# TP1 - BCC 362

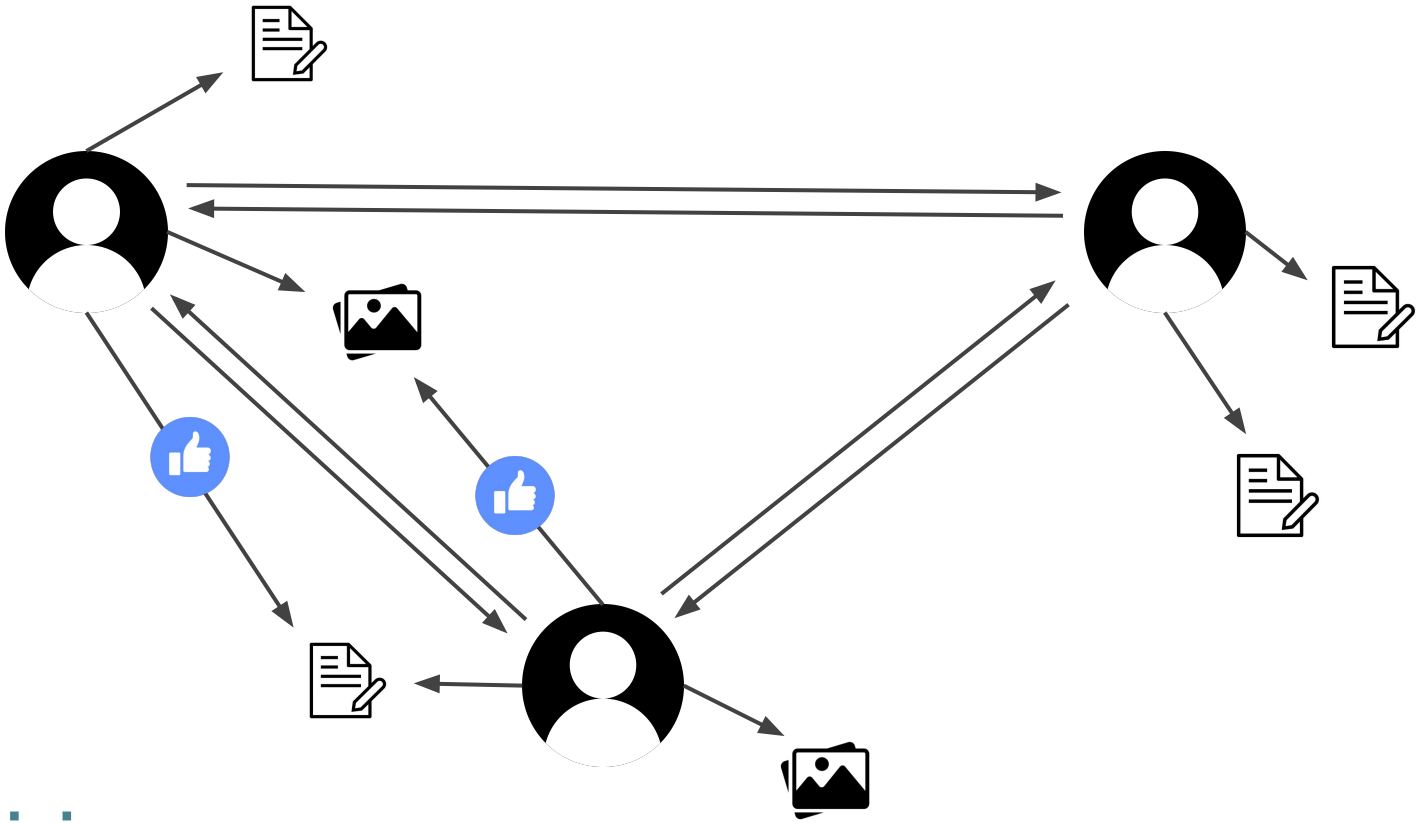
# O Middleware

- Graphx | Apache Spark

- É um componente para computação paralela de grafos. Trata-se de uma estrutura de **processamento de grafos distribuídos** que roda sobre o Spark.

