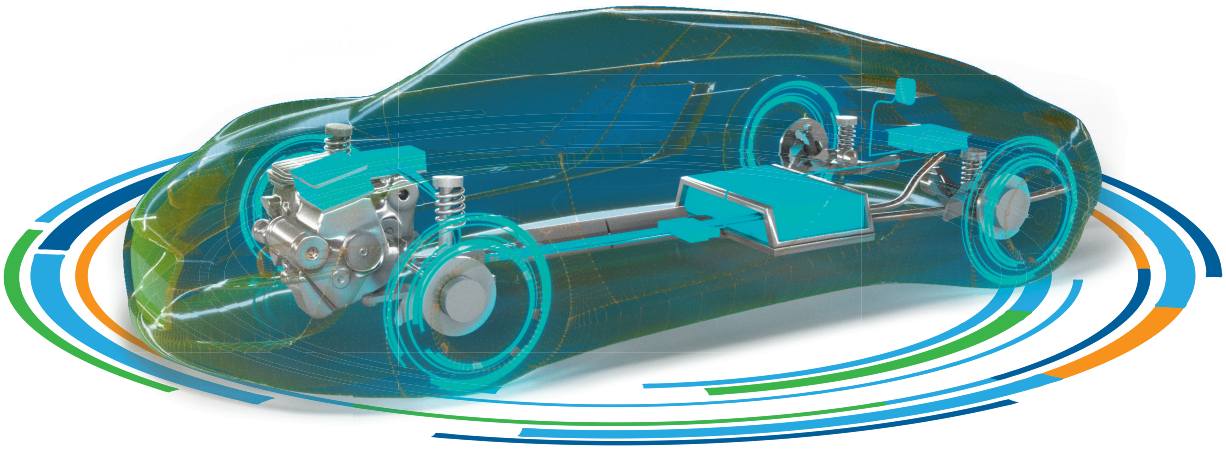
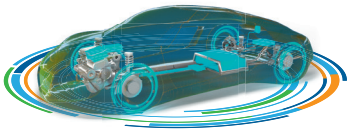
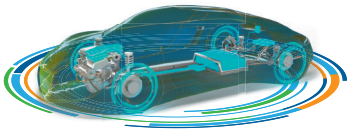


INTERNATIONAL SYMPOSIUM ON DEVELOPMENT METHODOLOGY



Fast analysis of worldwide distributed endurance run data

Dr. Tobias Abthoff – NorCom Information Technology GmbH & Co. KGaA



Abstract

Classical endurance run is required to take over more & more tasks and thereby faces many new challenges:

- *Data rates are increasing fast: Modern endurance run vehicles already collect about 3.000 channels, some of them measured at high frequencies*
- *Ultrafast turn-over-times and data quality management measures are required: The gap between classical endurance run and instrumented development vehicles is closing. Todays engineers need to analyse test drive result data within a few hours after the drive has finished – no matter on which part of the world testing has been performed. Additionally – due to test fleet costs – data quality issues like e.g. data logger misconfigurations or sensor faults need to be identified and fixed DURING and not after an endurance run campaign.*
- *Endurance runs become increasingly "KPI-driven": Testing a complete vehicle for a given number of miles or hours is no longer enough. Instead specific load duty cycles must be performed and monitored for individual components.*
- *Endurance run locations need to be more and more flexible: One must be able to quickly set-up und tear-down endurance run sites to cope with seasonal requirements or to be able to switch local subcontractors easily.*

All the above leads to a modern data-driven endurance run that must be managed and analysed using big data systems. However, due to its very nature, endurance runs are distributed all over the world in many different scenarios and climates. Thus, getting data back "home" or having fast online connectivity is a non-trivial and therefore an often tedious and expensive task.

We will present a comprehensive endurance run systems blueprint that addresses all the challenges named above and allows car manufacturers to make the most of their valuable endurance run data.

The system can orchestrate high-performance analyses on globally distributed data sets without moving them. This grants engineers immediate, world-wide data access, allows them to draw instant conclusions from analytics results and loop those insights back into the endurance run field by altering and optimizing test proceedings.

