



## **A aula interativa do Módulo 2 - Bootcamp Engenheiro de Dados começará em breve!**

**Atenção:**

- 1) Você entrará na aula com o microfone e o vídeo DESABILITADOS.**
- 2) Apenas a nossa equipe poderá habilitar seu microfone e seu vídeo em momentos de interatividade, indicados pelo professor.**
- 3) Utilize o recurso Q&A para dúvidas técnicas. Nossos tutores e monitores estarão prontos para te responder e as perguntas não se perderão no chat.**
- 4) Para garantir a pontuação da aula, no momento em que o professor sinalizar, você deverá ir até o ambiente de aprendizagem e responder a enquete de presença. Não é necessário encerrar a reunião do Zoom, apenas minimize a janela.**

# Armazenamento de Dados

Primeira Aula Interativa

Prof. Ricardo Brito Alves

# Nesta aula



- ☐ Apresentação do Professor.
- ☐ Trabalho Prático.
- ☐ Tópicos da Disciplina e Temas Interessantes.

# Apresentação do Professor

---

# Apresentação do Professor



Ricardo Brito Alves

## Formação Acadêmica:

- Graduado em Ciência da Computação pela Pontifícia Universidade Católica de Minas Gerais ,1994.
- Especialista em Gestão de Negócios pela Una, 2008.
- MBA em Gestão Estratégica de Projetos pela Una, 2009.
- Mestrado em Engenharia Elétrica pela Pontifícia Universidade Católica de Minas Gerais, 2018.
- Doutorando em Ciência da Computação pela Universidade Federal de Minas Gerais, 2024.

# Apresentação do Professor

Ricardo Brito Alves

Experiência Profissional:

- Atua há vários anos no setor de tecnologia, com desenvolvimento de projetos de Software.
- Desde 2002 atua com projetos de Data mining e BI.
- Atua há 7 anos na área de Inteligência Artificial.
- Ocupa atualmente o cargo de IT Manager em uma empresa de tecnologia e é docente de cursos de pós-graduação em tecnologia.

# Trabalho Prático

---

# **Tópicos da Disciplina e Temas Interessantes**

---



# Material





[https://drive.google.com/drive/u/1/folders/1sqqGG2eNmrr\\_39pRPUKY095-A10\\_cnnQ](https://drive.google.com/drive/u/1/folders/1sqqGG2eNmrr_39pRPUKY095-A10_cnnQ)

