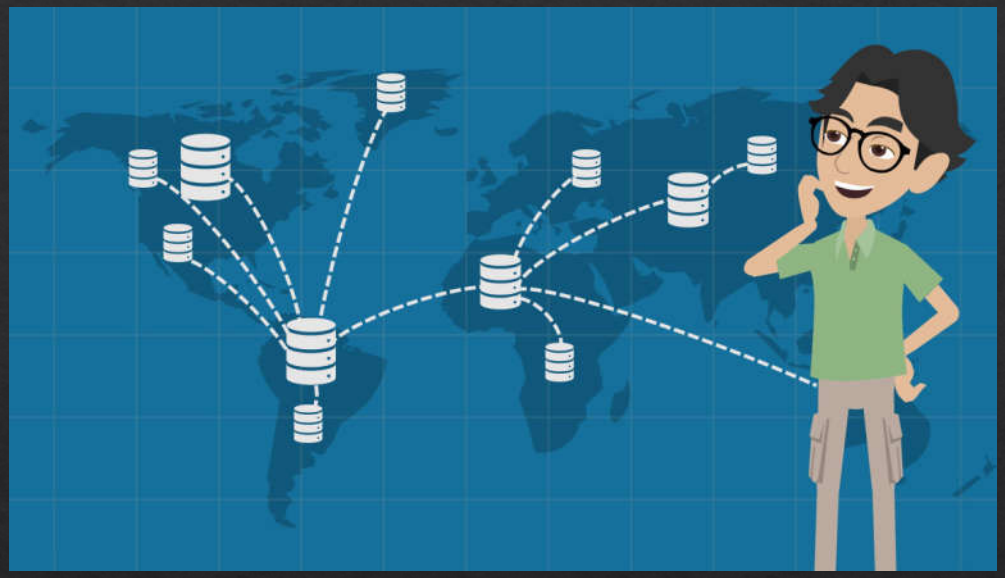

Formação Engenheiro de Dados

RDD

PROF. FERNANDO AMARAL
www.datascientist.com.br

RDD

- São recalculados a cada operação, não persistidos em memoria (Lazzy Evaluation)
  - Arquitetura pra grandes volumes de dados
- Para reuso, usar função RDD. persist() (diferentes formas, inclusive disco)

Lazy Evaluation

A Execução só começa quando é realmente necessária

Aplicado a transformações e carga de dados

Persistência

- Um RDD é imutável e recriado sempre que uma ação é executada
- Podemos persistir um RDD de diversas formas:
  - MEMORY_ONLY
  - MEMORY_ONLY_SER
  - MEMORY_AND_DISK
  - MEMORY_AND_DISK_SER
  - DISK_ONLY

Transformações e Ações

Transformações:

- Geram um novo RDD
- É mantida uma referência de dependências entre RDDs (lineage graph)

Ações:

- Cálculo exibido no console

Principais Transformações

Transformação	Descrição
`filter()`	Aplica uma filtro
`map()`	Aplica uma função
`sample()`	Gera subconjunto
`distinct()`	Retorna elementos úncios
`intersection()`	Retornar interseção de dois RDDs
`subtract()`	Subtrai conteúdo de um RDD
`cartesian()`	Gera o produto cartesiano de 2 RDDs
`union()`	Gera RDD com elementos de 2 RDDs

Principais Transformações para Chave-Valor

Transformação	Descrição
keys()	Retorna apenas as chaves
values()	Retorna apenas os valores
groupByKey()	Agrupa por chaves
sortByKey()	Ordena por chave

Principais Ações

Ação	Descrição
collect()	Retorna todo o RDD
count()	Conta o número de elementos
countByValue()	Contagem agrupada
take()	Retorna o número de elementos passados como parâmetro
top()	Retorna os primeiros elementos de acordo com o parâmetro
reduce()	Combina elementos usando computação paralela
`mean()`	Calcula a média
`sum()`	Calcula a soma
`min()`	Menor Valor
`max()`	Maior Valor
`variance()`	Calcula a Variância