Aggregating Millions of Data per Second in Real-time
MSc Thesis
Würth Phönix

The target of the project is to analyze the Timely Dataflow framework written in rust so to create a real time data aggregation and/or computing system and keep this data in a cache front end for dashboard web applications. As a database I was thinking of redis. The timely dataflow framework is based on following Microsoft Silicon Valley Research: http://research.microsoft.com/pubs/201100/naiad_sosp2013.pdf

The big advantage of this framework is the lack of an infrastructural overhead like for example those of Apache Spark, Flink or Hadhoop. Another big advantage, which we are aiming at, is creating the real time aggregation that is not yet possible through Hadhoop, Spark, Flink frameworks, because a large part is based on the batch data processing instead of the real time data processing in a distributed stream environment that we would like to create with the timely dataflow in rust.

The data types we would like to analyze are above all IT infrastructure metrics, for example TCP/IP Network metrics, I/O metrics, or CPU metrics a.s.o. As a starting point we were thinking of using NetFlow. Our data is stored on Netflow and the data volume is more than 2 Billions records per second. We would like to start analyzing the basis of NetFlow in order to visualize for example the network utilization based on the protocols (http, mail, remote desktop, streaming video/audio, ...) or to visualize who is communicating with whom, the amount of data exchanged, in which direction the communication is heading a.s.o. This data analysis should happen in real time to provide the users with an instant visualization of the aggregated data dashboard and not after hours of batch calculations.