Processamento de fluxo

Após a captura das mensagens em tempo real, é necessário processá-las para filtragem, agregação e preparação dos dados para análise. Os dados processados são então gravados em um coletor de saída.

+ Aprendizado de máquina

Algoritmos de aprendizado de máquina podem ser aplicados aos dados preparados para construir modelos preditivos ou classificatórios.

+ Armazenamento de dados analíticos

Os dados processados estão disponíveis para consulta por meio de ferramentas analíticas, podendo ser armazenados em um data warehouse relacional ou em tecnologias NoSQL de baixa latência.

As ações de armazenar, manipular e analisar dados são atividades convencionais para sistemas computacionais, porém, elas assumem uma importância ainda maior quando lidamos com grandes volumes de dados diariamente, no entanto muitas das vezes os formatos e/ou as estruturas desses dados não são compatíveis. Nesse cenário, o *Quadro 1* representa os tipos de armazenamento mais comuns encontrados em soluções para uma arquitetura de Big Data.

Armazenamento:

| Tipo de Armazenamento | Descrição | Tecnologias |
|-----------------------|---|------------------|
| HDFS | Sistema de arquivos distribuídos para armazenar grandes volumes de dados. | Hadoop |
| NoSQL | Bancos de dados não relacionais para armazenar dados semiestruturados e não estruturados. | HBase, Cassandra |
| Data Lakes | Repositórios de dados em grande escala para armazenar dados de diversas fontes. | Hive |

Quadro 1: Tipo de Armazenamento de Dados. Fonte: O autor (2024).

| Modelos de Arquitetura para Processamento de Dados:

Existem diferentes modelos de arquitetura para processamento de dados, cada um com suas características e aplicabilidades específicas. Entre os mais comuns estão o *Batch* (Processamento em Lote), o *Stream* e o *Lambda*, cada um oferecendo abordagens distintas para lidar com o processamento de informações.

No modelo de *Batch*, as informações são coletadas e armazenadas para posterior processamento em lotes. Segundo REIS (2018), o processamento batch é utilizado para análises sobre dados históricos, no qual a necessidade de acessar seu resultado não tem limitação rígida de tempo, ou seja, é possível esperar minutos ou horas pela entrega do resultado. Esse método é amplamente utilizado em diversas atividades do cotidiano, como na leitura de medidores de água e luz, além de transações comerciais com cartões de crédito e débito. Por exemplo, ao fazer a leitura do consumo de água, os dados são coletados e armazenados temporariamente até que sejam processados posteriormente na sede da empresa.

Já o modelo de *Stream* se destaca pelo processamento em tempo real de dados que estão em fluxo contínuo. É utilizado em situações em que a análise de informações em tempo real é essencial, como em sistemas de monitoramento de tráfego ou detecção de fraudes financeiras. Nesse caso, os dados são processados à medida que são gerados, permitindo uma tomada de decisão mais rápida e ágil. Na *Figura 4*, é mostrado um comparativo entre os modelos de processamento em *Batch* e de *Stream*.

