

# BIG DATA



LÁB DATA



FUNDAÇÃO  
INSTITUTO DE  
ADMINISTRAÇÃO

# **Disciplina: Aplicações de Big Data com Hadoop**

## **Tema da Aula: YARN**

### **Coordenação:**

Prof. Dr. Adolpho Walter  
Pimazzi Canton

Profa. Dra. Alessandra de  
Ávila Montini

**Profa. Rosangela de Fátima Pereira**

**Junho de 2016**

# Currículo

## Formação

- Mestrado em Engenharia de Computação pela Escola Politécnica da Universidade de São Paulo (Poli-USP) (em andamento)
- Especialização em Tecnologia Java pela Universidade Tecnológica Federal do Paraná (UTFPR) (2011)
- Tecnologia em Análise e Desenvolvimento de Sistemas pela UTFPR (2011)
- Bacharelado em Administração de Empresas pela Universidade Estadual do Norte do Paraná (UENP) (2007)

## Experiência

- Professora de Big Data Analytics em empresas e programas de MBA - FIA (2013 - atual)
- Pesquisadora no Laboratório de Arquitetura e Redes de Computadores (LARC) – USP (2013 - atual)
- Professora de cursos de engenharia na UTFPR (2011 -2012)
- Analista de sistemas na BSI Tecnologia (2009-2010)

LinkedIn: <https://br.linkedin.com/pub/rosangela-de-fatima-pereira/68/a10/b56>

Apaixonada por **Big Data!**

# Objetivo da Aula

Apresentar ao aluno características do Ecossistema Hadoop e fundamentos do YARN

# Conteúdo da Aula

- Revisão da aula anterior
- Ecossistema Hadoop
- YARN
- Considerações finais

# Aula anterior

Conjunto de classes que podem ser herdadas para utilização de funcionalidades pré-definidas

O conjunto de classe é dividido nos seguintes componentes:

- Hadoop Common
- Hadoop Distributed File System (HDFS)
- Hadoop MapReduce
- Hadoop YARN

# Aula anterior

Os seguintes passos serão executados:

1. Implementar as classes da aplicação
2. Gerar um arquivo JAR da aplicação
3. Executar o JAR no ambiente Hadoop
4. Visualizar o resultado da aplicação

# Aula anterior

Serão implementadas 3 classes:

ContaPalavrasMap.java

ContaPalavrasReduce.java

ContaPalavrasDriver.java

- Para facilitar o desenvolvimento das classes foi criado um projeto inicial chamado **ContaPalavras** contendo a estrutura-base das classes.



# Conteúdo da Aula

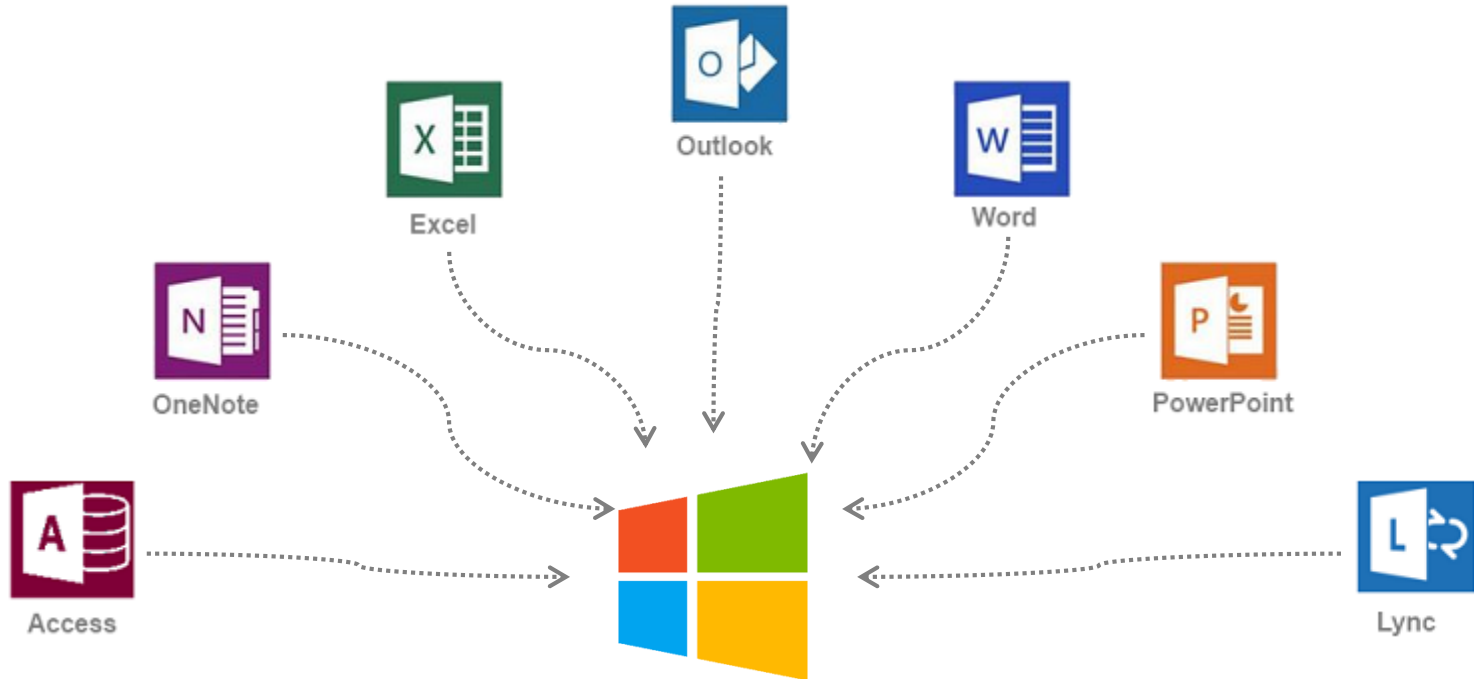
- Revisão da aula anterior
- **Ecosistema Hadoop**
- YARN
- Considerações finais

