Information Systems Management, 32:200–209, 2015
Copyright © Taylor & Francis Group, LLC
ISSN: 1058-0530 print / 1934-8703 online
DOI: 10.1080/10580530.2015.1044338

Augmenting Data Warehouses with Big Data

Nenad Jukić[1], Abhishek Sharma[1], Svetlozar Nestorov[1], and Boris Jukić[2]
[1]*Quinlan School of Business, Loyola University Chicago, Chicago, Illinois, USA*
[2]*School of Business, Clarkson University, Potsdam, New York, USA*

In the past decade, corporations are increasingly engaging in efforts whose aim is the analysis and wide-ranging use of big data. The majority of academic big data articles have been focused on methods, approaches, opportunities, and organizational impact of big data analytics. In this article, the focus is on the ability of big data (while acting as a direct source for impactful analysis) to also augment and enrich the analytical power of data warehouses.

Keywords big data; databases; data warehouses; MapReduce

INTRODUCTION

Today's organizations are dealing with increasing amounts and complexities of data. In addition to managing and processing traditional data sources, such as operational databases and data warehouses, in the past decade corporations are increasingly engaging in efforts whose aim is the analysis and wide-ranging use of big data. Much has been written about the big data phenomenon and the majority of academic work in this field has been focused on methods, approaches, opportunities, and organizational impact of big data analytics. These articles present big data as a source that, when properly processed and analyzed, has the potential for discovery of new knowledge offering fresh and actionable insights for corporations and other organizations. In this article, the focus will be on the ability of big data (while acting as a direct source for impactful analysis) to also augment and enrich the analytical power of data warehouses.

Big data is one the major buzzwords of this decade. In this article, the hype is sifted through and the big data phenomenon is examined through the lens of information systems theory and practice. There is no doubt that the world is in the midst of the explosion of data capture opportunities resulting in a flood of unstructured data at an extremely low grain (Höller et al., 2014). As stated by McAfee and Brynjolfsson, 2012, p. 3, ". . . mobile phones, online shopping, social networks, electronic communication, GPS, and instrumented machinery all produce torrents of data as a by-product of their ordinary operations." The ever-increasing creation of massive amounts of data through an extensive array of several new data generating sources has prompted organizations, consultants, scientists, and academics to direct their attention to how to harness and analyze big data (Goes, 2014). It would not be a hyperbole to claim that big data is possibly the most significant "tech" disruption in business and academic ecosystems since the meteoric rise of the internet and the digital economy (Agarwal & Dhar, 2014).

However, in the midst of this unbridled enthusiasm, the ever growing chatter about big data movement often contains an increasingly common misconception that standard data warehouse architecture is somehow incompatible with big data. This type of claim is grounded in reasoning that the fundamentally different nature of big data is irreconcilable with data warehouse architecture. For example, one of the commonly mentioned reasons for the alleged incompatibility of traditional data warehousing approaches and new big data methodologies is the notion that data warehouses are primarily focused on "stock" or fixed data supply oriented on the past and not suited for real time streaming analytics (Davenport, Barth, & Bean, 2012). Moreover, some even consider that the entire idea of a data warehouse in the era of big data is obsolete. According to Devlin, 2012, p. 3, ". . . the current hype around 'big data' has caused some analysts and vendors to declare the death of data warehousing, and in some cases, the demise even of the relational database."

In this article, the authors demonstrate how, instead of supplanting data warehouses, big data serves as an additional data analysis initiative, often in a symbiotic relationship with data warehouses. Through use of clear examples we will illustrate the added value of combining these two types of corporate data analytics initiatives.

The rest of this article is organized as follows. In the next section, the definitions of operational databases and data warehouses is examined, and the definition of big data in juxtaposition with these standard definitions is clarified, and in the subsequent section, a running example is given that illustrates and explains the differences between these three types of corporate data repositories. In the penultimate section, the running

Address correspondence to Nenad Jukić, Quinlan School of Business, Loyola University Chicago, 16 E. Pearson St., Chicago, IL 60611, USA. E-mail: njukic@luc.edu

example is used to illustrate the synergies of combining big data and data warehousing efforts, and a conclusion is given in the last section.

CORPORATE DATA REPOSITORIES

Today's organizations are inundated with data of various formats and sizes, from documents, spreadsheets, and presentations to large corporate data sets. Generally speaking, large corporate data sets can be categorized into three broad categories: operational database data sets, data warehouse data sets, and big data sets. The following will give a brief overview of these categories.

Operational Databases

Operational databases are modeled and structured data sets that collect and present operational information in support of daily business procedures and processes, such as increasing the balance on the customer's credit card account after a credit card purchase is made or issuing an e-ticket to a customer who purchased a flight on a travel web site. Creating an operational database involves the process of requirement collection that specifies in advance what data will be stored in the future operational database, and in what manner. The requirement collection is followed by conceptual and logical database modeling. Conceptual database modeling creates a conceptual database model that visualizes collected requirements. Typically this process uses the Entity–Relationship (ER) modeling method, a well-known conceptual database modeling technique. Logical database modeling converts the conceptual model into a model that is implementable by a database management systems (DBMS) software. The most commonly used logical database model is the relational database model implementable by the relational DBMS (RDBMS) packages such as Oracle, IBM-DB2, Microsoft SQL Server, Microsoft Access, ans PostgreSQL, to mention a few.

Logical database model creates the structure of the database, which is captured in database metadata. The database metadata contains the names of tables, the names of table columns, data types, and sizes of database columns, etc. Once the operational database is implemented, the data is stored in it in a highly structured and organized way. All of the data must fit into the predesigned table and columns and it must be of the appropriate data type and size. The structured nature of the operational database, resulting from the modeling process, enables it to be queried and used in a straightforward way.

Data Warehouses

Data warehouses and data marts are modeled and structured data sets that collect and present analytical information in support of analytical tasks, such as establishing patterns of credit card use among various segments of customers, or revealing sales trends in the airline industry. Data warehouses and data marts are both structured analytical databases, with the only substantial difference being that data warehouses are larger in scope. For simplicity, this article focuses on data warehouses, but the presented concepts and principles are also applicable to data marts.

