Big Automotive Data
Leveraging large volumes of data for knowledge-driven product development

Mathias Johanson, Stanislav Belenki, Jonas Jalminger, Magnus Fant
Alkit Communications AB
Mölndal, Sweden
{mathias, stanislav, jonas, magnus}@alkit.se

Mats Gjertz
Volvo Car Corporation
Gothenburg, Sweden
mats.gjertz@volvocars.com

Abstract—To be successful in the increasingly competitive consumer vehicle market, automotive manufacturers must be highly responsive to customer needs and market trends, while responding to the challenges of climate change and sustainable development. One key to achieving this is to promote knowledge-driven product development through large scale collection of data from connected vehicles, to capture customer needs and to gather performance data, diagnostic data and statistics. Since the volume of data collected from fleets of vehicles using telematics services can be very high, it is important to design the systems and frameworks in a way that is highly scalable and efficient. This can be described as a Big Data challenge in an automotive context. In this paper, we explore the opportunities of leveraging Big Automotive Data for knowledge driven product development, and we present a technological framework for capture and online analysis of data from connected vehicles.

Keywords—big data; automotive telematics; analytics.

I. INTRODUCTION

The globalization of markets, resources and knowledge require product development companies to be highly responsive to customer needs and to environmental changes. In the automotive industry, the major challenges of limiting CO₂ emissions while delivering high quality products to a growing number of customers in expanding and highly heterogeneous markets require very efficient and powerful tools and methods to capture customer needs and also to gather performance data and statistics to improve product development.

In the testing, verification and validation phases of automotive product development, large volumes of measurement data are being gathered from fleets of connected test vehicles [1, 2]. With the advent of telematics systems and improved means of wireless vehicular communication more or less ubiquitously, the opportunities to capture and collect data has improved tremendously over the past few years. This has an enormous potential of improving automotive product development, by making reliable performance data, statistics and customer behavior information available as quickly and efficiently as possible in the development process. The ability to make good use of this valuable resource can be clearly identified as a key means to competitiveness in the automotive industry. The big challenge is how to be able to efficiently capture, collect, manage, analyze and make good use of the large volumes of data, i.e. to convert collected data into useful knowledge.

The term “Big Data” has recently been popularized, referring to data sets that are so large and complex that they are difficult to handle using conventional database management systems and traditional data processing tools. In this paper we will study Big Data applications in an automotive context. This involves a number of applications that can be expected to benefit from large scale capture and analysis of data from vehicles, driven by connectivity and onboard telematics services.

II. BIG DATA

Design of scalable and distributed data management systems has been the goal of the database research community for a long time. Initial approaches include distributed databases for update intensive applications, and parallel database systems for analytics-oriented tasks. Whereas parallel database systems have matured into large commercial systems, distributed database systems were never very successfully commercialized. Instead, different ad-hoc approaches to achieve scalability were developed. New database approaches to achieve scalability are sometimes collectively referred to as “NoSQL” solutions, to distinguish them from traditional relational database systems (RDBMS), which are usually based on the query language SQL. However, since SQL is frequently used as the query language also in NoSQL solutions, the term is sometimes interpreted as “Not only SQL.”

One breed of NoSQL database systems, motivated by changes in data access patterns of applications and the need to scale out to large clusters of distributed processing units, is the new class of systems referred to as Key-Value stores [3] which are now being widely adopted. Examples include Amazon's Dynamo and LinkedIn's Voldemort. A particularly popular type of key value store is a Document Store database, which relies on the basic concept of a document for encapsulating data. The document encoding can be any kind of structured data container, including XML, JSON or Word, etc., or in the automotive domain various measurement data
file formats, such as MDF. Examples of document-oriented database systems include MongoDB and Couchbase.

Another approach is column-oriented database management systems [4], which store data tables as sections of columns of data, rather than as rows of data, which most RDBMSs do. Examples include Google's BigTable and Apache Cassandra.

In the field of data analytics, the MapReduce paradigm [5], pioneered by Google, and its open source implementation Hadoop [6] have seen widespread adoption in both industrial and academic contexts. There are many initiatives to improve Hadoop based systems in terms of usability and performance, and to integrate it with mathematical and statistical analysis frameworks such as R [7, 8]. Apache Spark [9] is another analytics cluster computing framework, which is based on the same distributed file system as Hadoop (HDFS), but is not limited to the two-stage MapReduce paradigm, which makes it considerably faster for certain applications.

