



INSIGHT



hadoop

HADOOP

Também pode conter um subtítulo logo abaixo

OLÁ!

Sou Nicksson,

Atualmente, sou aluno do PPGCC-Doutorado na UFC e orientando do prof. Macêdo.



Vocês podem me encontrar em:

- nickssonarrais@gmail.com

PLANEJAMENTO DAS AULAS

- **16/05/2019** - HADOOP/MAPREDUCE
- **21/05/2019** - HADOOP/HDFS E YARN
- **23/05/2019** - AULA PRÁTICA

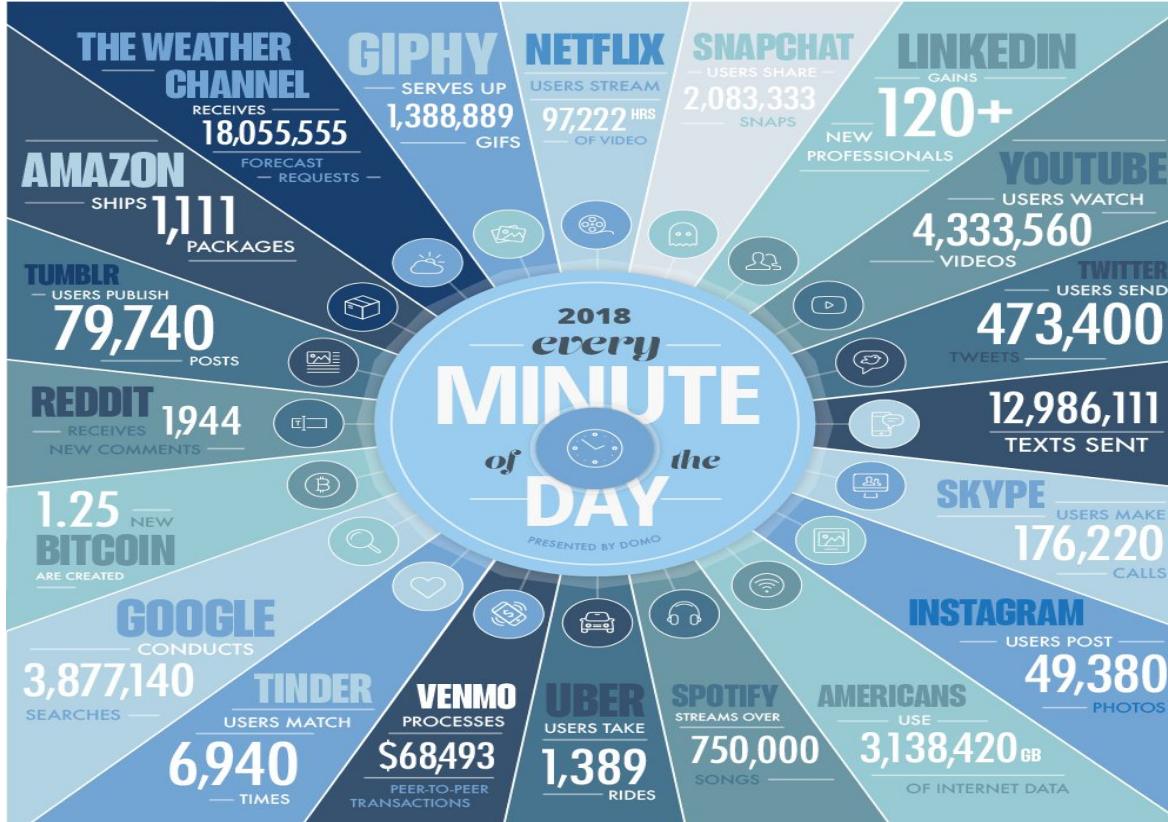


AGENDA

1. Introdução à Big Data
2. Hadoop
3. MapReduce
4. Exercício prático

1. O QUE É BIG DATA?

INTRODUÇÃO



CARACTERÍSTICAS BIG DATA

- Big data é o termo utilizado para enormes e complexos conjuntos de dados que são difíceis de processar usando aplicações de processamento de dados tradicionais.



CARACTERÍSTICAS DOS DADOS

- Dados estruturados:
 - Dados relacionais (esquema fixo);
- Dados semi-estruturados:
 - XML, Json;
- Dados não estruturados (normalmente, são a maioria):
 - PDF, Texto, Fotos, Vídeos, Word, Logs.

CARACTERÍSTICAS BIG DATA



MOTIVAÇÃO

- Big Data está em todo lugar:
 - Sistemas de recomendação Netflix e Amazon;
 - Mídias sociais como Facebook, Twitter, Youtube, Instagram, etc;
 - Gmail;
 - Uber;
 - Games onlines;
 - Governo;
 - Empresas financeiras.

DESAFIOS BIG DATA

- Captura de dados;
- Recuperação;
- Consulta;
- Compartilhamento;
- Análise;
- Visualização;
- etc.

FERRAMENTAS PARA BIG DATA



2. HADOOP

APACHE HADOOP

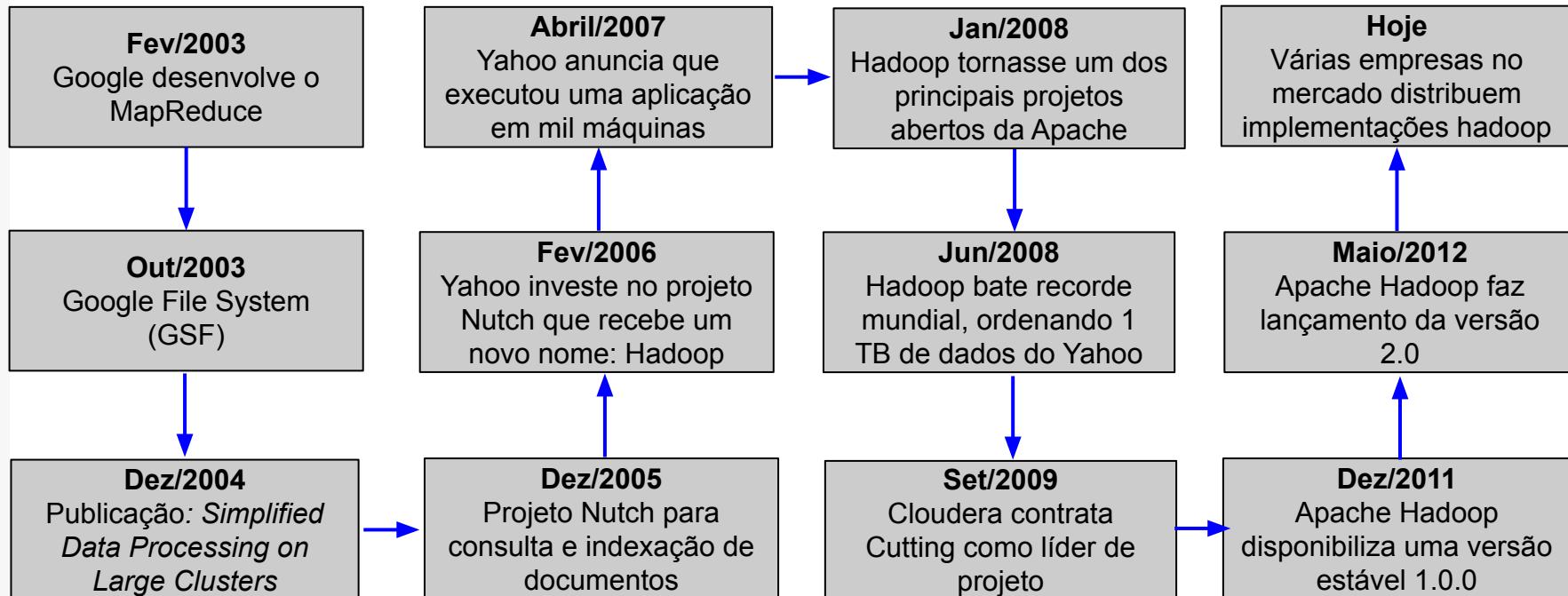
- Hadoop é uma plataforma de código aberto que fornece soluções para **armazenar** e **processar** grandes conjuntos de dados distribuídos, utilizando *clusters* de computadores com **hardware comum**.

HADOOP COMO SOLUÇÃO

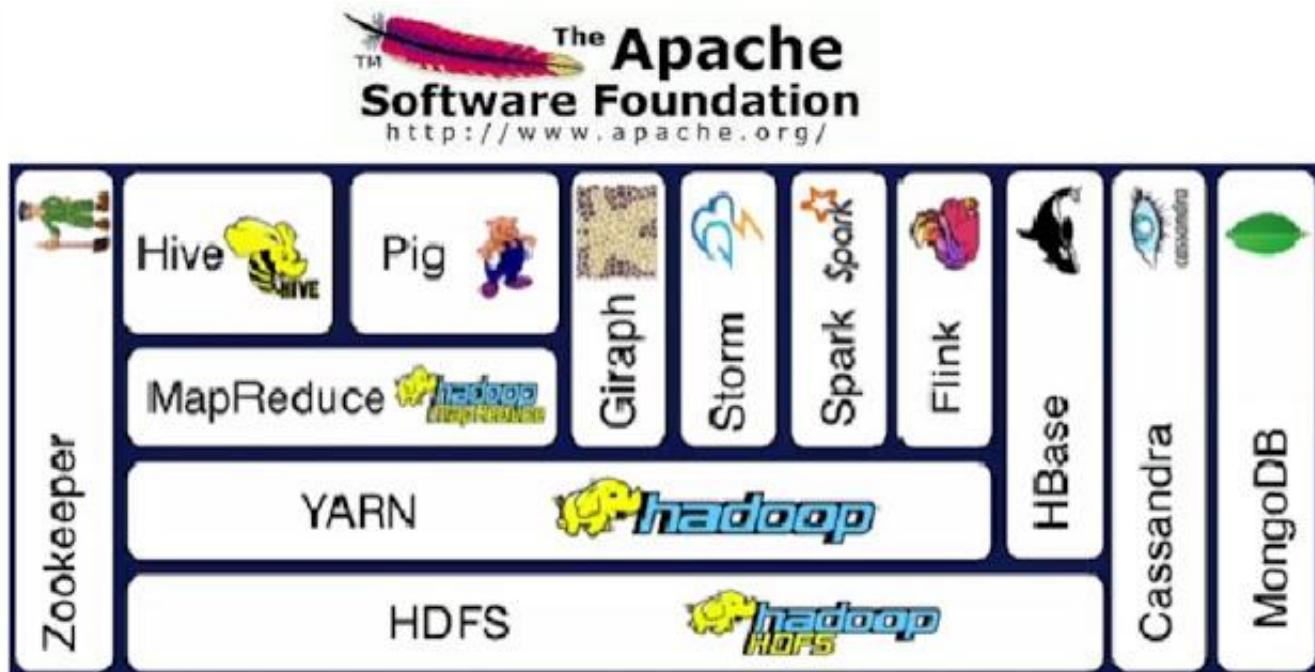
- Confiabilidade:
 - tolerância a falhas;
- Flexibilidade:
 - lida com todos os tipos de dados;
- Econômico:
 - demanda um hardware simples;
- Escalável:
 - integração a serviços nas nuvens.