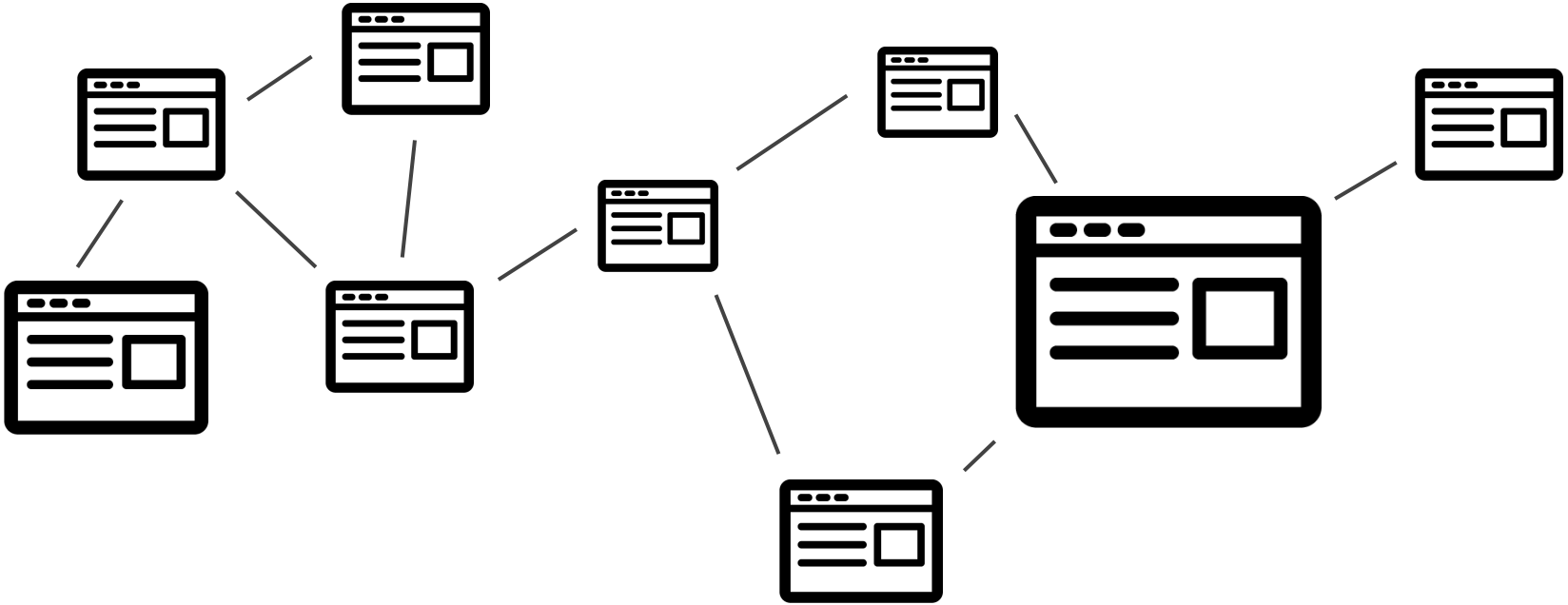
# GRAFOS

# GRAFOS



Redes sociais

# GRAFOS



Web graphs

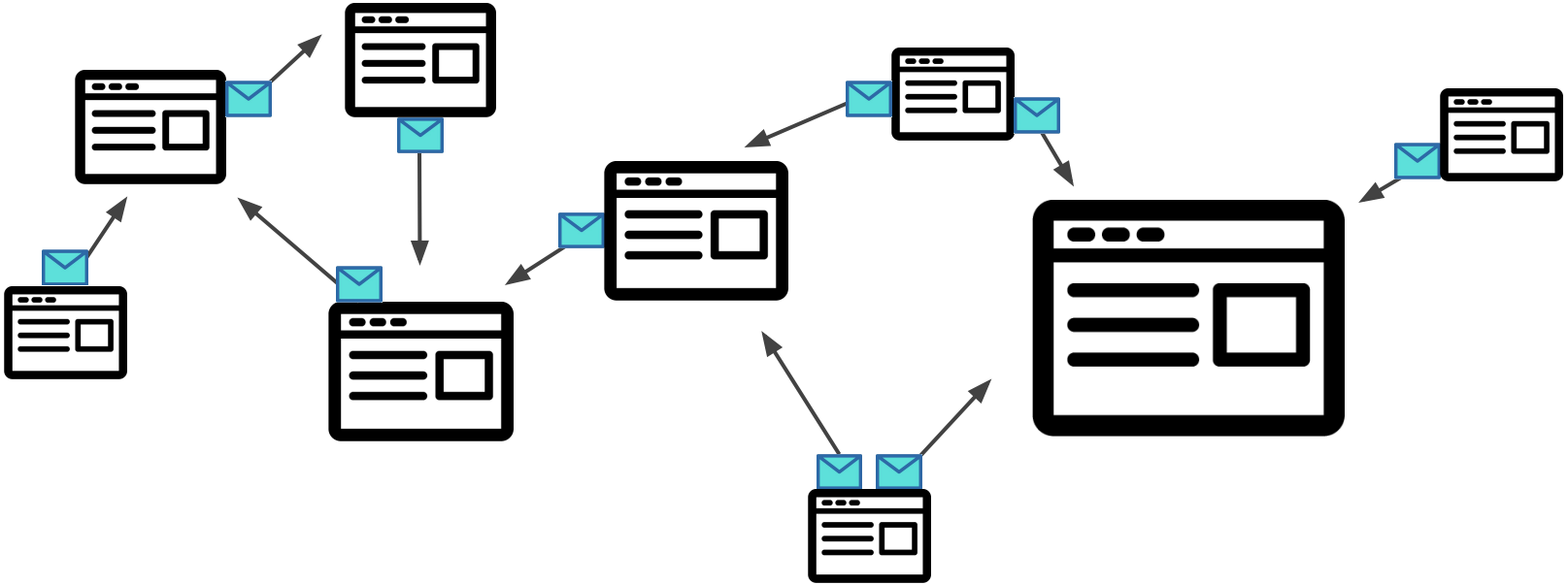
# GRAFOS

## Relação item-usuário

# ALGORITMOS PARA GRAFOS

# ALGORITMOS PARA GRAFOS

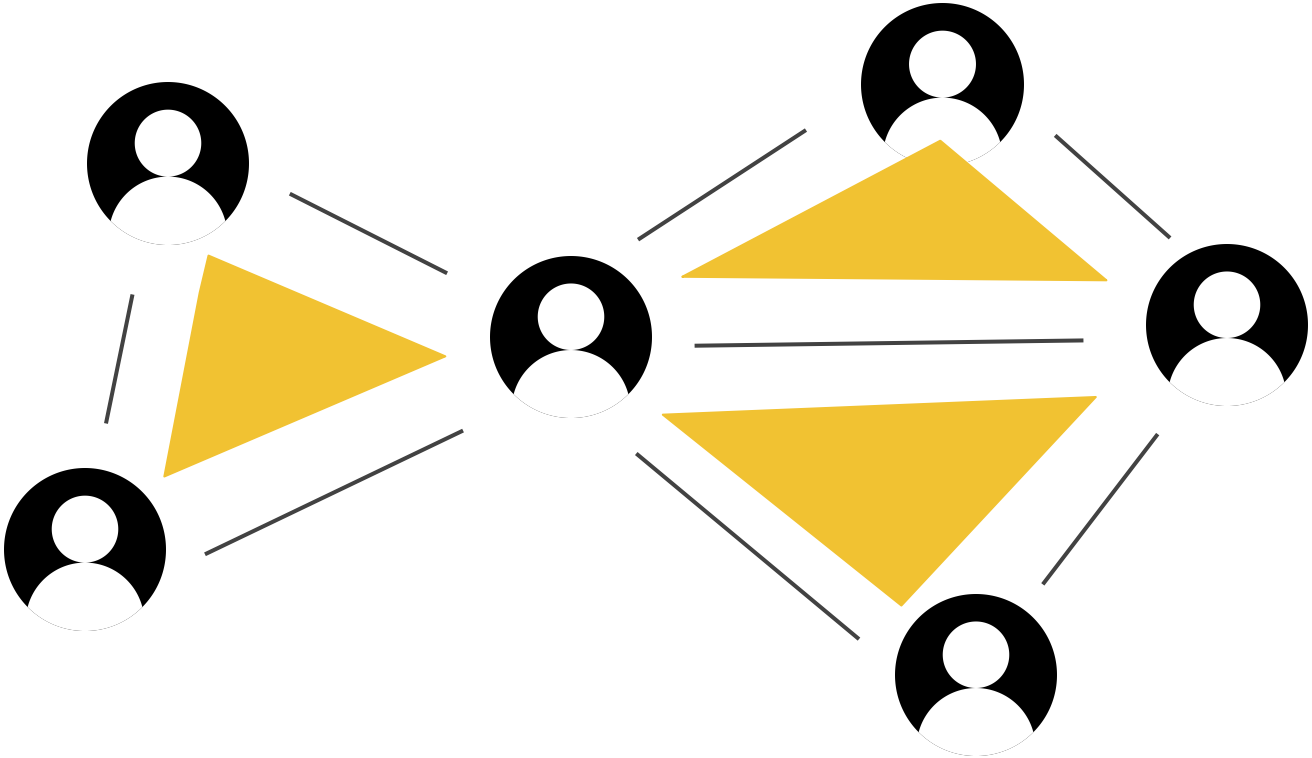
Apache Spark GraphX



## PageRank

# ALGORITMOS PARA GRAFOS

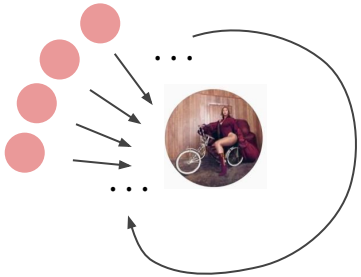
Apache Spark GraphX



## Triangle Counting

# DESAFIOS

## Vértices com grau alto

145.000.000
