On Next Generation Big Data Analytics Systems

A Position Paper by
Prof.-Dr. Volker Markl[1], *Technische Universität Berlin*
2nd BDEC (Big Data and Extreme-scale Computing) **International Workshop**
February 27-28, 2014 in Fukuoka, Japan

The last decade was marked by the digitalization of virtually all aspects of our daily lives. Today, businesses, government institutions, and science and engineering organizations, face an avalanche of digital data on a daily basis. All due in part to the decline in disk storage costs, the ever-increasing popularity of cloud storage services, and the ubiquitous availability of networked devices. At first glance this appears to be favorable for our increasingly networked society. However, in many ways it is a burden. Data is neither information, nor knowledge. Instead, data is of great value once it has been refined and analyzed, in order to address well-formulated questions, concerning problems of interest. It is only then that economic and social benefits can be fully realized.

These questions are generally reformulated and solved using techniques drawn from varying fields, including graph and network analysis, machine learning, mathematics, statistics, signal processing, and text processing, among others. Currently, data scientists, well versed in scalable data analysis methods, scalable systems programming, and knowledge in an application domain are needed to derive insight from big data. Unfortunately, data scientists are few in number. They are expensive and in high-demand. Consequently, this limits the amount of value that can currently be generated from big data for society as a whole. Moreover, despite the ever-increasing number of data science programs (at universities worldwide) and student enrollments, it will still be impossible to educate these so-called "jack-of-all-trades," as the required skills are both complex and diverse (as depicted in **Figure 1** below).

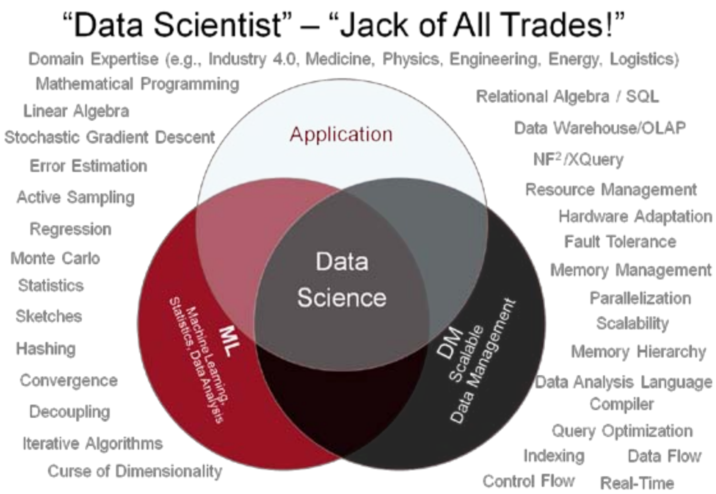

Figure 1. A Data Scientist is a "Jack-of-All-Trades"

Before "big data," the few programmers with MPI expertise, predominantly located in supercomputing centers were sufficient in number. For many decades, software engineers and general users in varying domains did not have to worry about scalability issues in their computing systems, thanks in part to higher-level programming languages, compilers, and database systems. In contrast, today's existing technologies have reached their limits due to big data requirements, which involve data volume, data rate and heterogeneity, and the complexity of the analysis algorithms, which go beyond relational algebra, employing complex user-defined functions, iterations, and distributed state.

[1] Email: volker.markl@tu-berlin.de

In the era of many-core processors, cloud computing, and NoSQL, we must ensure that well-established declarative language concepts (inherent in database systems) make their way into big data systems. In order to make this a reality, the research community will need to address the related challenges. For example, i) designing a programming language specification that does not require systems programming skills, ii) mapping programs expressed in this programming language to a computing platform of their own choosing, and iii) executing these in a scalable manner. In particular, this means devising execution strategies that are distributed, parallelized, and support in-memory technologies and out-of-core execution for data-intensive algorithm implementations.

In order to meet this challenge numerous scientific communities (e.g., *compiler*, *data analysis*, *database systems*, *distributed systems*) will have to come together. We will have to develop novel scalable algorithms and systems that are able to organize the data deluge and intelligently distill information to create value. To achieve this, declarative query languages must now be extended to support the specification of varying analysis methods (e.g., anomaly detection, classification, and clustering).

Furthermore, the power of declarative languages, namely, automatic optimization, parallelization, and adaptation of the same program to varying distributed systems & novel hardware architectures (depending on data distributions, data sizes, data rates, and system load) must be preserved. In this way, we will be able to overcome the current "stone age" in big data analytics. That is, algorithm specifications in systems that do not automatically optimize (e.g., MPI, MapReduce, and Hadoop), imperative languages (e.g., C), object-oriented languages (e.g., Java), and relational-oriented languages (e.g., SQL, XQuery, Pig, Hive, and JAQL) with non-tunable external driver programs, and technical computing systems (e.g., R and MATLAB) that do not scale.

With funding support from the EIT (European Institute of Innovation & Technology) ICT Labs and a DFG (German National Science Foundation) sponsored Research Unit involving TU Berlin, HU Berlin, and the Hasso-Plattner Institute in Potsdam, a large team of researchers has taken its first steps towards building a next generation big data analytics infrastructure called *Stratosphere*. Stratosphere and other next generation big data analytics systems, such as Spark and GraphLab will empower data scientists to conduct deep data analysis.

Stratosphere enables massively parallel in-situ data analytics using a programming model based on second order functions. It offers *Java* and *Scala* as programming language frontends, a scripting language called *Meteor*, and a graph processing frontend called *Spargel* with a Pregel-like interface. Key (Stratosphere) design points, include automatic optimization, parallelization, and hardware adaptation of complex data analysis pipelines from laptops to compute clusters. Additionally, native support for iterative data analysis programs, relational operators, and user-defined functions. Through the concepts of bulk and workset iterations, Stratosphere can process information extraction and integration operations together with deep analytics in a single system, subsuming many specialized systems for graph processing or machine learning in a single environment. Stratosphere is an open source-system (subject to the Apache 2.0 license). It runs standalone, natively in compute clusters, or without special installation in any Hadoop clusters via YARN.

Besides the current technologies available in the open source release of the Stratosphere system, our presentation at BDEC will also include innovative approaches to fault-tolerance, in particular, "optimistic fault-tolerance," which does not impose overhead in the error-free execution and can algorithmically compensate for faults in many important classes of advanced data analysis algorithms. Currently, Stratosphere is the only system for big data analytics that contains a query optimizer for advanced data analysis programs that go beyond the relational algebra. It optimizes relational operations jointly with user-defined functions and iterative programs, utilizing varying compiler and database technologies. In this manner, Stratosphere enables data scientists to focus on their respective problem and relieves them from scalable systems programming details. For more information visit www.stratosphere.eu.