# Ecossistema Hadoop

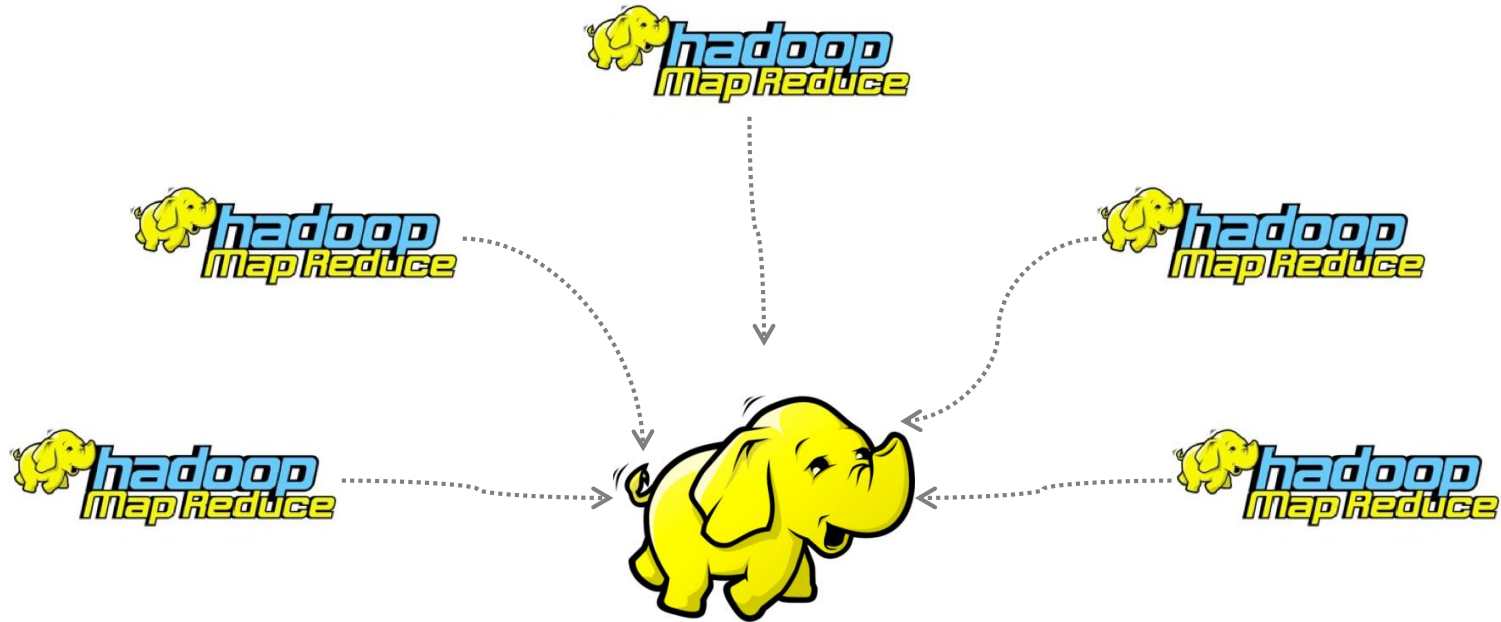
## Hadoop

“Um sistema operacional para Big Data”

# Ecosystem Hadoop



# Ecosystem Hadoop

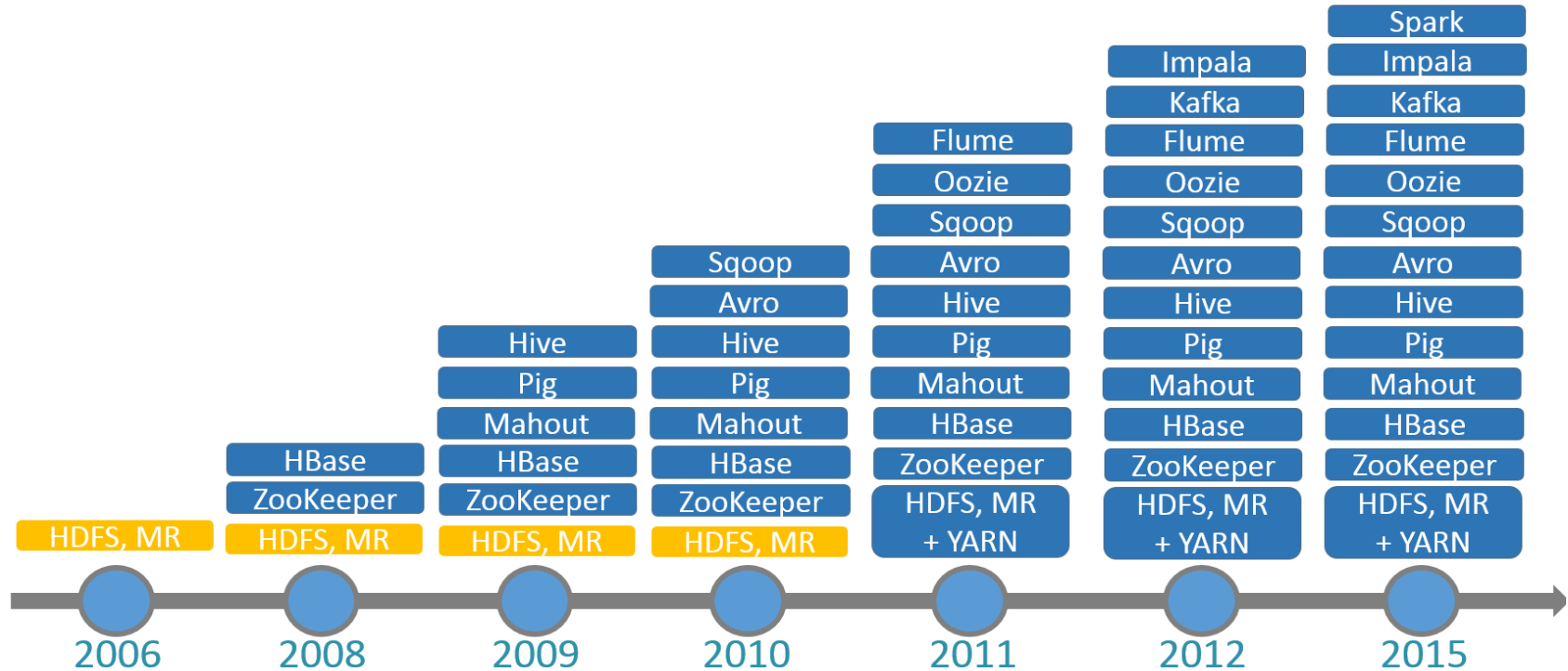


# Ecosystem Hadoop

Com a adoção do **Hadoop**, as empresas foram percebendo a necessidade de novas tecnologias e novas técnicas para facilitar e aperfeiçoar o armazenamento e processamento dos dados

Com isso, **Hadoop** foi evoluindo com o passar dos anos...

# Ecosystem Hadoop



# Ecossistema Hadoop

Preciso gerenciar os processos executados no Hadoop por inúmeros usuários e serviços...

# Ecossistema Hadoop

## ZooKeeper

- Serviço para coordenação de processos de aplicações distribuídas
- Oferece alta disponibilidade às aplicações
- Desenvolvido no Yahoo!
- Não é de uso exclusivo do Hadoop





# Ecosystem Hadoop

Preciso armazenar meus dados em um banco de dados  
NoSQL...

# Ecosystem Hadoop

## HBase

- Banco de dados NoSQL do Hadoop
- Versão open-source do Google Big Table
- Acelera o processo de escrita dos dados
- Orientado a colunas

APACHE  
**HBASE**

# Ecossistema Hadoop

Preciso de uma linguagem em script para processar meus dados...

# Ecosystem Hadoop

## Pig

- Linguagem em formato de script para manipulação de dados - PigLatin
- Converte o código em MapReduce jobs
- Oferece operações padrões de processamento de dados, filter, join, group-by, order-by
- Linguagem de fácil compreensão e manutenção
- Desenvolvido originalmente no Yahoo!



# Ecosystem Hadoop

Preciso de uma linguagem similar à SQL...