nós

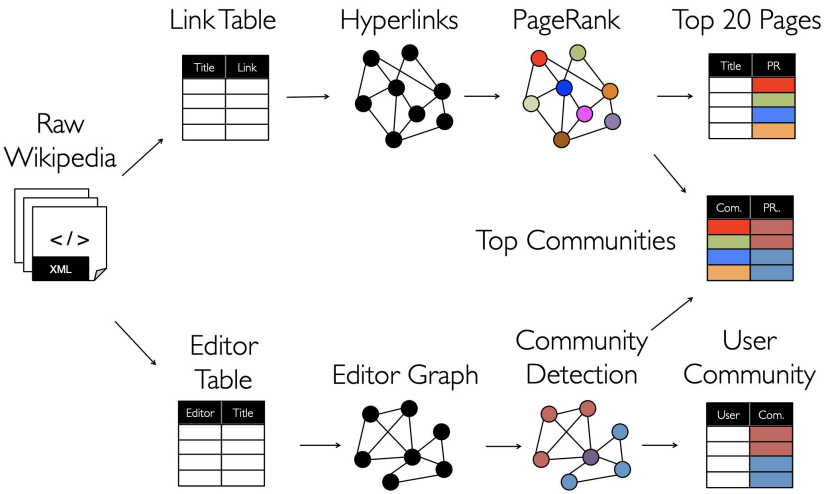
DESAFIOS

**ARMAZENAMENTO:** como armazenar grafos quando todos seus nós não cabem em uma máquina só?

**API:** como expor vértices vizinhos ao paralelismo?

# DESAFIOS

## Pipelines complexos




# SOLUÇÃO

**Incorporar um sistema orientado a tabelas (SPARK) ao processamento de grafos**

## DESAFIOS

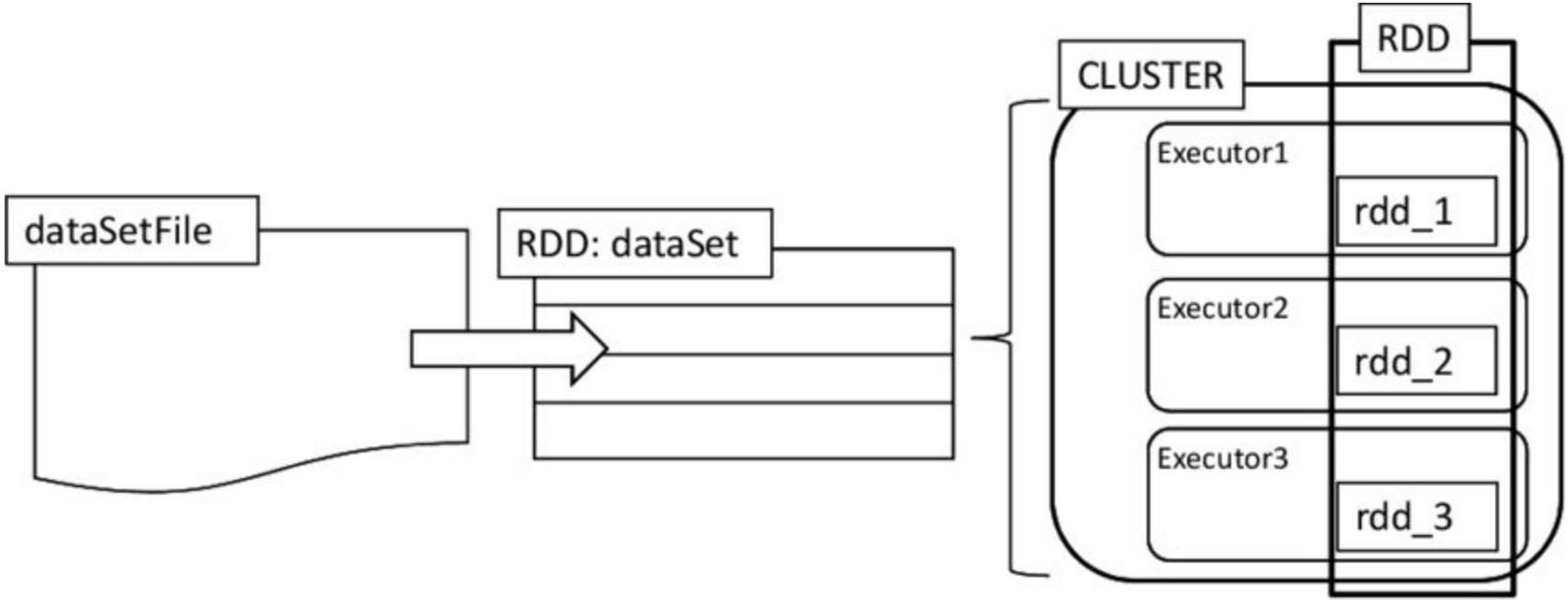
**ARMAZENAMENTO:** como armazenar grafos como tabelas?

