

Conceptos de Procesamiento in-memory con Spark

Computación de Alto Desempeño
Diego Alberto Rincón Yáñez MCSc
diego-rincon@javeriana.edu.co

Procesamiento in-memory



Diego Alberto Rincón Yáñez MCSc.
Twitter: @d1egoprog.

Historia

- Nació Originalmente para soluciones de BI
- Mejora de las soluciones de consultas tradicionales en disco
- Acceso a procesamiento de 64 bits



Definición

En cómputo in-memory, los datos son conservados en random Access memory (RAM), en lugar de discos. Adicionalmente de ser procesada en paralelo.

Columnas de computación in-memory

- Almacenamiento en RAM
- Procesamiento paralelo y distribuido



Diego Alberto Rincón Yáñez MSc.
Twitter: @d1egoprog.

Habilitadores

Velocidad en el despliegue

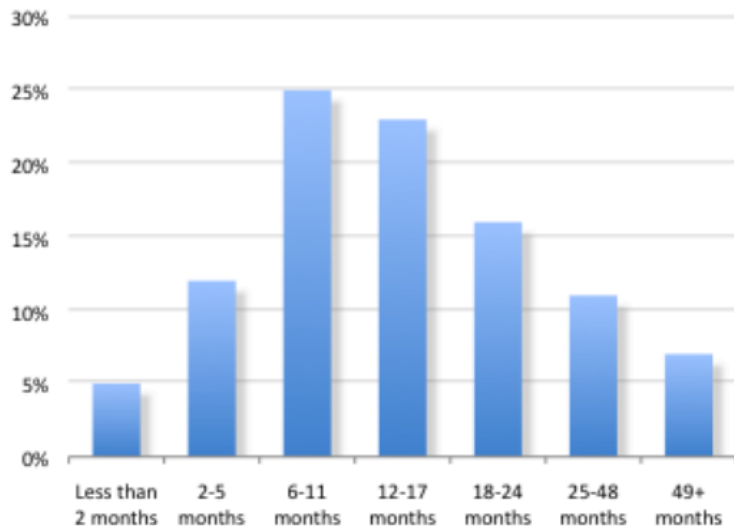
Remueve la complejidad de construir una capa de rendimiento basada en discos.

Bajar las peticiones y requerimientos de IT si se habilitan aplicaciones de procesamiento in-memory

Bajo desempeño en consultas baja la adopción del usuario.



Tiempos de Entrega en Proyectos de Analítica Tradicionales



Average implementation time:

17 months in total; 5 months to deploy the 1st usable analytic application

Mean annual expenditure on BI software:

\$1.1 million for companies with >1,000 employees

Project Success Rate:

31% success rate; at best

Meeting needs - right data to the right person:

only 36% are confident that reports and dashboards deliver the right data to the right person at the right time.

IDC Business Intelligence Survey



Diego Alberto Rincón Yáñez MCSc.
Twitter: @d1egoprog.

Velocidad en el Análisis

Visualización interactiva basada en aplicaciones analíticas con un fuerte componente de front-end.

Interactuar con datos visualizados requiere de tareas intuitivas, lo que habilita a usuarios más inexperimentados.

Bajo desempeño en consultas baja la adopción del usuario.



Habilitadores



Diego Alberto Rincón Yáñez MCSc.
Twitter: @d1egoprog.

Retos Técnicos

- Gran Impacto en tiempos de I/O y Recursos de red.
- Agregación a consultas produce tiempos de respuesta elevados.

Today	IT Pain	Business Pain
More Data Gb to Tb	Higher System Costs	Impact to financial performance
More end users 100's to 1000's	Slower Performance	Delayed decisions or responses
More Complex queries Ad hoc reports and analysis	Missed reporting windows	Missed deadlines and penalties



Diego Alberto Rincón Yáñez MCSc.
Twitter: @d1egoprog.