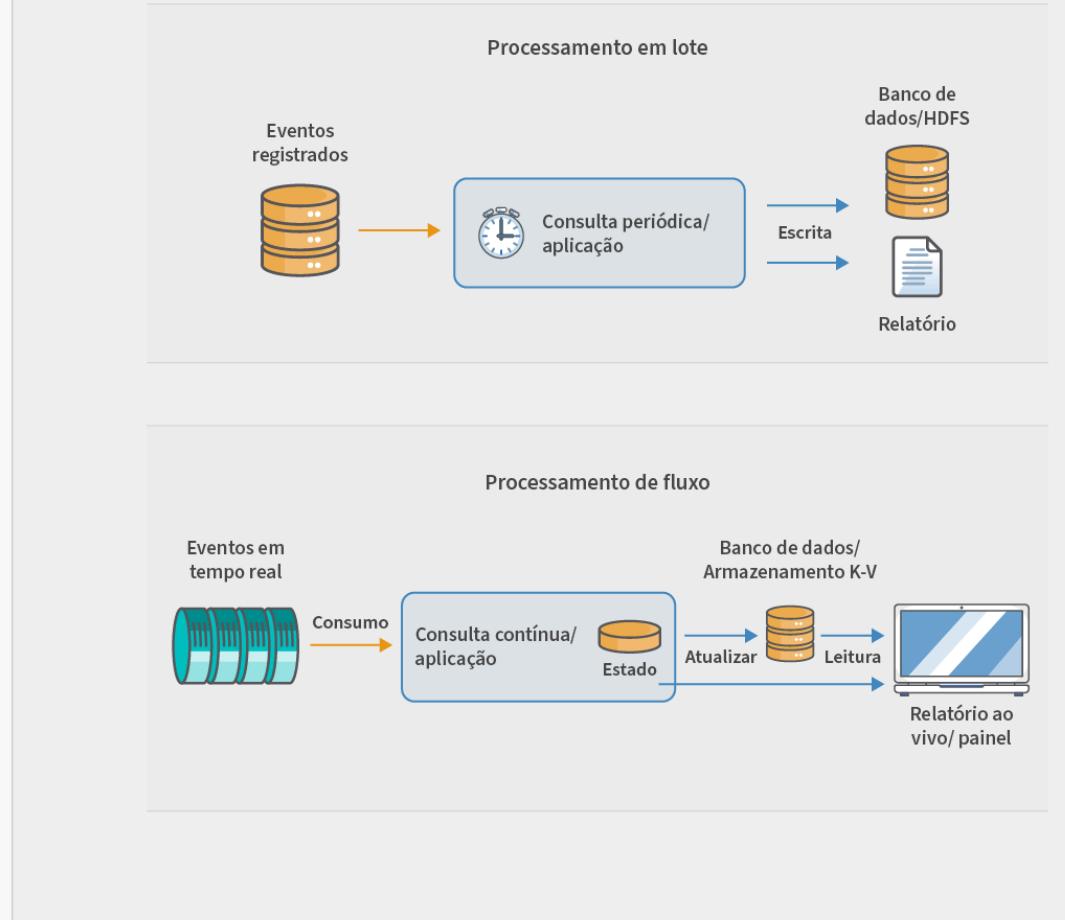


Figura 4: Comparativo dos modelos de Processamento Batch e Stream. Fonte: Adaptado de K21 Academy.

Por fim, a arquitetura Lambda, representada na *Figura 5*, combina elementos dos modelos Batch e Stream, oferecendo uma abordagem híbrida que busca aproveitar as vantagens de ambos. Nesse modelo, os dados são processados em lotes para análises históricas e em tempo real. Essa combinação permite uma maior flexibilidade e eficiência na manipulação de dados, sendo amplamente adotada em ambientes que requerem tanto análises em tempo real quanto análises retrospectivas.

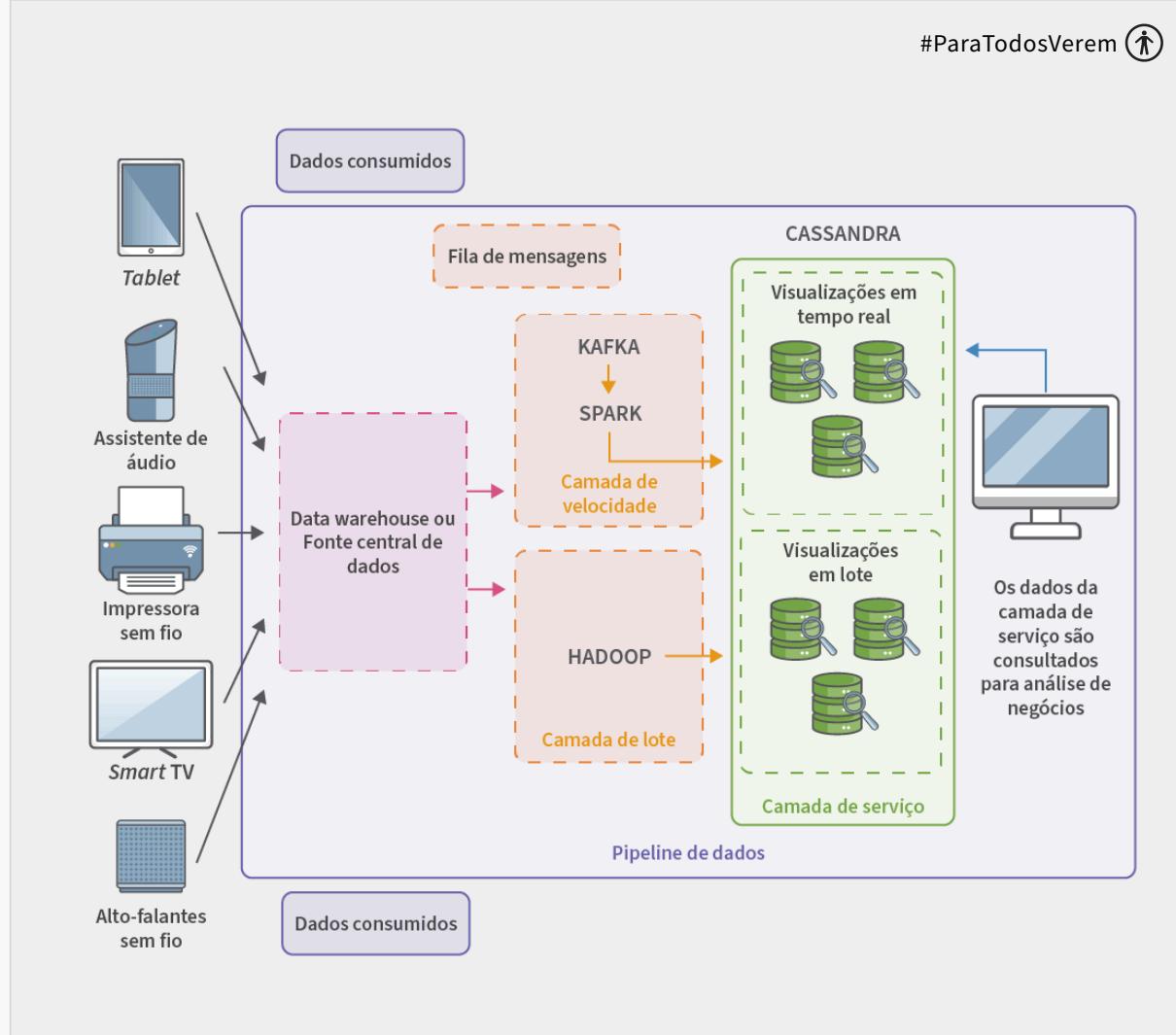


Figura 5: Modelo Lambda. Fonte: Adaptado de CAZTON, 2023.