Name ↑

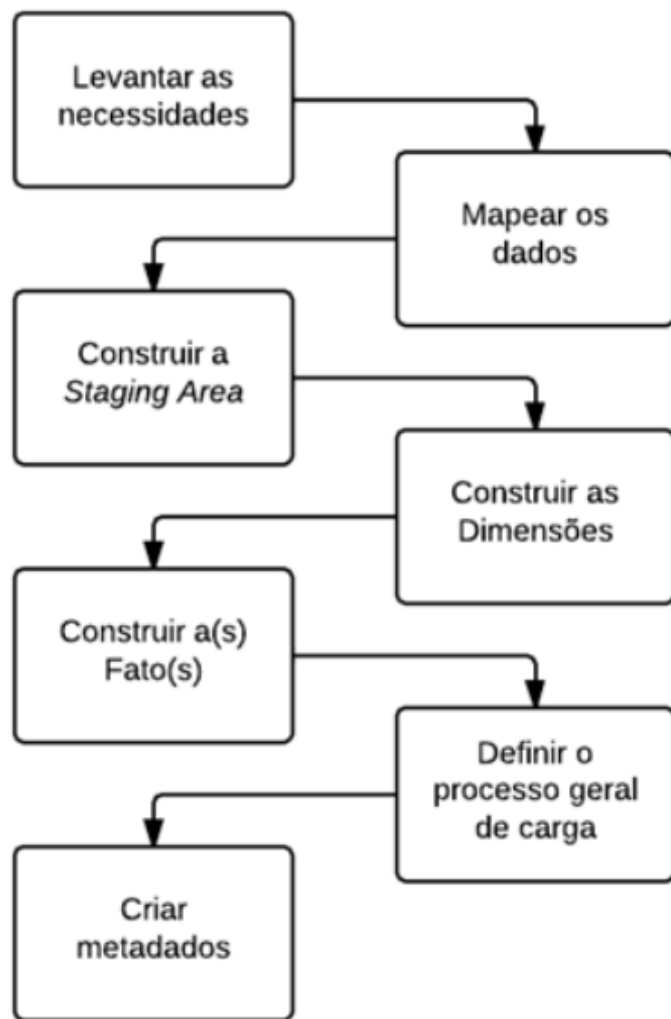
 Desafio

 Pratica MongoDB

 Pratica Mysql

 Trabalho prático

# Etapas na Construção de um DW



# Características dos Dados

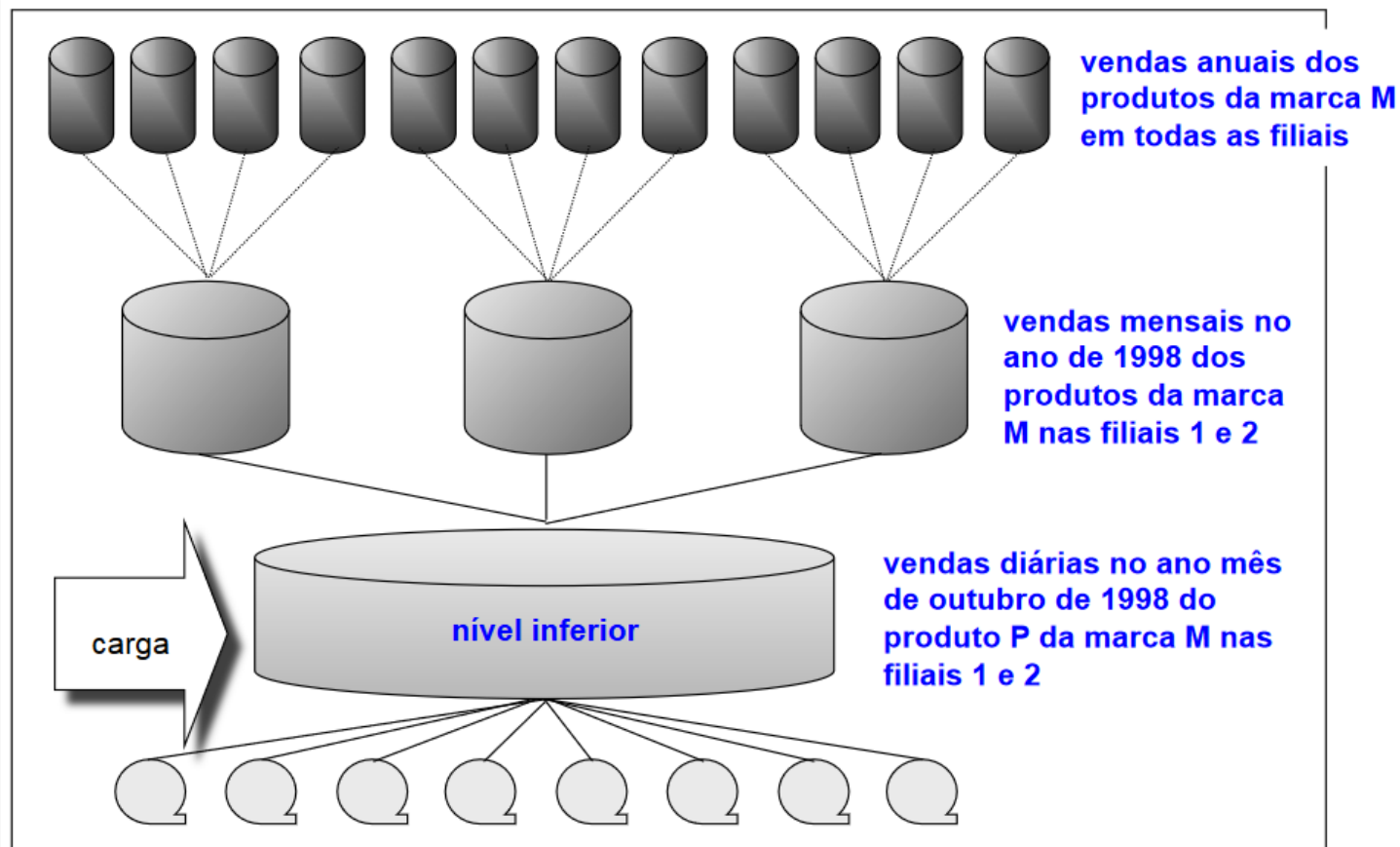
## São não-voláteis

- O conteúdo do DW permanece estável por longos períodos de tempo.

## São históricos

- Relevantes a algum período de tempo, por exemplo: usualmente dados relativos a um grande espectro de tempo (5 a 10 anos) encontram-se disponíveis.