An important point to make about Big Data is that it is not merely about data volume. Sometimes Big Data is described as spanning three dimensions: Volume, Velocity and Variety (the three V’s of Big Data, originally proposed by Doug Laney in a 2001 paper [10]). Frequently, a fourth V is added, for “Veracity,” and sometimes also a fifth for “Value.” In this multi-dimensional definition, “volume” refers to the size of the data sets, “velocity” highlights the need for time-sensitive processing of some data, “variety” implies that data can be of many different kinds with widely different characteristics, and “veracity” refers to the problem of deciding whether data from different sources can be trusted. Ultimately, the goal of any Big Data application is to create added value in terms of increased revenue, new services, higher quality products, or other benefits, which is captured by the fifth V representing “value.”

The three original V’s are present in Gartner’s oft-cited definition of Big Data [11]:

“Big Data is high-volume, high-velocity and high-variety information assets that demand cost-effective, innovative forms of information processing for enhanced insight and decision making.”

III. BIG DATA IN AN AUTOMOTIVE CONTEXT

To arrive at a definition of Big Automotive Data, we will explore the dimensions of the five V model in the context of automotive data application scenarios.

A. Volume of Automotive Data

To get a feeling for what volumes of data can be expected in a Big Automotive Data context, we have performed some estimations of the size of data sets for a number of relevant application scenarios. The estimations should be seen as rough order-of-magnitude approximations in order to illustrate the need for unconventional data processing and storage techniques.

1) Extensive CAN bus monitoring

The first application scenario is an extensive CAN\(^1\) bus monitoring scenario, intended to serve as an example of the magnitude of data that results if data is captured at a high rate in a fleet of vehicles. Let us assume that there are five CAN buses in a modern vehicle, where each CAN bus supports communication at 500 kilobits per second (kbps), and let us furthermore assume an average bus load of 50%. For an application monitoring CAN data in customer vehicles, let us assume the vehicles are on average used for one hour each day. In a test vehicle, we assume the vehicle is used for 8 hours every day. Then, for a customer vehicle fleet of one million vehicles, the total size of the CAN signal data set will grow by about 560 terabytes per day, or about 200 petabytes per year. For test vehicle applications (where extensive CAN bus monitoring is commonplace already today), a fleet of 1000 test vehicles will produce 4.5 terabytes of data per day, or equivalently, around 1.6 petabytes per year.

2) Remote Diagnostic Read-Out

A Diagnostic Read-Out (DRO) application is a concept whereby a chunk of vehicle related data is accessed from a vehicle using the vehicle’s diagnostic system. In a usage scenario whereby DRO can be performed remotely using onboard telematics services, DRO can be performed on a regular basis to find faults (by reading Diagnostic Trouble Codes, DTC), and to access statistics and performance data. Currently, DRO is typically performed when a vehicle is brought in for service to a repair shop, whereupon the DRO data set is uploaded to the data warehouse of the car manufacturer. In a future remote DRO scenario, where a fleet of one million customer vehicles are read out once a day, assuming a data chunk size of 100 kilobytes, will result in an aggregate data volume increase of about 100 gigabytes per day, or about 36 terabytes per year. For a test vehicle fleet of 1000 vehicles, assuming 10 read outs per day and a data chunk size of one megabyte (reflecting the fact that testing and validation typically require more extensive data at higher sampling rates), the corresponding numbers are 10 gigabytes per day or 36 terabytes per year. Although less dramatic than the CAN monitoring scenario, this is still a lot of data to manage.

3) State-of-Health

As a final example, an automotive State-of-Health application monitors a number of carefully selected parameters in the vehicles, and triggers data upload to allow the status of the vehicle to be assessed. This typically requires substantially lower data rates. For a vehicle fleet of one million customer vehicles, the estimated data size grows by about one gigabyte per day (i.e. 365 gigabytes per year). For a test vehicle fleet of 1000 vehicles, the corresponding

\(^1\) CAN – Controller Area Network, is an in-vehicle communication bus technology heavily used for broadcasting of sensor signal values between Electronic Control Units, and for diagnostic services
estimates are 100 megabytes per day or 36 gigabytes per year. The numbers are summarized in Table 1.