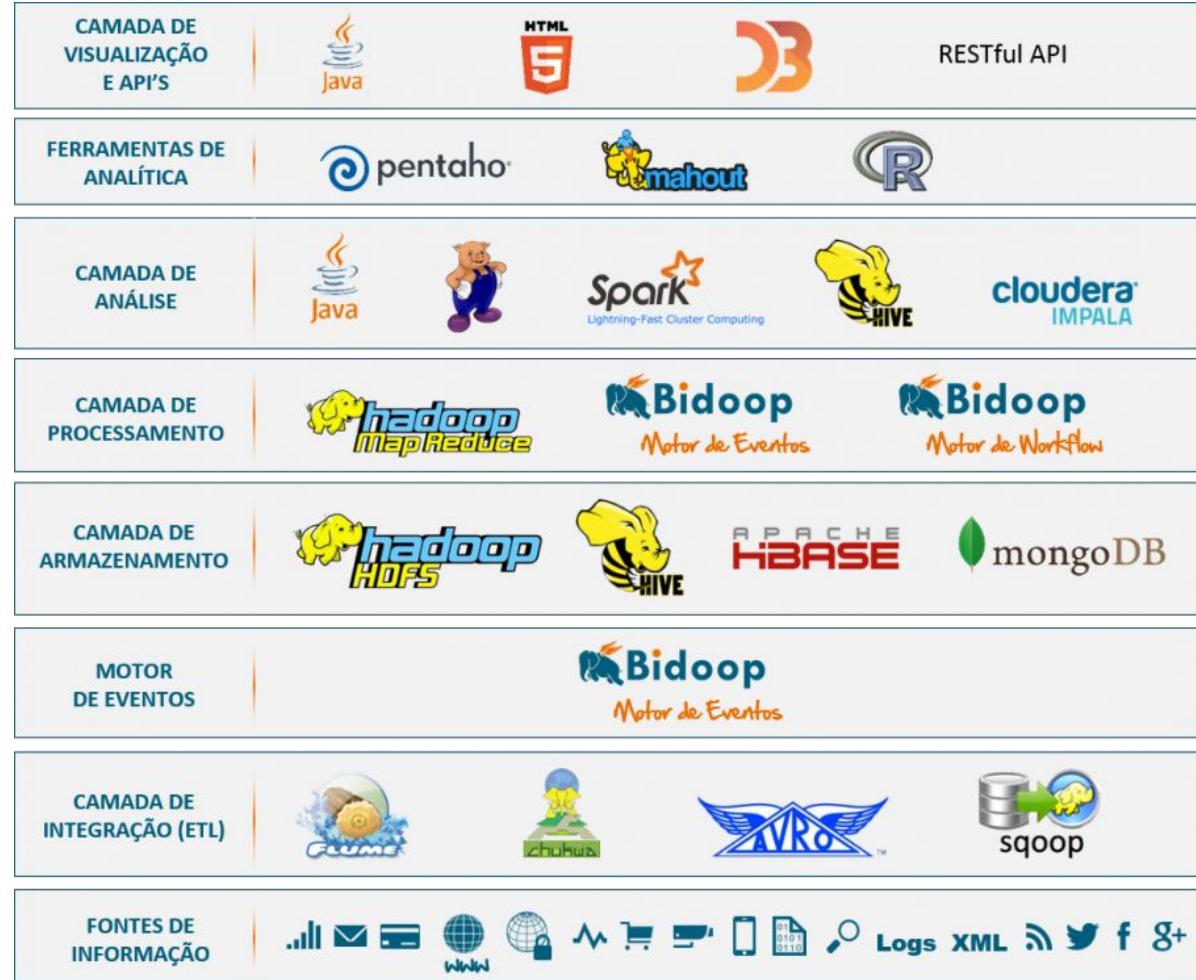


HISTÓRIA



ECOSSISTEMA HADOOP





MÓDULOS PRINCIPAIS



Hadoop v1.0

MapReduce
Data Processing
& Resource Management

HDFS
Distributed File Storage



Hadoop v2.0

MapReduce

**Other Data
Processing
Frameworks**

YARN

Resource Management

HDFS

Distributed File Storage

3. MAPREDUCE

PROBLEMA

- A capacidade de armazenamento nos discos aumentou consideravelmente, mas a velocidade de acesso aos dados não aumentou na mesma proporção.



PROBLEMA

- Em 1990:
 - armazenavam 1370 MB de dados;
 - taxa de transferência de 4,4 MB/s;
 - 5 minutos para uma leitura completa.

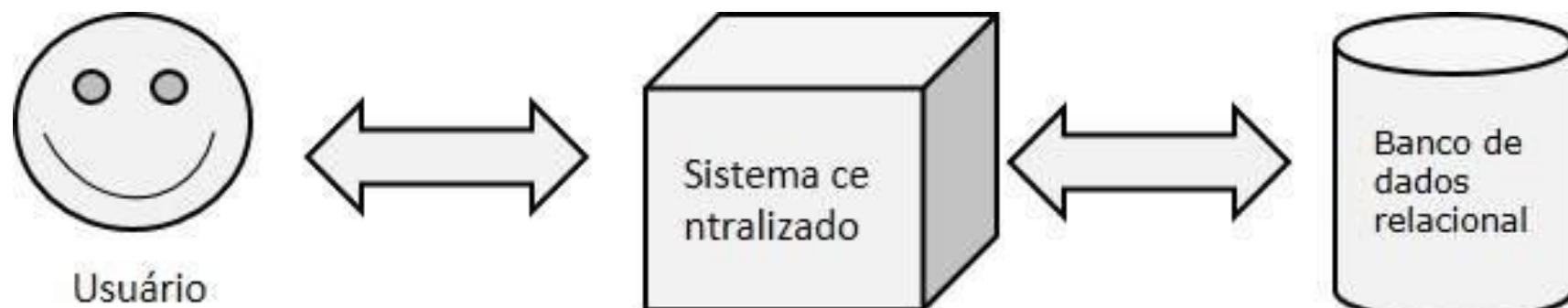


PROBLEMA

- Em 1990:
 - armazenavam 1370 MB de dados;
 - taxa de transferência de 4,4 MB/s;
 - 5 minutos para uma leitura completa.
- Em 2010:
 - armazenavam um terabyte;
 - taxa de transferência de 100MB/s;
 - horas para uma leitura completa.



SISTEMAS TRADICIONAIS



MAPREDUCE

- Um modelo de programação, introduzido pela Google, para suportar computação paralela e distribuída em grandes coleções de dados;

MAPREDUCE

- Um modelo de programação, introduzido pela Google, para suportar computação paralela e distribuída em grandes coleções de dados;
- O modelo abstraí o problema de leitura e escrita no disco, transformando os dados em um conjunto de **chave** e **valor**;

MAPREDUCE

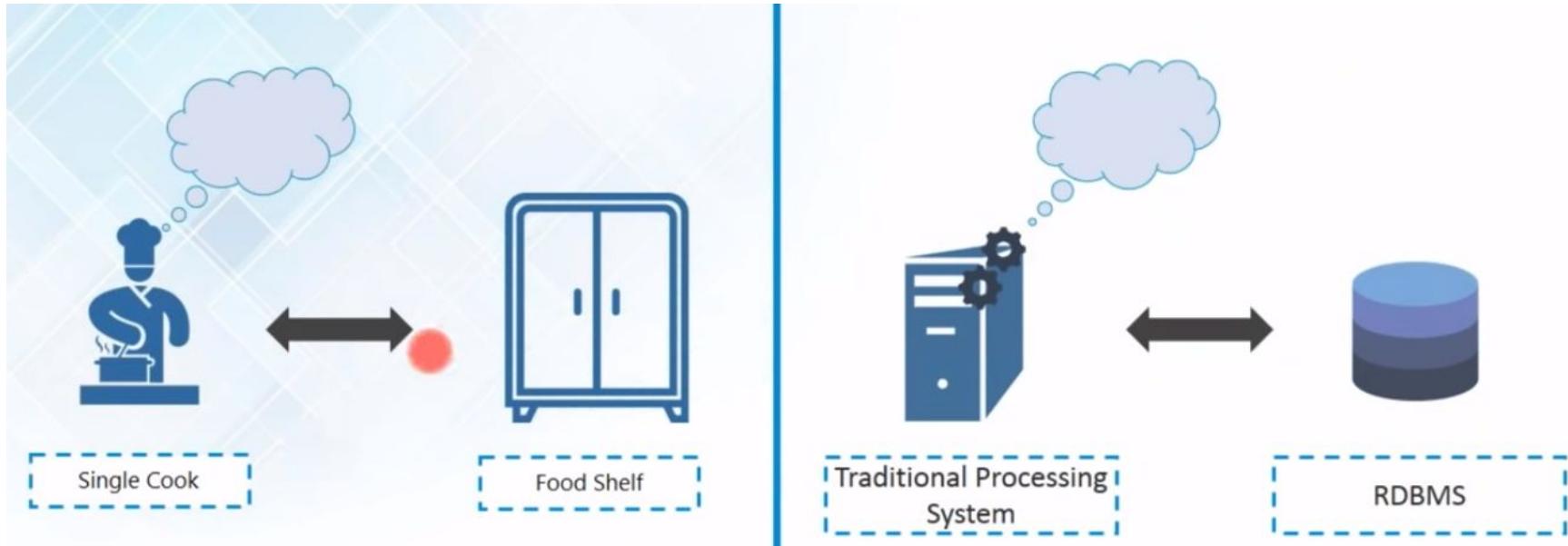
- Um modelo de programação, introduzido pela Google, para suportar computação paralela e distribuída em grandes coleções de dados;
- O modelo abstraí o problema de leitura e escrita no disco, transformando os dados em um conjunto de **chave** e **valor**;
- Um ponto importante é a interface entre o **map** e o **reduce**, inspirada pelas mesmas funções usadas na programação funcional (Lisp).