The automotive development cycle is predicated on different iterations of progress assessment in consecutive stages, which means recording and evaluating data. The first stages in an early development phase are usually carried out using a small fleet of instrumented development prototype vehicles. Those tests mostly take place on campus or close to manufacturers' development centers. Up to this point collection, transfer, management and analysis of generated data was and is technically feasible.

In a later, more mature stage of development, the passel of testing vehicles grows and more and more spreads locally over development providing engineers with the data that has been collected worldwide in a timely manner for analysis. With the introduction of more and more sensors, the test vehicles are pushing the limit of the bandwidth of the Ethernet infrastructure and are generating huge quantities of globally distributed data every day. It is not realistic to first move this data to a central cluster and then start the analysis. In this article, we show how our solution - the Distributed Query Engine and the Data Lifecycle Management - based on the big data platform DaSense, can be used to perform high-performance analysis on globally distributed data sets without moving them. This allows users to access data immediately, in order to evaluate it and schedule test runs quicker.

The time-critical analysis of large amounts of data

During ongoing test drives, large quantities of measured data accumulate over a longer period of time at various globally distributed locations. Time lags are costly so these data must be evaluated promptly in order to incorporate the newly acquired information into the development process.

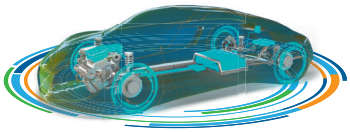
Time-consuming transport of large data sets into a centralised environment is not possible in this tight time frame. Only through the immediate evaluation, test runs can be planned quickly and development costs can be reduced.

The software presented here helps to solve these huge challenges. Based on DaSense, a big data and advanced analytics/deep learning platform, it is tailored to the needs of the automotive industry.

The DaSense feature of working with globally distributed data has been patented by NorCom and, together with a major car manufacturer, the solution has been successfully tested in multiple locations.

DaSense big data development platform

DaSense[1] is a Hadoop[2]-based Big Data technology platform that scales analytics in a computing cluster with Apache Spark[3] for data up to the petabyte range. It makes use of the fact that, in a first step, only the data of test drive under investigation is needed for the analysis of that test drive. Hence, this part of the analysis can run on multiple clusters at the same time in a massively parallel manner and the number of test runs that can be evaluated simultaneously is limited only by the size of the compute clusters. In the subsequent step, the individual partial results are combined into an overall analysis and used for further steps, e.g. a visualization.



An analysis which is often essential in practice, which is already covered without any programming by DaSense, is the event search. Here, limits are set for various signal and metadata, in order to identify events in the data (e.g., switching operations, temperature ranges, etc.). In the next step, these events are identified within all available measurement data and the results are returned per test drive. This result is very small compared to the whole dataset, so the basic idea of DaSense is: „Move the algorithm, not the data". Instead of moving the data, the evaluation is transferred to the data. In the final step, these individual results are then combined into an overall statistic.

A single event can be represented by only two values - the beginning and the ending time. Several events in the same test run are then combined in a time interval list. In DaSense, this is defined as an exchange format in order to merge several applications (apps) into a Big Data capable workflow. This enables the search app to pass on the found events to various other analysis apps (e.g. for a classification), to visualization apps (e.g. to a histogram app) or to the export app, which makes the events available in various file formats so they can be edited through other tools (see Figure 1). The data flow remains largely virtual, a key design feature for scaling to large data.

Figure 1: In DaSense, only relevant events, not all data, are passed on for analysis

Evaluations that are not covered by the standard apps of DaSense can be realized at any time through a comfortable programming interface. For this purpose, DaSense provides a Python-based domain-specific language (DSL), which is optimized for the evaluation of distributed time series. In the interactive development environment, complex analysis can be implemented, in which DaSense automatically takes care of the parallelisation. The resulting analysis can be easily and self-sufficiently turned into productive apps for future execution (see Figure 2).

Distributed analysis with the Distributed Query Engine

A limitation in the traditional approach of big data with Hadoop is that the data that is to be evaluated must be available in a cluster site, and there is no provision for distributing data across multiple sites. The Distributed Query Engine (DQE) extends this approach to the next logical step: running analysis on multiple clusters at the same time (see Figure 3). In this case it does not matter if the connected clusters are on-premise or cloud solutions. Every single cluster has an instance of DaSense installed that links together into a large network. The analysis can therefore be executed across multiple data centres.