<table>
<thead>
<tr>
<th>Application</th>
<th>Fleet</th>
<th>Per day</th>
<th>Per year</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAN bus monitoring</td>
<td>Customer</td>
<td>560 TB</td>
<td>206 PB</td>
</tr>
<tr>
<td>CAN bus monitoring</td>
<td>Test</td>
<td>4.5 TB</td>
<td>1.6 PB</td>
</tr>
<tr>
<td>Remote Diagnostic Read-Out</td>
<td>Customer</td>
<td>100 GB</td>
<td>36 TB</td>
</tr>
<tr>
<td>Remote Diagnostic Read-Out</td>
<td>Test</td>
<td>10 GB</td>
<td>3.6 TB</td>
</tr>
<tr>
<td>State-of-Health</td>
<td>Customer</td>
<td>1 GB</td>
<td>365 GB</td>
</tr>
<tr>
<td>State-of-Health</td>
<td>Test</td>
<td>100 MB</td>
<td>36 GB</td>
</tr>
</tbody>
</table>

B. Velocity of Automotive Data

Some automotive applications require quick response times for processing of data, whereas others have more relaxed real-time requirements. Examples of delay-sensitive applications include collaborative active safety functions or autonomous driving services, whereby vehicles communicate safety-related data used by in-vehicle components to other vehicles, to make safety related decisions. For such use cases, the processing of the data must typically be performed with very low latency.

In general, use cases that are based on positioning of vehicles typically are time-sensitive in different degrees, since the position is constantly updated when the vehicle is moving. The need for real-time processing, or near real-time processing can therefore be seen to apply for Big Automotive Data applications.

C. Variety of automotive data

Automotive data sets are very diverse. Common types of data are time-based signals, such as CAN bus signals from the multitude of onboard sensors in the vehicle, scalar data, such as DTCs or parameters that are accessed through diagnostic services, images, video, statistical data, legal and administrative data and more. Moreover, there is a plethora of different data formats and standards in use in the automotive industry.

Variety of data is definitely a relevant issue for big automotive data sets.

D. Veracity of automotive data

Veracity of data is highly relevant in the automotive context, in order to guarantee security and safety in using customer vehicles and connected services.

To ensure the relevance and quality of data captured for the benefit of knowledge-driven product development, provenance and traceability of data is very important. For instance, when carrying out performance tests of vehicle components or subsystems, it is of vital importance that the data captured originate from the correct components, with the correct configuration, software versions, etc.

Another aspect of veracity in this context is that automotive companies tend to be very secretive about their engineering data, requiring sophisticated information security mechanisms when communicating data over public network infrastructures.

E. Value of automotive data

There is a broad range of possible benefits enabled by Big Automotive Data services. New aftermarket services and product features can be designed based on information resources generated from data captured from connected vehicles, aggregated and analyzed using cloud-based data processing and management services. This includes predictive and preventive maintenance services, various infotainment services, active safety and autonomous driving support services, to name just a few. In addition to novel aftermarket services, product development processes will benefit from access to large volumes of data captured from test vehicles as well as customer vehicles. This makes it possible to make better informed decisions in more or less all stages of the product development process, from the early concept development to testing, validation and verification. This is what we refer to as knowledge-driven product development, which is the focus of this paper and the main driving force behind the technological framework described in Section IV.

In addition to the value of Big Automotive Data in supporting knowledge-driven product development and as an enabler for new services, we can also identify a more direct value of the data itself. Automotive companies are increasingly exploring the opportunities of selling carefully selected and processed data sets to third parties. Potential customers of this kind of data sets are for instance road administration authorities, insurance companies and automotive e-service developers. With this perspective, Big Automotive Data becomes a new source of revenue for the automotive companies.

F. Properties of Big Automotive Data

As we have seen, the five V model of Big Data applies well in an automotive context, so Big Automotive Data can be loosely defined simply as Big Data for automotive applications. To probe a little deeper into the distinguishing characteristics of Big Automotive Data - to see what might make it different from other Big Data applications - we can immediately identify a few salient points.

The data sources are typically mobile (i.e. moving vehicles), requiring wireless communication networks in many cases to collect data.

Compared to many other Big Data applications, there is often (but not always) an emphasis on time-series data, typically originating from sensors connected to the in-vehicle communication buses (e.g. CAN bus signals). This is particularly pronounced for automotive testing and validation
Automotive applications that have a direct impact on the performance of vehicles are generally very concerned with safety issues. Automotive technology and systems in general have a very strong safety focus, and this is reflected also for Big Automotive Data applications.

IV. BAuD: A BIG DATA FRAMEWORK FOR AUTOMOTIVE TELEMATICS AND ANALYTICS

To explore the opportunities of leveraging Big Automotive Data for knowledge-driven product development, we have developed a prototype telematics and analytics framework in the BAuD project. The design of the BAuD framework is based on a number of use cases identified at Volvo Car Corporation, with an emphasis on R&D use cases. The core functionality developed so far is concerned with capturing and analyzing data from vehicles for use in the product development process. The hypothesis is that Big Automotive Data is a resource that can be exploited to improve product quality and reduce time to market. Although this is the initial focus of the BAuD framework, we also foresee use cases where the information resources are exploited for novel aftermarket services.

