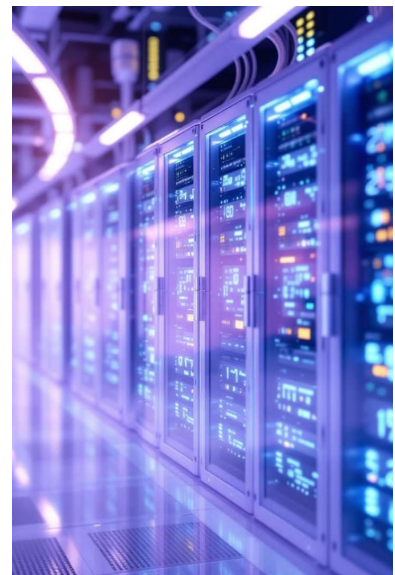


# Pré-Processamento e Armazenamento de Big Data

*Prof. Edson Pedro Ferlin*

## Introdução

- 1** O que é?  
Fases essenciais para garantir que os dados sejam limpos, estruturados e armazenados eficientemente.
- 2** Melhora a qualidade  
Prepara os dados para análises futuras mais precisas.
- 3** Otimiza o desempenho  
Possibilita o processamento distribuído mais eficiente.
- 4** Armazenamento eficiente  
Facilita a recuperação de dados quando necessário.



## Processamento Distribuído e Paralelo

### Processamento Distribuído

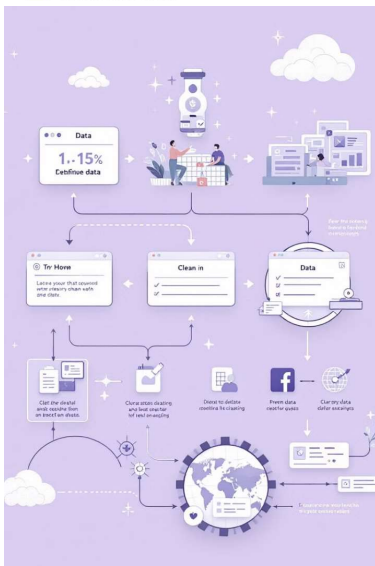
Execução de tarefas em múltiplos nós simultaneamente.  
Possibilita lidar com grandes volumes sem sobrecarregar uma máquina.

### Processamento Paralelo

Divide tarefas em partes executadas simultaneamente.  
Melhora desempenho e escalabilidade.

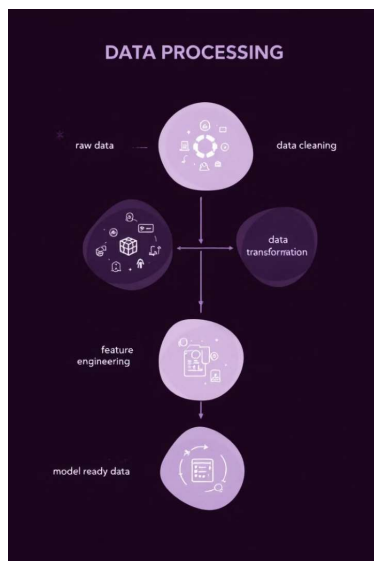
### Exemplos

Hadoop MapReduce processa dados em paralelo.  
Apache Spark acelera com processamento em memória.



## Pré-processamento de Dados

Preparar os dados para análise, garantindo que estejam limpos, formatados corretamente livres de ruídos.



## Técnicas de Pré-processamento de Dados

### Limpeza de Dados

Remoção de dados duplicados, valores nulos ou inconsistentes.

### Transformação de Dados

Normalização ou padronização de variáveis para escala similar.

### Extração de Características

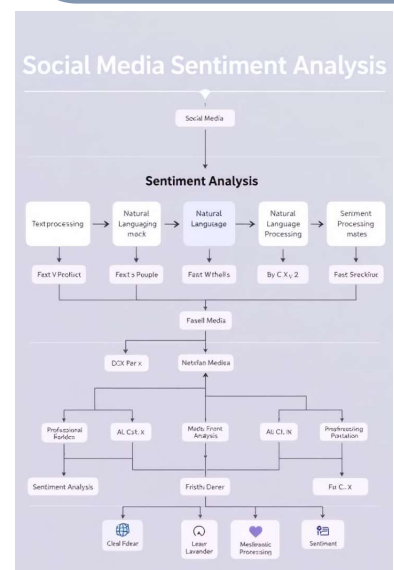
Criação de novas variáveis para melhorar o modelo.

### Redução de Dimensionalidade

Técnicas para reduzir variáveis mantendo a variabilidade.

## Exemplo de Pré-processamento de Dados

- 1 **Limpeza**  
Remover stopwords, URLs e caracteres especiais dos textos de redes sociais.
- 2 **Transformação**  
Converter texto em minúsculas e aplicar tokenização.
- 3 **Codificação**  
Transformar palavras em vetores numéricos (TF-IDF, Word2Vec).