**COMPUTAÇÃO:** como expressar operações em grafos como operações em tabelas (map, reduce, join, etc)?

**API:** como apresentar essas abordagens ao usuário?

# SPARK

# RESILIENT DISTRIBUTED DATASETS (RDDs)



# RDDs E DFS




# SETUP BÁSICO

Spark

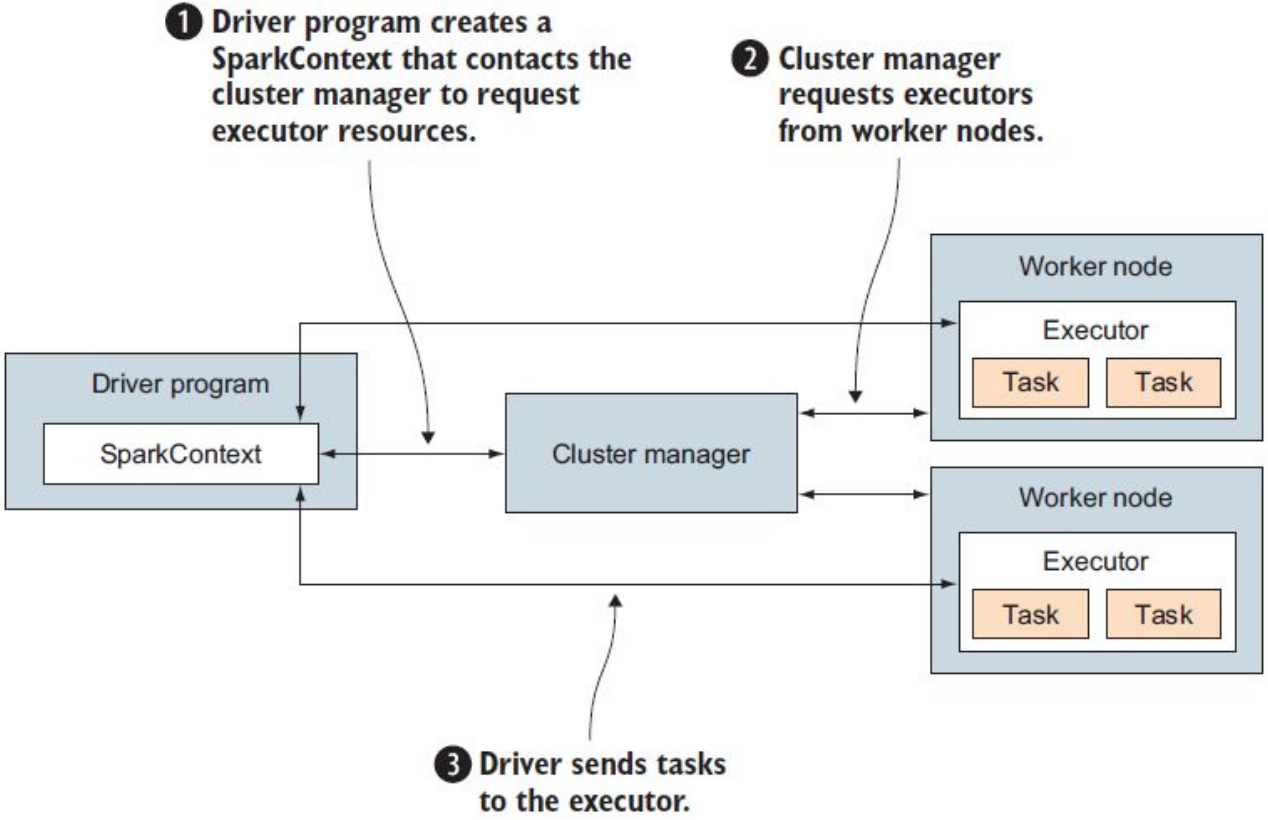
**Distributed Storage**

- HDFS
- Cassandra
- S3
- Other cloud storage vendors

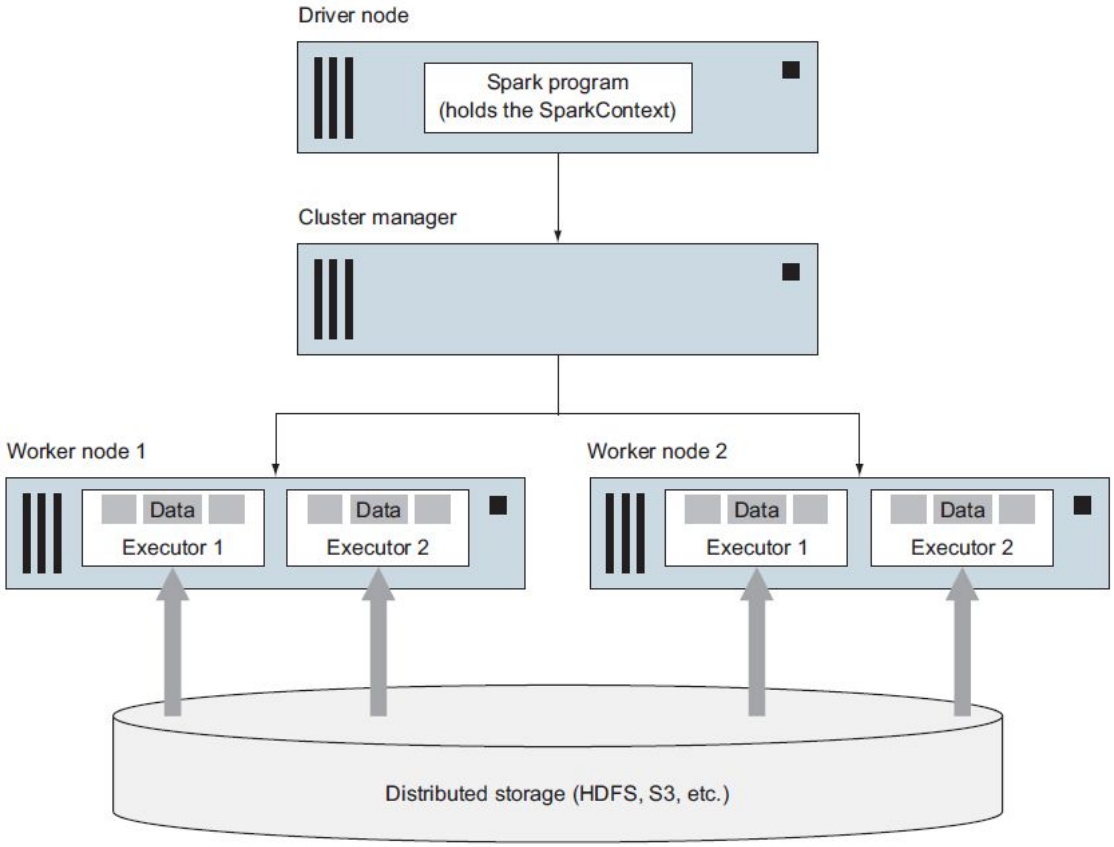
**Cluster Manager**

- Standalone
- YARN
- Mesos

# FUNCIONAMENTO BÁSICO



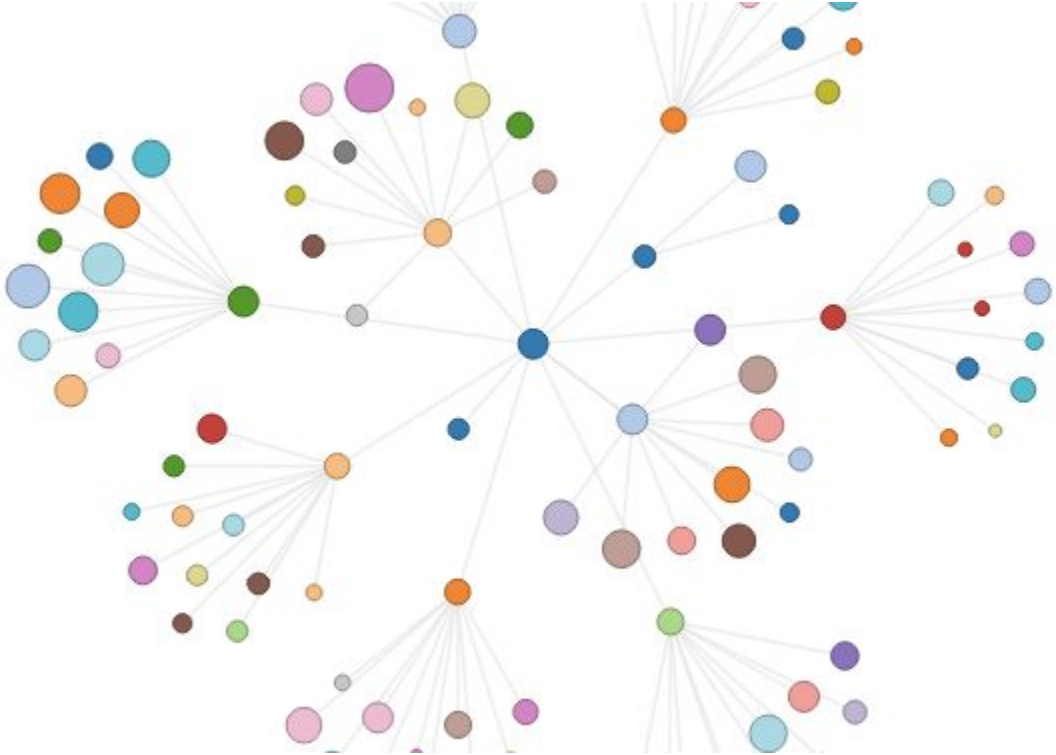
# FUNCIONAMENTO BÁSICO



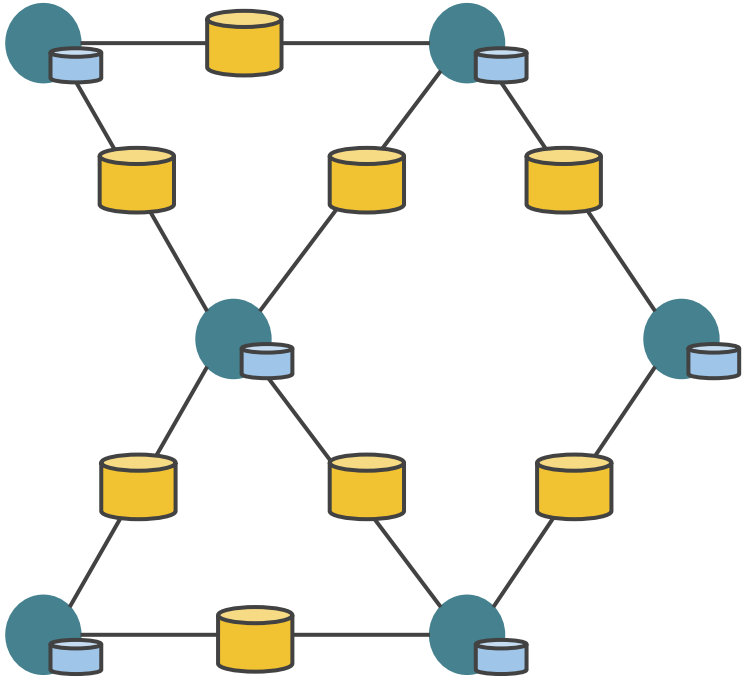
# GRAPHX

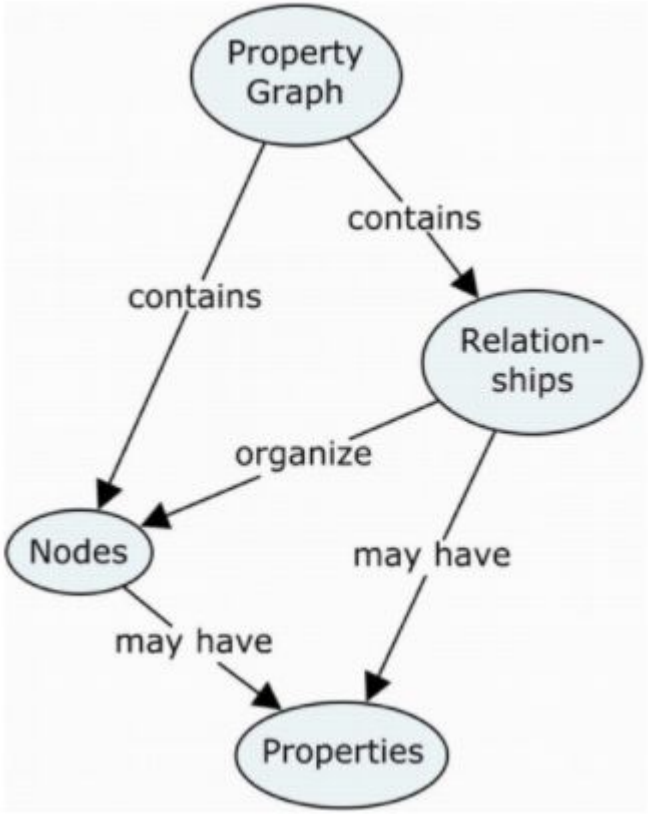
# PROPERTY GRAPHS

Apache Spark GraphX

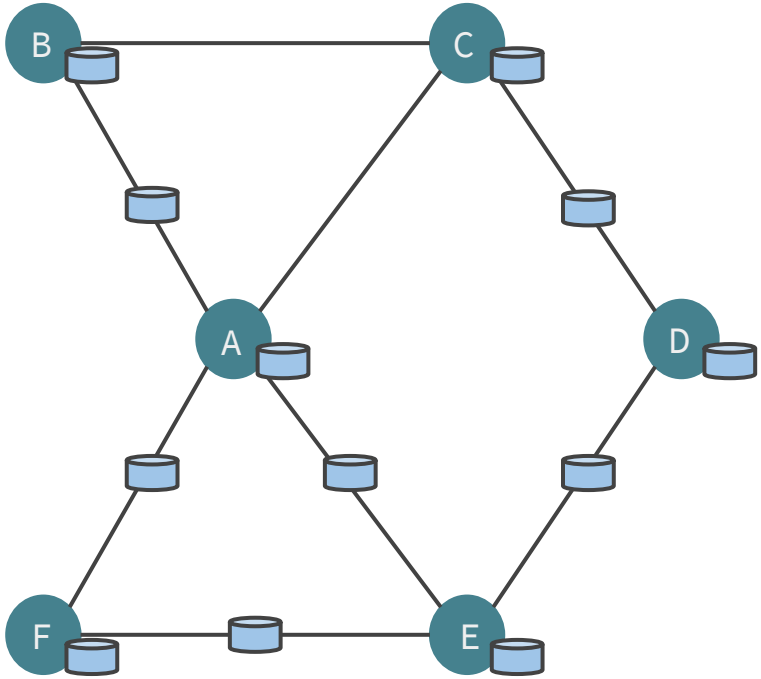


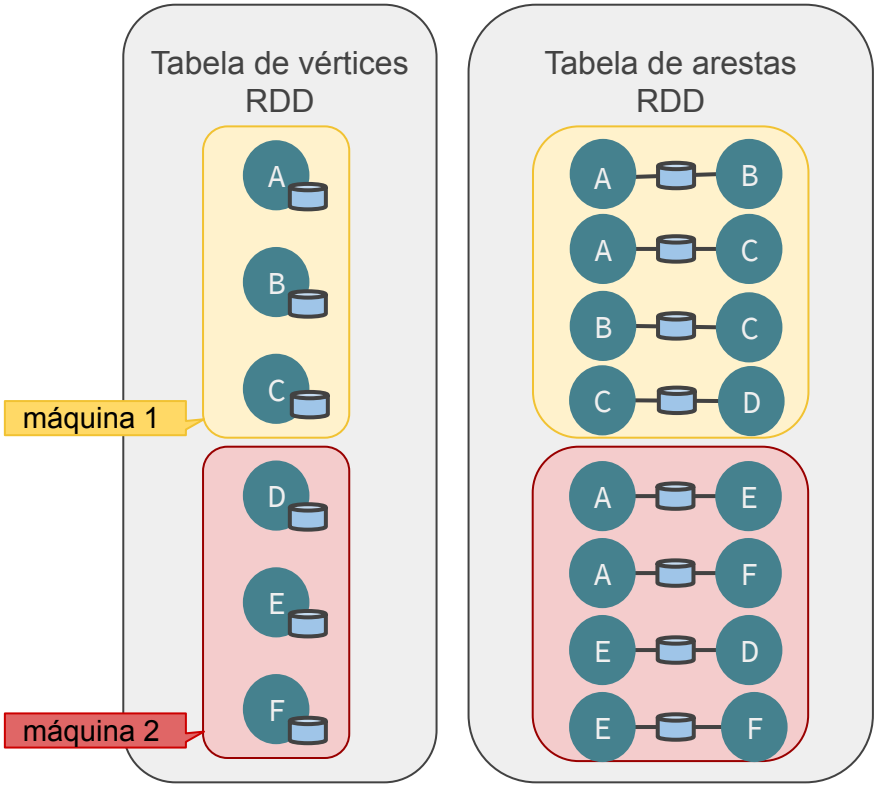
# PROPERTY GRAPHS





# PROPERTY GRAPHS



# PROS / CONS

Apache Spark GraphX

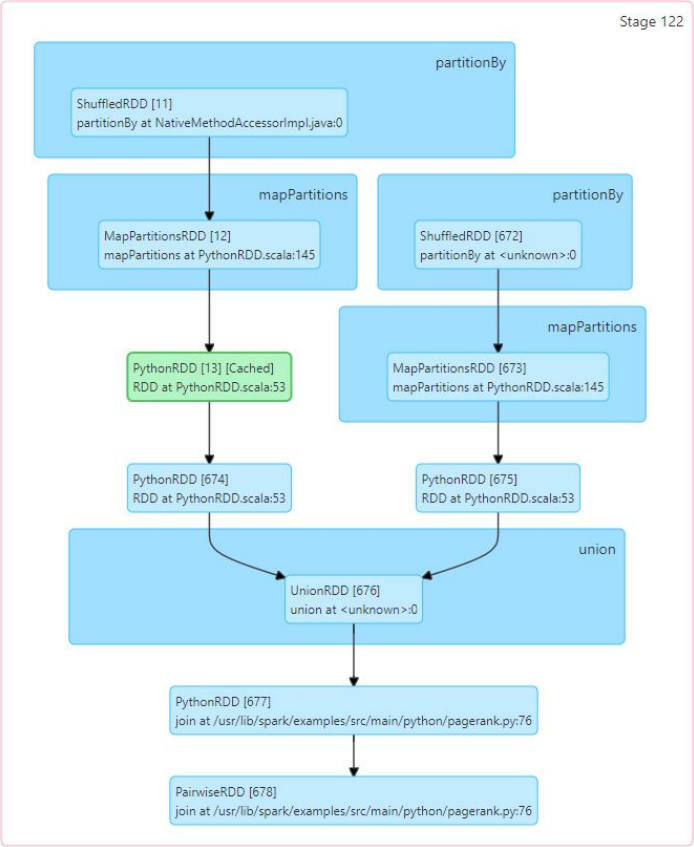
## Vantagens

- Trata a estrutura de dados tanto como grafos comuns, quanto como conjuntos separados de arestas e vértices que podem ter operações feitas de forma paralela (*map*, *join*, *transform*).
- Lida bem com *property graphs*.

## Desvantagens

- Latência alta
- Alto consumo de memória
- Não é eficiente quando há um grande número de iterações, pois pode haver sobrecarga de RDDs (Fonte).

# DAG (DIRECTED ACYCLIC GRAPH)



# Onde é usado

- Netflix - recomendação (junto com machine learning)
- https://pt.slideshare.net/SessionsEvents/ehtsham-elahi-senior-research-engineer-personalization-science-and-engineering-group-at-netflix-at-mlconf-sea-50115
-

# A API

# Graphx

Apache Spark GraphX

org.apache.spark.graphx

## Graph

Related Docs: object Graph | package graphx