BIG DATA E OS SISTEMAS TRADICIONAIS



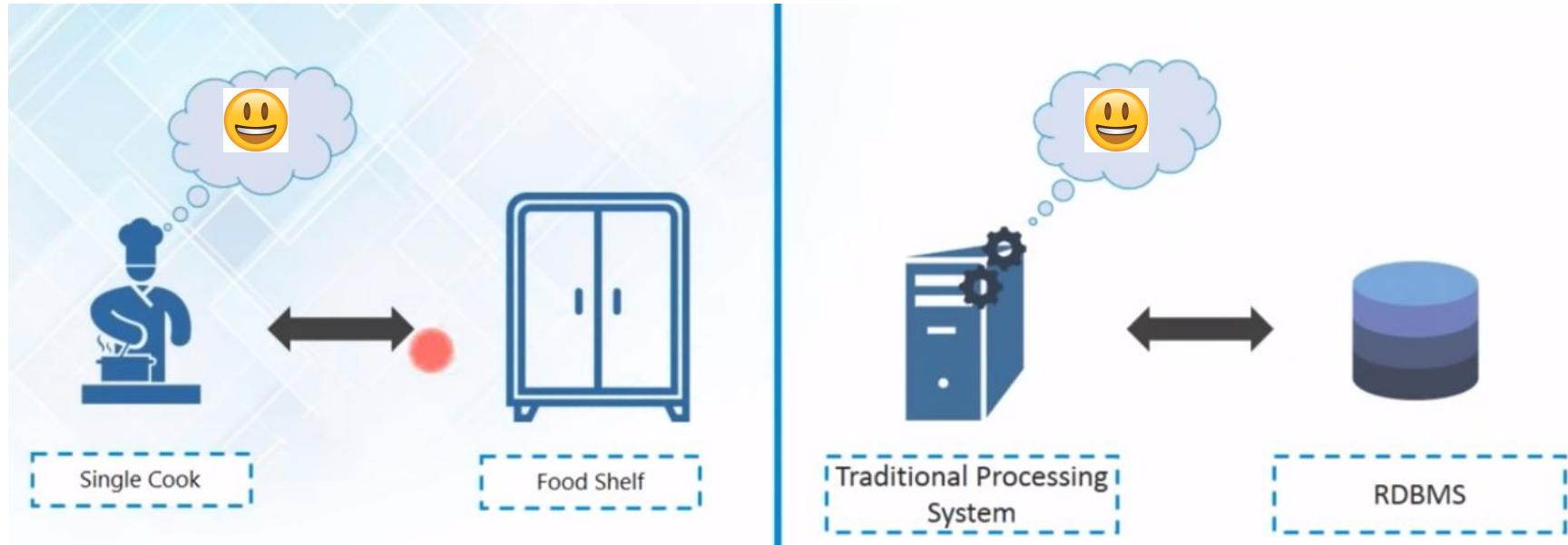
CENÁRIO TRADICIONAL

- Dois pedidos por hora.
- Dados estruturados gerados a uma taxa regular.



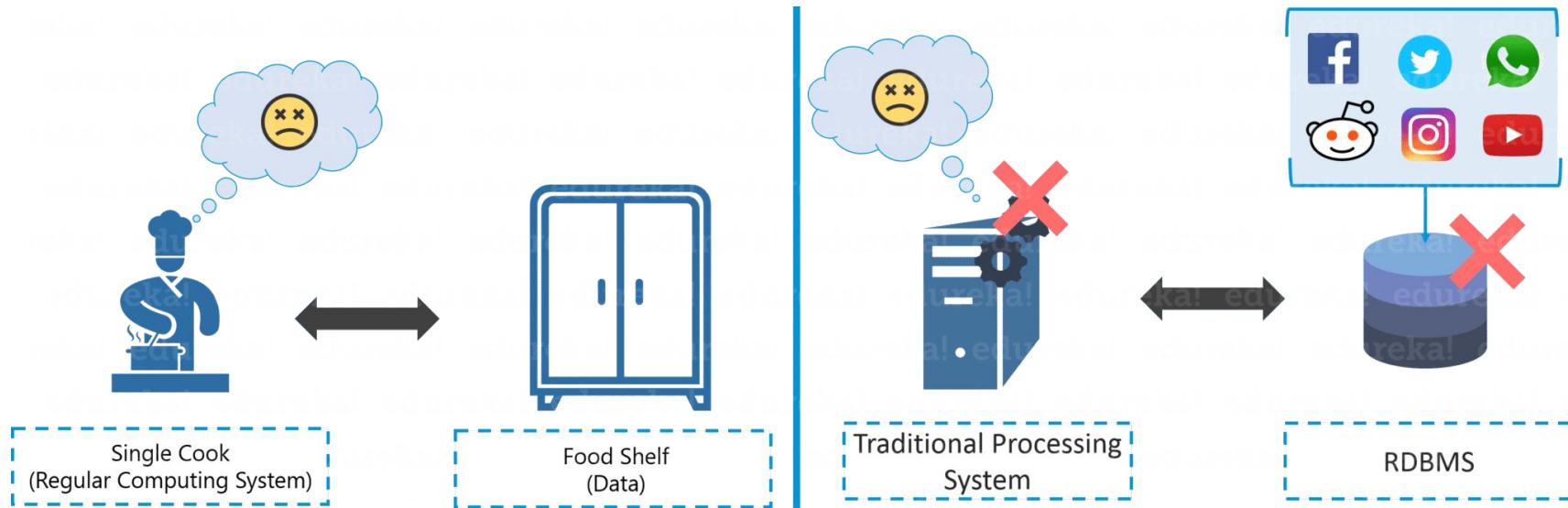
CENÁRIO TRADICIONAL

- Dois pedidos por hora.
- Dados estruturados gerados a uma taxa regular.



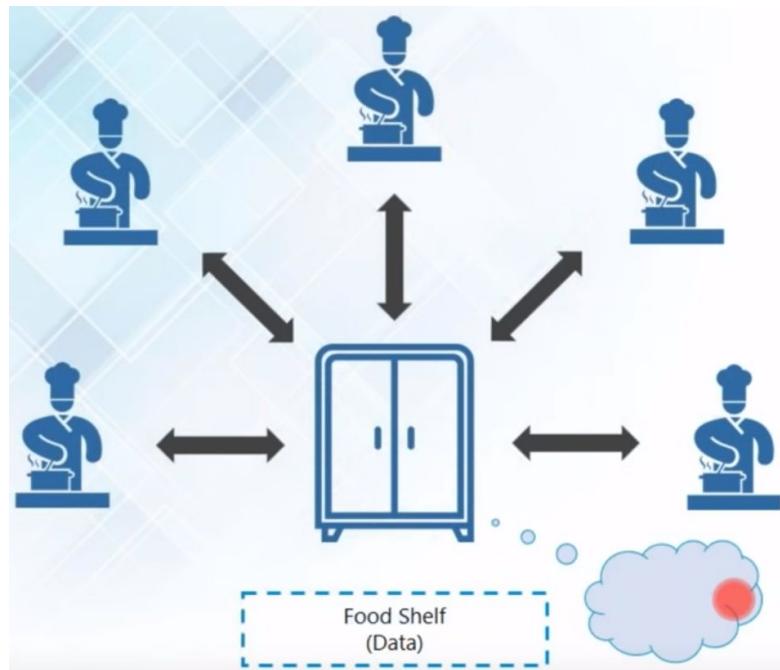
CENÁRIO TRADICIONAL

- 10 pedidos **online** por hora;
- Dados heterogêneos, em grande parte não estruturados, gerados numa taxa alarmante;



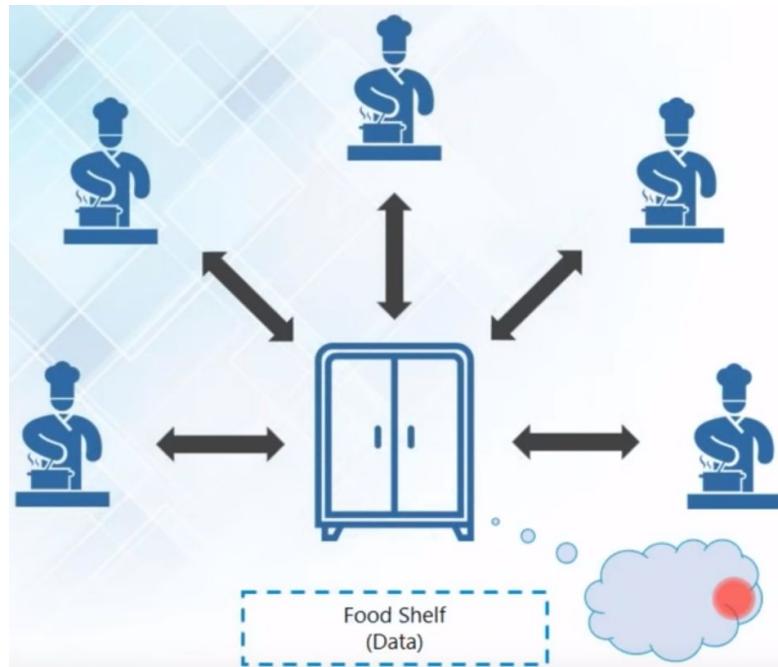
SOLUÇÃO INICIAL - DISTRIBUIR TAREFAS

- Múltiplos cozinheiros;