# Ecossistema Hadoop

## Hive

- Linguagem de abstração de alto nível
- Utiliza **HiveQL**, uma linguagem similar à **SQL**
- Gerar MapReduce Jobs
- Desenvolvido pelo Facebook



# Ecossistema Hadoop

Preciso mover os dados para o HDFS no momento que eles são gerados...

# Ecosystem Hadoop

## Flume

- Serviço distribuído, confiável e de alta disponibilidade
- Transfere grande volume de dados no momento que eles são gerados
- Utilizado para inserção de arquivos (logs da web, logs da rede, etc.)





# Ecossistema Hadoop

Preciso processar dados em formato de tabelas...

# Ecossistema Hadoop

## Sqoop

- Sqoop significa “SQL para Hadoop”
- Importa tabelas do banco de dados relacionais para HDFS e vice-versa
- Utiliza MapReduce para importar os dados
- Suporta banco de dados Oracle



# Ecosystem Hadoop

Preciso processar rapidamente as mensagens coletadas das fontes de dados...

# Ecosystem Hadoop

## Kafka

- Serviço de distribuição de mensagens
- Oferece alta vazão de dados, replicação, particionamento de tarefas e tolerância a falhas
- Muito utilizado para coleta de logs e processamento de streaming



# Ecosystem Hadoop

Preciso executar técnicas de aprendizado de máquina em modo distribuído...

# Ecossistema Hadoop

## Mahout

- Linguagem que facilita a execução de algoritmos de aprendizado de máquina
- Fornece suporte às seguintes técnicas:
  - **Filtragem colaborativa**: sistemas de recomendações
  - **Agrupamento (*clustering*)**: permite identificar grupos de acordo com características similares
  - **Classificação**: técnicas para categorizar itens não-classificados a uma determinada categoria
  - **Frequent Itemset Mining**: analisa itens em um grupo e identifica quais deles normalmente aparecem juntos



# Ecossistema Hadoop

Preciso processar outras aplicações além do modelo de processamento MapReduce...

# Ecosystem Hadoop

## Hadoop 1.x

- Falta de suporte para outros paradigmas e modelos de programação em Big Data
  - Processamento em lote, streaming, in-memory, em grafos,...
- Problemas de escalabilidade em cluster com mais de 4.000 nós
- Falta de flexibilidade na configuração de slots dos nós escravos



# Conteúdo da Aula

- Revisão da aula anterior
- Ecossistema Hadoop
- YARN
- Considerações finais

# YARN

## Principais diferenças entre Hadoop 1.x e Hadoop 2.x (YARN):

- Divisão das duas funcionalidades do JobTracker:
  - Gerenciamento de recursos
  - Gerenciamento/monitoramento de jobs

# YARN

## Principais diferenças entre Hadoop 1.x e Hadoop 2.x (YARN):

- Divisão das duas funcionalidades do JobTracker:
  - Gerenciamento de recursos
  - Gerenciamento/monitoramento de jobs
- MapReduce se torna uma das aplicações do Hadoop

