Apache Spark

O que é o Spark

- Plataforma de computação em Cluster rápida, tolerante a falhas e de propósito geral
- 100x mais rápido que Mapreduce em memória e 10x mais rápido que Mapreduce em disco
- Desenvolvido em Scala
- Aplicações em Java, Scala, Python e R

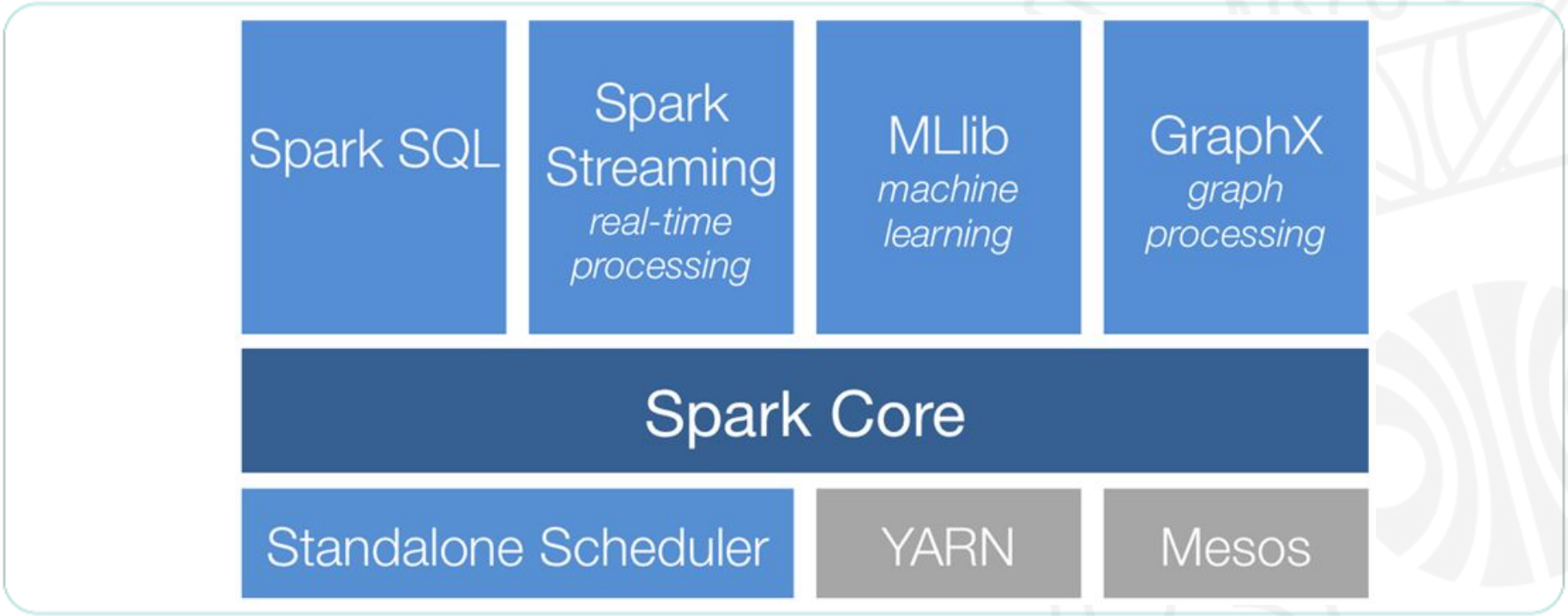
O que é o Spark

- Bibliotecas para SQL, Streaming, Machine Learning e grafos
- Processamento em larga escala
- Cluster, StandAlone, Docker ou Cloud
- Muito utilizado para ETL e análise de dados
- Processamento de PBs de dados