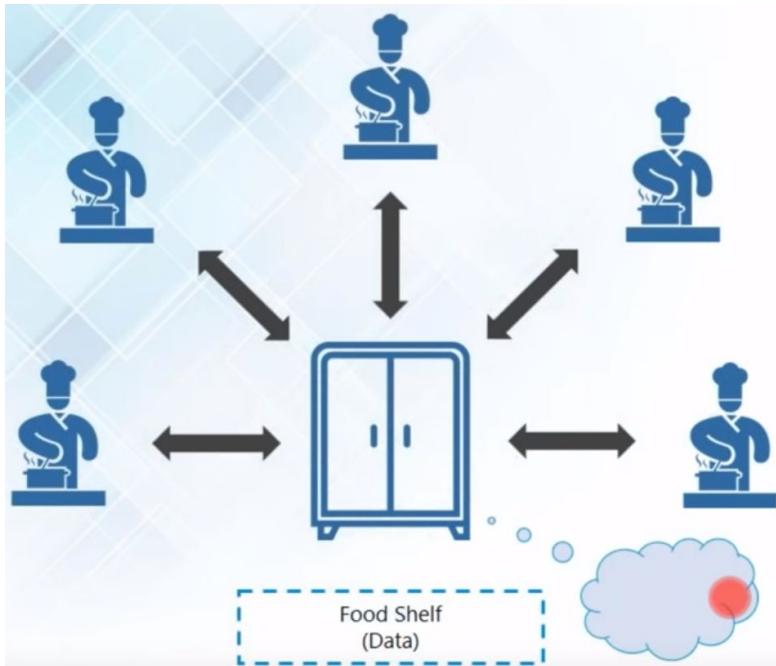
SOLUÇÃO INICIAL - DISTRIBUIR TAREFAS

- Múltiplos cozinheiros;
- ***Delay no acesso aos alimentos:***



SOLUÇÃO INICIAL - DISTRIBUIR TAREFAS

- Múltiplos cozinheiros;
- ***Delay no acesso aos alimentos:***

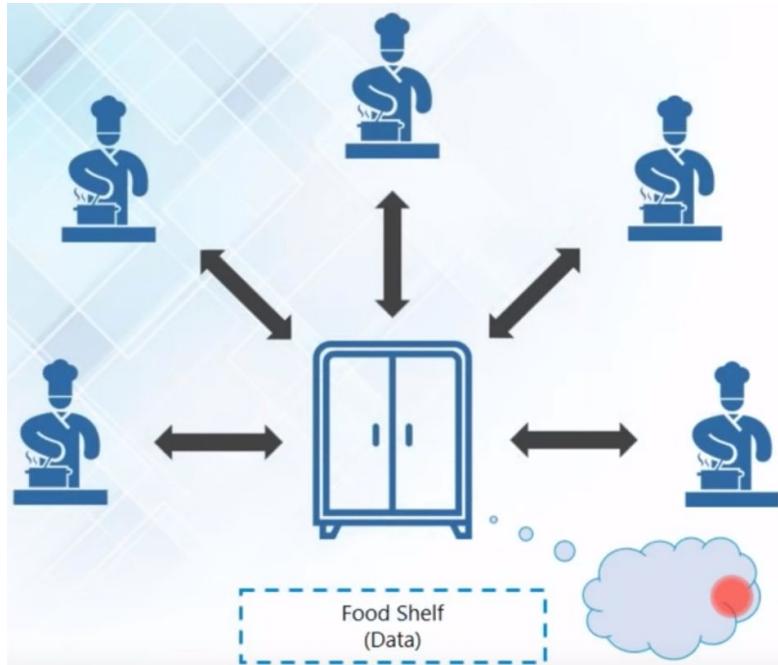


- Múltiplas máquinas e uma base de dados;



SOLUÇÃO INICIAL - DISTRIBUIR TAREFAS

- Múltiplos cozinheiros;
- ***Delay no acesso aos alimentos;***

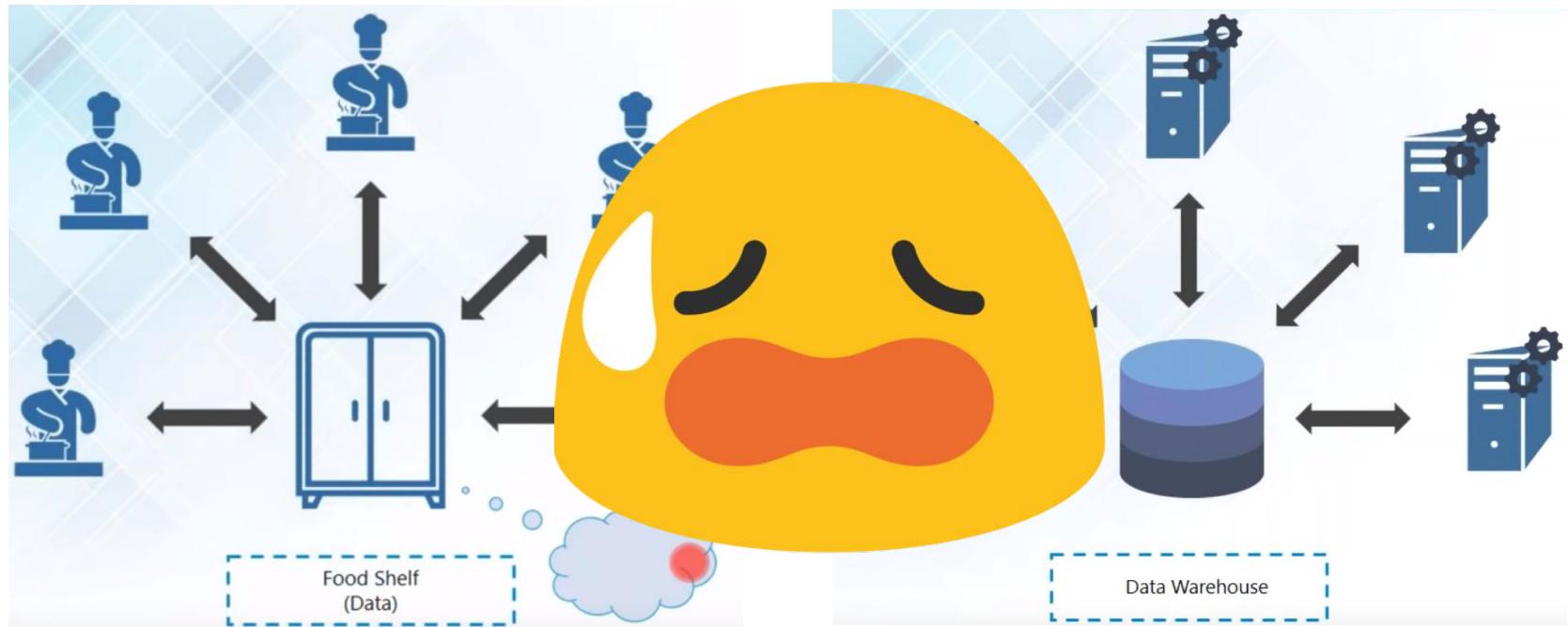


- Múltiplas máquinas e uma base de dados;
- ***Overhead na rede;***

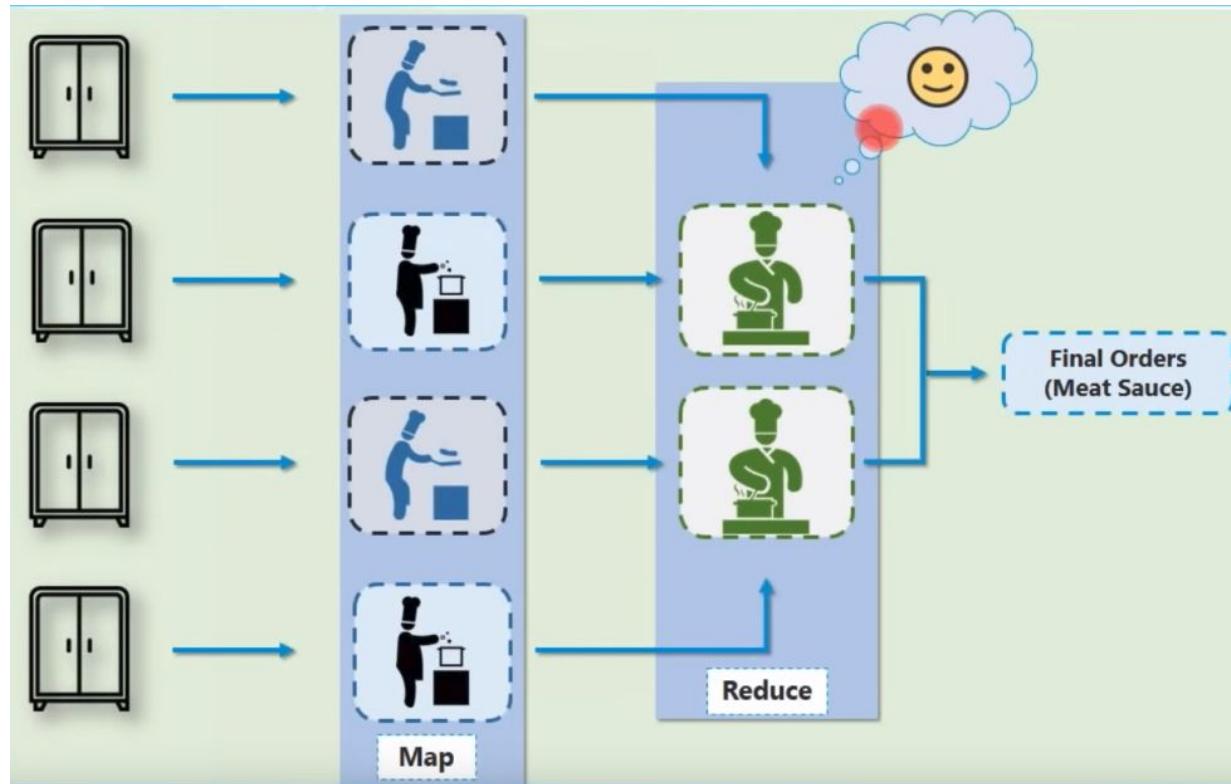


SOLUÇÃO INICIAL - DISTRIBUIR TAREFAS

- Múltiplos cozinheiros;
- ***Delay no acesso aos alimentos;***
- Múltiplas máquinas e uma base de dados;
- ***Overhead na rede;***