Data warehouses store and maintain analytical data separately from operational databases. Figure 1 shows a high-level view of the architecture of a data warehousing system. The analytical data in a data warehouse or data mart is periodically retrieved from various data sources. Typical data sources are internal corporate operational databases, while other data sources can include external data and big data sources (discussed in the next subsection).

The data from data sources is brought into the data warehouse or data mart via the process called ETL, which stands for extraction, transformation and load. The ETL infrastructure extracts analytically useful data from the chosen data sources, transforms the extracted data so that it conforms to the structure of the data warehouse (while ensuring the quality of the transformed data) and then loads the data into the data warehouse.

Like operational databases, data warehouses are also modeled and highly structured data sets. The requirements for the data warehouse are collected in advance, visualized, and implemented as a structured data model. Techniques such as ER Modeling or Dimensional Modeling (a specialized data warehousing modeling method resulting in so called star schemas) are used during the modeling process. The data warehouse logical data model is typically implemented by an RDBMS, such as Oracle, an IBM-DB2, or one of the specialized data warehousing RDBMS packages, such as Teradata or Greenplum. As with operational databases the data warehouse metadata captures the modeled structure of the data warehouse. Once the data warehouse is implemented its data is stored in a structured and organized way which enables it to be analyzed and used in a straightforward fashion.

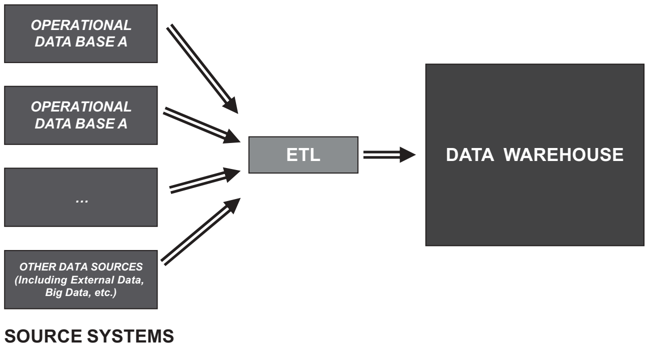


FIG. 1. High-level view of data warehouse architecture.

Big Data

Big data refers to data sets in corporations and organizations that contain massive volumes of diverse and rapidly growing data that is not formally modeled (Jukic, Vrbsky, & Nestorov, 2013). Big data is typically unstructured (such as text, video, audio) or semi-structured (such as web-logs, e-mails, tweets). Unstructured data, such as a text document, has no explicit structure. Semi-structured data may involve some structure. For example, a collection of e-mails has a simple structure that distinguishes the sender, recipient, subject and body of the message, but the message itself is unstructured. Unstructured data, semi-structured data, low-structure data, and lightly structured data, are all various terms that describe data with little or no formal metadata. The lack of formal metadata in big data sets is caused by the fact that, unlike operational databases and data warehouses, the repositories of big data sets (typically flat files) are not created through the process of formal database modeling. Consequently, big data is not straightforwardly organized into tables with rows and columns.

In literature big data has been described as characterized by three, four, five, six, or even seven Vs: volume, variety, velocity, veracity, variability, value, and visualization (Biehn, 2013; Goes, 2014; Hitzler & Janowicz, 2013; Knilans, 2014; Laney, 2001; Normandeau, 2013; Sicular, 2013). While expressive, describing big data with simple Vs can also be somewhat confusing and misleading. The whole cottage industry that has sprouted around multiplying and further extending the meaning of those terms has been under criticism (Grimes, 2013). As described in the following paragraphs, these Vs stated on their own, without a context or an example, can be applied to operational databases and data warehouses as well and, therefore, are not completely and exclusively descriptive of big data.

Volume refers to the large size of data. It is true that big data sets are large, but that is not a property that uniquely describes big data sets. Many operational databases can be voluminous in size as well. In fact the first Very Large Data Bases (VLDB) conference was held in 1975, and this conference still remains one of the most eminent venues for the timely dissemination of research and development results in the field of database management (www.vldb.org). Data warehouses, especially ones with fine granularity of data such as a line-item of a transaction, are also voluminous in size, as they contain a massive amount of detailed data over a large time horizon (often years' worth of data).

Variety refers to the abundance of different types of big data sources. Big data can include various kinds of data originating from smart devices, web-logs, public records, social media, and numerous other types of data outlets (Franks, 2012; Jukic et al., 2013). However, operational databases and data warehouses can also include a variety of types of sources. For example it has become common for retailers to combine in their data warehouses structured sales data from their operational databases with acquired structured weather data to observe how weather patterns influence customer purchases.

Velocity refers to the high speed of incoming data into big data sets. A big data repository that receives and stores tweets or GPS data from smart phones certainly experiences high velocity of incoming data. However, a similar claim can be made for an operational database or an active data warehouse that receives real time (or near real time) transaction data about credit card purchase for a credit card company.

Veracity of the big data refers to the data quality issues such uncertainties, biases, noise, and abnormality in data (Normandeau, 2013). For example it is quite certain that a number of tweets will contain incomplete, misspelled, and non-uniformly abbreviated text. The standard data quality measures of accuracy, completeness, uniqueness, conformity, timelines, and consistency are more likely to be lower in many of the big data sets than in typical formally modeled and structured operational databases and data warehouses. But that is not to say that operational databases and data warehouses do not experience data quality issues at all. In fact data scrubbing and cleansing are common procedures in maintenance and operation of databases and data warehouses, as low quality data, such as misspelled or incomplete data, appears routinely in structured and modeled data repositories as well.