O

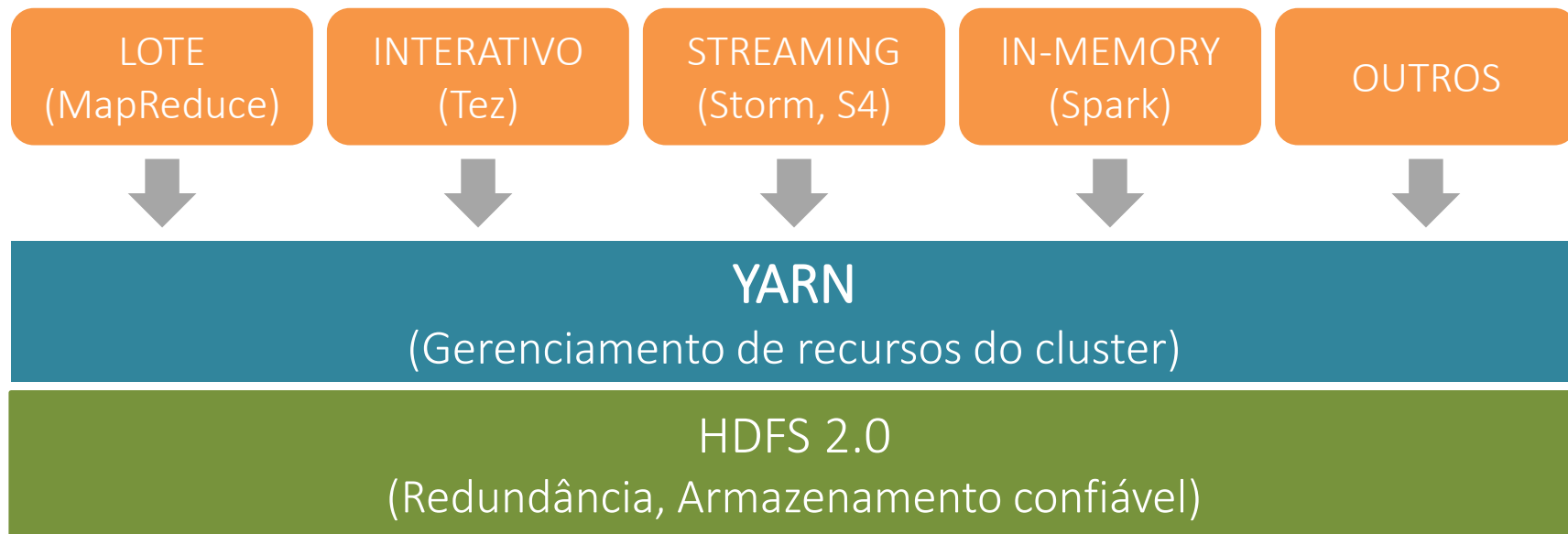
# YARN

## Principais diferenças entre Hadoop 1.x e Hadoop 2.x (YARN):

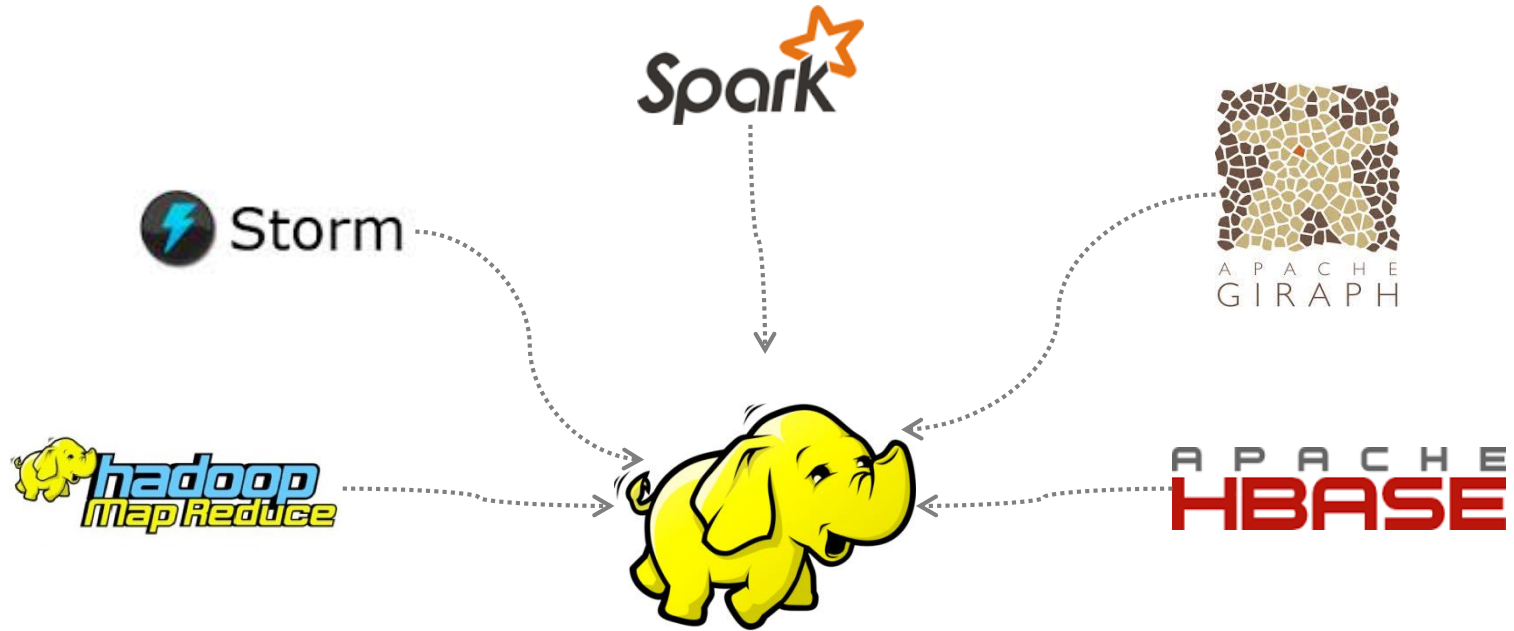
- Divisão das duas funcionalidades do JobTracker:
  - Gerenciamento de recursos
  - Gerenciamento/monitoramento de jobs
- MapReduce se torna uma das aplicações do Hadoop
- Remoção de slots fixos: recursos dos nós são alocados para as aplicações somente quando for requisitado

# YARN

*YARN – Yet Another Resource Negotiator*



# YARN



# YARN

YARN = *Yet Another Resource Negotiator*

Apache Hadoop YARN: Yet Another Resource Negotiator

Vinod Kumar Vavilapalli<sup>h</sup>    Arun C Murthy<sup>h</sup>    Chris Douglas<sup>m</sup>    Sharad Agarwal<sup>i</sup>  
Mahadev Konar<sup>h</sup>    Robert Evans<sup>y</sup>    Thomas Graves<sup>y</sup>    Jason Lowe<sup>y</sup>    Hitesh Shah<sup>h</sup>  
Siddharth Seth<sup>h</sup>    Bikas Saha<sup>h</sup>    Carlo Curino<sup>m</sup>    Owen O'Malley<sup>h</sup>    Sanjay Radia<sup>h</sup>  
Benjamin Reed<sup>f</sup>    Eric Baldeschwieler<sup>h</sup>

<sup>h</sup>: hortonworks.com, <sup>m</sup>: microsoft.com, <sup>i</sup>: inmobi.com, <sup>y</sup>: yahoo-inc.com, <sup>f</sup>: facebook.com

## Abstract

The initial design of Apache Hadoop [1] was tightly focused on running massive, MapReduce jobs to process a

programming frameworks onto YARN viz. Dryad, Graph, Hoya, Hadoop MapReduce, REEF, Spark, Storm, Tez.

<http://br.hortonworks.com/blog/apache-hadoop-yarn-wins-best-paper-award-at-socc-2013/>

# YARN

YARN é responsável por:

- Gerenciamento de recursos do cluster
  - Alocação de recursos
  - Controles de segurança
  - Monitoramento de cargas de trabalho
  - Operações de administradores
- Escalonamento de tarefas

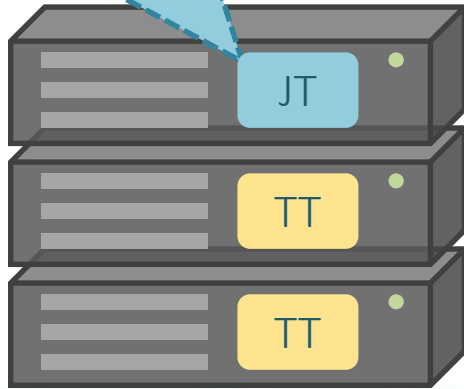


# YARN

## Anteriormente

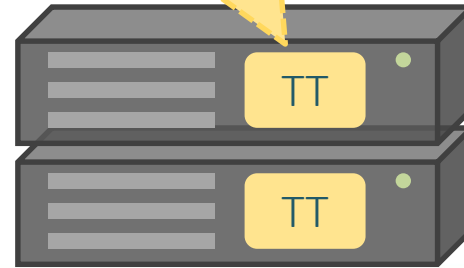
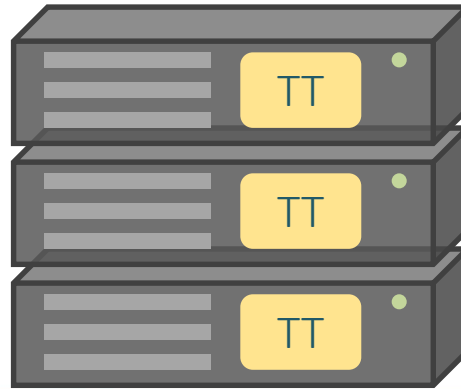
### JobTracker (JT)

- Gerenciador de recursos
- Escalonador de jobs



### TaskTracker (TT)

- Executam as tarefas MapReduce

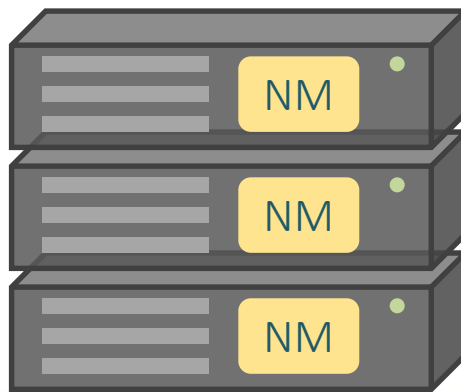
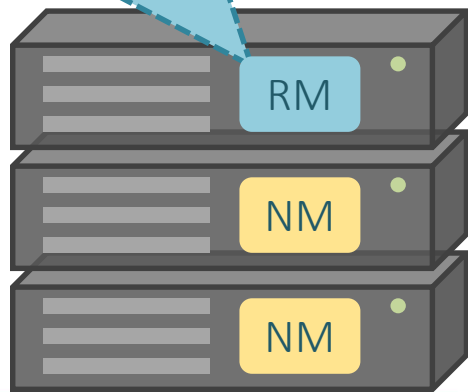