Beneficios de Procesamiento in-memory

- Mejoras dramáticas en el desempeño
- Diferente aproximación a “caching”
- Almacenamiento más grande en RAM
- Conservar los datos en memoria mejora el rendimiento en ordenes de magnitudes.
- Optimizado para procesos de ML y micro-batch.
- Provee ejecución rápida para trabajos iterativos.
- Los datos están listos la momento de analizarlos.
- Ideal para administración de riesgos en tiempo real y detección de fraudes.



Diego Alberto Rincón Yáñez MSc.
Twitter: @d1egoprog.

Desventajas de Procesamiento in-memory

- Falta de soporte para procesos en tiempo real.
- Problemas con archivos pequeños (más eficiente pocos archivos grandes a muchos archivos pequeños).
- No hay manejador de archivos (debe usar otra plataforma como HDFS).
- Costoso, el acceso a todos los datos en memoria puede llegar a crear cuellos de botella en el proceso de carga.
- Menor cantidad de algoritmos (MLlib)
- Latencia, para altas cargas de procesamiento en streaming se recomienda usar Apache Flink





Apache SPARK



Diego Alberto Rincón Yáñez MSc.
Twitter: @d1egoprogram

¿Qué es?

Es un sistema de computación en clúster que provee un API de alto nivel en JAVA, Scala Python y R.

Puede acceder datos desde HDFS, Cassandra, Hbase, Hive, etc...



Diego Alberto Rincón Yáñez MSc.
Twitter: @d1egoprog.

Historia

- Creado en el 2009 en UC Berkeley R&D Lab, luego AMPLab
- Creado bajo licencia BSD
- En 2013 fue donado a Apache Software Foundation.
- En 2014 se convirtió en proyecto de alto nivel



Ecosistema de Spark



...

Spark Core

SQL

Streamming

MLlib

GraphX

SparkR



Diego Alberto Rincón Yáñez MCSc.
Twitter: @d1egoprog.

Introducción a RDD

Es la estructura de datos fundamental de SPARK, es una colección inmutable de objetos.

- **Resilient:** Tolerante a fallos, es capaz de reconstruirse, en caso de fallos o particiones dañadas debido a fallos de los nodos.
- **Distributed:** Los datos residen en múltiples nodos.
- **Dataset:** representa los registros de los datos en si trabajo, el usuario puede cargarlos desde JSON, CSV, Texto, base de datos, a través de JDBC sin estructura específica.



¿Porque RDD?

- Algoritmos Iterativos
- Herramientas de Data mining interactivas
- DSM (Distributed Shared Memory)
- Evaluado de forma perezosa (Solo cuando es necesitado)
- Distribuida por Naturaleza
- Tolerante a Fallos



RDD vs DSM

Lec/Esc: las lecturas pueden ser a nivel granular fino o grueso

Consistencia: Debido a su inmutabilidad cualquier cambio es permanente.

Tolerancia: Debido a que cada vez que se transforma se crea un RDD nuevo su recuperación es casi transparente.

Lec/Esc : Las operaciones son a nivel granular fino

Consistencia: El programador debe seguir las reglas de consistencia.

Tolerancia: Se aplica gracias a guardados de tipo checkpoint.