# Características dos Dados



# Granularidade

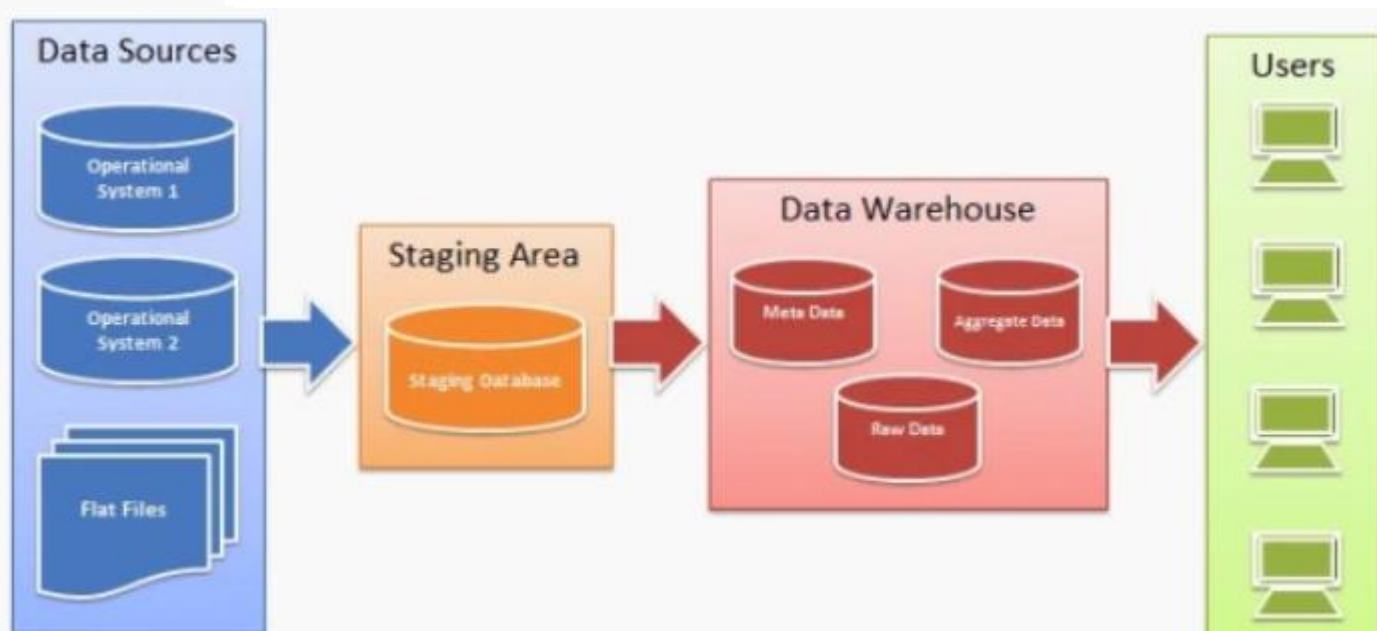
Grau de detalhamento em que os dados são armazenados em um nível.

Questão de projeto muito importante, pois:

- Impacta no volume de dados armazenado.
- Afeta as consultas que podem ser respondidas.

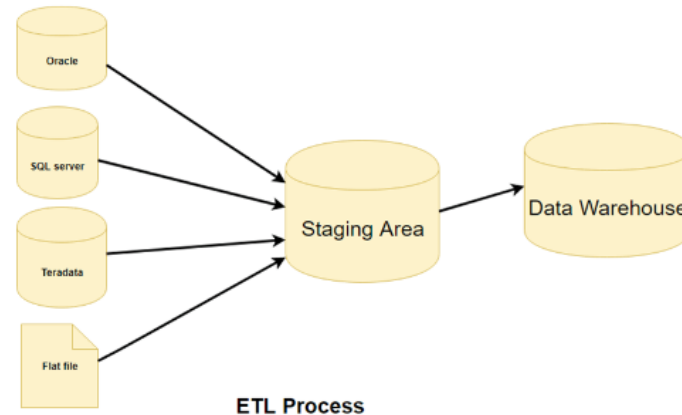
# Stage Area

É uma área de tratamento, padronização e transformação das informações operacionais para carga na arquitetura de dados BI (DW, ODS, DM).



# Stage Area

ST1 (staging 1)



## Características:

1. Query de carga idêntica à tabela de origem.
2. Fonte de Origem é a tabela origem.
3. Levam-se todos os campos da tabela origem.
4. Não existe chave primária na ST1.
5. Método de carga escolhido Truncate.
6. A cada processamento a tabela será esvaziada e carregada novamente.