A separate job engine was implemented for the DQE, with which even complex job workflows can be mapped. The

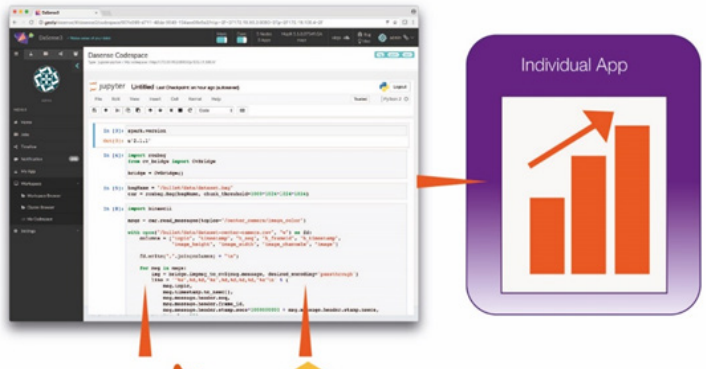


Figure 2: In DaSense, queries can be programmed and, if needed, apps for reuse can be extracted.

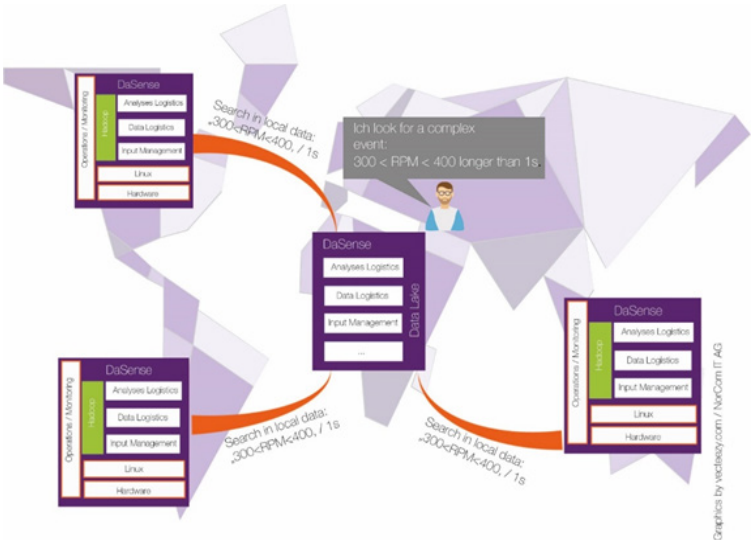


Figure 3: Distributed Query Engine - DQE allows you to run analytics on datasets that are distributed not only across multiple machines but even across multiple datacenters

Figure 4: Data Lifecycle Management - data is always ready for analysis on at least one DaSense instance. After transport, redundant data may be deleted.

DQE takes care of the planning of the individual jobs and the dispatch of the algorithms. If an exchange of raw data can not be prevented - for example, if signals from different test drives are to be compared directly with one another - the DSL offers the functionality to automatically trim and compress the raw data to the necessary level.

This approach has another crucial advantage: in many countries there are legal restrictions, which make it difficult or even prevent the transfer of raw data across national borders. With the DQE, the endurance runs can also be evaluated in this case without the data leaving the place of origin.

Data-Lifecycle-Management

With the help of Data Lifecycle Management, DaSense is able to track the location of the data at all times so that if necessary, it can plan the physical transport of the data. Even if the globally distributed data is available immediately for analysis purposes, a later transport of the data can prove itself to be very useful. Above all, in order to ensure the reliability of the partly mobile local clusters, the accumulated data will usually continue to be transferred to a central cluster.

The Data Lifecycle Management keeps track of where the data is at a given time to ensure the distributed analysis at all times (see Figure 4). The data is not removed from a site until it has been completely transferred to another site and no analysis is performed on the data set at that time. As a result, the data can be moved on „slow" paths (e.g. by mail) and are available for evaluations at any time.

Combining the components

The „Distributed Query Engine" and the „Data Lifecycle Management" represent two independent enhancements of the DaSense platform, but they only reach their full potential once reasonably combined in production of a running distributed data collection, exchange and evaluation process.