A. BAuD Applications

For a typical R&D use case, the BAuD framework is used as follows. A stakeholder in the automotive development process has a specific question regarding how the product is used or how some subsystem is performing. To gain knowledge about the particular phenomenon of interest, an engineer designs a measurement task, defining what data will be captured, and then assigns this measurement task to be executed in a fleet of test vehicles. The measurement assignment is uploaded to a server and scheduled for download to the target vehicles. In parallel, an analytics task is defined, describing what kind of analysis will be performed on the data collected by the corresponding measurement task. Depending on the type of analysis requested, the result of the analytics task can be different kinds of visualizations or reports, assembled and made available to the users through a web-based user interface.

B. BAuD Framework Architecture

The BAuD framework is a complex technological platform designed to support the applications discussed above, with a specific design goal of being flexible and extensible to cater for new needs and novel applications. Moreover, the framework has been designed for scalability to large numbers of connected vehicles and large volumes of captured data. Although the primary focus is heavily on pre-production test vehicles and R&D use cases, the emphasis on scalability will allow also aftermarket use cases of millions of connected vehicles to be supported by the platform. A schematic overview of the architecture is shown in Fig. 1.

The core components of the BAuD framework are:

- A telematic service platform allowing wireless access to data from vehicles in use,
- A cloud-based back-end infrastructure, including application programming interfaces to provide controlled access to the information resources and framework services,
- A Task Manager, handling the execution of measurement and analytics services,
- A Data Broker mechanism, handling the relay of data from data sources to data sinks,
- An analytics service architecture, enabling automated data-driven analysis of data originating from connected vehicles,
- A web-based user interface front-end.

For the main BAuD use case described above, the Task Manager component of the architecture shown in Fig. 1 keeps track of all measurement and analytics tasks, and configures the data broker to forward the data to the proper analytics service as it is uploaded from the vehicles of the fleet.

The telematic service layer handles the capture and upload or streaming of data from connected vehicles to the back-end infrastructure. The presentation layer and user interface components provide the means by which the end-users and administrators of the system access the framework services.

The core component of the telematics framework is a Linux-based data capture and communication unit installed in vehicles. The unit executes measurement tasks that support data capture both by passive in-vehicle communication bus monitoring and active diagnostic services. The captured data is uploaded to the cloud-based infrastructure using 2G, 3G or 4G wireless mobile data.
communication. The telematics unit also provides additional services such as GPS-based positioning.

The telematics software architecture is designed in a modular fashion with a strong emphasis on portability, for the purpose of stepwise integration into future production vehicle ECU architectures. This means that a common telematic service platform for both aftermarket services and R&D services will be possible to realize. There is also a strong emphasis on security in the design, to prevent malicious unauthorized access to both in-vehicle information sources and the cloud infrastructure.

Since mobile wireless data communication can be expensive, and since the available bandwidth in many parts of the world is still very limited, data compression techniques are used to reduce the required network bandwidth. To some extent there is also data reduction due to onboard preprocessing analytics. These mechanisms are mainly statistical processing functions, like histogram generation, which can be performed directly in the telematics units, before data is uploaded.

The communication architecture can handle both bulk upload of data and streaming of data without the need to store it on the solid state disks of the telematics units. For most data capture services, the data is stored to disk while the measurement assignment is running, and then pre-processed and uploaded when the ignition of the vehicle is turned off. (The triggering of data upload is configurable, but the most common trigger is ignition off.) The streaming mode is used for time-sensitive applications, such as positioning services where it is important to show the current location of moving vehicles.

D. BAuD Analytics Framework

The analytics framework of the BAuD architecture is based on a data-driven approach, whereby data sets uploaded from connected vehicles are automatically analyzed based on an analytics task definition, and the result of the analysis is incrementally refined and stored in a knowledge base. Typically, the measurement tasks and the analytics tasks are designed in concert, since the input signals referenced in the analytics task must be the result of the corresponding measurement task.

The core of the analytics framework implemented so far is concerned with analysis of CAN bus signals and diagnostic data, such as Diagnostic Trouble Codes, originating from the measurement assignments executed in the in-vehicle telematics units. The captured data sets are uploaded using telematics services and accessed by the analytics framework in the in form of MDF files. MDF (Measurement Data Format) is a standardized binary data format widely used in the automotive industry, which supports trigger events, data conversion formulas, and sample reduction. It provides a very compact representation of time series data (compared to e.g. text-based formats) and as such is highly suitable for the data capture and telematics services. However, due to the binary representation and the need for data conversion, extracting signal values from MDF files requires non-negligible processing resources and can therefore be time consuming for large files. Specifically, when a large number of files need to be processed for one analysis, which is often the case, we see the need for parallel processing in a distributed computing environment to achieve scalability and performance.