## Técnicas de Armazenamento

### Em Blocos

Divisão em blocos distribuídos. Ex: HDFS.

### Em Objetos

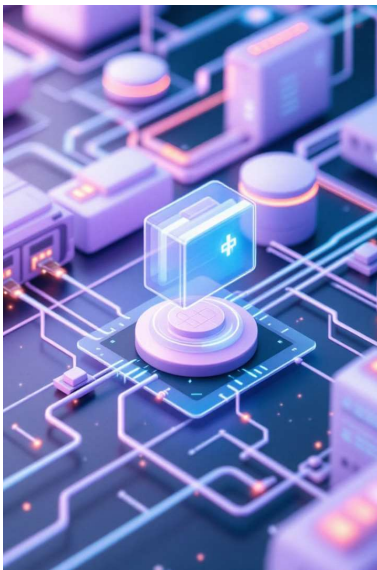
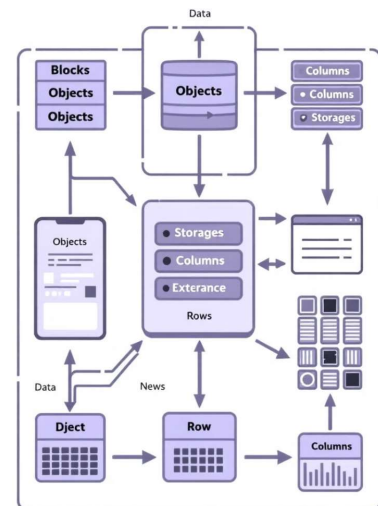
Ideal para dados não estruturados. Ex: Amazon S3.

### Em Colunas

Melhora desempenho analítico. Ex: Apache Parquet.

### Em Linha

Para transações rápidas. Ex: MySQL, PostgreSQL.



## O que é Armazenamento Distribuído?

Os dados são armazenados em múltiplos nós, garantindo alta disponibilidade e escalabilidade.

	DISTRIBUTED STORAGE MANAGER (ARCHITECTURE)	DISTRIBUTED STORAGE SYSTEMS	DATA DISTRIBUTION SYSTEM	DISTRIBUTED STORAGE ROUTER
Distributon	From destination storages	Delta Database	Local Guard	Data line
olopmanics	Database for active storages	Netas	Archiving	Inactive
Lotul	RTA	Neting Systems	Pre-ewary	Proctio
ributions	Comma-idea forrage systems	Compl-anitates byle?	Local-convaition	Localie Recheads
ataners	Prouced deapre decomposed converage (eppa and utra)	Not defect of communication recovery	Retri-fications crage, lation (recovery)	Deshtat-enduat
tabases	Position distrib-uge (converage inuacion)	Host-ic Edilidnary (converage display system)	Concecion partitionation	Method Formad
onifications	Centiona distapag & districion-actadman	Just inelictorage (actadman)	Premser Cupidisy	Unatid-c-ctadman
tababats	Contribution frauce of herida canaring	Real-Comunition	USA	Netas
ficurnies	Distributional-actadman (net)	Distributional-actadman (net)	Local-Platue-actadman	Net-ic-Dyatem-ic-actadman
riblerals-actadman	Versione-storage systems	Patl-icoms	Real-ic-Terica	Cent-ic-Local
Nades	Distributional-actadman (net)	New-Comer	Maxier	Now-ic-actadman
tabases	Distributional-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)
hibitation	Non-ic-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)
hibition	Compl-ic-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)
omwoies	Compl-ic-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)	Net-ic-actadman (net)

## Armazenamento Distribuído



HDFS

Sistema de arquivos distribuído do Hadoop. Armazena grandes volumes em múltiplos nós redundantes.



HBase

Banco NoSQL distribuído baseado no HDFS. Armazena dados em tabelas sem rigidez relacional.



Cassandra

Banco NoSQL orientado a colunas. Alta disponibilidade e tolerância a falhas.

## Comparação HDFS vs. HBase vs. Cassandra

Característica	HDFS	HBase	Cassandra
Modelo	Sistema de arquivos distribuído	NoSQL baseado em tabelas	NoSQL orientado a colunas
Escalabilidade	Horizontal	Horizontal automática	Alta horizontal
Tipo de Dados	Dados brutos	Dados estruturados	Dados estruturados

## Infraestrutura de TI

Na publicação (**A Infraestrutura para a Tecnologia da Informação: será que está correta?**)

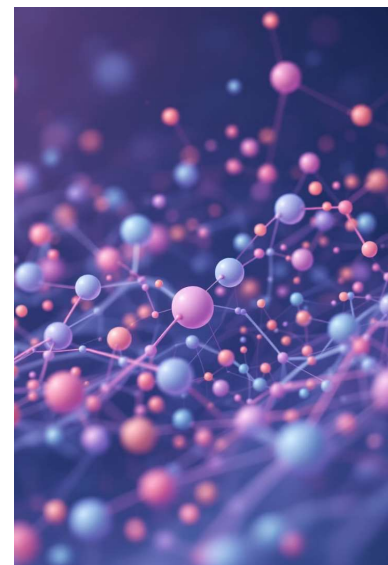
(link: <http://professorferlin.blogspot.com/2013/09/a-infraestrutura-para-tecnologia-da-3646.html>)

temos uma reflexão sobre Infraestrutura de TI nas empresas.