```
abstract class Graph[VD, ED] extends Serializable
```

The Graph abstractly represents a graph with arbitrary objects associated with vertices and edges. The graph provides basic operations to access and manipulate the data associated with vertices and edges as well as the underlying structure. Like Spark RDDs, the graph is a functional data-structure in which mutating operations return new graphs.

| | |
|---|---|
| **VD** | the vertex attribute type |
| **ED** | the edge attribute type |

- Property graph: multi grafo direcionado com objetos anexados a cada vértice e aresta.
- São imutáveis, distribuídos e tolerante a falhas.
- Cada partição do grafo pode ser recriada em uma máquina diferente caso haja falha.

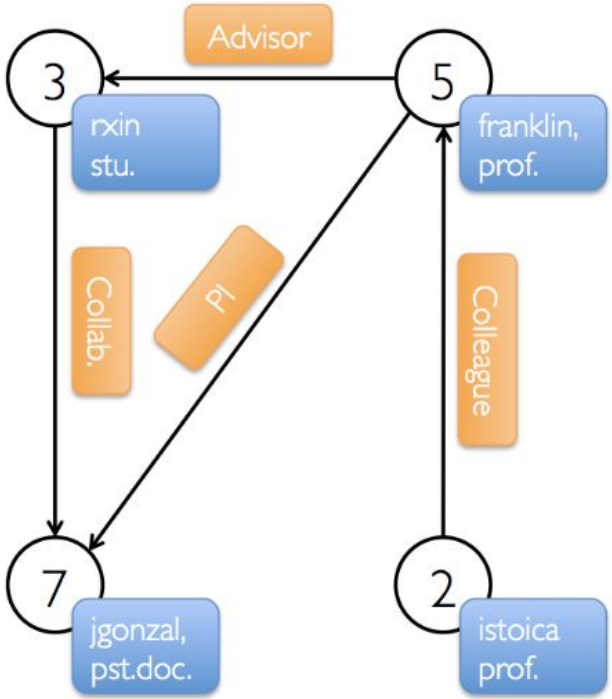
# Graphx

```
class Graph[VD, ED] {
    val vertices: VertexRDD[VD]
    var edges: EdgeRDD[ED]
}
```

# Graphx - exemplificando

APACHE Spark GraphX

## Property Graph




## Vertex Table

| Id | Property (V) |
|---|---|
| 3 | (rxin, student) |
| 7 | (jgonzal, postdoc) |
| 5 | (franklin, professor) |
| 2 | (istoica, professor) |

## Edge Table

| SrcId | DstId | Property (E) |
|---|---|---|
| 3 | 7 | Collaborator |
| 5 | 3 | Advisor |
| 2 | 5 | Colleague |
| 5 | 7 | PI |

# Graphx - exemplificando

Vertex Table

| Id | Property (V) |
|---|---|
| 3 | (rxin, student) |
| 7 | (jgonzal, postdoc) |
| 5 | (franklin, professor) |
| 2 | (istoica, professor) |

```
// Create an RDD for the vertices
val users: RDD[(VertexId, (String, String))] =
  sc.parallelize(Array((3L, ("rxin", "student")), (7L, ("jgonzal", "postdoc")),
                       (5L, ("franklin", "prof")), (2L, ("istoica", "prof"))))
```