1) Scaling the analytics framework with Spark

As our investigation in section III reveals, the volumes of data collected from connected vehicles can potentially be very large, which calls for a distributed and scalable approach to both storage and data processing. To achieve this, we have chosen Apache Spark as the Big Data processing platform for BAuD, considering its superiority over Hadoop MapReduce in case of machine learning and iterative data mining applications. The architecture is based on MLlib, Spark's machine learning library, and SparkR, which is R for Spark, to perform analytics operations requested by the user. With this approach, the Automotive Data Broker submits MDF files into a HDFS file system, which is Hadoop's distributed file system, as a data source for Spark. Since Spark operates by RDD - Resilient Distributed Datasets - a mapping function is required to convert a set of MDF files into a set of RDDs. In our design, a set of MDF files is defined by a set of vehicles identified by VIN (Vehicle Identification Number), a time range (Δt), and a set of CAN signals (s₁ to sₙ). When the analysis assignment returns, the results are submitted to the presentation layer of the system.

The conceptual design is illustrated in Fig. 2.

E. Presentation layer / User interface

To interact with the framework/system a user interface and an administrative infrastructure is needed. In this particular case we have chosen to use a web based user interface. The rationale behind this decision is based on several factors, including end-user familiarity with web-based interfaces, simplified remote access to the system in presence of corporate firewalls, and simplified software maintenance. Another strongly contributing factor is the
multitude of frameworks and toolkits available to build web front ends, which speeds up user interface development and promotes reusability of code. For this reason we have chosen the framework SmartGWT, which is based on Google Web Toolkit, an open source set of tools supporting development and deployment of complex JavaScript front-end applications in Java.

For users or administrators of the system to be able to manage a large number of vehicles, some user-interface mechanism is needed to handle these data sources in an aggregated form. Therefore, we have introduced the concept of resource groups. A resource group is a way to aggregate a set of data sources to be handled in a uniform manner through the user interface. In the example of configuring the telematic units of a set of vehicles, the administrator simply manipulates the configuration of the resource group and when saving the configuration it is fanned out to all vehicles in the resource group. Another example is when a user is about to start a new measurement task on a set of vehicles. Instead of selecting the individual vehicles, a resource group is selected and the system makes sure that the measurement task is propagated to each vehicle of the group. The notion of resource groups has been found to be a very powerful abstraction, which can be used throughout almost all user interface parts of the system.

Yet another effect of resource groups is that grouping of data sources can be used to handle a set of vehicles based on geographical location, vehicle model, target market or other selection criteria. Therefore, the administration of vehicle fleets can be separated from the administration of measurement tasks.

F. Scalability issues

As the number of data sources (i.e. the number of vehicles) grows, there are a number of scalability challenges to address in the design of the BAuD framework. The main server-side approach to handle the large volumes of data produced by measurement tasks in a large fleet of vehicles is to scale out the corresponding processing and analytics tasks over a cluster of computational units using a distributed computing framework.

To improve the scalability of the communication architecture, the upload of data from vehicles is designed using a multi-stage approach, whereby the data is uploaded to different access network servers depending on the geographical location of the vehicle. Data is then successively aggregated by the broker into the core of the cloud-based server architecture. The downloading of measurement tasks and configuration data to vehicles is also performed using a hierarchical, multi-tiered architecture.

Apart from technological scalability issues, we have also identified many usability and user interface related scalability challenges. The approach we have employed to allow users to manage large numbers of vehicles and other resources in a scalable and efficient way is based on aggregation of similar resources into resource groups.

V. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper we have explored the opportunities and challenges of leveraging Big Automotive Data for knowledge-driven product development. The BAuD framework, a scalable and efficient Big Automotive Data platform including integrated telematics and analytics services, is currently being evaluated in two case studies conducted at Volvo Car Corporation. The two case studies are focused on active safety development and battery performance for hybrid vehicles respectively.

As part of our future work we will explore how the BAuD framework can be extended to capture not only objective measurement data from connected vehicles, but also subjective usage information from customers. We intend to do this by designing a smartphone app, which will allow specialized questionnaires to be presented to selected customers, to capture the usage experience and feed the subjective data into the BAuD analytics framework.

ACKNOWLEDGMENT

This work was co-funded by VINNOVA, the Swedish Governmental Agency for Innovation Systems.

REFERENCES