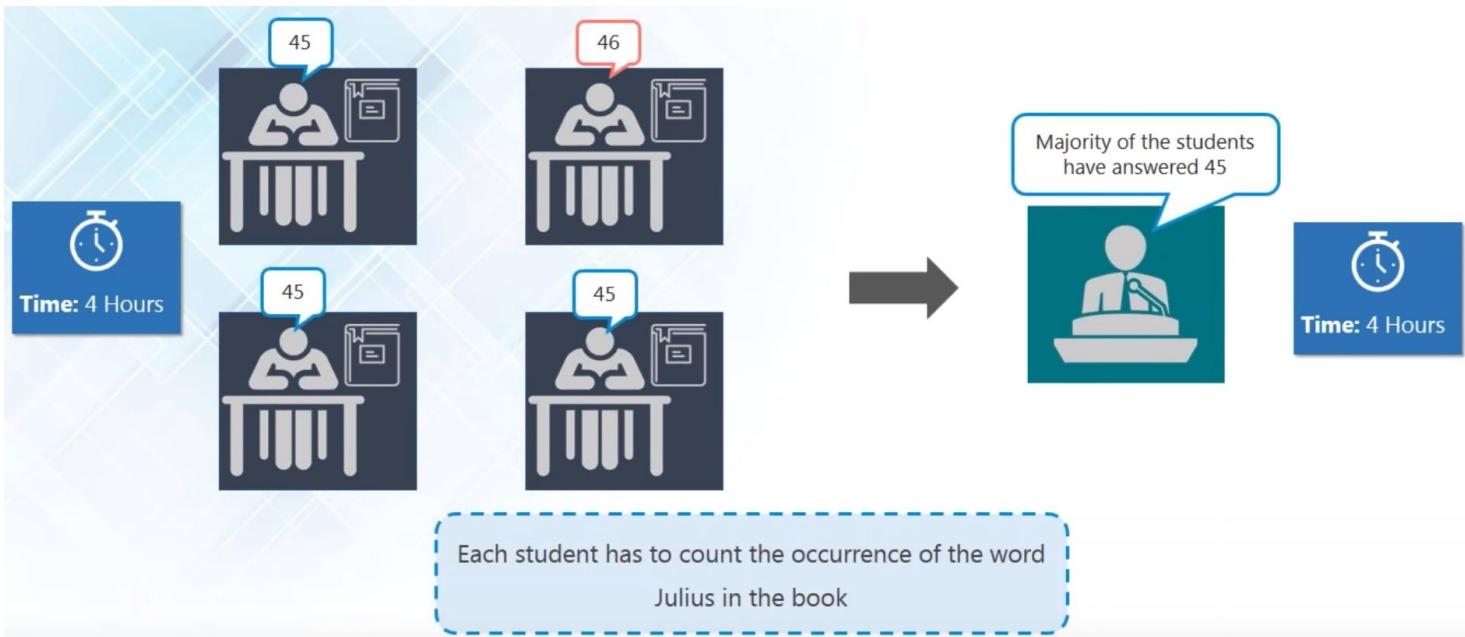


SOLUÇÃO EFETIVA - DISTRIBUIR E PARALELIZAR TAREFAS

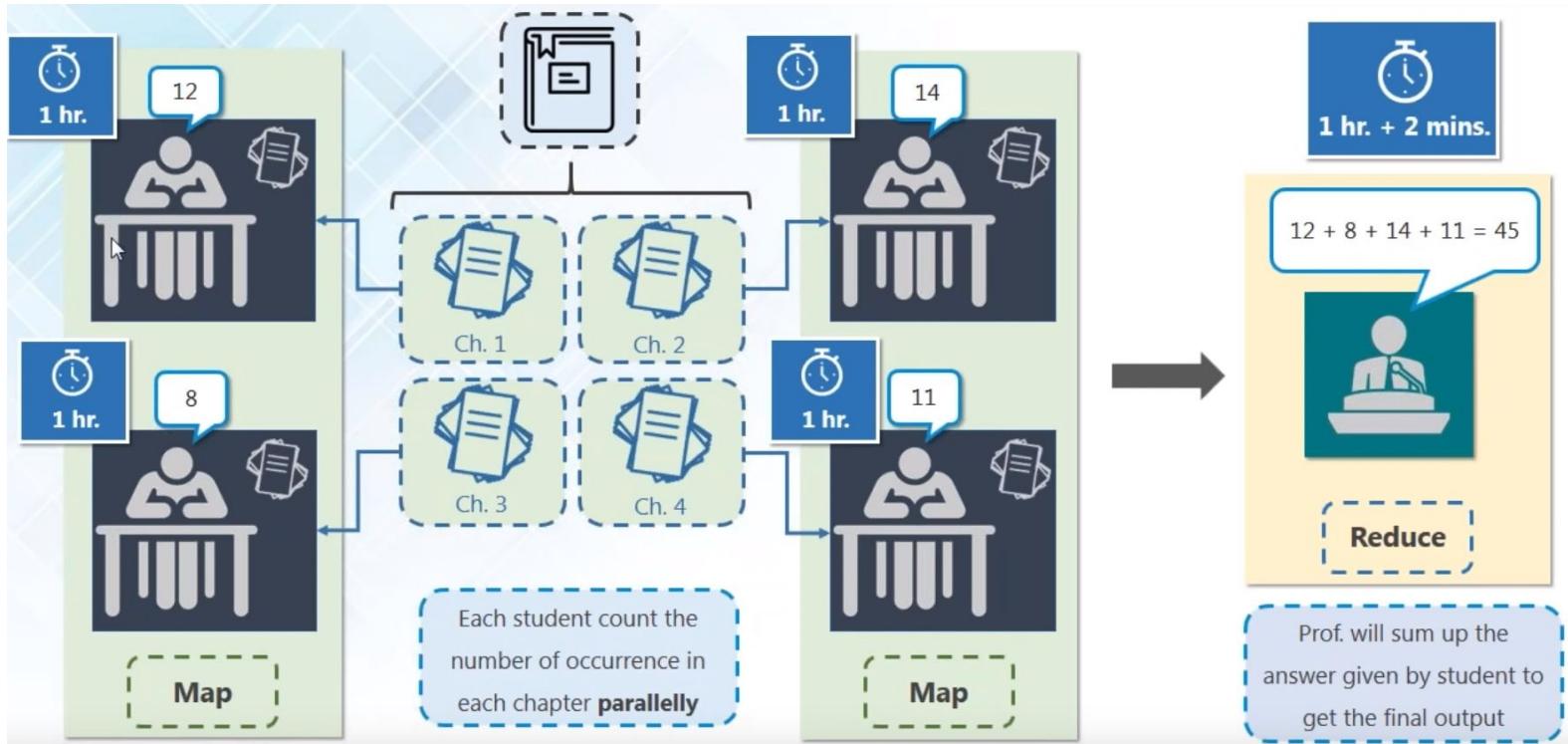


SOLUÇÃO EFETIVA - Exemplo 2

- Estudante precisam contar um determinada palavra em um livro;