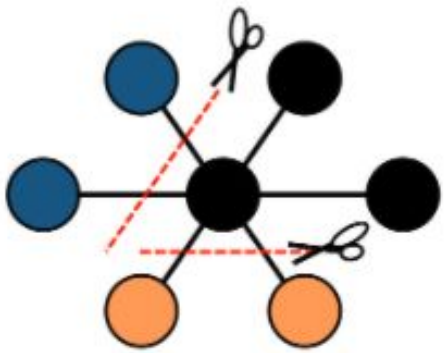
# Graphx - exemplificando

Edge Table

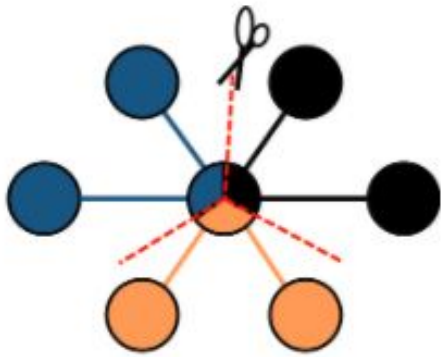
| SrcId | DstId | Property (E) |
|---|---|---|
| 3 | 7 | Collaborator |
| 5 | 3 | Advisor |
| 2 | 5 | Colleague |
| 5 | 7 | PI |

```
// Create an RDD for edges
val relationships: RDD[Edge[String]] =
  sc.parallelize(Array(Edge(3L, 7L, "collab"),    Edge(5L, 3L, "advisor"),
                       Edge(2L, 5L, "colleague"), Edge(5L, 7L, "pi")))
```

# PROPERTY GRAPHS - otimizações

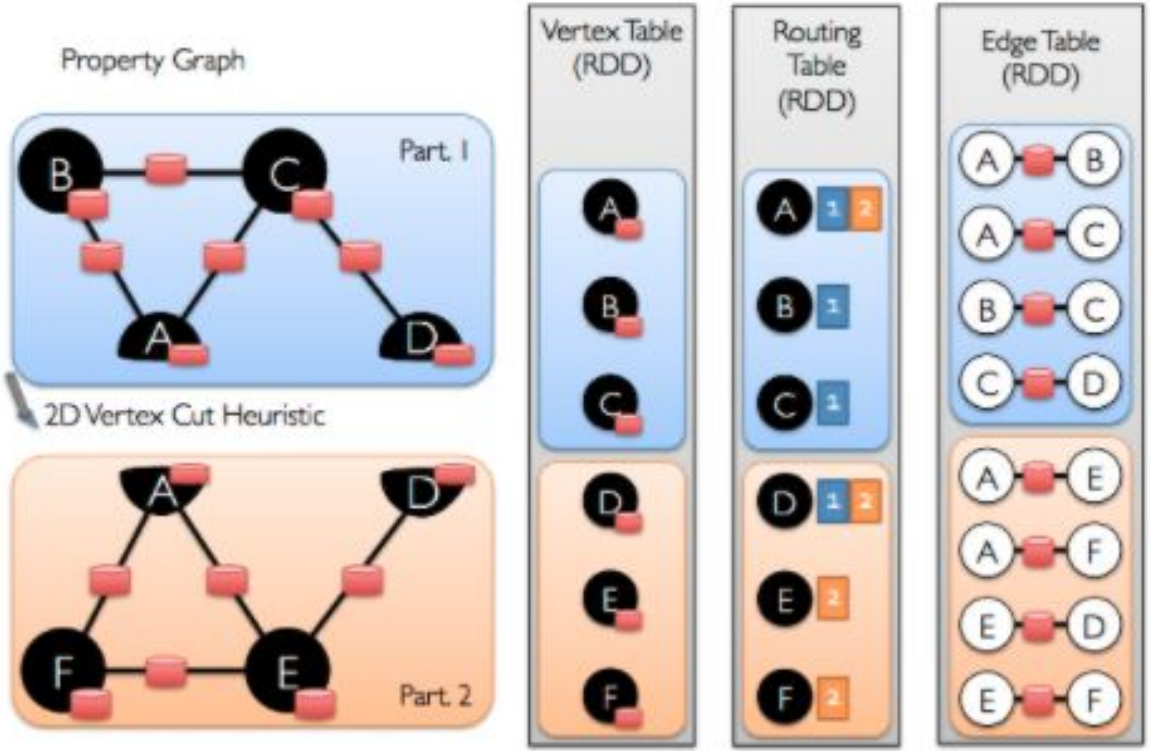


Edge Cut



Vertex Cut

# PROPERTY GRAPHS - otimizações



# PROPERTY GRAPHS - otimizações

org.apache.spark.graphx

## PartitionStrategy

Companion trait PartitionStrategy

```
object PartitionStrategy extends Serializable
```

Collection of built-in PartitionStrategy implementations.

***Source*** PartitionStrategy.scala

Linear Supertypes

Filter all members

### Value Members

```
def fromString(s: String): PartitionStrategy
```
Returns the PartitionStrategy with the specified name.

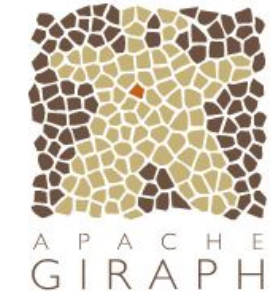
```
object CanonicalRandomVertexCut extends PartitionStrategy with Product with Serializable
```
Assigns edges to partitions by hashing the source and destination vertex IDs in a canonical direction, resulting in a random vertex cut that colocates all edges between two vertices, regardless of direction.

```
object EdgePartition1D extends PartitionStrategy with Product with Serializable
```
Assigns edges to partitions using only the source vertex ID, colocating edges with the same source.

```
object EdgePartition2D extends PartitionStrategy with Product with Serializable
```
Assigns edges to partitions using a 2D partitioning of the sparse edge adjacency matrix, guaranteeing a `2 * sqrt(numParts)` bound on vertex replication.

```
object RandomVertexCut extends PartitionStrategy with Product with Serializable
```
Assigns edges to partitions by hashing the source and destination vertex IDs, resulting in a random vertex cut that colocates all same-direction edges between two vertices.


# FRAMEWORKS RELACIONADOS

# Pregel

- Sistema da Google, alternativo ao Spark GraphX, para processamento de grafos de larga escala;
- Escalável e tolerante a falhas;
- Alimenta o PageRank da ferramenta de pesquisa;
- Inspiração para o Apache Giraph, que o Facebook usa para analisar seu grafo de rede social;

# Apache Giraph

Apache Spark GraphX

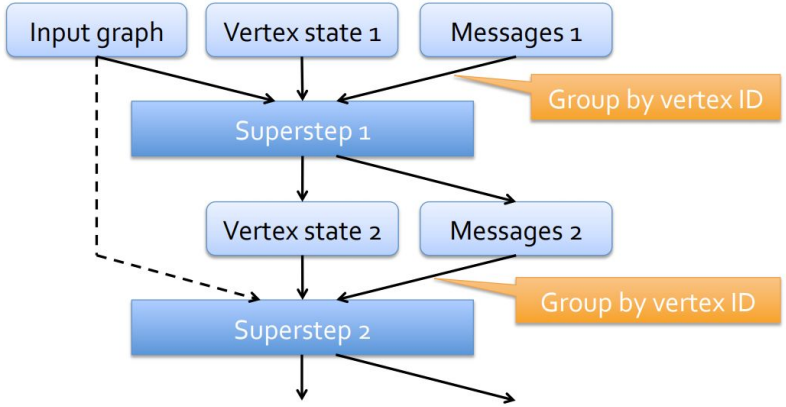
- Usado no Facebook para analisar o grafo de rede social formado pelos usuários e suas conexões;
- Giraph se originou como a contraparte open-source para o Pregel;
- Adiciona vários recursos além do modelo Pregel básico, incluindo computação mestre, agregadores fragmentados, entrada de conjunto de arestas, computação fora do núcleo;

# Pregel & GraphX- comparação

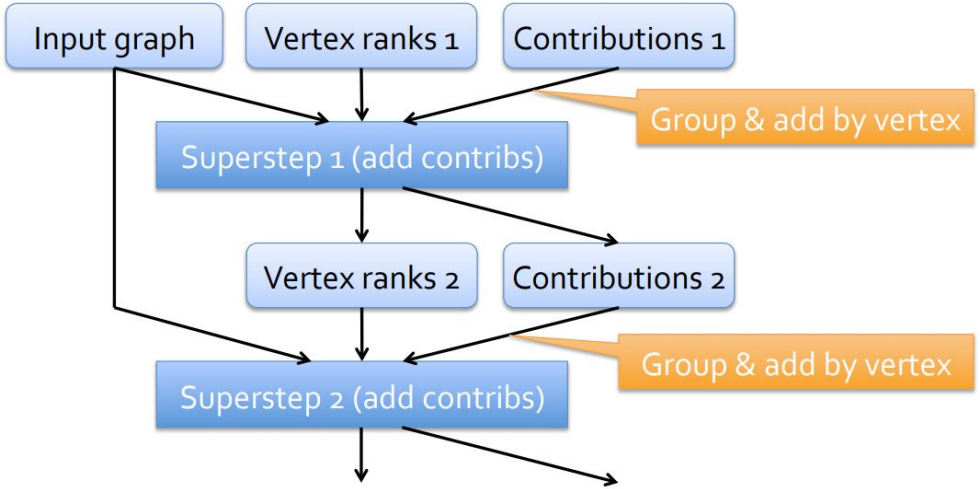
- Expõem API específicas para simplificar programação em grafos
- Se baseiam no padrão em que a computação depende apenas dos vizinhos
- Expressam os jobs como grafos de operadores de alto nível
  - O sistema escolhe como dividir cada operador em tarefas e onde executar cada tarefa
  - Executam partes para recuperação de falhas duas vezes
- GraphX roda o modelo do Pregel por trás de suas tabelas de vértices e arestas

