# Apache Spark is here to stay

**Giovanna Roda[a], Lukas Linauer[a], Dieter Kvasnicka[a,b]**

[a] *TU Wien, TU.it*
[b] *VSC Research Center*

Spark began as a research project at the UC Berkeley in 2009 and was released to the open source the following year [2]. The aim was to improve and exceed the *MapReduce* computing engine while maintaining its benefits: scalability and fault-tolerance.

MapReduce is a programming paradigm and framework designed to process massive amounts of data on a distributed computing architecture based on a split-apply-combine strategy. While at its core are the customary *mapper* and *reducer* functions from functional programming (similar to the MPI *scatter* and *reduce* operations), MapReduce refers to the whole system responsible of partitioning the input data, scheduling task execution on a cluster and handling failures [1].

One of the main performance bottlenecks of MapReduce is the mapper's output being written to disk. In addition to that, every data transformation requires a new job, which may become costly for complex pipelines. Spark addresses these issues with its fundamental data abstraction: the RDD (Resilient Distributed Dataset).

RDDs are immutable, partitioned collections of records that can be operated on in parallel. Lazy evaluation, as well as in-memory processing of RDDs, are the core features of Spark, that enable it to outperform MapReduce by orders of magnitude.

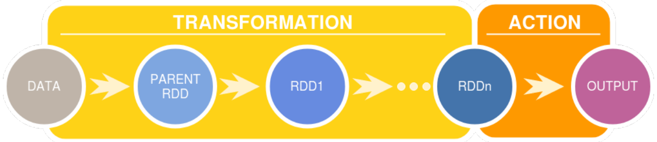

**Fig.1:** RDD: transformations are evaluated lazily, i.e. they won't be executed until an action is performed.

Born within the Apache Hadoop ecosystem with its standard resource manager YARN (Yet Another Resource Negotiator), Spark can also be deployed as standalone on any cluster, including Kubernetes- and Slurm-managed clusters.

Is Apache Spark here to stay? Spark is a general-purpose tool for distributed computing and thanks to its SQL and Dataframe APIs, as well as libraries for Graph computations and Machine Learning, it is popular among data scientists. Additionally, scientific fields such as bioinformatics, high energy physics, and geosciences increasingly rely on the use of Spark.

Spark is available on the Little Big Data cluster at the TU Wien in a Hadoop environment (864 cores with hyper-threading, `http://lbd.zserv.tuwien.ac.at`), as well as on the High Performance clusters of the Vienna Scientific Cluster (≈ 50000 cores, `http://vsc.ac.at`), where a standalone Spark cluster can readily be launched by a Slurm script that takes care of allocating the necessary resources.

## References

[1] Dean J., Ghemawat S.. MapReduce: Simplified Data Processing on Large Clusters, OSDI'04: Sixth Symposium on Operating System Design and Implementation, 137–150, 2004.

[2] Zaharia M., M. Chowdhury M.,Franklin M.J., Shenker S., Stoica I.. 2010. Spark: cluster computing with working sets. In Proceedings of the 2nd USENIX conference on Hot topics in cloud computing (HotCloud'10).