In the first step an analysis using the DaSense Query Optimizer is evaluated and, if necessary, optimization is reached using the following modules:

- *Algebraic Library: algebraic rules for modifying an analysis.*
- *Cost Prediction: a series of functions that predict the cost of calculating an analysis.*

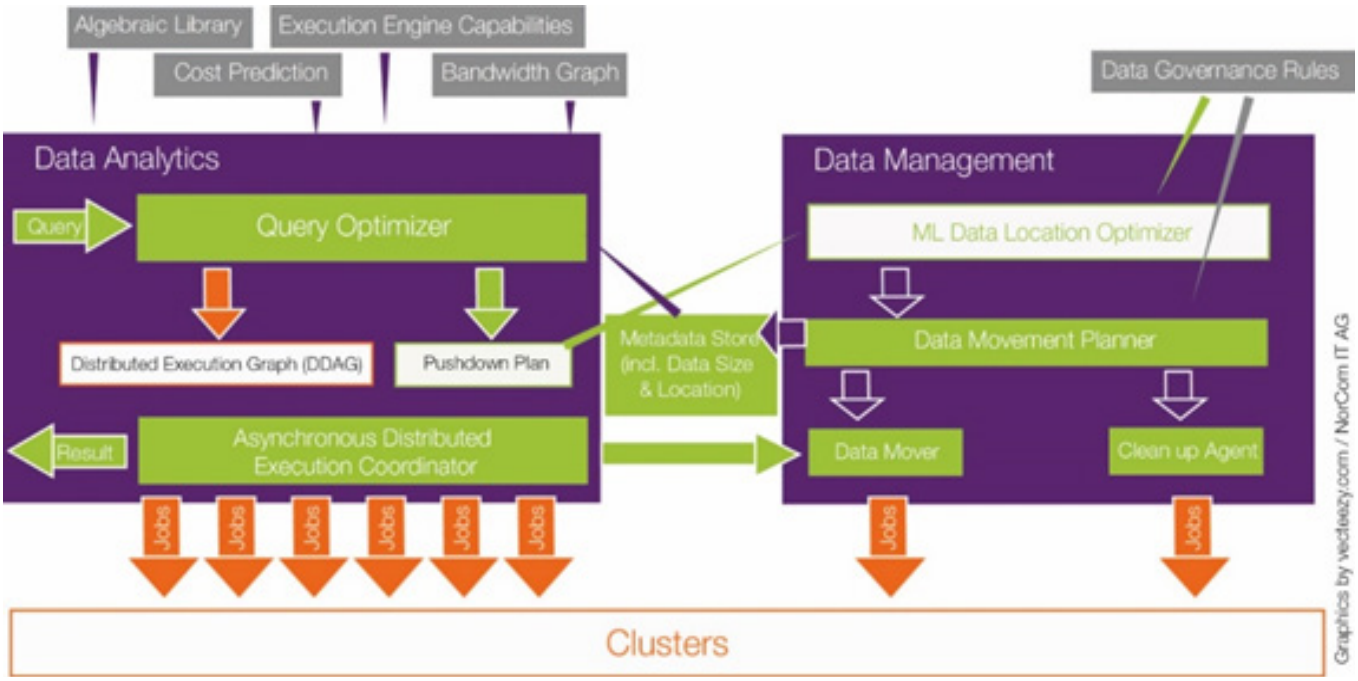
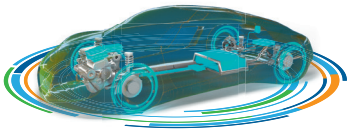


Figure 5: By using different components, the runtime of the analyzes can be optimized and data moved accordingly.



- *Execution Engine Capabilities: for each site information such as the number of free resources or technical specifications of the available hardware can be retrieved.*
- *Bandwidth Graph: the available bandwidth for input and output at each site.*
- *Metadata Store: database for managing metadata of analysable data. This includes e.g. the data size.*

The Query Optimizer uses a predefined cost function to forecast the costs of an analysis, based on factors such as: the expected runtimes, the need for resources, the network bandwidth, etc. The aim of the Query Optimizer is to modify a given analysis using the algebraic rules so that costs are minimized.