# Pregel & GraphX- comparação

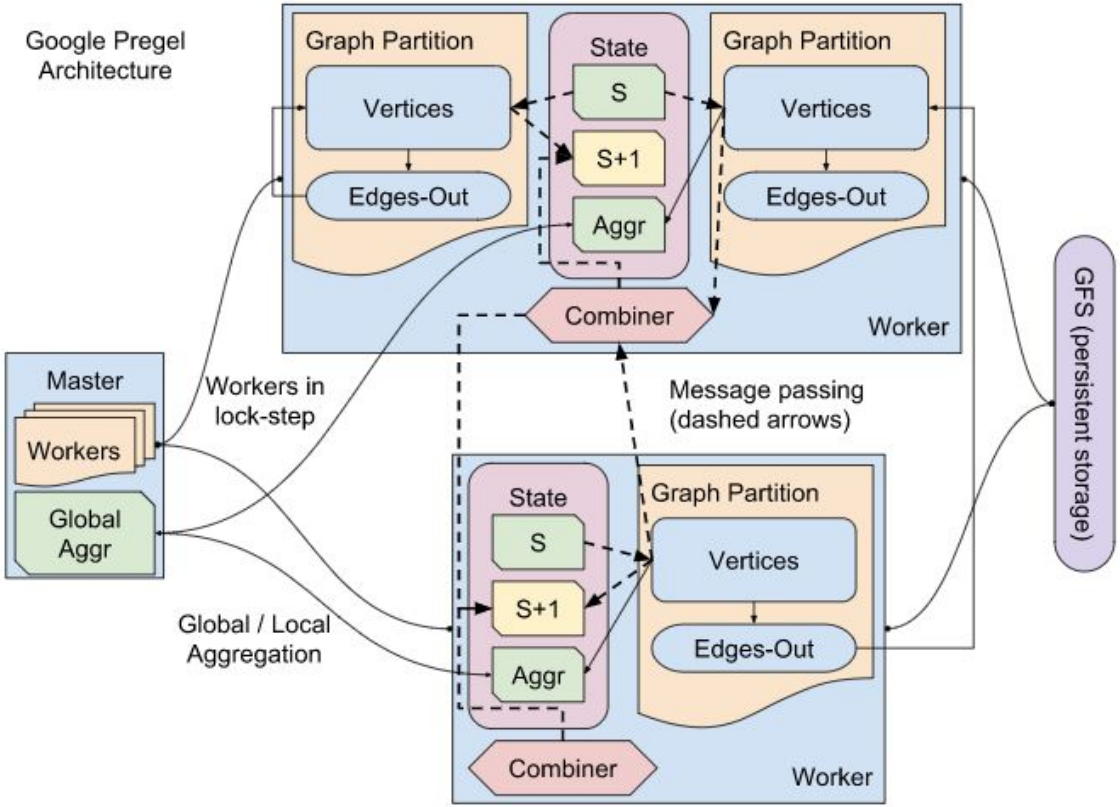
## Pregel Data Flow



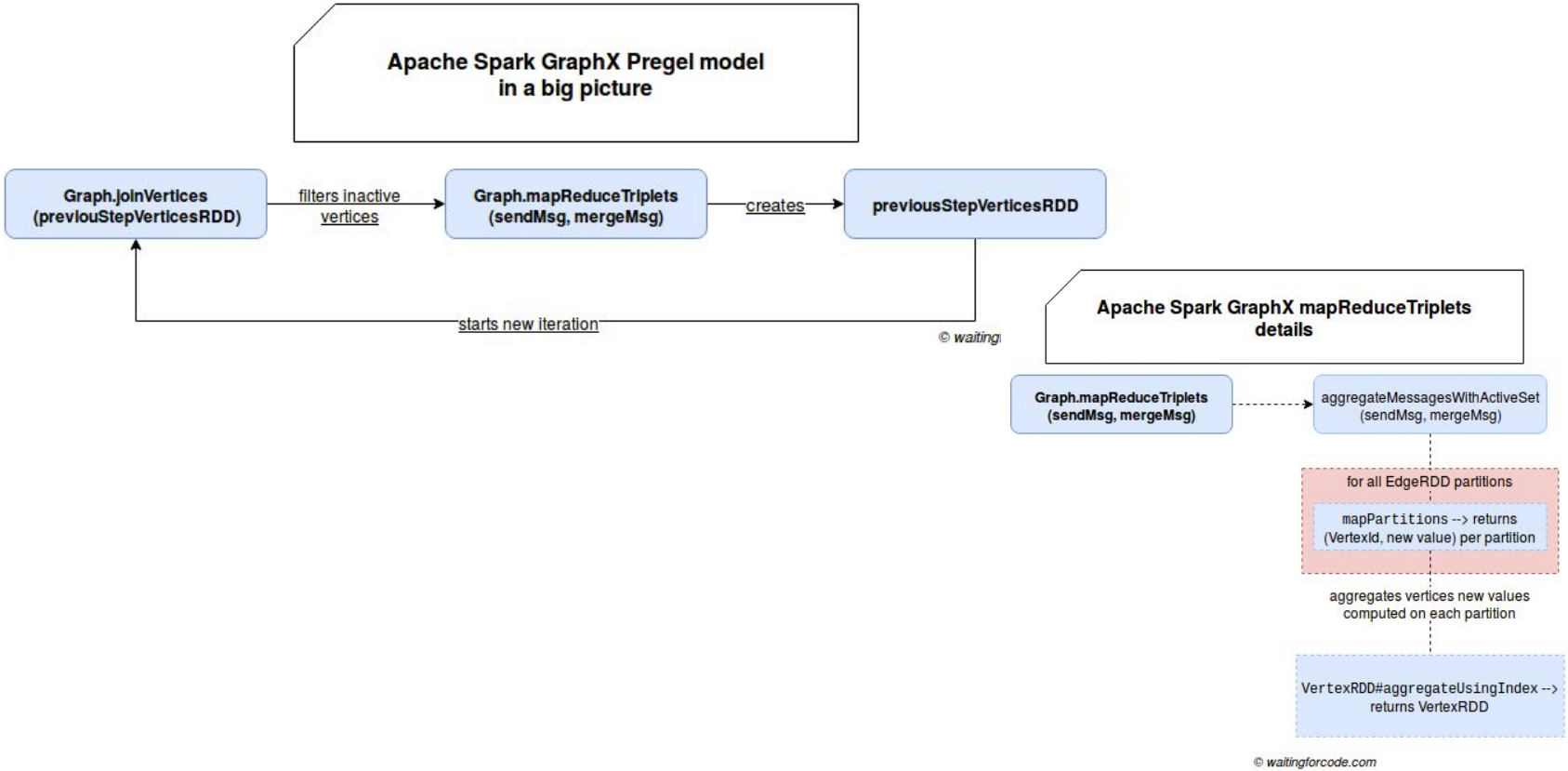

## PageRank in Pregel



# Pregel & GraphX- comparação



# Pregel & GraphX- comparação



# FUNCIONAMENTO

# Funcionamento

debora_barbosa@graphx-cluster-m: /usr/lib/spark/data/graphx/bcc362 - Google Chro...