Manejando RDD's

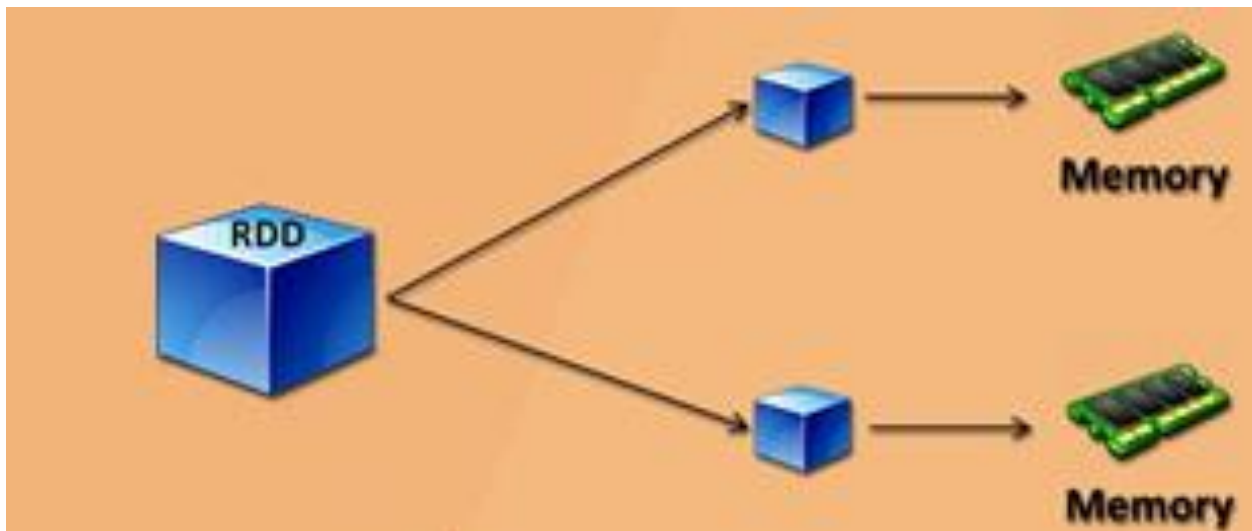
- **Métodos de Operación**
 - **Transformación:** Crea un RDD a partir de uno existente, como resultado de una operación
 - **Acción:** Retorna el resultado final para escritura de un dataset
- **Métodos de Creación**
 - Colecciones Paralelizadas
 - Data Sets Externos
 - RDD's Existentes



Métodos de Acción

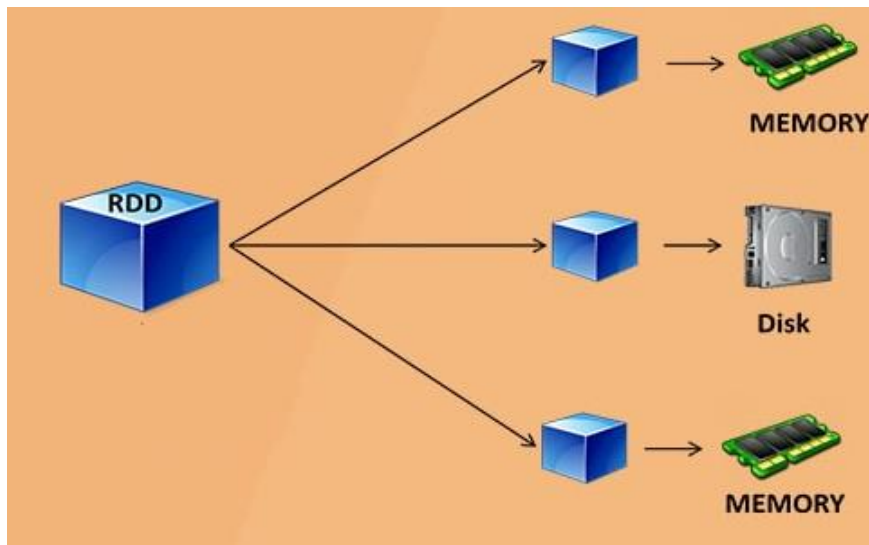
Sólo en Memoria

```
rdd.persist(MEMORY_only)
```