Variability of big data refers to the possibility of interpreting the data in different ways. For example a stream of tweets or a collection of GPS signal recordings are open to various interpretations as to what kind of meaning and knowledge they contain. In general variability of interpretations is a common issue with big data sets, but again not solely exclusive to it. Structured information, such as, for example, a database containing state-wide collection of individual high-school GPA data can also be open to interpretation

Value of big data refers to the usefulness and actionability of the information extracted from the big data sets. As stated by Franks (2012), much of big data is of little use and one of the main issues in big data management is how to recognize valuable big data sets or portions of big data sets. In many cases, searching for value in big data requires approaches that are explorative in nature, i.e., they may or may not yield valuable results (Franks, 2012). Operational databases and data warehouses, as formally modeled data sets, are more uniformly likely to be of value, because they are (ostensibly) designed based on requirements for achieving something useful with that data. However, many organizations underutilize their databases and data warehouses to a point where there are instances of unused or barely used tables and columns in such repositories. In fact, identifying and deleting unused tables and columns is one of the standard tasks of database administrators (Biju & Bryla, 2002).

Visualization of big data refers to the necessity for illustrative and rich visualization of big data sets in many cases to grasp the meaning of it. However, the same can be said of many operational databases and data warehouses. Rich and innovative visualization of data is a growing trend for both big data and formally modeled data sources.

To avoid vagueness in describing big data with simple Vs, a more precise description of big data using such Vs would have to state that big data refers to the massive volumes of diverse and rapidly growing data sets which, when compared with most databases and data warehouses:

- are much less structured, with little or no metadata;
- in general exhibit larger volume, velocity, and variety;
- have higher likelihood of having data quality (veracity) issues;
- have higher possibility of different interpretations (variability);
- need a more explorative and experimental approach to yield value (if any);
- are as likely (if not even more so) to benefit from rich and innovative visualization.

In the section "Insurance Company—Big Data Set," a concrete example that illustrates the Vs of a big data set as compared to an operational database set and data warehouse set is given.

Standard database and data warehousing technologies, such as RDBMS and Structured Query Language (SQL), cannot adequately capture the unstructuredness and heterogeneity of big data and, therefore, cannot be used to effectively process big data sets. Instead, several approaches have emerged in order to deal with such data, with the most prominent one being the MapReduce framework, and Hadoop as the most popular implementation of the MapReduce framework. The big data processing and Hadoop will be discussed later in the section "Insurance Company—Big Data Set," but first, an example of all three types of large corporate data sets within the same organization will be given. This example will be used as a running scenario to discuss the augmentation of data warehouses with big data in the penultimate section.

RUNNING EXAMPLE—CORPORATE DATA REPOSITORIES

To illustrate the three categories of large corporate data sets—operational database, data warehouse, and a big data set— an abbreviated and simplified example of a car insurance company is used. The example is simplified for brevity and clarity.

Insurance Company—Operational Database

To start, an operational database is presented. Figure 2 illustrates a claims operational database within this company. The top part of Figure 2 shows the ER diagram for the claims operational database. The middle part of Figure 2 shows the relational schema for the claims operational database, mapped from the ER diagram, and the bottom part of the figure shows sample data in the claims operational database.

This operational database stores the data that supports day-to-day business processes of the insurance company. For example, when the company issues a new car policy, it records information about the new car in the CAR table, information about the customer in the CUSTOMER table if the policy holder is a new customer, information about the new policy in the CARPOLICY table, and finally, the fact that the customer is the policy holder is recorded in the INCLUDES table. Note that every piece of data that is generated by this business transaction is not only captured and stored but also modeled and expected. Furthermore, the use of the data is already known and no additional raw information is kept beyond what appears in the operational tables. In contrast, as will be illustrated later in this article, big data is typically stored in its original un-modeled format and may be processed multiple times for different business purposes (Kimball, 2012).

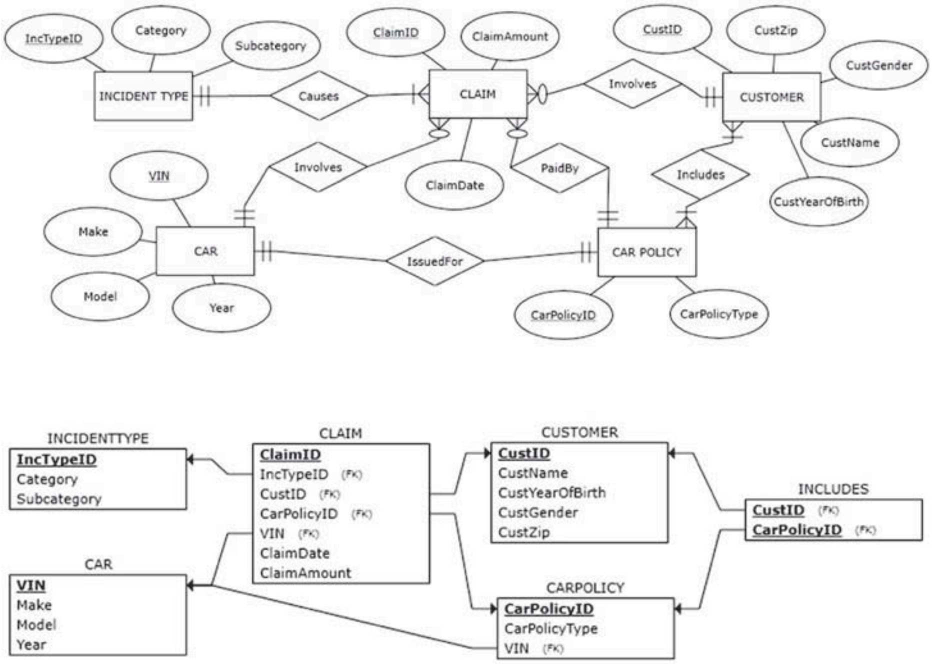
The next subsection shows how the claims data is organized in the data warehouse.

Insurance Company—Data Warehouse

The insurance company developed the data warehouse whose subject of analysis is claims. The main source for this data warehouse is the claims operational database shown in Figure 2. Another source for this data warehouse is an external data source showing a "blue book" value for various cars. In particular, the external source shows that a market value for 2008 Ford Escape is $9,000 and for 2009 Honda Civic is $7,500. This external source will be used to add more information to the data warehouse.

Figure 3 shows the insurance company data warehouse whose subject of analysis is claims and whose sources are operational database shown in Figure 2 and the "blue book" value external source. Top part of Figure 3 shows the star schema for the data warehouse and the bottom part shows sample data.