ssh.cloud.google.com/projects/gcp-graphx-sd/zones/southamerica-east1-b/instances/graph...

```
debora_barbosa@graphx-cluster-m:/usr/lib/spark/data/graphx/bcc362$ ls
README.md  cit-HepTh-abstracts.tar.gz  cit-HepTh.txt
debora_barbosa@graphx-cluster-m:/usr/lib/spark/data/graphx/bcc362$ hdfs dfs -put
cit-HepTh.txt bcc362
debora_barbosa@graphx-cluster-m:/usr/lib/spark/data/graphx/bcc362$ hdfs dfs -ls b
cc362
Found 1 items
-rw-r--r--   2 debora_barbosa hadoop    5576648 2021-02-22 22:44 bcc362/cit-HepTh
.txt
debora_barbosa@graphx-cluster-m:/usr/lib/spark/data/graphx/bcc362$
```

ssh.cloud.google.com/projects/gcp-graphx-sd/zones/southamerica-east1-b/instances/graphx-cluster-m?useAdminProxy=true&authuser...

```
GNU nano 2.9.3                    cit-HepTh.txt

# Directed graph (each unordered pair of nodes is saved once): Cit-HepTh.txt
# Paper citation network of Arxiv High Energy Physics Theory category
# Nodes: 27770 Edges: 352807
# FromNodeId    ToNodeId
1001    9304045
1001    9308122
1001    9309097
1001    9311042
1001    9401139
1001    9404151
1001    9407087
1001    9408099
1001    9501030
1001    9503124
1001    9504090
1001    9504145
1001    9505025
1001    9505054
1001    9505105
1001    9505162
1001    9506048
1001    9506112
1001    9506144
1001    9507050
1001    9507158
1001    9508094
1001    9508155
1001    9510142
1001    9510225
1001    9510234
1001    9511030
1001    9511171
1001    9601108
1001    9602022
1001    9602114
1001    9603003

^G Get Help    ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify    ^C Cur Pos    M-U Undo
^X Exit        ^R Read File    ^\ Replace     ^U Uncut Text  ^T To Spell   ^_ Go To Line  M-E Redo
```

Não seguro | 34.95.148.125:18080/history/application_1614206849165_0003/executors/

- [ ] Off Heap Memory
- [ ] Peak JVM Memory OnHeap / OffHeap
- [ ] Peak Execution Memory OnHeap / OffHeap
- [ ] Peak Storage Memory OnHeap / OffHeap
- [ ] Peak Pool Memory Direct / Mapped
- [ ] Resources
- [ ] Resource Profile Id

## Summary

| | RDD Blocks | Storage Memory | Disk Used | Cores | Active Tasks | Failed Tasks | Complete Tasks | Total Tasks | Task Time (GC Time) | Input | Shuffle Read | Shuffle Write | Excluded |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Active(9)** | 0 | 0.0 B / 11.6 GiB | 0.0 B | 8 | 1 | 0 | 7936 | 7937 | 20 min (1.4 min) | 130.7 MiB | 360.3 MiB | 363 MiB | 0 |
| **Dead(1)** | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 1 | 1 | 6 s (0.2 s) | 675 KiB | 0.0 B | 352.3 KiB | 0 |
| **Total(10)** | 0 | 0.0 B / 12.8 GiB | 0.0 B | 9 | 1 | 0 | 7937 | 7938 | 20 min (1.4 min) | 131.4 MiB | 360.3 MiB | 363.4 MiB | 0 |

## Executors

Show 20 entries

Search:

| Executor ID | Address | Status | RDD Blocks | Storage Memory | Disk Used | Cores | Active Tasks | Failed Tasks | Complete Tasks | Total Tasks | Task Time (GC Time) | Input | Shuffle Read | Shuffle Write | Logs |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| driver | graphx-cluster-m.c.gcp-graphx-sd.internal:37727 | Active | 0 | 0.0 B / 1.8 GiB | 0.0 B | 0 | 0 | 0 | 0 | 0 | 0.0 ms (0.0 ms) | 0.0 B | 0.0 B | 0.0 B | |
| 1 | graphx-cluster-w-0.c.gcp-graphx-sd.internal:43175 | Dead | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 1 | 1 | 6 s (0.2 s) | 675 KiB | 0.0 B | 352.3 KiB | stdout stderr |
| 2 | graphx-cluster-w-2.c.gcp-graphx-sd.internal:40085 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 1 | 0 | 4231 | 4232 | 9.5 min (40 s) | 130.7 MiB | 215.9 MiB | 265.4 MiB | stdout stderr |
| 3 | graphx-cluster-w-1.c.gcp-graphx-sd.internal:46527 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 79 | 79 | 24 s (0.3 s) | 0.0 B | 2.8 MiB | 2.4 MiB | stdout stderr |
| 4 | graphx-cluster-w-0.c.gcp-graphx-sd.internal:34099 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 82 | 82 | 26 s (0.5 s) | 0.0 B | 3 MiB | 2.5 MiB | stdout stderr |
| 5 | graphx-cluster-w-3.c.gcp-graphx-sd.internal:44139 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 80 | 80 | 30 s (0.5 s) | 0.0 B | 2.9 MiB | 2.5 MiB | stdout stderr |
| 6 | graphx-cluster-w-3.c.gcp-graphx-sd.internal:37297 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 79 | 79 | 28 s (0.6 s) | 0.0 B | 2.9 MiB | 2.3 MiB | stdout stderr |
| 7 | graphx-cluster-w-1.c.gcp-graphx-sd.internal:46207 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 83 | 83 | 26 s (0.3 s) | 0.0 B | 3.1 MiB | 2.5 MiB | stdout stderr |
| 8 | graphx-cluster-w-2.c.gcp-graphx-sd.internal:42305 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 3220 | 3220 | 7.7 min (41 s) | 0.0 B | 126.7 MiB | 83.1 MiB | stdout stderr |
| 9 | graphx-cluster-w-0.c.gcp-graphx-sd.internal:33975 | Active | 0 | 0.0 B / 1.2 GiB | 0.0 B | 1 | 0 | 0 | 82 | 82 | 24 s (0.5 s) | 0.0 B | 2.9 MiB | 2.4 MiB | stdout stderr |

Showing 1 to 10 of 10 entries

Previous 1 Next

Não seguro | 34.95.148.125:18080/history/application_1614206849165_0003/stages/

Spark 3.1.1 Jobs Stages Storage Environment Executors PythonPageRank application UI

## Stages for All Jobs

**Completed Stages:** 124
**Skipped Stages:** 78
**Failed Stages:** 1

### Completed Stages (124)

Page: 1 2 >

2 Pages. Jump to 1 . Show 100 items in a page. Go

| Stage Id | Description | | Submitted | Duration | Tasks: Succeeded/Total | Input | Output | Shuffle Read | Shuffle Write |
|---|---|---|---|---|---|---|---|---|---|
| 123 | reduceByKey at /usr/lib/spark/examples/src/main/python/pagerank.py:80 | +details | 2021/02/24 23:37:24 | 12 s | 124/124 | | | 3.1 MiB | 4.7 MiB |

Não seguro | 34.95.148.125:18080/history/application_1614206849165_0003/stages/stage/?id=123&attempt=0

Spark 3.1.1 Jobs Stages Storage Environment Executors PythonPageRank application UI

# Details for Stage 123 (Attempt 0)

**Resource Profile Id:** 0
**Total Time Across All Tasks:** 26 s
**Locality Level Summary:** Node local: 112; Rack local: 12
**Shuffle Read Size / Records:** 3.1 MiB / 7441
**Shuffle Write Size / Records:** 4.7 MiB / 30543
**Associated Job Ids:** 0

- DAG Visualization
- Show Additional Metrics
- Event Timeline
- Enable zooming

Scheduler Delay
Task Deserialization Time
Shuffle Read Time
Executor Computing Time
Shuffle Write Time
Result Serialization Time
Getting Result Time

Tasks: 124. 2 Pages. Jump to 1 . Show 100 items in a page. Go


## Summary Metrics for 124 Completed Tasks

| Metric | Min | 25th percentile | Median | 75th percentile | Max |
|---|---|---|---|---|---|
| Duration | 0.1 s | 0.1 s | 0.2 s | 0.2 s | 2 s |
| GC Time | 0.0 ms | 0.0 ms | 0.0 ms | 0.0 ms | 1 s |

00:46
25/02/2021

| número de iterações | tempo de execução |
|---|---|
| 1 | 26s |
| 10 | 50s |
| 50 | 6.7min |
| 100 | 10min |

# Referências

Michael Malak, Robin East - Spark GraphX in Action (2016, Manning Publications) - libgen.lc

https://www.youtube.com/watch?v=Y7hq5MudV9M GraphX: Graph Analytics in Spark - Ankur Dave (UC Berkeley)
Ankur Dave: ankus is a third-year phd student advised by ion stoica in the UC Berkeley AMPLab. He's a Spark commiter and a maintainer for GraphX

https://spark.apache.org/docs/latest/graphx-programming-guide.html#vertexrdds - graphx programming guide

# Referências

https://stanford.edu/~rezab/classes/cme323/S16/notes/Lecture16/Pregel_GraphX.pdf - pregel

https://blog.acolyer.org/2015/05/26/pregel-a-system-for-large-scale-graph-processing/

https://www.waitingforcode.com/graphx/iterative-algorithms-pregel-apache-spark-graphx/read

https://www.doc.ic.ac.uk/~nuric/sysadmin/how-does-google-pregel-work.html#pregel