University of Fribourg / Faculty of Science and Medicine / Department of Informatics

Dependency-Driven Analytics

Ruslan Mavlyutov

Large companies operate increasingly complex infrastructures to collect, store, and analyze vast amounts of data. The exhaust of these infrastructures may consist of petabytes of system logs daily. These logs faithfully capture all relevant aspects of the infrastructure, applications, and data life cycle. However, their sheer size and loose structure make them very challenging and expensive to access and analyze.

To address this problem, we built a system — currently in use at Microsoft — that automatically extracts from the logs a compact, higher-level graph representation capturing entities (e.g., jobs, files, machines) and their dependencies (e.g., job reads file), and provide this to users as a compass to navigate this ocean of raw data. We argue that the log-analysis scenario presented above is just one example of a broadening class of data analytics we call Dependency-Driven Analytics (DDA). In DDA, raw data are (automatically) preprocessed to extract a semantically rich and compact graph structure that captures key entities from the data along with their relationships. We expect DDA to emerge in various settings with massive volumes of loosely structured data produced by uncoordinated parties.

In this thesis, we investigate various areas where DDA may dramatically improve the state of the art. We create and evaluate a system that embodies DDA principles for the log- processing scenario presented above and call its result the Provenance-Telemetry Graph (PTG). In addition, we introduce new ways to automatically extract, store and query large-scale DDA data. We start by presenting DDA, focusing on the automatic extraction of a dependency graph from large volumes of mostly unstructured data, to guide users in efficiently navigating and querying the data.

We make this picture concrete by presenting the PTG graphs that result from a DDA analysis for a specific use case: log analytics. We show how PTG was used at Microsoft to dramatically improve job scheduling in massive computational clusters.

Next, we turn our attention to the problem of efficient construction, storage and querying of DDA-like systems. We present a software stack called LogAn to automatically infer schema and data-mining templates for Java-based distributed software primarily targeting Apache Hadoop. Then, we introduce a series of algorithms to compactly store and efficiently query interval data (which is dominant in DDA systems) called CINTIA. We also presented 3DBG, a novel storage paradigm and database system to store and query genomic data in a three-dimensional space that is using CINTIA at its core. Finally, we make a systematic comparison of the most common data structures used to manage URI data, which is another type of data that is widely used in DDA systems to identify entities. We evaluate a series of data structures in terms of their read/write performance and memory consumption before concluding our work.

Jury:
Prof. Dr. Philippe Cudré-Mauroux, University of Fribourg (Switzerland), thesis supervisor.
Prof. Dr. Pascal Felber, University of Neuchatel (Switzerland), external examiner.
Dr. Carlo Curino, Microsoft CISL (USA), external examiner.
Prof. Dr. Edy Portmann, University of Fribourg (Switzerland), president of the jury.