Dimensional modeling technique used for developing star schemas distinguishes two types of tables—facts and dimensions. Fact tables contain measures related to the subject of analysis. In this case, CLAIMS Fact table contains a numeric measure claim amount. Fact tables are connected to dimension tables that provide the basis for analysis of the subject. In this case dimensions CALENDAR, CUSTOMER, CARPOLICY, and INCIDENTTYPE provide the basis for analysis of claim amount. The primary keys of dimensions, known as surrogate keys, are systems-generated keys whose purpose is to improve the efficiency and handling of updates of records in dimensions. Other than the primary key columns of dimensions, all of the other data shown in Figure 3 is directly sourced or derived from the claims operational database shown in Figure 3, except for the BlueBookValue column, which is sourced from the external "blue book" source. Structuring and consolidating data from various sources into a data warehouse in this fashion allows business analysts engage in detailed and efficient analysis in a productive fashion. The necessary analytical data



CLAIM

ClaimID	CarPolicyID	IncTypeID	VIN	CustID	ClaimDate	ClaimAmount
C-111	CP11	IT1	1FM1111	111	1-Jan-2015	$400
C-222	CP12	IT2	JHM2222	333	1-Jan-2015	$900

CUSTOMER

CustID	CustName	CustYrOfBirth	CustGender	CustZip
111	Tina	1972	Female	60137
222	Tony	1970	Male	60611
333	Pam	1971	Female	35401

CAR

VIN	Make	Model	Year
1FM1111	Ford	Excape	2008
JHM2222	Honda	Civic	2009

INCIDENTTYPE

IncType	Category	Subcategory
IT1	Multicar Collision	Fender Bender
IT2	Single Car Collision	Front Pole Hit

CARPOLICY

CarPolicyID	CarPolicyTyp	VIN
CP11	Collision	1FM1111
CP12	Full Coverage	JHM2222

INCLUDES

CarPolicyID	CustID
CP11	111
CP12	222
CP12	333

FIG. 2. Claims operational database.

is consolidated in one place and structured for analysis. This allows analysts to focus all their efforts on analyses, instead of having to use much of their time rummaging through different data sets.

For example, analysts can use standard Online Analytical Processing (OLAP) tools connected to the data warehouse to examine the relationships between the day of the week when a claim is filed and the make, model, and year of the car and the zip code of the customer. This work can be done quickly and efficiently, without spending any time on data preparation, since all pertinent data is organized for analysis and available at one location, i.e., the data warehouse.

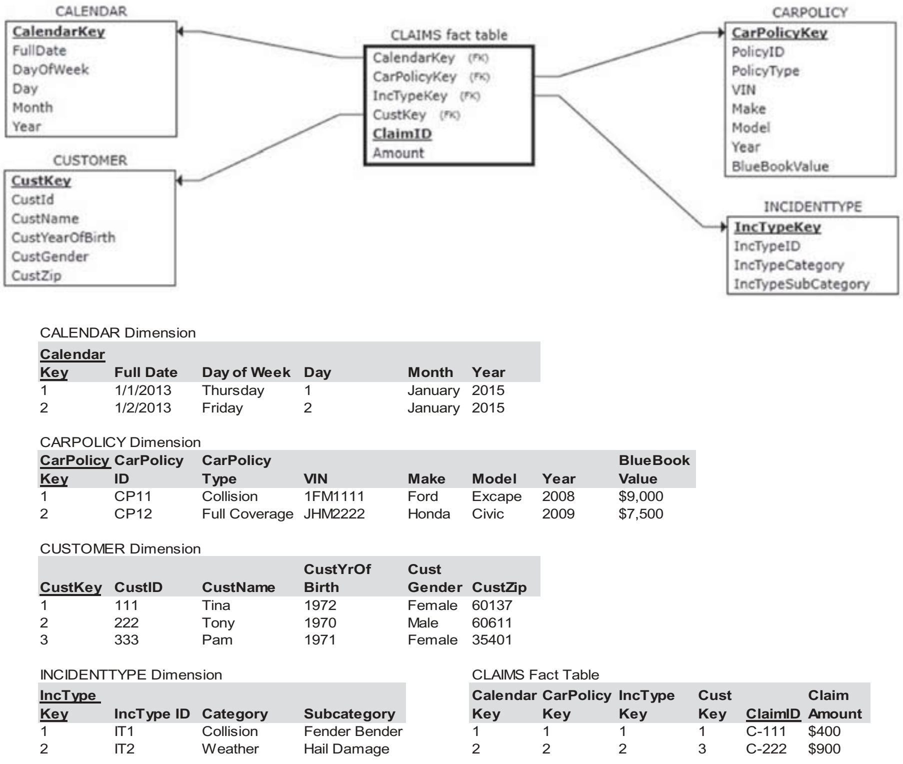
The next subsection shows an example of a big data set for this insurance company.

Insurance Company—Big Data Set

The insurance company in this scenario is an adopter of car telemetry technology. The company requires its customers to install a car telemetry device in their vehicles, similar to Garmin or TomTom car navigation devices. The data generated by these devices is collected and stored by the insurance company as a big data set. It is then used for adjusting insurance policies by increasing or reducing premiums based on driving data.