# YARN

## Atualmente

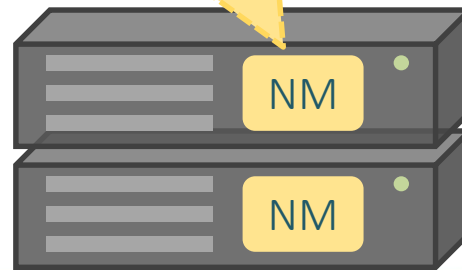
### ResourceManager(RM)

- Gerenciador de recursos
- Escalonador global de recursos



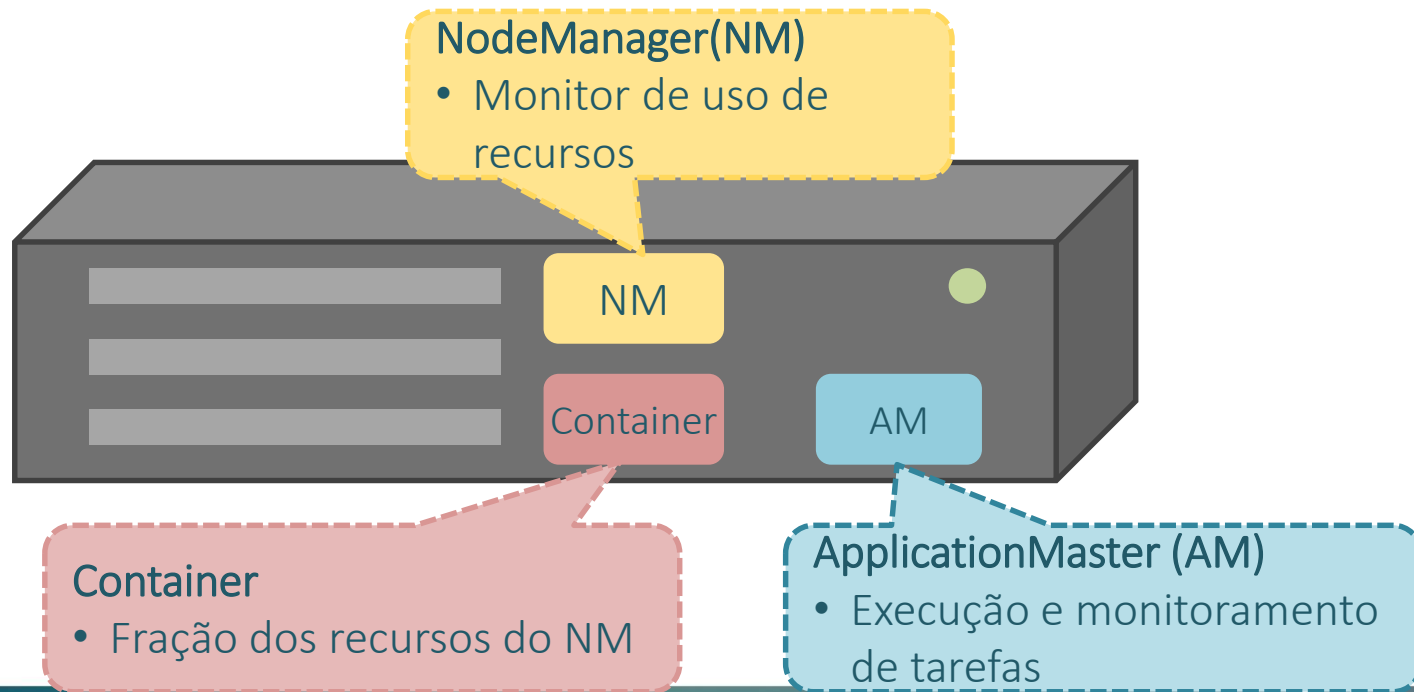
### NodeManager(NM)

- Monitor de uso de recursos



# YARN

Atualmente



# YARN

Aplicação: um job submetido ao Hadoop (ex. MapReduce job)

# YARN

**Aplicação:** um job submetido ao Hadoop (ex. MapReduce job)

**Container:** unidade de alocação de recursos (ex.: c1 = 1 GB, 2 CPU)  
substitui os slots fixados das tarefas map e reduce

# YARN

**Aplicação:** um job submetido ao Hadoop (ex. MapReduce job)

**Container:** unidade de alocação de recursos (ex.: c1 = 1 GB, 2 CPU)  
substitui os slots fixados das tarefas map e reduce

**Resource Manager:** escalonador global de recursos

# YARN

**Aplicação:** um job submetido ao Hadoop (ex. MapReduce job)

**Container:** unidade de alocação de recursos (ex.: c1 = 1 GB, 2 CPU)  
substitui os slots fixados das tarefas map e reduce

**Resource Manager:** escalonador global de recursos

**Node Manager:** gerencia o ciclo de vida do container, monitora os recursos do container (1 por máquina)

# YARN

**Aplicação:** um job submetido ao Hadoop (ex. MapReduce job)

**Container:** unidade de alocação de recursos (ex.: c1 = 1 GB, 2 CPU)  
substitui os slots fixados das tarefas map e reduce

**Resource Manager:** escalonador global de recursos

**Node Manager:** gerencia o ciclo de vida do container, monitora os recursos do container (1 por máquina)

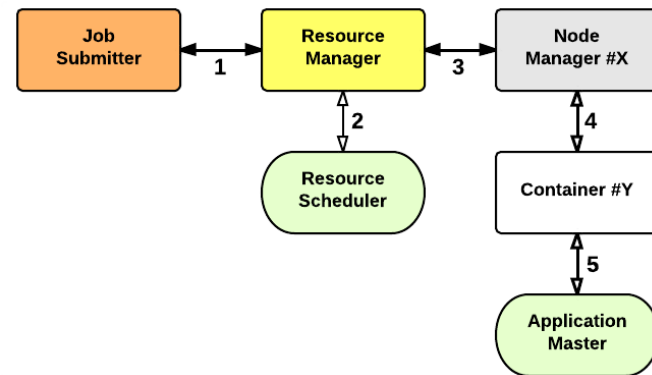
**Application Master:** gerencia a execução e escalonamento das tarefas (1 por aplicação)