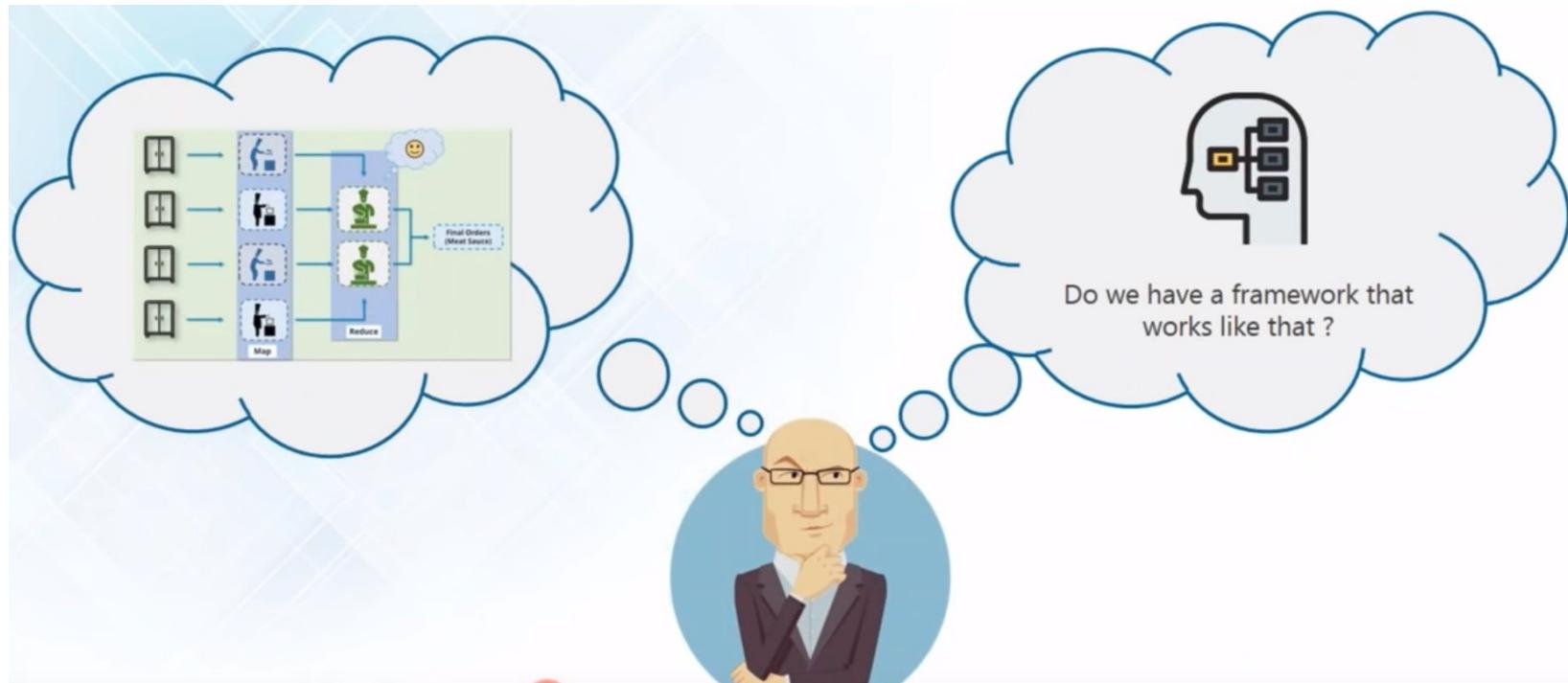
SOLUÇÃO EFETIVA - Exemplo 2



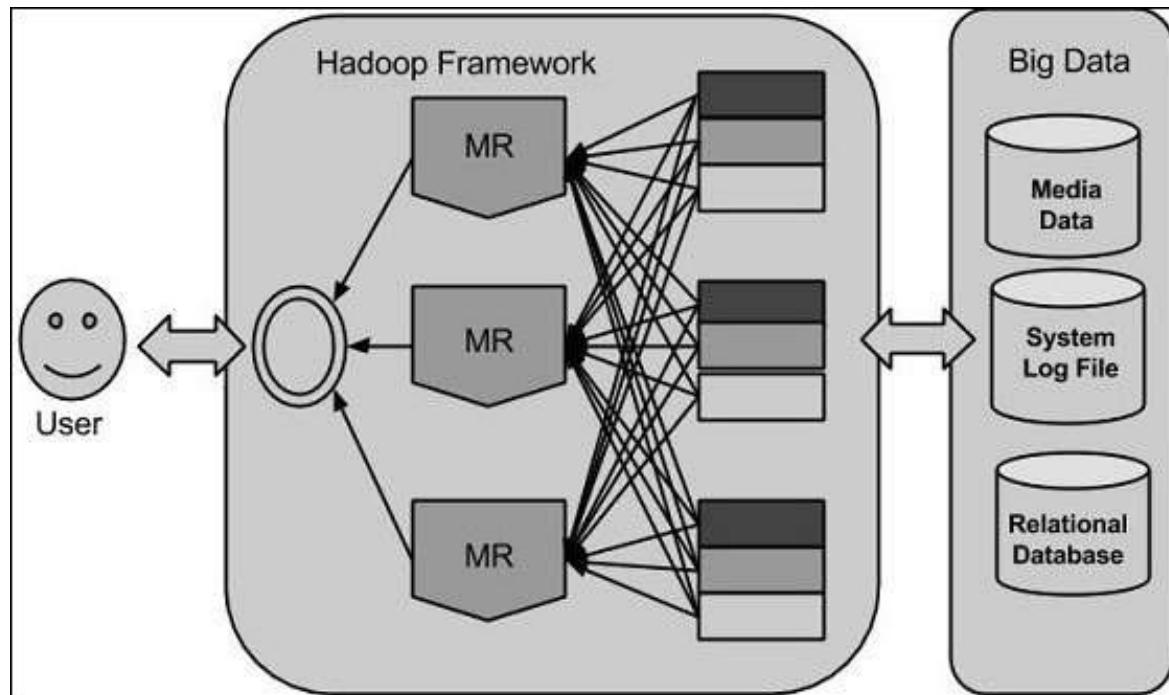
PROBLEMAS NA SOLUÇÃO EFETIVA

- Falhas de hardware:
 - Replicar dados (cópias redundantes);
- Combinação de dados de discos diferentes:
 - Alguns sistemas de arquivos permitem que dados sejam combinados a partir de várias fontes, mas isso ainda é considerado desafiador.