# Stage Area



## ST2 (staging 2)

### Características:

1. Query de carga com transformações e cálculos.
2. Fonte de origem é a ST1;
3. Levam-se os campos que serão utilizados nas Dimensões e Fatos;
4. Existe chave primária na ST2. A chave montada é a chave de negócio;
5. Método de carga escolhido Update/Insert;
6. A latência dessa tabela será de todo o período de carga do DW.
7. A cada processamento a tabela será atualizada com alterações de registros já existentes e com novos registros.



# Stage Area

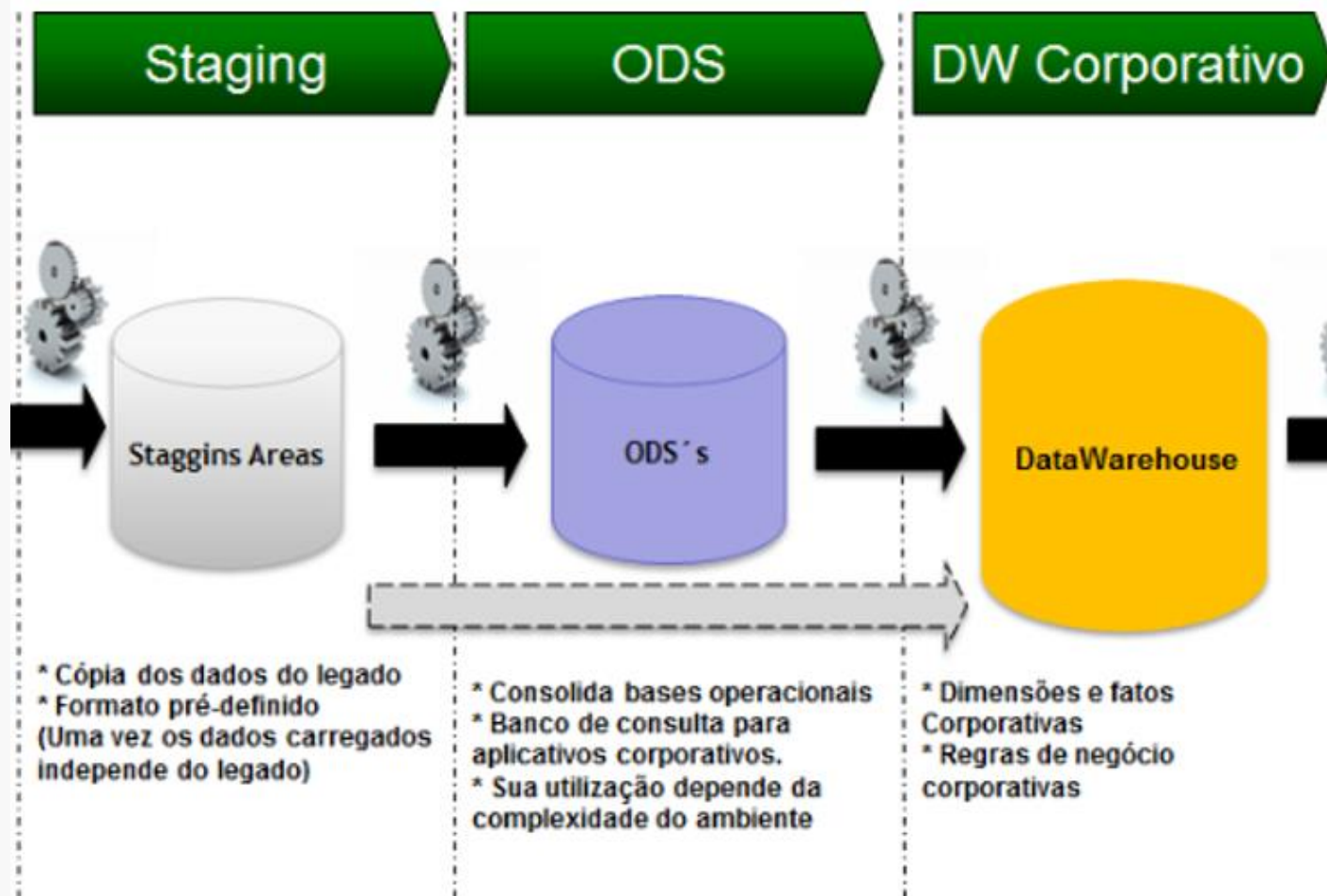


**ST2 Aux (staging 2 aux):** tem como objetivo otimizar o processo de carga diário.

## **Características:**

1. Query de carga com transformações e cálculos.
2. Fonte de origem é a ST1.
3. Levam-se os campos que serão utilizados nas Dimensões e Fatos.
4. Existe chave primária na ST2. A chave montada é a chave de negócio.
5. Método de carga escolhido Truncate.
6. Latência diária.
7. A cada processamento a tabela será esvaziada e carregada novamente.

