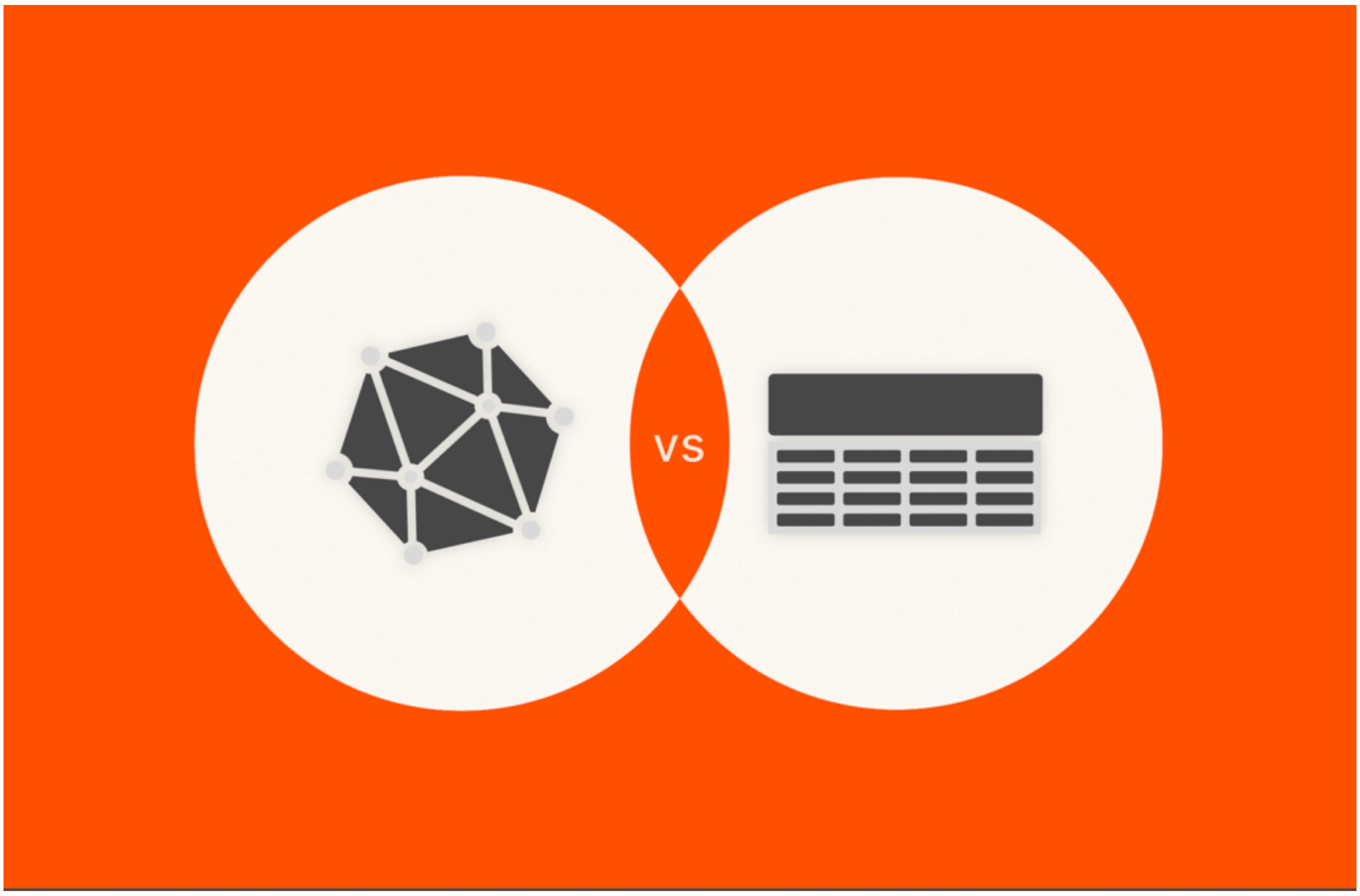
RDD vs. DataFrame: What's The Difference?



To speed up performance in data analytics, Apache Spark uses two storage organization strategies: resilient distributed datasets (RDDs) and DataFrames. RDDs use collections of data across multiple nodes, while DataFrames distribute data in columns, similar to a relational database table. Here, we will discuss RDD vs. DataFrame and what the differences are.

What Is RDD?

The original API for Apache Spark was RDD, which is a collection of data objects across nodes in an Apache Spark cluster. The distributed data speeds up delivery to end users, but RDD's immutable functionality is what makes it fault-tolerant. Immutability avoids data corruption from updates by creating new data from existing data rather than overwriting it. The main feature in RDD is immutable data distributed across numerous servers. RDD also performs computations in memory for better performance.

How Do RDDs Work?

RDDs work by distributing partitioned data across multiple servers represented as unstructured blocks of data. Because the data is immutable, it's never updated but recreated when changes are made. Developers query the database using the RDD API, mainly for data like media or large blocks of text.

Developers working with RDDs do not need to determine or define a structure. The API returns a data set in a developer-defined structure such as a JSON or CSV file. Partitions of data can be stored in memory or on disk, depending on performance. Even with in-memory computations, the immutable feature can harm performance since data must be recreated entirely rather than updated.

What Is a DataFrame?

The next step in Apache Spark data development is DataFrames. A DataFrame is similar to a standard database table where the schema is laid out into columns and rows. Developers familiar with structured databases will appreciate the DataFrame API in Apache Spark. Data is laid out in columns, and queries can be optimized for performance.

How Do DataFrames Work?

DataFrames work by storing data in structured columns. Every column has an identifier used to retrieve and filter data in developer queries. Because of their structured nature, DataFrames are used in several libraries and APIs to query and store data. For example, the popular Python library pandas uses DataFrames to retrieve data in analysis projects.

Storing data requires developers to set a type, and indexes on columns commonly used to query data use indexes to speed up performance. Data can be updated, added to the DataFrame structure, or created from an imported file. Think of a DataFrame as a spreadsheet of information that can be used to store millions of records.

RDD vs. DataFrame

RDD is beneficial for applications using unstructured data. For example, you would use RDD in Apache Spark for analytics and machine learning. A DataFrame uses structured data, so it's best used when you know the data type for each column and can fit data into a predefined column. Both solutions can work with structured and unstructured data, but the solution you choose depends on your use case.

If you don't know the structure of your data and need analytics calculations, RDD is the best choice. RDD is often used with Java, Scala, R, and Python.

DataFrames are best used with structured data (although they can also work with unstructured data). They're often used with files or exports in JSON and CSV formats. Java, Scala, R, and Python can also work with DataFrames.

Conclusion

To work with Apache Spark and RDD and DataFrames, your data pipelines need a solution to store it. Pure Storage® FlashBlade® is a fast performance storage solution built specifically for distributed data and Apache Spark. Utilize it to build applications for Apache Spark using structured or unstructured data. No matter your preferred programming language or application, Pure Storage data storage solutions can support your resource needs.

Achieving

IT Agility

How the Pure Storage data management platform delivers tangible outcomes for customers.

Read the Ebook

Written By:

Jacob Yothment