After the optimisation, a Distributed Execution Graph (DEG) is created whose components are to be executed at different locations. This task is taken over by the Asynchronous Distributed Execution Coordinator. The coordinator sends parts of the DEG to the individual sites, aggregates the results and ultimately returns them to the user.

A key factor in cost optimisation is data locality. Similar to Hadoop, DaSense also follows the paradigm of avoiding unnecessary data transfers. This is where data lifecycle management comes into play. First, it manages information about the location and size of the data. On the other hand, data movements can be initiated via the Data Movement Planner.

The Movement Planner knows a set of rules - either manually defined or derived by means of machine learning methods from data movement patterns - that can be interpreted by the Planner. The data mover eventually performs copy operations; a clean-up agent is responsible for removing data. Intelligent planning of data movement can significantly reduce the cost of frequently performed analysis (see Figure 5).

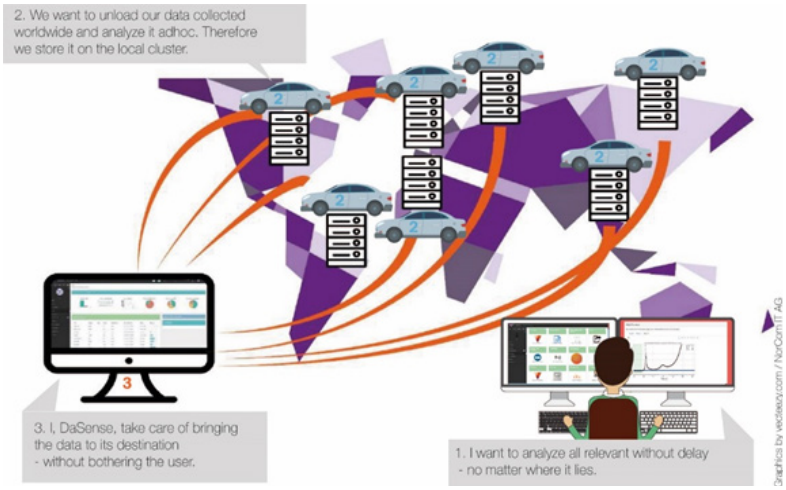


Figure 6: In continuous operation, measurement data is generated in large quantities distributed globally. Nevertheless, the analysis should be possible on all data without delay, before the data is transferred to a central cluster. If necessary, the data can still be transferred to the central cluster without downtime.

Application in continuous operation

In vehicle endurance, more measured data are recorded within a shorter timeframe. As the vehicles are used worldwide, it is becoming increasingly difficult to transfer the resulting measurement data to the head office for analysis. In one project we tested a distributed measurement technique together with an OEM to solve the new challenges in continuous operation (see Figure 6).

Measuring data boxes are used, where the loggers installed in the endurance vehicles are unloaded directly on site during shift changes. The measurement data boxes do not only serve as temporary storage, but also function as flexible, transportable and globally integrable analysis platforms. The basis for this is the DaSense analysis method described above, which considerably reduces the transmission bandwidth requirements and enables the evaluation of even larger amounts of data almost in real time.

The implementation of the project took place in two phases. In phase 1, DaSense was installed in the central data center, as well as on a measurement data box. DaSense automatically transforms data into the data center and on the box into a big-data-capable format, providing in depth analysis to users for fast search, reporting, and root cause[4]. In phase 2, distributed analytics were implemented using DaSense across data center and measurement data box.

Outlook

With DaSense, we have developed a data measurement solution that enables fast and scalable remote analysis of testing, development fleet and field fleet data, even when located across geographically distributed. This makes DaSense ideal for use even beyond the endurance run, such as for the worldwide development of algorithms for autonomous driving[5].

References

[1] www.dasense.de; ATZelektronik 05/2016, Big Data Technologien in der Fahrzeugentwicklung, Dr. T. Abthoff

[2] http://hadoop.apache.org/

[3] https://spark.apache.org/

[4] MTZ 12/2016, Interaktive Big Data Analytik in der Motorenentwicklung, Dr. T. Abthoff

[5] https://www.springerprofessional.de/big-data-for-assisted-and-autonomous-driving/15757862