# ODS



# Surrogate Key



A Surrogate Key nada mais é que o campo de Primary Key da dimensão.

O que é uma Primary Key?

É a coluna utilizada para identificar cada linha na tabela de forma única.

**A Surrogate Key é uma chave artificial e auto incremental.**

A palavra artificial vem do tipo, porque ela não existe em lugar nenhum, não está lá no transacional como a Natural Key, ela é criada no Data Warehouse.

E é auto incremental porque toda vez que é chamada, troca de número, então ela começa com 1 e vai indo para 2, 3, 4, e assim por diante.

- Tem as características de uma Primary Key.
- É utilizada para referenciar a dimensão na fato.
- É auto incremental.
- É uma chave artificial.
- Não se repete.



# Carga das Dimensões



As cargas das dimensões serão originadas a partir da ST2.

## Características:

1. Query apenas de leitura da ST2, pois as transformações já foram feitas.
2. Fonte de Origem é a ST2 Aux para carga diária e a ST2 para carga full.
3. Altera-se algum nome de campo para se adequar as regras da corporação de acordo com o Dicionário de Dados.
4. Para cada chave de negócio será gerada uma SRK.
5. A SRK é um campo numérico sequencial.

# Slowly Changing Dimension

## Tipo 1

O valor anterior é sobreposto pelo valor atual, perdendo-se o histórico. Usado principalmente para correção de informações, como nome de segurado e descrição de produto. Exemplo de uma tabela fornecedor.

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State
123	Abc	Acme Supply Co	CA

# Slowly Changing Dimension

## Tipo 2

Supplier_Key	Supplier_Code	Supplier_Name	Supplier_State	START_DATE	END_DATE
123	Abc	Acme Supply Co	CA	01-Jan-2000	21-Dez-2004
124	Abc	Acme Supply Co	IL	22-Dez-2004	

# Slowly Changing Dimension



## Tipo 3

Supplier_Key	Supplier_Code	Supplier_Name	Original_Supplier_State	Effective_Date	Current_Supplier_State
123	Abc	Acme Supply Co	CA	22-Dez-2004	IL

# OLAP



O OLAP é uma interface com o usuário e não uma forma de armazenamento de dados, porém se utiliza do armazenamento para poder apresentar as informações.

Os métodos de armazenamento são:

- ROLAP (OLAP Relacional): Os dados são armazenados de forma relacional.
- MOLAP (OLAP Multidimensional): Os dados são armazenados de forma multidimensional.
- HOLAP (OLAP Híbrido): Uma combinação dos métodos ROLAP e MOLAP.

Os métodos mais comuns de armazenamento de dados utilizados pelos sistemas OLAP são ROLAP e MOLAP. O ROLAP usa a tecnologia RDBMS (Relational DataBase Management System), na qual os dados são armazenados em uma série de tabelas e colunas. Enquanto o MOLAP usa a tecnologia MDDB (MultiDimensional Database), onde os dados são armazenados em arrays multidimensionais.

Microsoft SQL Server oferece suporte a todos os três modos de armazenamento básico.