# YARN

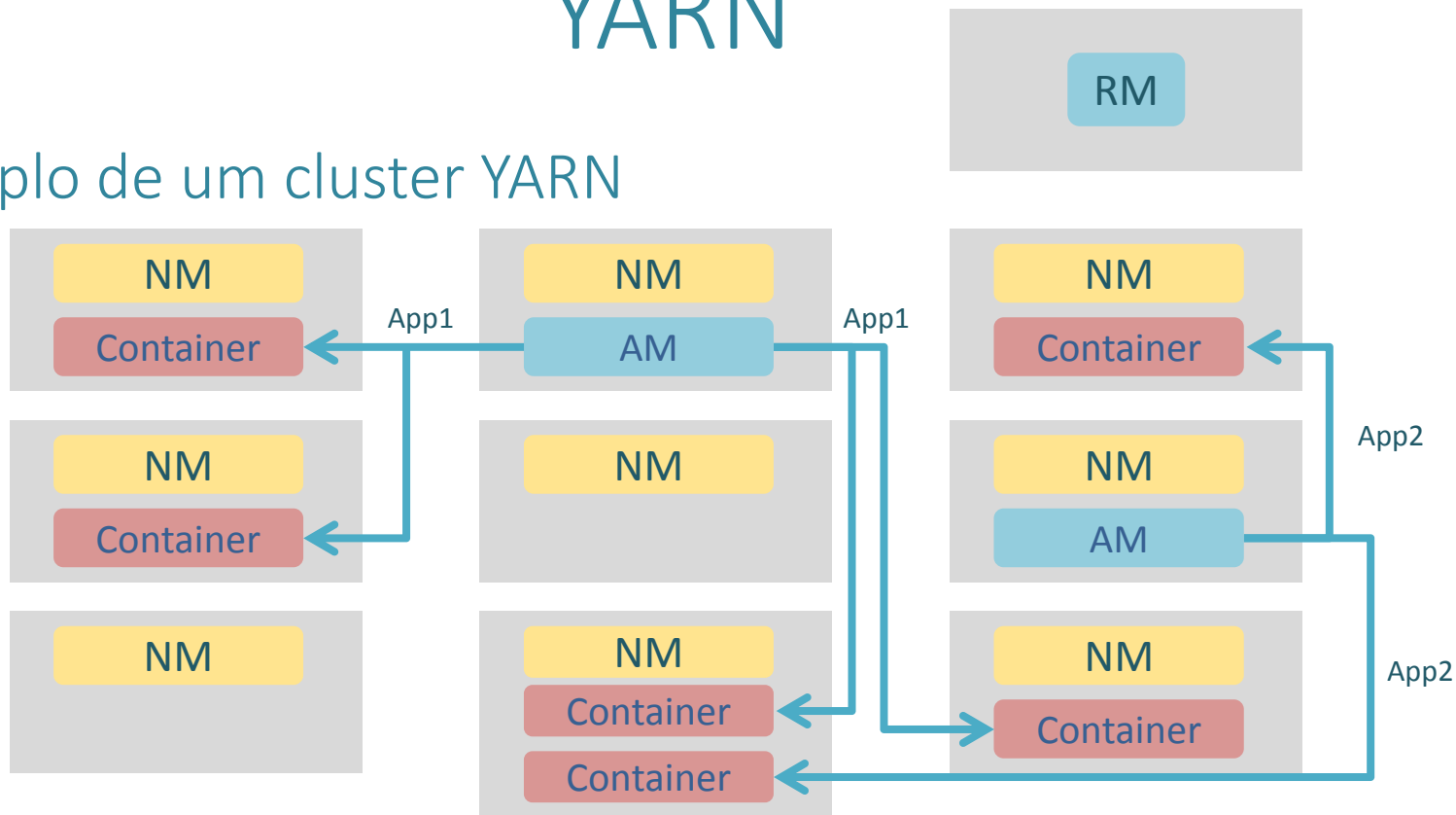
## Fluxo de execução de uma aplicação no YARN

1. Um cliente submete uma **aplicação** (job) para o **Resource Manager**
2. O **Resource Manager** aloca um **Container**
3. O **Resource Manager** faz contato com o **Node Manager**
4. O **Node Manager** inicia o **Container**
5. O **Container** executa a **Application Master**



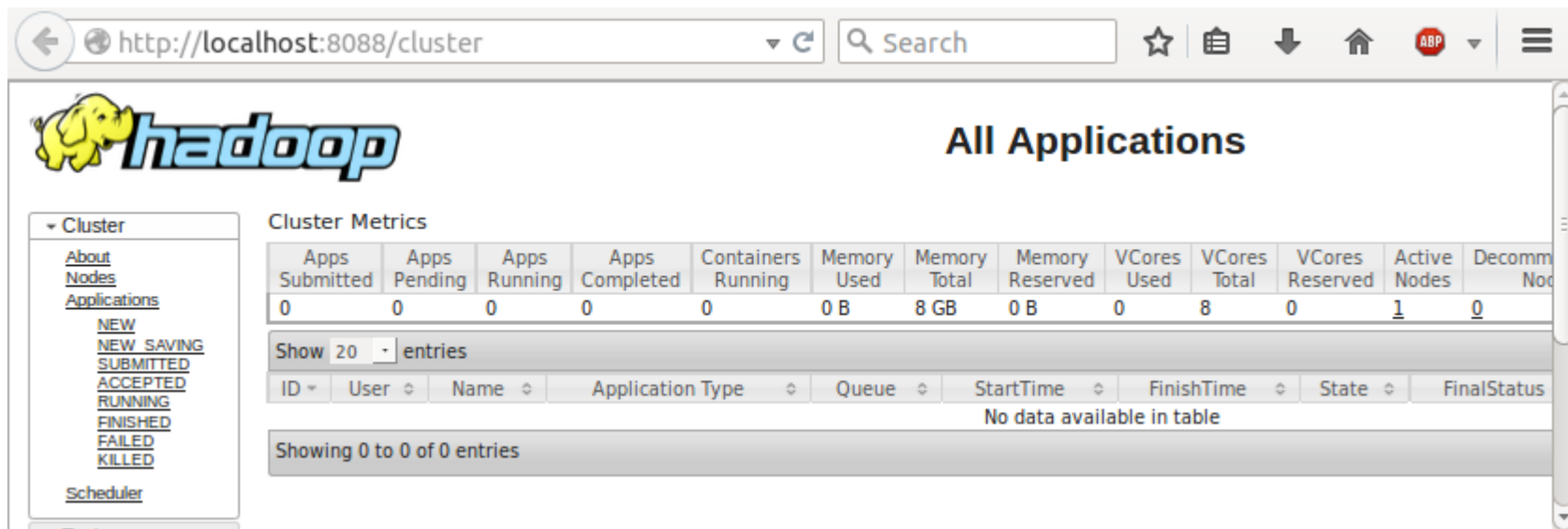
# YARN

## Exemplo de um cluster YARN



# YARN

Interface Web do Resource Manager (*http://rmhost:8088*)



The screenshot displays the Hadoop YARN web interface in a browser window. The address bar shows `http://localhost:8088/cluster`. The page features the Hadoop logo and the title "All Applications". On the left, a sidebar menu includes links for "Cluster", "About", "Nodes", "Applications", and "Scheduler". The "Applications" link is selected, showing a list of application states: NEW, NEW SAVING, SUBMITTED, ACCEPTED, RUNNING, FINISHED, FAILED, and KILLED. The main content area displays "Cluster Metrics" with a table of various resource metrics. Below this, there is a section for "Showing 0 to 0 of 0 entries" with a "Show 20 entries" dropdown and a table header for application details. The table is currently empty, displaying "No data available in table".