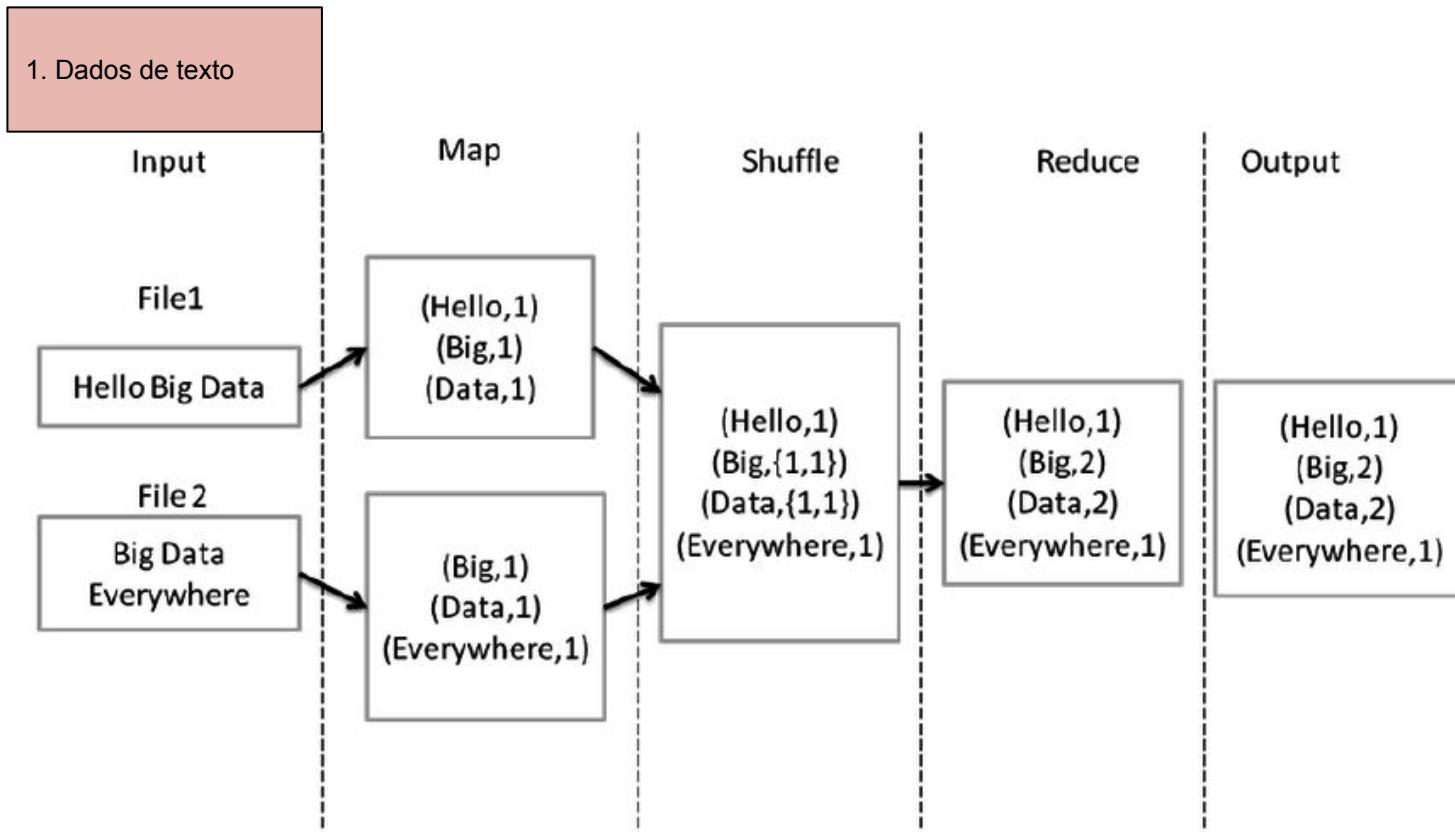
SOLUÇÃO EFETIVA - DISTRIBUIR E PARALELIZAR TAREFAS



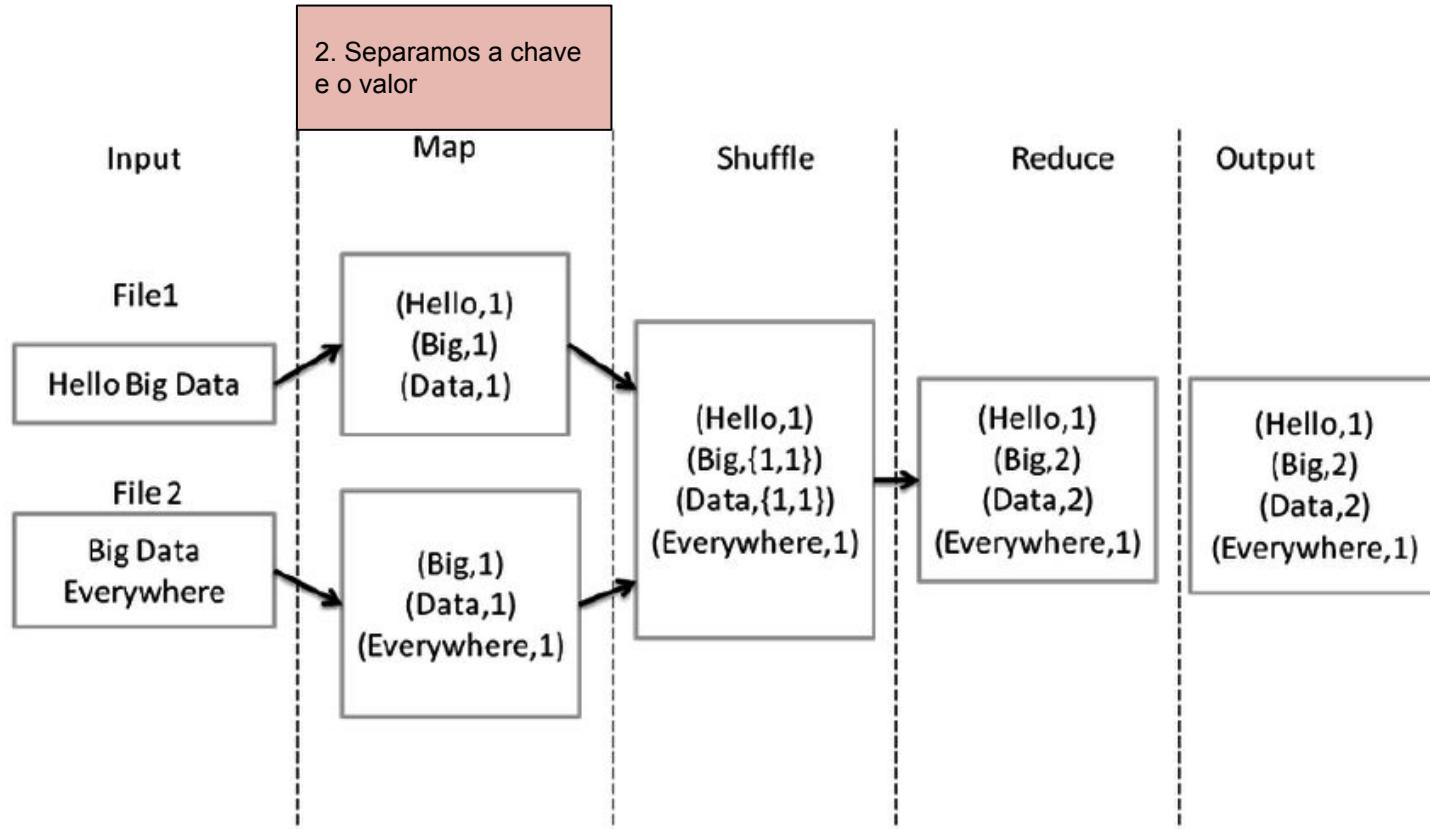
SISTEMA PROPOSTO



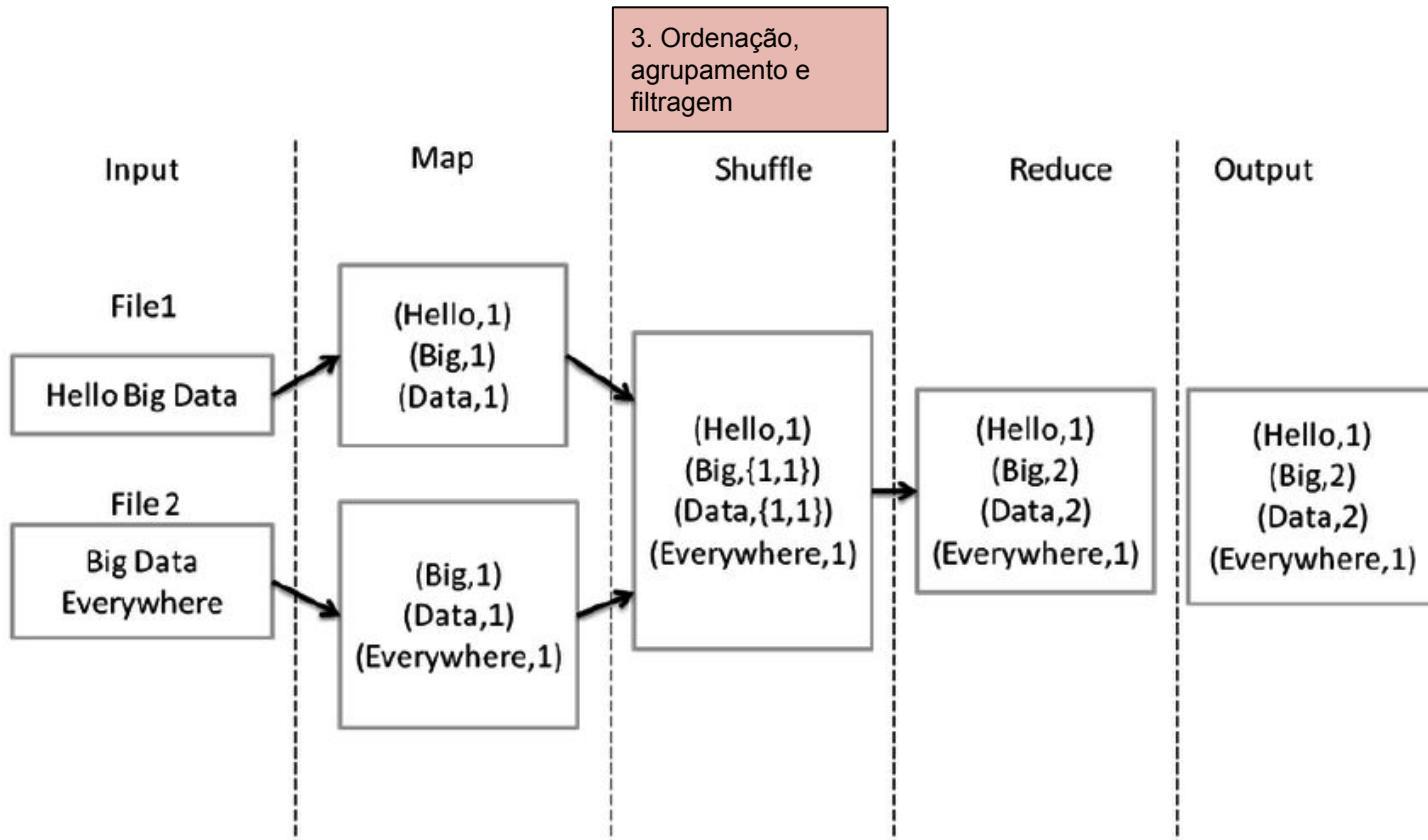
MAP + REDUCE - Exemplo 01



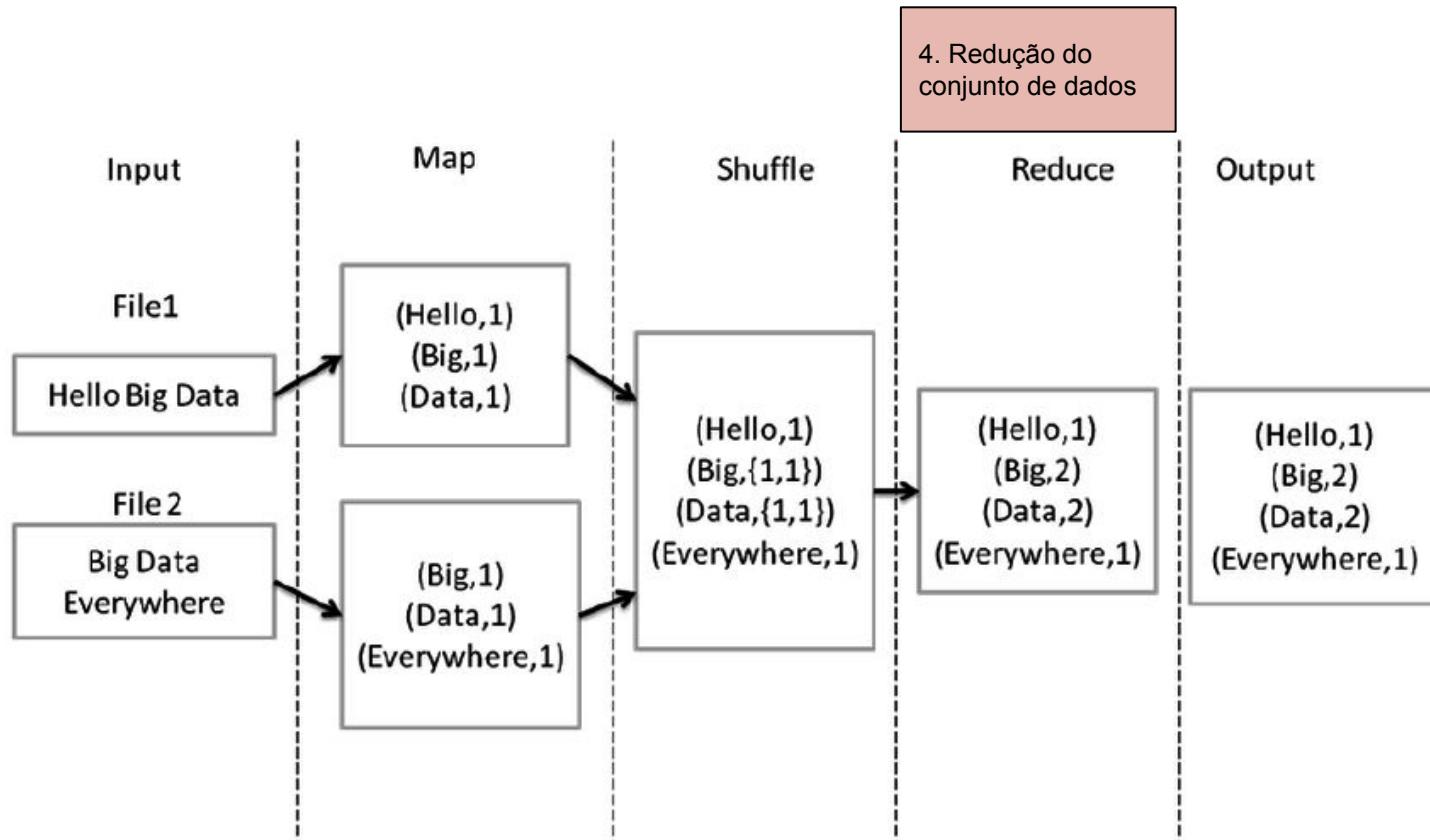
MAP + REDUCE - Exemplo 01



MAP + REDUCE - Exemplo 01

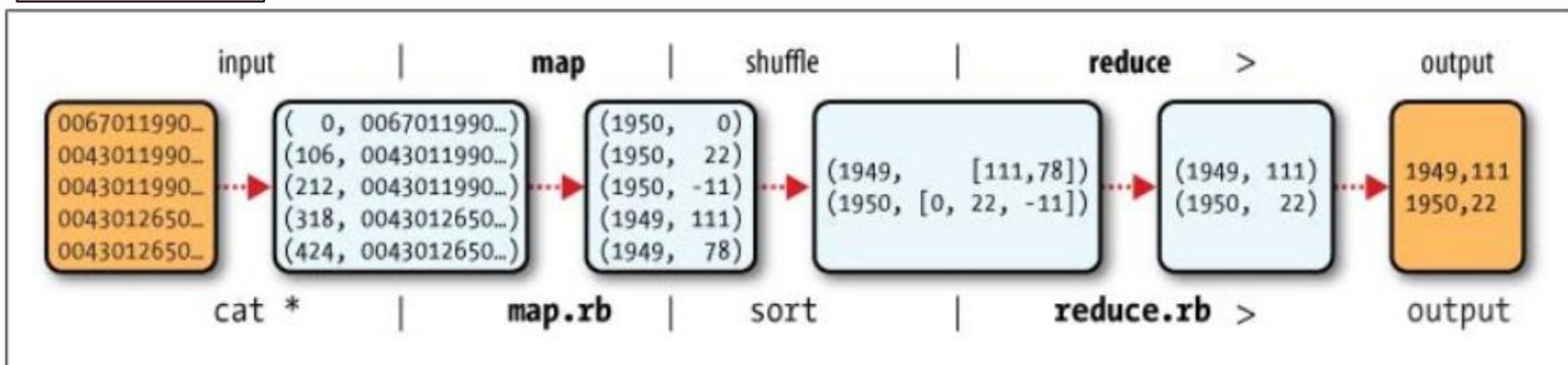


MAP + REDUCE - Exemplo 01



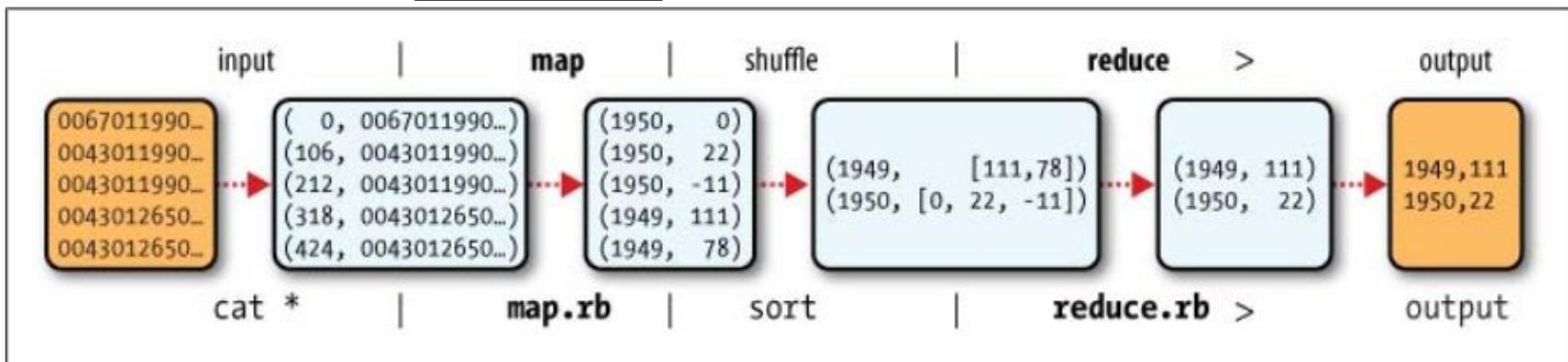
MAP + REDUCE - Exemplo 02

1. Dados Meteorológicos

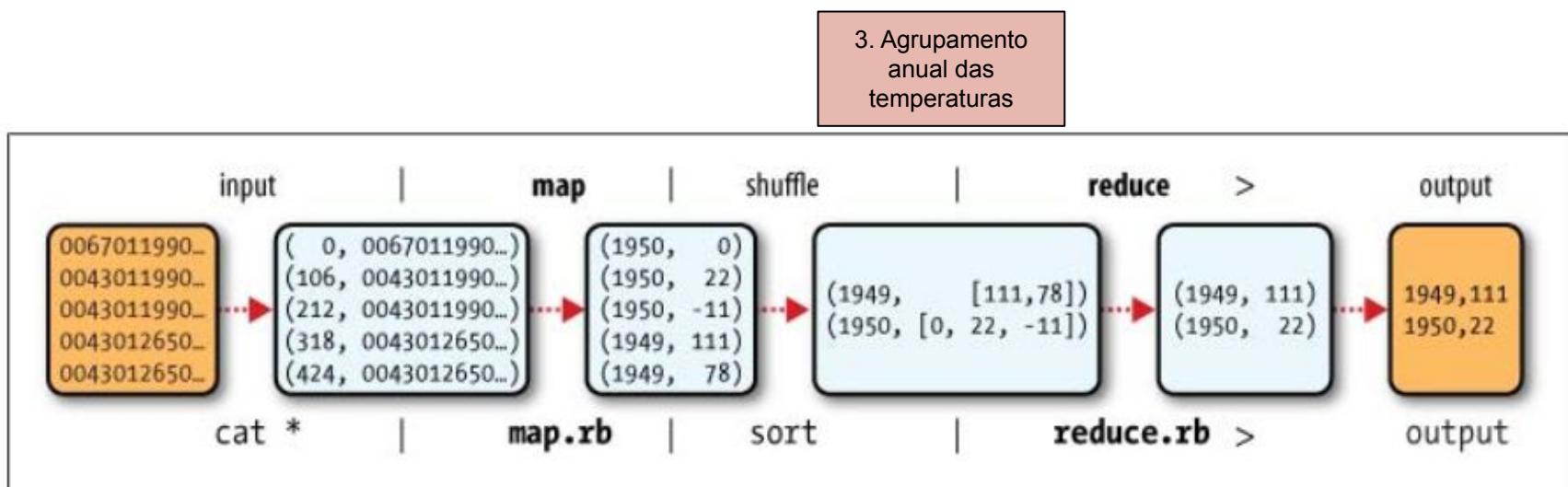


MAP + REDUCE - Exemplo 02

2. Defino a chave e o valor

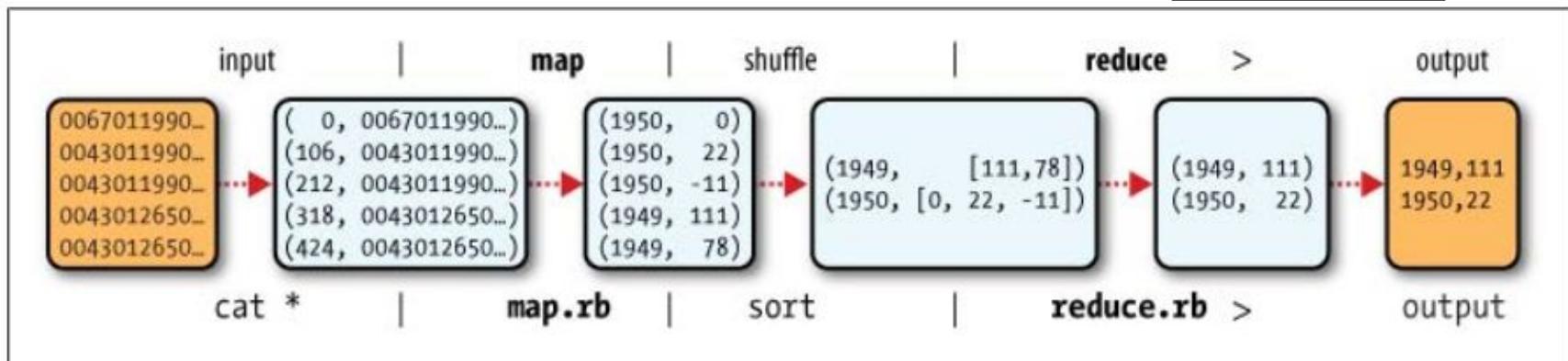


MAP + REDUCE - Exemplo 02



MAP + REDUCE - Exemplo 02

4. Reduzo o conjunto
separando a
temperatura máxima



COMPARAÇÃO COM RDBMS

- MapReduce:
 - analisar um banco de dados completamente de forma descontínua;
 - atende aplicações onde dados são gravados uma única vez e lidos muitas vezes;
 - suporta dados semi-estruturados e não estruturados, uma vez que a chave é definida pelo programador.
- RDBMS:
 - consultas e atualização em baixa latência em pequenos subconjunto de dados;
 - dados gravados e consultados muitas vezes;
 - dados estruturados.

COMPARAÇÃO COM RDBMS

	Traditional RDBMS	MapReduce
Data size	Gigabytes	Petabytes
Access	Interactive and batch	Batch
Updates	Read and write many times	Write once, read many times