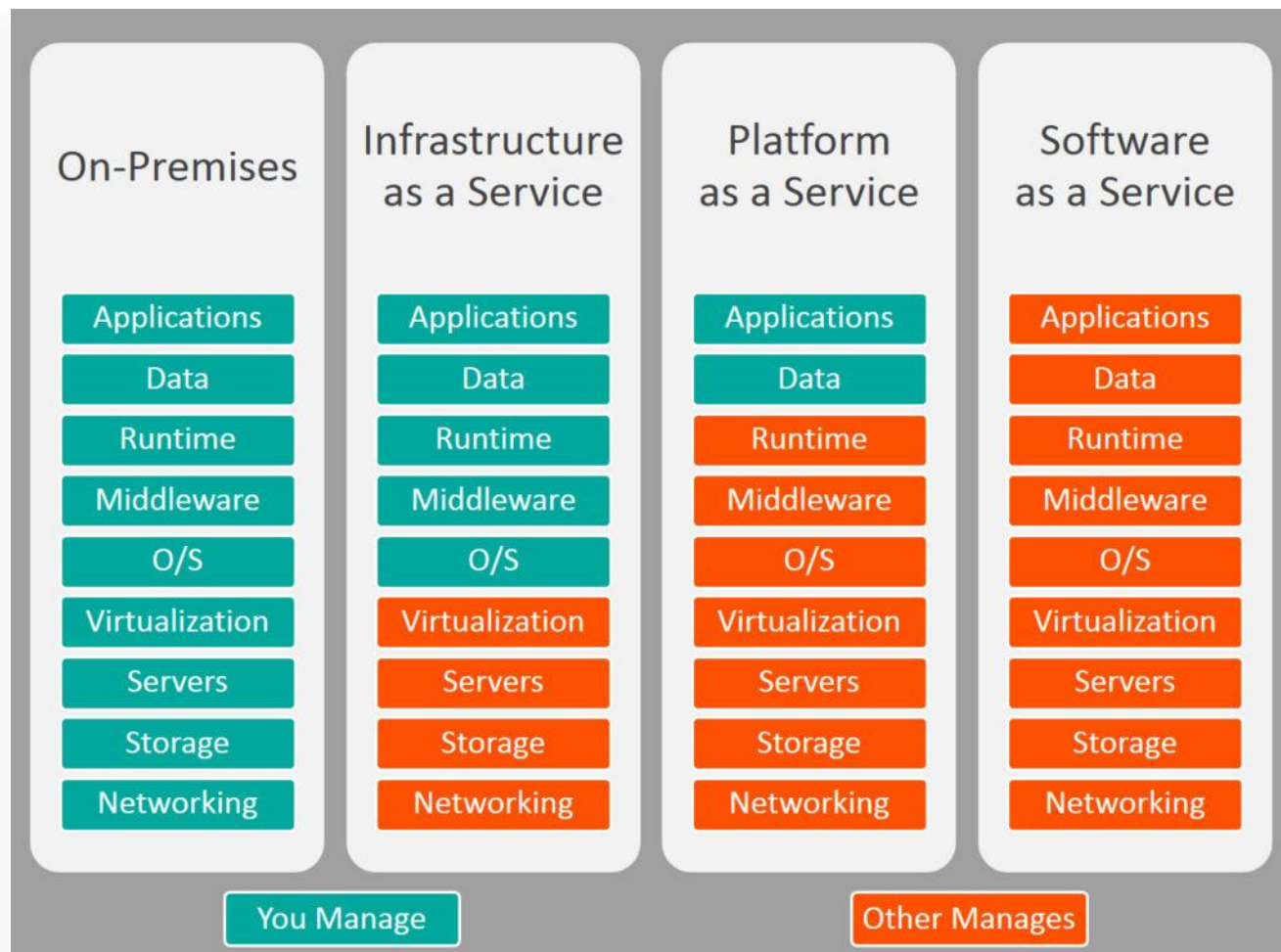
# OLAP



ROLAP é mais indicado para DATA WAREHOUSE pelo grande volume de dados, a necessidade de um maior número de funções e diversas regras de negócio a serem aplicadas.

MOLAP é mais indicado para DATA MARTS, onde os dados são mais específicos e o aplicativo será direcionado na análise com dimensionalidade limitada e pouco detalhamento das informações.

# Computação em Nuvem



# Arquiteturas Monolíticas

***Com as arquiteturas monolíticas, todos os processos são altamente acoplados e executam como um único serviço.***

Isso significa que se um processo do aplicativo apresentar um pico de demanda, toda a arquitetura deverá ser escalada. A complexidade da adição ou do aprimoramento de recursos de aplicativos monolíticos aumenta com o crescimento da base de código. Essa complexidade limita a experimentação e dificulta a implementação de novas ideias.