**Cluster Metrics**

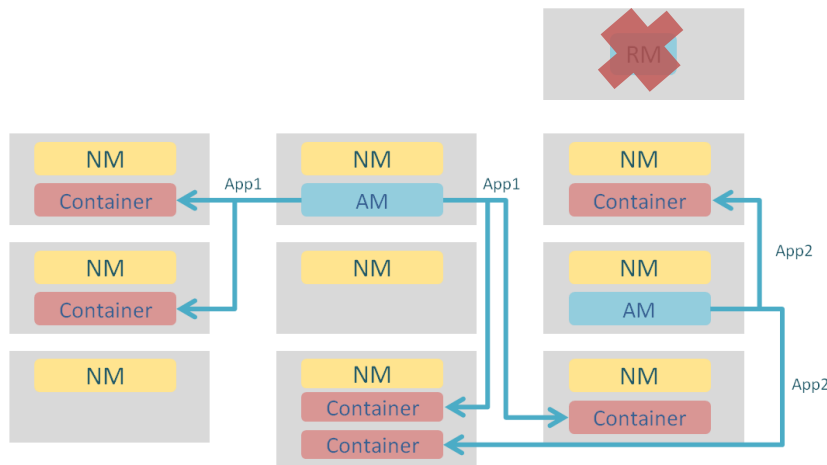
Apps Submitted	Apps Pending	Apps Running	Apps Completed	Containers Running	Memory Used	Memory Total	Memory Reserved	VCores Used	VCores Total	VCores Reserved	Active Nodes	Decomm Nodes
0	0	0	0	0	0 B	8 GB	0 B	0	8	0	1	0

Showing 0 to 0 of 0 entries

ID	User	Name	Application Type	Queue	StartTime	FinishTime	State	FinalStatus
No data available in table								

# YARN

- Resource Manager é um ponto único de falha do YARN
  - Se esse vier a falhar, todo o cluster fica indisponível até que ele seja novamente iniciado



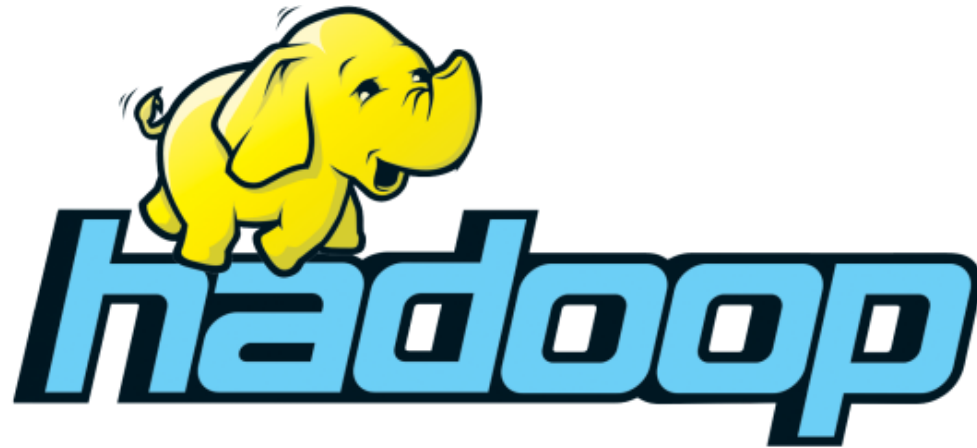
# YARN

- Solução: Resource Manager High Availability (RM HA)
  - Permite executar RMs redundantes
  - Arquitetura Active/StandBy
  - Oferece uma recuperação de falhas rápida e automática
  - Utiliza ZooKeeper para determinar qual RM deve permanecer ativo

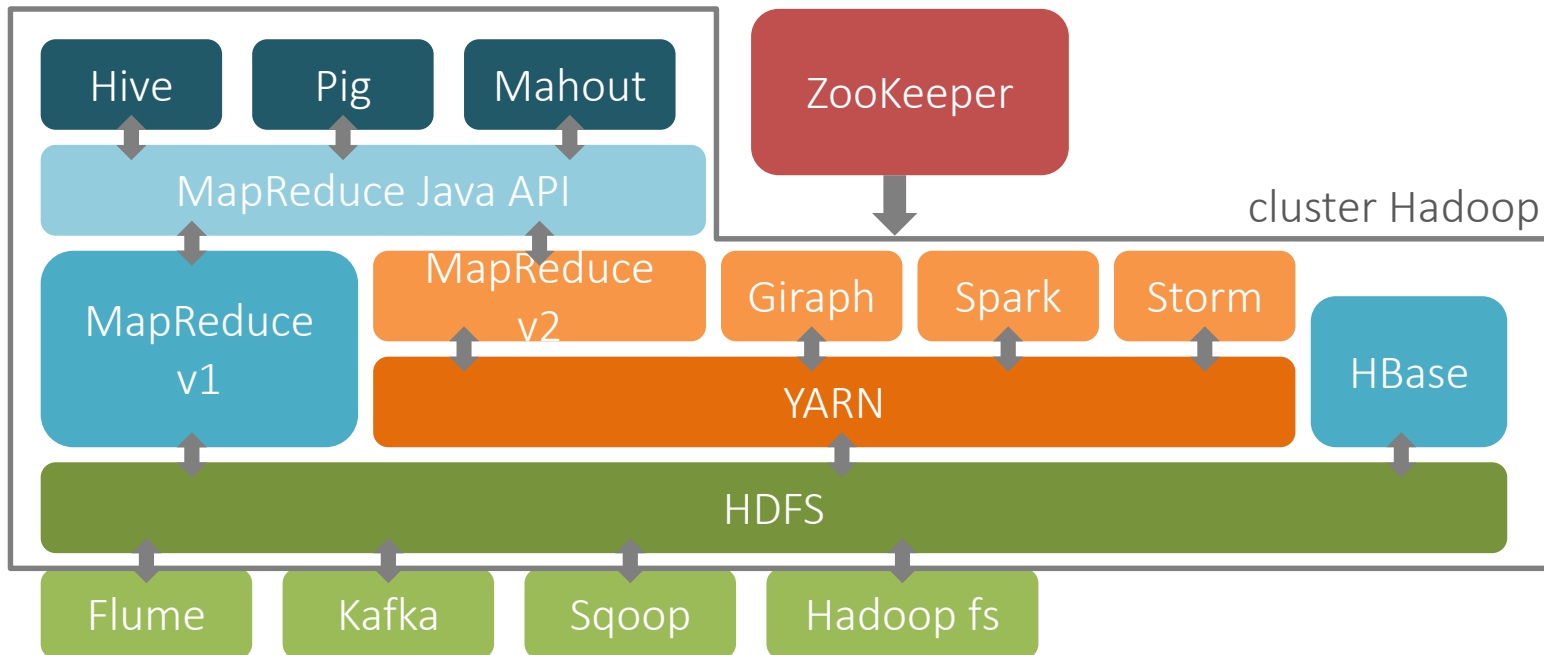
# Ecosystem Hadoop

Preciso de uma plataforma para atuar com Big Data...