Car telemetry devices record the speed and position of a car at all times. A small sample of data generated by such device is shown here:

```
1.1.2015; 10:00:00

[(41.879632, −89.064769), 0], [(41.879632, −89.064769), 0],
[(41.879632, −89.064769), 0],

[(41.898933, −89.061731), 3], [(41.900747, −89.064872), 6], . . .

. . .

1.1.2015; 13:25:00

[(41.898239, −87.628423), 0], [(41.898239, −87.628423), 0],
[(41.897311, −87.627119), 5],
```



CALENDAR Dimension

Calendar					
Key	Full Date	Day of Week	Day	Month	Year
1	1/1/2013	Thursday	1	January	2015
2	1/2/2013	Friday	2	January	2015

CARPOLICY Dimension

CarPolicy	CarPolicy	CarPolicy					BlueBook
Key	ID	Type	VIN	Make	Model	Year	Value
1	CP11	Collision	1FM1111	Ford	Excape	2008	$9,000
2	CP12	Full Coverage	JHM2222	Honda	Civic	2009	$7,500

CUSTOMER Dimension

			CustYrOf	Cust	
CustKey	CustID	CustName	Birth	Gender	CustZip
1	111	Tina	1972	Female	60137
2	222	Tony	1970	Male	60611
3	333	Pam	1971	Female	35401

INCIDENTTYPE Dimension

IncType			
Key	IncType ID	Category	Subcategory
1	IT1	Collision	Fender Bender
2	IT2	Weather	Hail Damage

CLAIMS Fact Table

Calendar	CarPolicy	IncType	Cust		Claim
Key	Key	Key	Key	ClaimID	Amount
1	1	1	1	C-111	$400
2	2	2	3	C-222	$900

FIG. 3. Claims data warehouse.

[(41.897105, −87.626878), 12], [(41.896594, −87.626792), 18], . . .

. . .

1.1.2015; 18:33:05

[(41.791636, −87.582976), 0], [(41.792100, −87.581989), 7], [(41.794307, −87.583534), 16],

[(41.793188, −87.580057), 27], [(41.794083, −87.579628), 39], ...

This sample contains the telemetry data for first five seconds of three trips of one car in use. For each trip, the date and time stamp is followed by the information about the longitude, latitude, and speed of the car recorded for each second of the trip. For example, the first car trip shown in the sample started at Latitude 41.879632 and Longitude −89.064769. For the first three seconds of the car in operation during this trip the driver kept the car in place and then slowly started moving, with speed in the fourth second being three miles per hour and the speed in the fifth second being six miles per hour.

This data is used by the insurance company to observe and analyze driving behavior of its customers. This type of data is not appropriate for processing using standard database processing technology, such as RDBMS. Instead, the insurance company can, for example, use Hadoop technology using MapReduce framework. The following is a brief description of MapReduce approach.

MapReduce computation has two phases: the map phase and the reduce phase. During the map phase a program (usually written in Java or another general purpose language) is used to map every record in the data set (e.g., set of signals for each individual trip) to zero or more so called key-value pairs. A key can be a single value (such as a word, phrase, integer, or a URL) or a more complex value. For example, suppose that the insurance company wants to find out how many instances of slow, medium, and fast acceleration and deceleration has a driver performed during his or her trip. The insurance company defines slow acceleration when speed increases, in a short period of time (e.g., 5 seconds) by less than 10 mph, medium acceleration when speed increases by between 10 and 20 mph, and fast acceleration when the speed increases more than 20 mph. Deceleration is defined using the same thresholds for speed decreases. Performing this task using MapReduce is illustrated in Figure 4.

Map function parses through the set of signals resulting from a trip and calculates the number of accelerations or decelerations of different types. Given a set of signals for an individual trip, a map function can output up to six pairs, where each pair consists of a key indicating the type of acceleration/deceleration, and a value that shows how many times that type of acceleration/deceleration occurred during

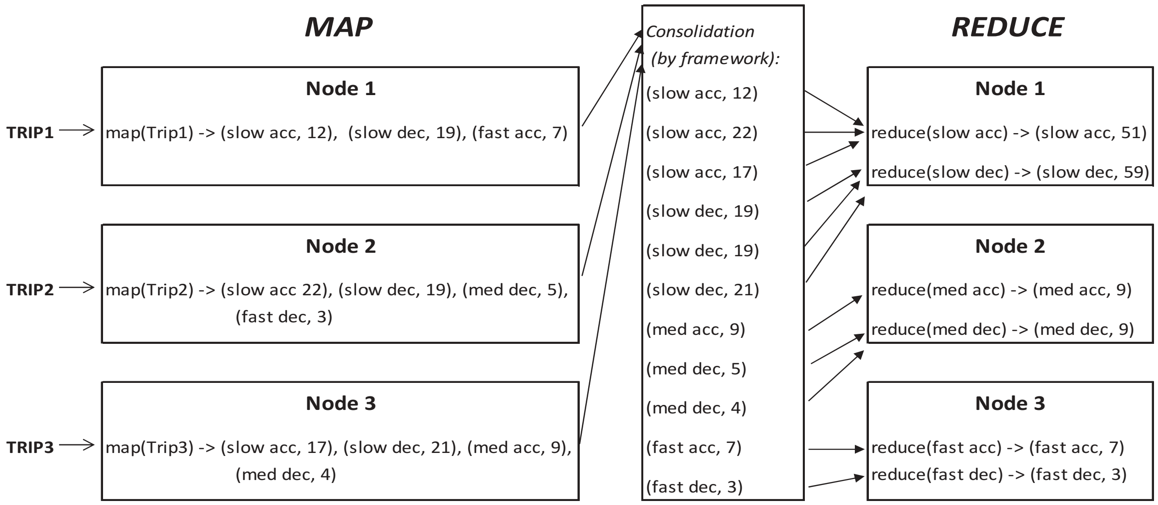

MAP
Consolidation (by framework):
REDUCE
Node 1
TRIP1
map(Trip1) -> (slow acc, 12), (slow dec, 19), (fast acc, 7)
Node 2
TRIP2
map(Trip2) -> (slow acc 22), (slow dec, 19), (med dec, 5), (fast dec, 3)
Node 3
TRIP3
map(Trip3) -> (slow acc, 17), (slow dec, 21), (med acc, 9), (med dec, 4)
(slow acc, 12)
(slow acc, 22)
(slow acc, 17)
(slow dec, 19)
(slow dec, 19)
(slow dec, 21)
(med acc, 9)
(med dec, 5)
(med dec, 4)
(fast acc, 7)
(fast dec, 3)
Node 1
reduce(slow acc) -> (slow acc, 51)
reduce(slow dec) -> (slow dec, 59)
Node 2
reduce(med acc) -> (med acc, 9)
reduce(med dec) -> (med dec, 9)
Node 3
reduce(fast acc) -> (fast acc, 7)
reduce(fast dec) -> (fast dec, 3)

FIG. 4. MapReduce processing the telemetry data.

that trip. For example, assume that the first trip had 12 instances of slow acceleration, 19 instances of slow deceleration, and 7 instances of fast acceleration, and that there were no instances of medium acceleration or of medium and fast deceleration in the first trip. Therefore, the map function applied to first trip returned three key-value pairs. The same type of processing occurs for other trips.