***As arquiteturas monolíticas aumentam o risco de disponibilidade de aplicativos, pois muitos processos dependentes e altamente acoplados aumentam o impacto da falha de um único processo.***

# Arquitetura de Microserviços

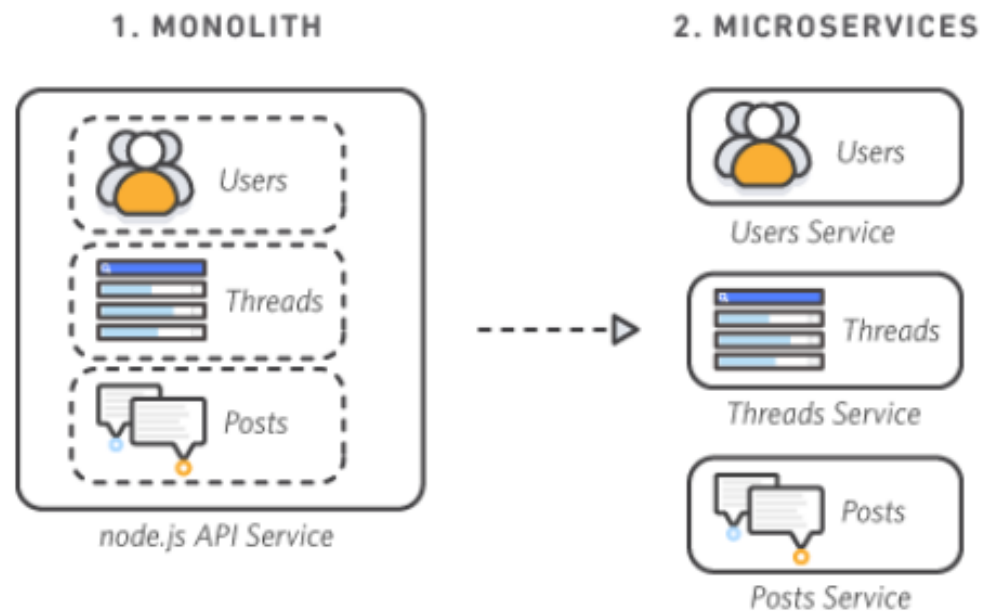


*Com uma arquitetura de microserviços, um aplicativo é criado como componentes independentes que executam cada processo do aplicativo como um serviço.*

Esses serviços se comunicam por meio de uma interface bem definida usando APIs leves. Os serviços são criados para recursos empresariais e cada serviço realiza uma única função. Como são executados de forma independente, cada serviço pode ser atualizado, implantado e escalado para atender a demanda de funções específicas de um aplicativo.