## O que é Indexação de Dados em Big Data?

Processo de criação de estruturas para acelerar consultas sobre grandes volumes de dados.



## Indexação de Dados em Big Data

### Objetivo

Melhorar o tempo de resposta e a eficiência na consulta de grandes volumes de dados.

### Técnicas

Índices B-tree e hash: Utilizados para melhorar a busca de dados em sistemas de arquivos ou bancos de dados.

Indexação invertida: Utilizada em sistemas de busca e mineração de texto. Aplicada em buscas textuais (Elasticsearch, Solr).

Índices compostos: Combinam múltiplos campos de dados para melhorar o desempenho das consultas.

## Ferramentas de Indexação de Dados

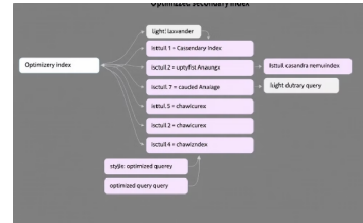
**Apache Solr:** Plataforma de busca baseada em Apache Lucene, especializada em indexação e busca de grandes volumes de dados.

**Elasticsearch:** Sistema de busca distribuído que permite indexação e consulta eficiente de dados em tempo real.

## Exemplo de Indexação de Dados

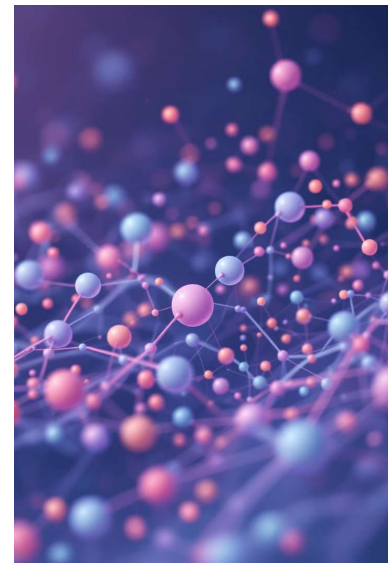
### Indexação em Cassandra

Utiliza índices secundários para melhorar o desempenho das consultas em campos não primários.



## O que é Compressão de Dados em Big Data?

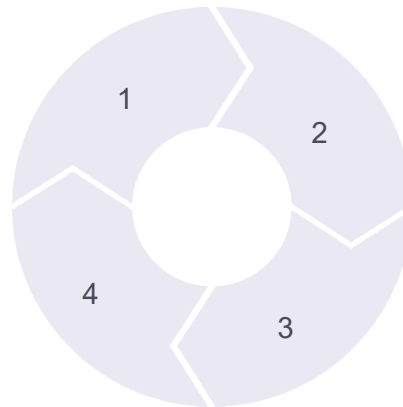
Redução do tamanho dos arquivos para economizar espaço e melhorar a eficiência da transmissão de dados.



## Compressão de Dados em Big Data

### Objetivo

Reduzir volume de dados,  
economizar espaço e otimizar  
transferências.



### Benefícios

Redução de custos e melhor  
desempenho na leitura e transmissão.

## Técnicas de Compressão de Dados

### Sem Perda (Lossless)

Mantêm 100% da integridade dos  
dados.

Exemplos: Gzip, Snappy, LZO

### Com Perda (Lossy)

Remove informação irrelevante para  
reduzir tamanho.

Exemplos: JPEG, MP3, MPEG

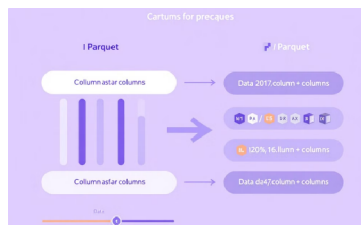
## Ferramentas de Compressão de Dados

**Snappy:** Algoritmo de compressão rápido e eficiente, usado em sistemas como Apache Hadoop e Apache Cassandra.

**Gzip:** Algoritmo comum para compressão de arquivos de texto.

**Brotli:** Algoritmo de compressão mais eficiente, usado em navegadores web e servidores.

## Exemplo de Compressão de Dados



### Compressão com Parquet

Formato de armazenamento columnar que usa compressão para armazenar grandes volumes eficientemente.

## Combinando Indexação e Compressão

A combinação de indexação e compressão otimiza tanto o armazenamento quanto a recuperação.



## Contato



[eferlin@live.com](mailto:eferlin@live.com)



(BLOG) [professorferlin.blogspot.com](http://professorferlin.blogspot.com)

(SITE) [professorferlin.com.br](http://professorferlin.com.br)

(YOUTUBE) [ProfEdsonPedroFerlin](https://www.youtube.com/ProfEdsonPedroFerlin)