Because the output of each pair depends on a single input record, the input data (containing records depicting many trips) and the work of creating output pairs can be distributed among many computing nodes. Once the results are computed on different nodes, they are consolidated by the framework. The reduce step then gathers all records with the same key and generates a single output record for every key. As shown in Figure 4, the reduce step can also be distributed among many compute nodes.

Implementations of MapReduce framework, such as Hadoop, automatically perform tasks, such as distributing the data among nodes, collecting the output of the map phase, sorting it, and distributing it among the compute nodes executing the reduce phase. The programmer only has to write the map and reduce functions. These functions can vary in terms of complexity, from very simple ones (such as simple parsing and counting) to fairly complex ones embedding data mining algorithms. Same data sets can be processed using different map and reduce functions, looking for different types of information from the data. That is why the original data set after being processed for one particular purpose, is often kept intact and available for additional types of analyses.

The small telemetry data sample used here is a miniscule portion of a massive file exemplary of the term big data. Figure 4 shows how such big data set can be processed using Hadoop. This data set can also be used to illustrate with an actual example, the Vs listed in the section “Big Data,” that are often used to define the term big data.

If the insurance company has hundreds of thousands of customers, policies and claims, the size of its operational database or data warehouse can be quite voluminous. However, those sizes are orders of magnitude smaller than the enormous *Volume* of telemetry generated data.

Telemetry data is just one example of the *Variety* of possible different types of big data sources. In addition to telemetry data, big data can include various kinds of data originating from different sources such as sensors, RFID tags, smart phones, gaming consoles, social media, and many others.

The driving telemetry data example is also indicative of the term *Velocity*, as many instances of data from devices issued to all of the customers are streaming into the big data set every second. The velocity of incoming data into the telemetry big data set is much higher than the pace of adding new claims or customers to the operational database or data warehouse.

The telemetry data set can be prone to data quality, i.e., *Veracity*, issues. An example of dealing with data *Veracity* in this big data set would be a series of faulty signals sent and recorded due to spotty GPS coverage, which would lead into inaccuracies in the telemetry data set.

Variability of data interpretations of this big data set refers to different interpretations of data. For example, variability of data interpretations would have to account for a difference between a vehicle not operating and a vehicle in a tunnel, which would be both depicted by the absence of data in the data set, but would have to be interpreted differently.

The telemetry example can also be used to illustrate the role of *Visualization* in a big data set. For example, superimposing the set on a geographical map and showing longitudinal data for each 5-minute period directly preceding every incident that led into a claim for auditing purposes, would be an example of useful utilization of this data set enabled by visualization.

The telemetry data set is an example of a big data set that can provide *Value*, as was indicated with several examples so far. However, not all of the value of this data set is as obvious and self-evident. In the next section, additional value of this data set will be elaborated on, that is not apparent at first glance.

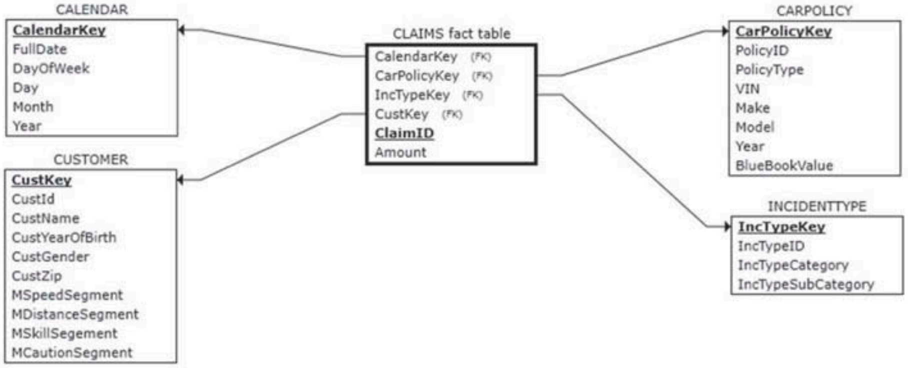
RUNNING EXAMPLE—BIG DATA AS A DWH SOURCE

The telemetry data can be used to observe driving habits and incidents of individual drivers in order to build individual pricing models and offer premium discounts to the safest ones while increasing premiums for repeat offenders. In addition to this immediate operational use, this big data set can be used to enrich the analytical power of the data in the data warehouse. For example, the telemetry data can be processed using a different set of map functions that are designed to identify fine segments of driving habits among customers and recognize which driver fits into which segment. The car telemetry data can be actually much more complex than what is shown in the previous section, and it can be supplemented with additional information about placement of road signs and lights, speed limits, road types, etc.

Suppose that the insurance company developed measures that, based on precise telemetry data of each driver, can evaluate with a score of 1 to 10, drivers in four different segments depicting four aspects of their driving behavior: speed, distance, skill, and caution. A speed value of 1 would indicate a driver who typically drives at very low speeds, and a speed value of 10 would indicate a driver who typically drives at very high speeds. A distance value of 1 would indicate a driver who typically drives very short distances, and a distance value of 10 would indicate a driver who typically drives very long distances. A skill value of 1 would indicate a driver with very low skills, and a skill value of 10 would indicate a supremely skilled driver. The reduce function for this segment would, for example, use the information in the data set that illustrates drivers' reflexes, agility of merging into traffic, etc. A caution value of 1 would indicate a driver abandoning any sense of caution, and a caution value of 10 would indicate an extremely cautious driver. The reduce function for this segment would, for example, use the information about drivers' behavior regarding speed limit, road signs, etc.

Once the information about drivers' values for various measured segments is calculated from the big data set, this information can then be added to the customer dimension of the data warehouse as shown in Figure 5.