# Ecosystem Hadoop



# Ecosystem Hadoop

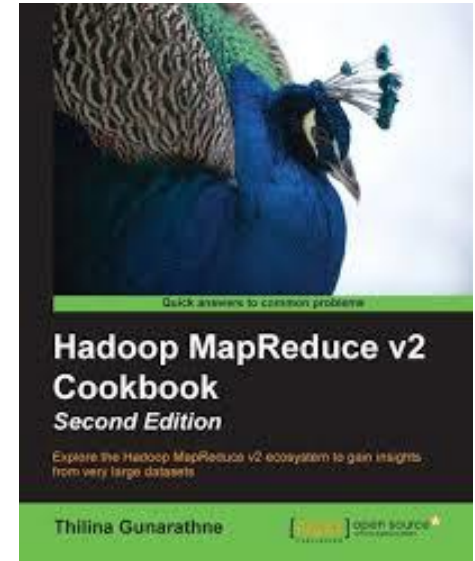
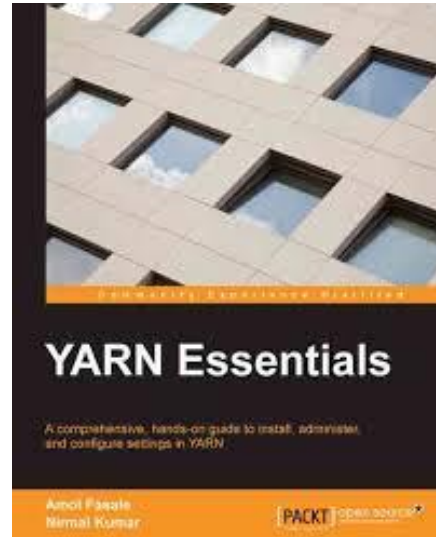
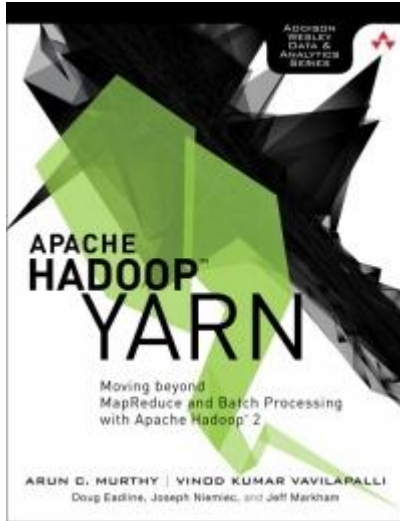




# Conteúdo da Aula

- Revisão da aula anterior
- Ecossistema Hadoop
- YARN
- **Considerações finais**

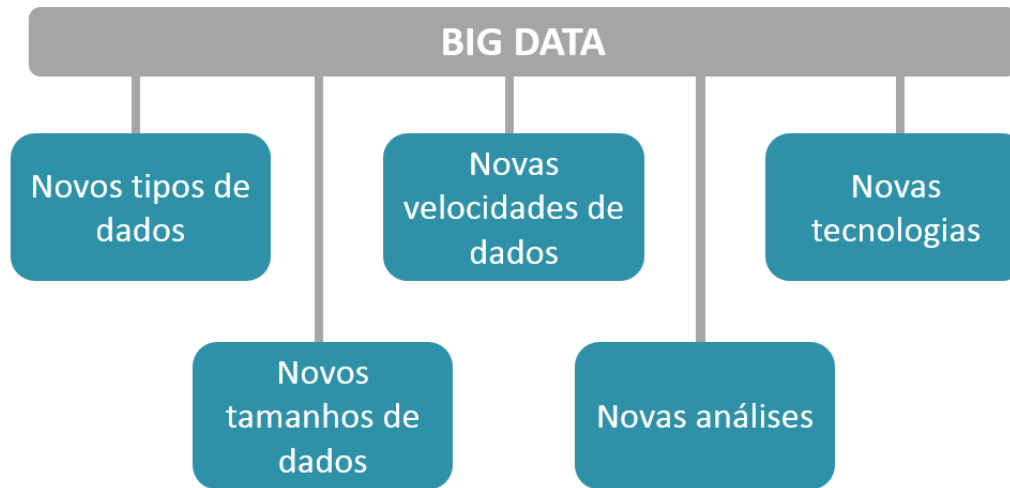
# Livros sobre YARN



# Considerações finais

Lembre-se...

Big Data requer inovação!



# Considerações finais

Recomendações de grupos de Big Data

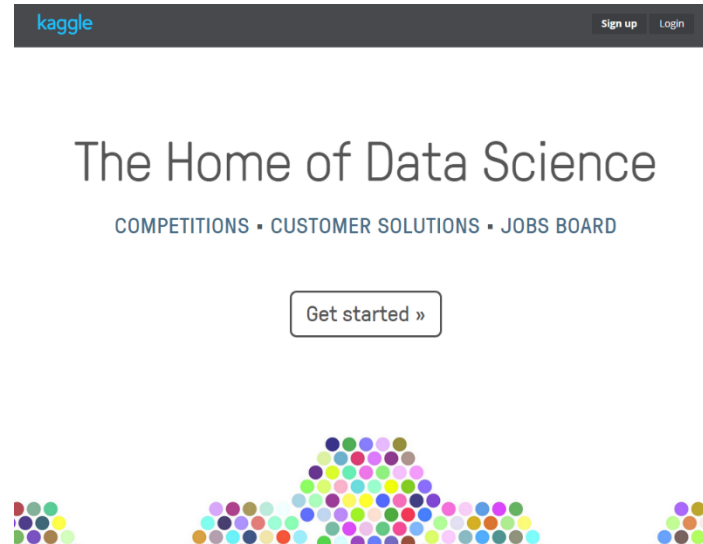
[www.meetup.com](http://www.meetup.com)



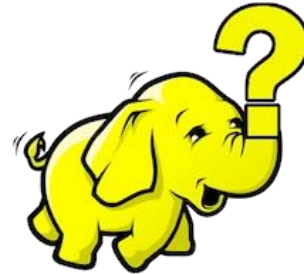
# Considerações finais

Recomendações de sites de Big Data

<https://www.kaggle.com/>



# Perguntas



rpereira@larc.usp.br

# Referências Bibliográficas

WHITE, Tom. **Hadoop: The definitive guide**. " O'Reilly Media, Inc.", 2012.

VAVILAPALLI, Vinod Kumar et al. **Apache hadoop yarn: Yet another resource negotiator**.

In: Proceedings of the 4th annual Symposium on Cloud Computing. ACM, 2013. p. 5.

VAVILAPALLI, Vinod Kumar et al. **Apache hadoop yarn: Yet another resource negotiator**.

In: Proceedings of the 4th annual Symposium on Cloud Computing. ACM, 2013. p. 5.

KAMBURUGAMUVE, Supun et al. **Survey of Apache Big Data Stack**. 2013. Tese de

Doutorado. Ph. D. Qualifying Exam, Dept. Inf. Comput., Indiana Univ., Bloomington, IN.

# Referências Bibliográficas

GOLDMAN, Alfredo et al. **Apache Hadoop: conceitos teóricos e práticos, evolução e novas possibilidades**. XXXI Jornadas de atualizações em informática, p. 88-136, 2012.

MONTEITH, J. Yates; MCGREGOR, John D.; INGRAM, John E. **Hadoop and its Evolving Ecosystem**. In: IWSECO@ ICSOB. 2013. p. 57-68.

FASALE, Amol; KUMAR, Nirmal. **Yarn Essentials**. Packt Publishing Ltd, 2015.