Agora que você já conhece os tipos de arquiteturas para processamento de dados, vamos revisitar o exemplo do nosso estudo de caso do Walmart e analisar quais modelos de arquitetura seriam mais eficientes considerando as necessidades da empresa?

Dada a imensa escala das operações do Walmart e a importância de lidar com grandes volumes de dados, podemos concluir que uma abordagem híbrida como a *arquitetura Lambda seria a mais vantajosa*. Essa combinação de processamento em lotes e em tempo real oferece flexibilidade para enfrentar tanto desafios do passado quanto situações em tempo real, permitindo ao Walmart se adaptar rapidamente a eventos imprevistos, como o furacão Sandy, enquanto continua tomando decisões estratégicas baseadas em informações valiosas.

Portanto, a arquitetura Lambda se destaca como a escolha ideal para uma empresa com a magnitude e complexidade operacional do Walmart, capacitando-a a manter sua posição de liderança no mercado global de varejo.

| Conclusão

Na aula de hoje, exploramos a importância da Arquitetura de Software em Big Data e como ela serve como base para construir sistemas capazes de transformar dados em insights valiosos. Ao entender os conceitos, modelos e ferramentas dessa área, os profissionais se preparam para enfrentar desafios como o apresentado no caso do furacão Sandy, onde redes varejistas como a Walmart precisam antecipar e atender às demandas dos consumidores durante crises naturais.

Para superar esses desafios, é essencial estabelecer uma arquitetura de hardware/software adequada que suporte as demandas emergenciais, garantindo o sucesso do negócio. No caso específico da Walmart durante o furacão Sandy, a análise em tempo real seria crucial para prever e atender às demandas dos consumidores de forma ágil e eficaz. Portanto, um modelo de processamento em tempo real seria mais adequado para esse cenário, permitindo que a empresa reaja rapidamente às mudanças nas condições do mercado e nas necessidades dos clientes.

No entanto, é importante ressaltar que a escolha do modelo de processamento depende das especificidades do problema e das capacidades tecnológicas disponíveis. Cada situação exigirá uma avaliação cuidadosa para determinar a abordagem mais eficaz e a arquitetura mais adequada para enfrentar os desafios apresentados.

| Referências Bibliográficas

ALECRIM, E. Cluster: conceito e características. In: InfoWester. Cluster. InfoWester, 2023. Disponível em: <https://www.infowester.com/cluster.php>. Acesso em: 29 fev. 2024.

ANALYTICS BR. Big data para iniciantes. **Analytics BR**, 2023. Disponível em: <https://analyticsbr.com.br/big-data-para-iniciantes/>. Acesso em: 29 fev. 2024.

CAZTON CONSULTING. Lambda architecture. **Cazton Consulting**, 2023. Disponível em: <https://www.cazton.com/consulting/enterprise/lambda-architecture>. Acesso em: 29 fev. 2024.

INFOWESTER. Cluster. **InfoWester**, 2023. Disponível em: <https://www.infowester.com/cluster.php>. Acesso em: 29 fev. 2024.

GEEKSFORGEEKS. An overview of cluster computing. **GeeksforGeeks**, 2023. Disponível em: <https://www.geeksforgeeks.org/an-overview-of-cluster-computing/>. Acesso em: 29 fev. 2024.

K21 ACADEMY. Batch processing vs stream processing. **K21 Academy**, 2023. Disponível em: <https://k21academy.com/microsoft-azure/data-engineer/batch-processing-vs-stream-processing/>. Acesso em: 29 fev. 2024.

MICROSOFT. Arquiteturas de big data. **Microsoft Learn**, 2023. Disponível em: <https://learn.microsoft.com/pt-br/azure/architecture/guide/architecture-styles/big-data>. Acesso em: 29 fev. 2024.

MORAIS, I. S.; et al. **Introdução a Big Data e Internet das Coisas (IoT)**. Porto Alegre : Grupo A, 2018. E-book. ISBN 9788595027640. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788595027640/>. Acesso em: 29 fev. 2024.

PEREIRA, M. A.; et al. **Framework de Big Data**. Porto Alegre: Grupo A, 2020. E-book. ISBN 9786556900803. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9786556900803/>. Acesso em: 29 fev. 2024.

REIS, M. Arquitetura de referência para soluções de big data. **Blog Marco Reis**, 2023. Disponível em: <http://blog.marcoreis.net/>. Acesso em: 29 fev. 2024.



© PUCPR - Todos os direitos reservados.