With this added information analysts can analyze the frequency and monetary value of claims for customers in various measured segments: MSpeed Segment, MDistance Segment, MSkill Segment, and MCaution Segment. A possible finding



CALENDAR Dimension

Calendar Key	Full Date	Day of Week	Day	Month	Year
1	1/1/2013	Thursday	1	January	2015
2	1/2/2013	Friday	2	January	2015

CARPOLICY Dimension

CarPolicy Key	CarPolicy ID	CarPolicy Type	VIN	Make	Model	Year	BlueBook Value
1	CP11	Collision	1FM1111	Ford	Excape	2008	$9,000
2	CP12	Full Coverage	JHM2222	Honda	Civic	2009	$7,500

CUSTOMER Dimension

CustKey	CustID	CustName	CustYrOf Birth	Cust Gender	CustZip	MSpeed Seg	MDistance Seg	MSkill Seg	MCaution Seg
1	111	Tina	1972	Female	60137	9	9	8	8
2	222	Tony	1970	Male	60611	8	7	6	7
3	333	Pam	1971	Female	35401	7	4	6	7

INCIDENTTYPE Dimension

IncType Key	IncType ID	Category	Subcategory
1	IT1	Collision	Fender Bender
2	IT2	Weather	Hail Damage

CLAIMS Fact Table

Calendar Key	CarPolicy Key	IncType Key	Cust Key	ClaimID	Claim Amount
1	1	1	1	C-111	$400
2	2	2	3	C-222	$900

FIG. 5. Claims data warehouse—expanded with big data.

may be, for example, that people with the highest and lowest levels of caution have on average more claims than people in the middle of the caution segment. That kind of finding would simply be impossible without adding the information extracted from the big data into the data warehouse.

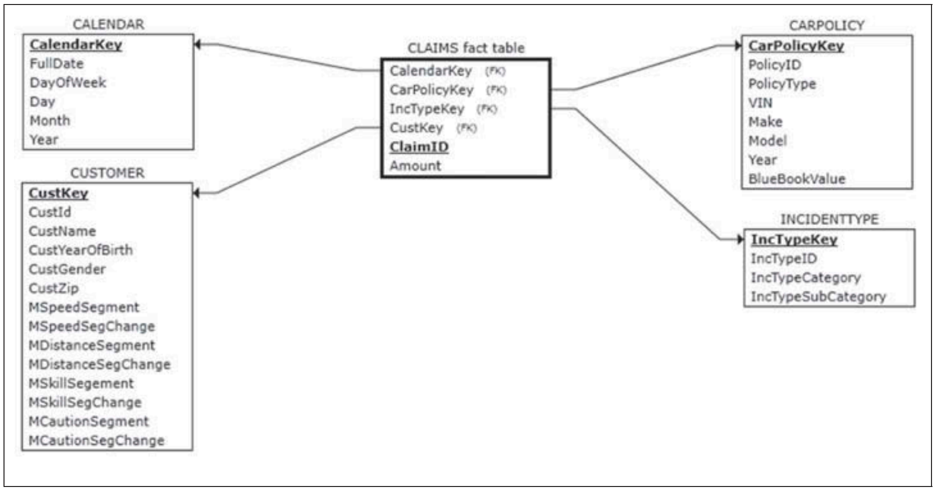
The telemetry data set offers layers of possible analytical use. In addition to segmenting customers into Caution, Skill, Speed, and Distance measured segments the Change column can be added to each of these columns. A change value of 0 would indicate that the customer has been in this segment from the first recording. A change value of −1 (or −2, −3, and so on) would indicate that the current value went down by 1 (or 2, 3, and so on) from the previous state, and a change value of +1 (or +2, +3, and so on) would indicate that the current value went up by 1 (or 2, 3, and so on). Such data opens up the possibility for much finer analysis of claims. Figure 6 shows the data warehouse with this additional information.

A possible finding based on this expanded data may be that a certain change in one or more measured segments is particularly likely to lead into incidents and claims. Such information can be used to issue warnings to drivers and prevent incidents from happening in the first place.

CONCLUSION

In this article, the big data phenomenon was examined in the context of corporate information systems and illustrated how big data fits with the existing paradigm of populating the data warehouse via ETL from a variety of data sources. First, the smorgasbord of big data definitions was surveyed from the academic and industry literature and enumerated the different characteristics attributed to big data. The analysis showed that many of these characteristics are not unique to big data but also appear in other corporate data repositories. Based on the analysis, a more focused and granular definition of big data was proposed.

In addition to setting the record straight on the definition and specificity of big data, the article also illustrated how big data fits with existing practices of the corporate data warehouse. The novel technologies, such as Hadoop, developed to tame the big data explosion, naturally become part of the arsenal of ETL developers. With these technological advancements, big data, despite its challenges, has become another type of source data set for the corporate data warehouse. As stated in Lopez and D'Antoni, 2014, p. 5, "... the modern enterprise data warehouse (EDW) needs to bring together the technologies and data



CALENDAR Dimension

Calendar Key	Full Date	Day of Week	Day	Month	Year
1	1/1/2013	Thursday	1	January	2015
2	1/2/2013	Friday	2	January	2015

CARPOLICY Dimension

CarPolicy Key	CarPolicy ID	CarPolicy Type	VIN	Make	Model	Year	BlueBk Value
1	CP11	Collision	1FM1111	Ford	Excape	2008	$9,000
2	CP12	Full Coverage	JHM2222	Honda	Civic	2009	$7,500

CUSTOMER Dimension

CustKey	CustID	CustName	CustYrOf Birth	Cust Gender	CustZip	MSpeed Seg	MSpeed SegCh	MDist Seg	MDist SegCh	MSkill Seg	MSkill SegCh	MCaut Seg	MCaut SegCh
1	111	Tina	1972	Female	60137	9	+1	9	0	8	-2	8	-3
2	222	Tony	1970	Male	60611	8	0	7	0	6	+1	7	0
3	333	Pam	1971	Female	35401	7	0	4	0	6	0	7	+1

INCIDENTTYPE Dimension

IncType Key	IncType ID	Category	Subcategory
1	IT1	Collision	Fender Bender
2	IT2	Weather	Hail Damage

CLAIMS Fact Table

Calendar Key	CarPolicy Key	IncType Key	Cust Key	Claim ID	Claim Amount
1	1	1	1	C-111	$400
2	2	2	3	C-222	$900

FIG. 6. Claims data warehouse—further expanded with big data.

required to support traditional business needs and stronger predictive analytics, leveraging large data sets.” In this article, an illustrative example that demonstrate how processed big data augments the existing analytical data and adds value to the organizations was presented. One of the most common big data mistakes made by corporations is treating big data as a completely separate and distinct issue (Franks, 2014). This article illustrated the benefits of an integrated approach where a big data initiative goes hand in hand with other data projects.