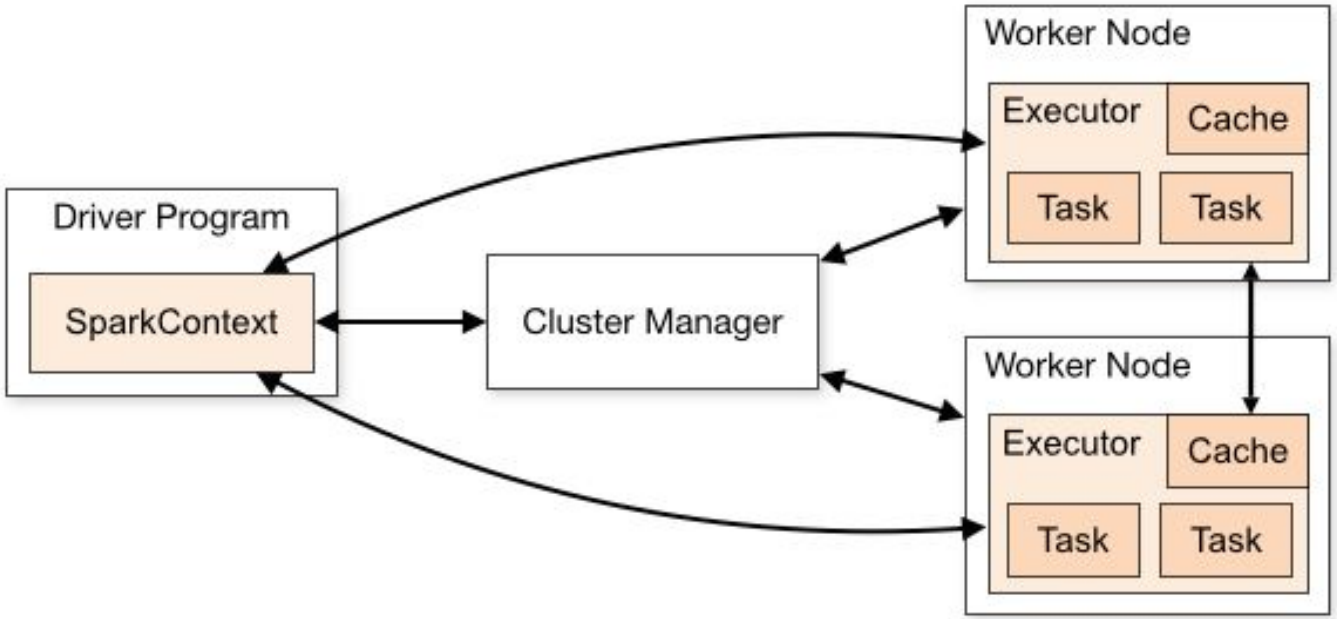
História

- Começou em 2009 como um projeto de pesquisa no UC Berkeley – AMPLab
- Os pesquisadores observaram que o MapReduce era ineficiente para empregos de computação iterativa e interativa
- Logo após a sua criação em 2009, já era 10-20 vezes mais rápido do que o MapReduce em alguns casos
- Além da UC Berkeley, os principais contribuidores incluem Databricks, Yahoo ! e Intel
- Foi aberto em março de 2010 e foi transferida para a ASF

Componentes



Arquitetura



Componentes de Execução

- Application: Qualquer aplicação submetida no Spark
- Job: Conjunto de transformações do RDD que são geradas em resposta a uma ação submetida pela aplicação
- Stage: Agrupamento de tasks que podem ser executadas independentes
- Taks: Uma unidade de trabalho que será enviada para um executor
- DAG: Grafo Acíclico Direcionado das dependências dos RDDs

Programamando em Spark

RDD

- Uma coleção distribuída imutável de objetos tolerantes a falhas que podem ser operados em paralelo.
- Resilient (tolerância a falha) Distributed (distribuído no cluster) Dataset (conjunto de dados)
- Criados a partir de outro RDD ou de um conjunto externo de dados (HDFS, Hbase, Hive, etc)

DataFrame

- Semelhante ao RDD
- Representa uma coleção de dados distribuídos imutáveis, como um RDD
- Permite a imposição de estrutura a uma coleção distribuída de dados
- Possui um esquema, o que significa que é algo que pode ser visto como uma coluna com um nome e um tipo
- Pode ser manipulado com SQL

DataSet

- Fornecer o melhor dos dois mundos (RDD e DataFrame)
- O estilo familiar de programação orientado a objetos e a segurança em tempo de compilação do RDD, mas com os benefícios de desempenho do otimizador de consulta do DataFrame
- Os conjuntos de dados também usam o mesmo mecanismo de armazenamento eficiente de pilha que o DataFrame.

Transformação

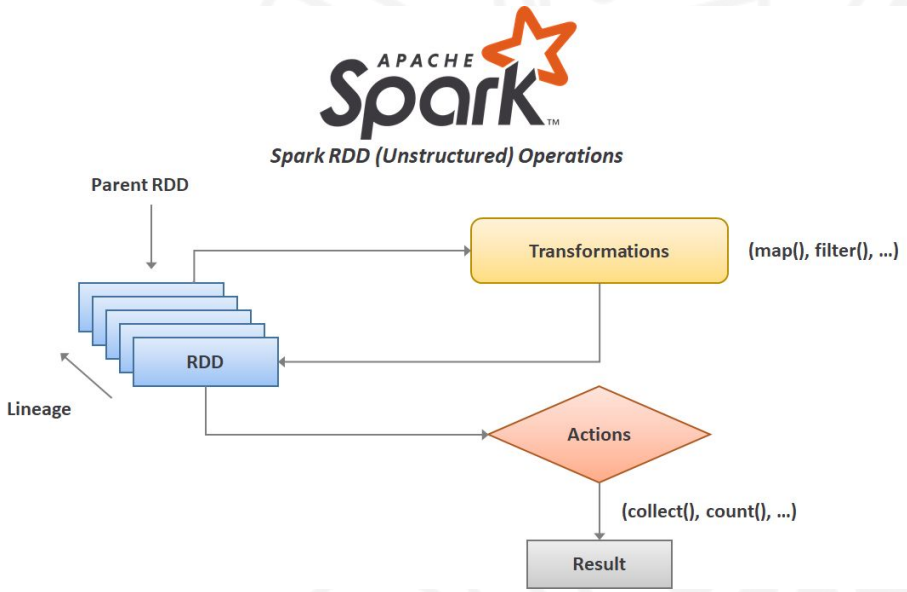
- Cria um RDD/DataFrame a partir de outro RDD/DataFrame
- Dois tipos de transformação: Wide e Narrow Dependencies
- Exemplos de transformações:
- map() - Aplica uma função a cada elemento no RDD e retorna um RDD do resultado
- filter() - Retorna um RDD com os elementos que correspondem a condição de filtro
- union() – Retorna um RDD contendo elementos de ambos os RDDs.

Ação

- Retorna o resultado ao driver ou salva em um sistema de armazenamento
- Exemplos de Ação:
- collect() - Retornar todos os elementos do RDD
- count() - Retorna o número de elementos do RDD
- take(10) - Retorna 10 elementos do RDD
- foreach(func) - Aplica a função fornecida a cada elemento do RDD

Lazy Evaluation

- As transformações não são executadas até que uma ação seja executada
- Spark internamente registra metadados para indicar que a operação foi solicitada
- Utilizado para reduzir o número de passos, agrupando operações



PUC Minas
Virtual