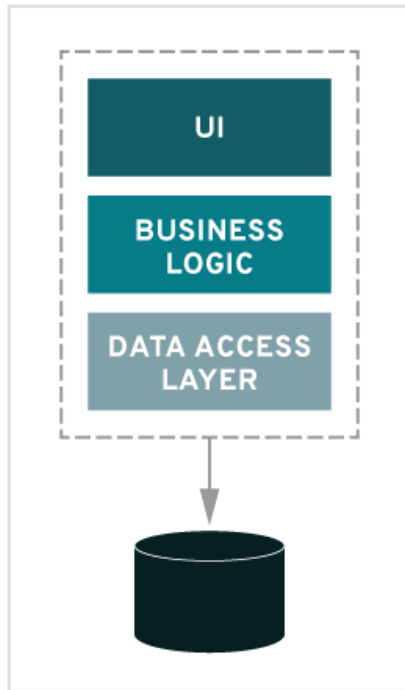


# Arquitetura de Microsserviços



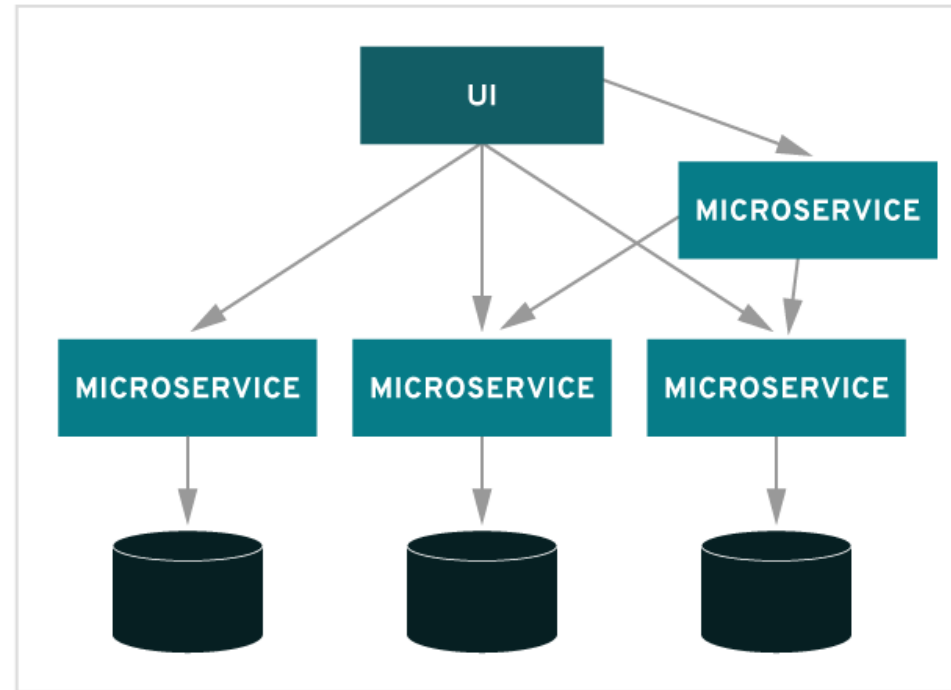
# Arquitetura

## MONOLITHIC



VS.

## MICROSERVICES



# Pentaho PDI



## **Pentaho Data Integration.**

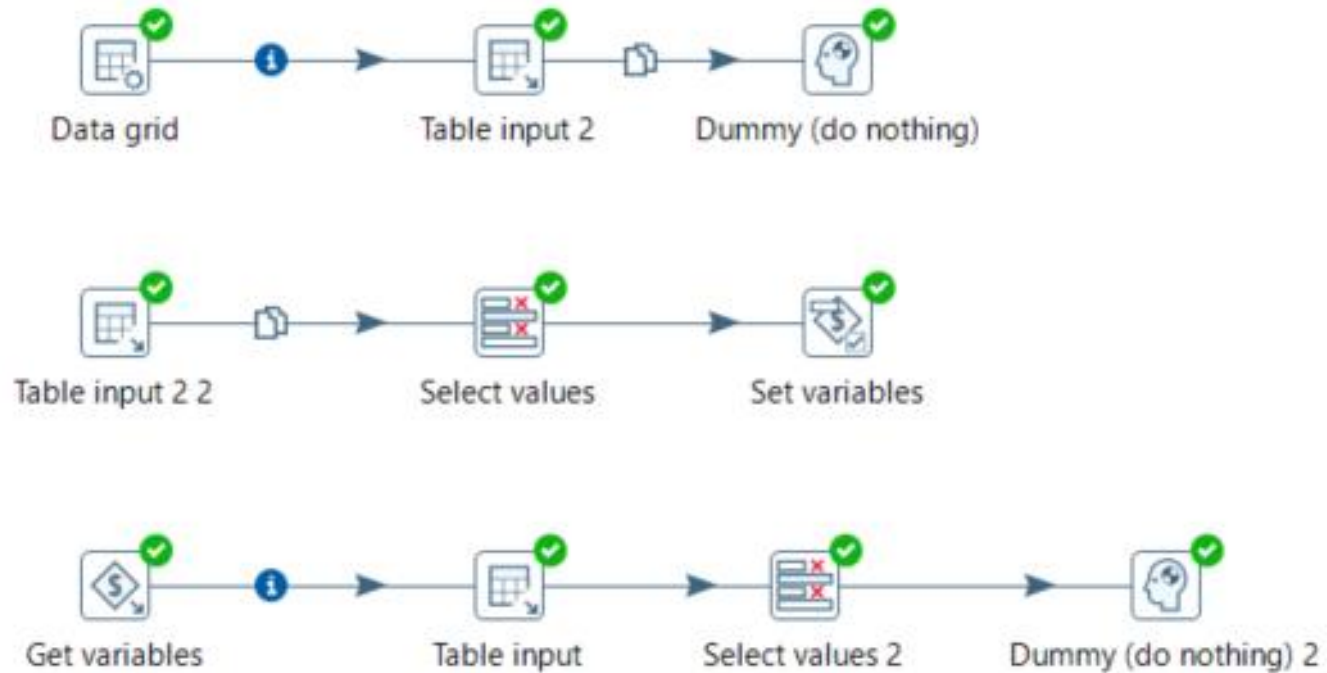
Utilização do Pentaho PDI ou outra ferramenta de ETL para migração de dados.

Por exemplo, podemos fazer uma suposta fusão entre duas lojas de e-Commerce, no qual o sistema de uma delas, armazenado em SQL Server, se tornará responsável por gerir as informações sendo necessária, portanto, uma migração de dados do MySQL para o SQL Server.

# Ensaio no Pentaho



Trabalhando com a passagem de parâmetros.





# Ensaio no Pentaho



Trabalhando com Rest.