Structure	Static schema	Dynamic schema
Integrity	High	Low
Scaling	Nonlinear	Linear

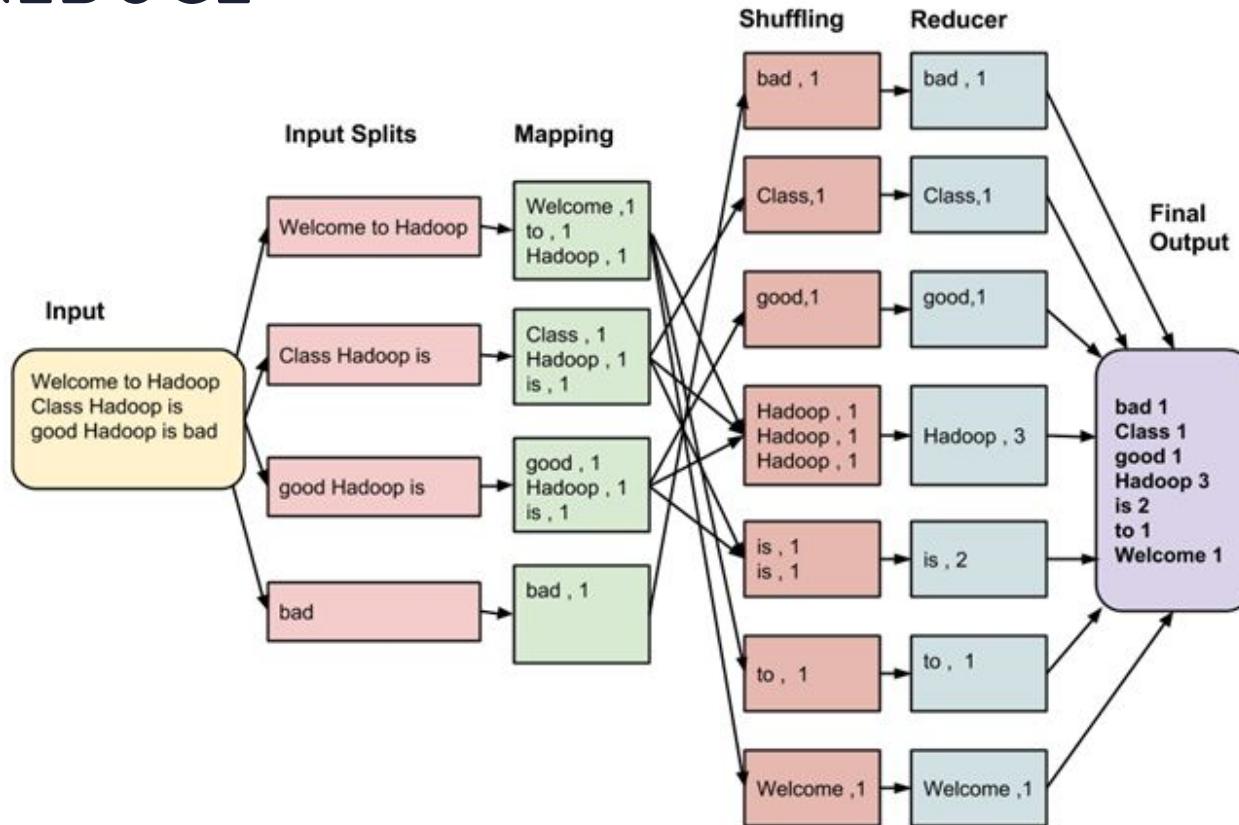
4. EXERCÍCIO PRÁTICO

MAPREDUCE

1. Abram o projeto /aula01-map-reduce/wordCount.ipynb

2. Enviar a atividade até o próximo **domingo 18/05**.

MAPREDUCE



REFERÊNCIAS PRINCIPAIS

- WHITE, Tom. **Hadoop: The definitive guide.** " O'Reilly Media, Inc.", 2012.
- RADTKA, Zachary; MINER, Donald. **Hadoop with Python.** O'Reilly Media, 2015.
- CHU, Cheng-Tao et al. Map-reduce for machine learning on multicore. In: **Advances in neural information processing systems.** 2007. p. 281-288.
- DEAN, Jeffrey; GHEMAWAT, Sanjay. MapReduce: a flexible data processing tool. **Communications of the ACM**, v. 53, n. 1, p. 72-77, 2010.

REFERÊNCIAS COMPLEMENTARES

- **Curso Edureka.** Hadoop Tutorial: All you need to know about Hadoop! Acesso em <<https://www.edureka.co/blog/hadoop-tutorial>>;
- **Curso Guru.** Big Data Hadoop Tutorial for Beginners: Learn in 7 Days!. acesso em <<https://www.guru99.com/bigdata-tutorials.html>>;
- **Curso Cetax.** Apache Hadoop Essentials. acesso disponível em <<https://www.cetax.com.br/curso-de-apache-hadoop-essentials/>>



OBRIGADO!

Dúvidas?

Você pode me encontrar em

- ▶ nickssonarais@gmail.com
- ▶ [\(85\) 9 9253-5715](tel:(85)99253-5715)