In summary, the contributions in this article are twofold: Elucidating the definition of big data in the context of the traditional corporate data repositories and identifying how big data sources and technologies fit with standard data warehousing practices.

AUTHOR BIOS

Nenad Jukić is a professor of Information Systems at the Quinlan School of Business at Loyola University Chicago. He conducts research in various information management-related areas, including database modeling and management, data warehousing, business intelligence, data mining, business analytics, big data, e-business, and IT strategy. His work has been published in numerous information systems and computer science academic journals, conference publications, and books. In addition to his academic work, he provides expertise to database, data warehousing, business intelligence, and big data projects for corporations and organizations that vary from startups to Fortune 500 companies and U.S. government agencies.

Abhishek Sharma is a database/business intelligence consultant and the founder of an IT consulting company, Awishkar, Inc. He is also an adjunct professor of Information Systems at the Quinlan School of Business at Loyola University Chicago. He has worked at various information technology positions in fields such as information management, banking/quantitative finance and instrumentation, process control, and statistical analysis in manufacturing environment. Parallel with his consulting work and teaching, he conducts research in a variety of fields, including database modeling and management, data warehousing, business intelligence, data mining, very large databases (VLDBs)/big data, and IT strategy.

Svetlozar Nestorov is an assistant professor of Information Systems at the Quinlan School of Business at Loyola University Chicago. Previously he worked at the University of Chicago as a senior research associate at the Computation Institute, an assistant professor of computer science, and a leader of the data warehouse project at the Nielsen Data Center at the Kilts Center for Marketing at the Booth School of Business. He is a co-founder of Mobissimo, a venture-backed travel search engine that was chosen as one of the 50 coolest web sites by *Time* magazine in 2004. His research interests include data mining, high-performance computing, and web technologies.

Boris Jukić is a professor of Information Systems and the Director of The Masters of Data Analytics Program at Clarkson University School of Business. Previously he was also an associate dean of graduate programs at Clarkson University School of Business. He conducts active research in various information technology related areas including e-business, data warehousing, data analytics, computing resource pricing and management, process and applications modeling, as well as IT strategy. His work has been published in a number of management information systems and computer science academic journals, conference publications, and books.

REFERENCES

Agarwal, R. & Dhar, V. (2014). Editorial—Big data, data science, and analytics: The opportunity and challenge for is research. *Information Systems Research*, *25*(3), 443–448. doi:10.1287/isre.2014.0546

Biehn, N. (2013). The missing V’s in big data: Viability and value. *Wired*. Retrieved January 10, 2015, from http://www.wired.com

Biju, T. & Bryla, B. (2002). *OCA/OCP: Oracle9i DBA fundamentals I study guide: Exam 1Z0-031*. New York, NY: John Wiley & Son.

Davenport, T. H., Barth, P., & Bean, R. (2012). How big data is different. *MIT Sloan Management Review*, *54*(1), 43–46.

Devlin, B. (2012). Will data warehousing survive the advent of big data? *O’Reilly Radar*. Retrieved January 10, 2015, from http://radar.oreilly.com

Franks, B. (2012). *Taming the big data tidal wave*. New York, NY: John Wiley & Son.

Franks, B. (2014). *The analytics revolution*. New York, NY: John Wiley & Son.

Goes, P. B. (2014). Big data and is research. *MIS Quarterly*, *38*(3), iii–viii.

Grimes, S. (2013). Big data: Avoid ‘Wanna V’ confusion. *Information Week*. Retrieved January 10, 2015, http://www.informationweek.com

Hitzler, P. & Janowicz, K. (2013). Linked data, big data, and the 4th paradigm. *Semantic Web*, *4*, 233–235.

Höller, J., Tsiatsis, V., Mulligan, C., Karnouskos, S., Avesand, S., & Boyle, D. (2014). *From machine-to-machine to the internet of things: Introduction to a new age of intelligence*. Amsterdam, The Netherlands: Elsevier.

Jukic, N., Vrbsky, S., & Nestorov, S. (2013). *Database systems—Introduction to databases and data warehouses*. Upper Saddle River, NJ: Pearson/Prentice Hall.

Kimball, R. (2012). New emerging best practices for big data—A Kimball group white paper. *Kimball Group*. Retrieved January 10, 2015, from http://www.kimballgroup.com

Knilans, E. (2014). *The 5 V’s of big data. Avnet Technology Solutions*. Retrieved January 10, 2015, from http://www.ats.avnet.com

Laney, D. (2001). 3-D data management: Controlling data volume, velocity, and variety. *META Group Report, File 949*, February 2001. Stamford, CT: META Group Inc.

Lopez, K. & D’Antoni, J. (2014). The modern data warehouse—How big data impacts analytics architecture. *BI Journal*, *19*(3), 8–15.

McAfee, A. & Brynjolfsson, E. (2012). Big data: The management revolution. *Harvard Business Review*, *90*(10), 61–68.

Normandeau, K. (2013). Beyond volume, variety, and velocity is the issue of big data veracity. *Inside Big Data*. Retrieved January 10, 2015, from http://insidebigdata.com

Sicular, S. (2013). Gartner’s big data definition consists of three parts, not to be confused with three “V”s. *Forbes*. Retrieved January 10, 2015, from http://www.forbes.com