Métodos de Acción – Memoria y Disco

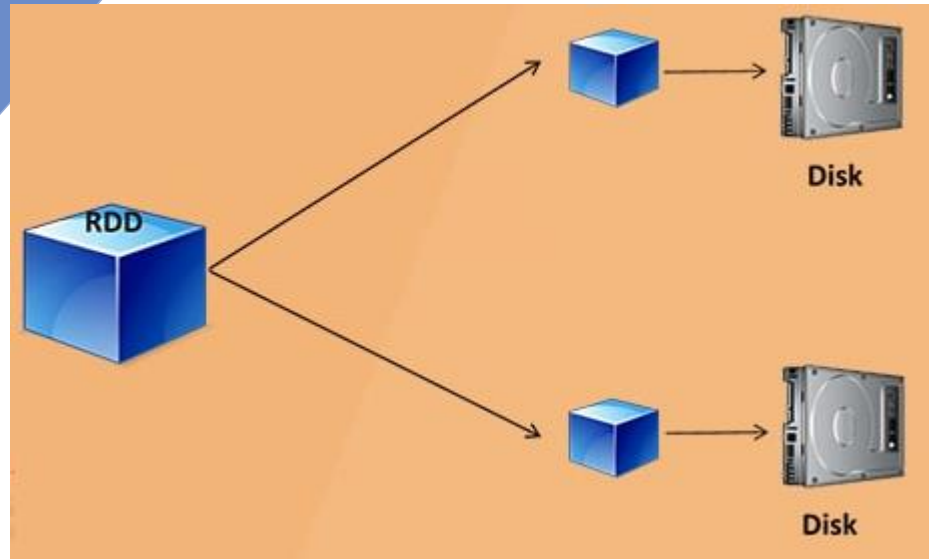
```
rdd.persist(MEMORY_AND_DISK)
```



Diego Alberto Rincón Yáñez MSc.
Twitter: @d1egoprog.

Métodos de Acción – Sólo en Disco

```
rdd.persist(DISK_ONLY)
```

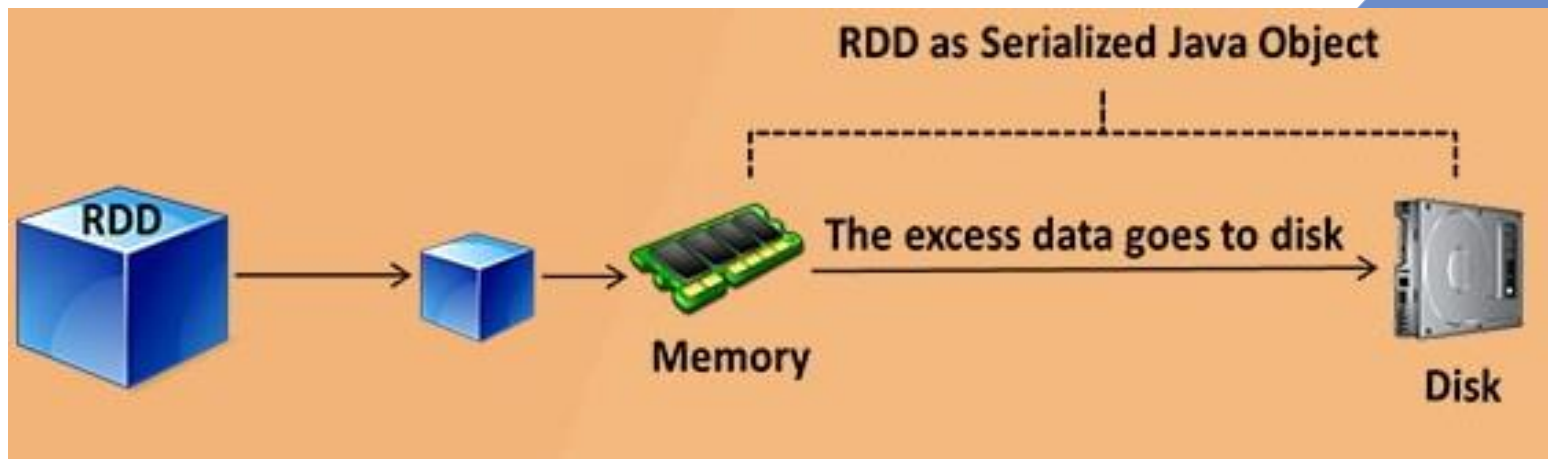


Diego Alberto Rincón Yáñez MCSc.
Twitter: @d1egoprogram

Métodos de Acción

Sólo en Memoria

```
rdd.persist(MEMORY_AND_DISK_SER)
```





Questions
are
guaranteed in
life;
Answers
aren't.

¿Preguntas?



Diego Alberto Rincón Yáñez MCSc.
Twitter: @d1egoprog.