International Journal of Multidisciplinary Research and Development
Online ISSN: 2349-4182, Print ISSN: 2349-5979, Impact Factor: RJIF 5.72
www.allsubjectjournal.com
Volume 3; Issue 3; March 2016; Page No. 129-136

Influence of big data in market its various aspect and dimension

[1] Vaibhav Srivastav, [2] Deepak Kumar Singh, [3] Anshuma Srivastava, [*4] Dr Anshu Srivastava

[1] MCA Scholar, Department of Computer Application, Babu Banarsi Das National Institute of Technology and Management, Lucknow, Uttar Pradesh, India.

[2] Department of Computer Application, Babu Banarsi Das National Institute of Technology and Management, Lucknow, Uttar Pradesh, India.

[3] Department of Information Technology, Babu Banarsi Das Northern Indian Institute of Technology College, Lucknow, Uttar Pradesh, India.

[4] Uttar Pradesh Rajarshi Tandon Open University

Email: anshu_qrat@yahoo.co.in

Abstract
According to the Wikipedia, big data "Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate". In other perceptions, the "4 Vs" that characterize it (i.e., volume, velocity, variety and veracity) or the "5 Vs" (adding versatility to the previous four) are responsible for the fact that on evaluating, planning for, and executing each phase of cloud migration, including the top pros and cons, and key cost, staffing and budgeting factors to consider pre-migration. [29] In essence, big data refers to the situation that more and more aspects and artifacts of everyday life, be it personal or professional, are available in digital form, e.g., personal or company profiles, social network and blog postings, buying histories, health records, to name just a few, that increasingly more data gets dynamically produced especially on the Internet and on the Web, and that nowadays the tools and techniques are available for evaluating and analyzing all that data in various combinations. Numerous companies already foresee the enormous business effects that analytical scenarios based on big data can have, and the impacts that it will hence have on advertising, commerce, and business intelligence (BI). This paper reviews the issues, techniques, and applications of big data, with an emphasis on future BI architectures.

Keywords: Big data Business and Market intelligence, Market analytics, Business systematic, Real-time analysis, Business Architecture

1. Introduction
The scenario you are describing is an Internet-of-Things (loT) one. With a small amount of traffic is a small city, Big Data is not needed. But when you need to:

- Deal with thousands or millions of devices
- Handle re-routing traffic when accidents happen
- Predict failures of devices (based on years of collected device events)
- Combine data from multiple device types (traffic lights, cameras, railroad crossings, draw-bridges, emergency vehicles) etc.

A Big Data system will be needed. Ever since the beginning of the digital age, data in digital form has received a growing importance, first primarily in the business domain and later also in the private domain. Big data is a broad term for data sets or complex that conventional data processing frameworks are not able to process due to growth of internet user. This paper is about the issues, techniques, and applications of big data, with an emphasis on future BI architectures.

In a recent statistics [1], Intel reported that in a single Internet minute, 639,800 GB of global IP data gets transferred over the Internet, which can be broken down into emails, app downloads, e-commerce sales, music listening, video viewing, or social network status updates, and this number will increase significantly over the next couple of years. This already is representative of one dimension of big data, its volume or size: data is considered big if it has reached TB or PB in size, and is typically so large that it exceeds a single organization's storage capacity. Other dimensions which have become common for characterizing big data, and which together with volume are called the "5 Vs of big data" [2], are the velocity or the speed with which data is produced and needs to be consumed, the variety data can have, and the veracity the data comes with (We note that the first three of these Vs are attributed to analyst Doug Laney [3] who now works for Gartner). Velocity refers to the fact that data often comes in the form of streams which do not give the respective consumer a chance to store them for whatever purpose, but to act on the data instantly. Variety means that data can come in different forms such as unstructured (e.g., text), semi-structured (e.g., XML documents), or structured (e.g., as a table), veracity refers to the fact the data may or may not be trustworthy or uncertain and versatility refers to Challenges include analysis, capture, data curation, search, sharing, storage, transfer, visualization, querying and information privacy. These characteristic properties of big data are summarized in Fig. 1.

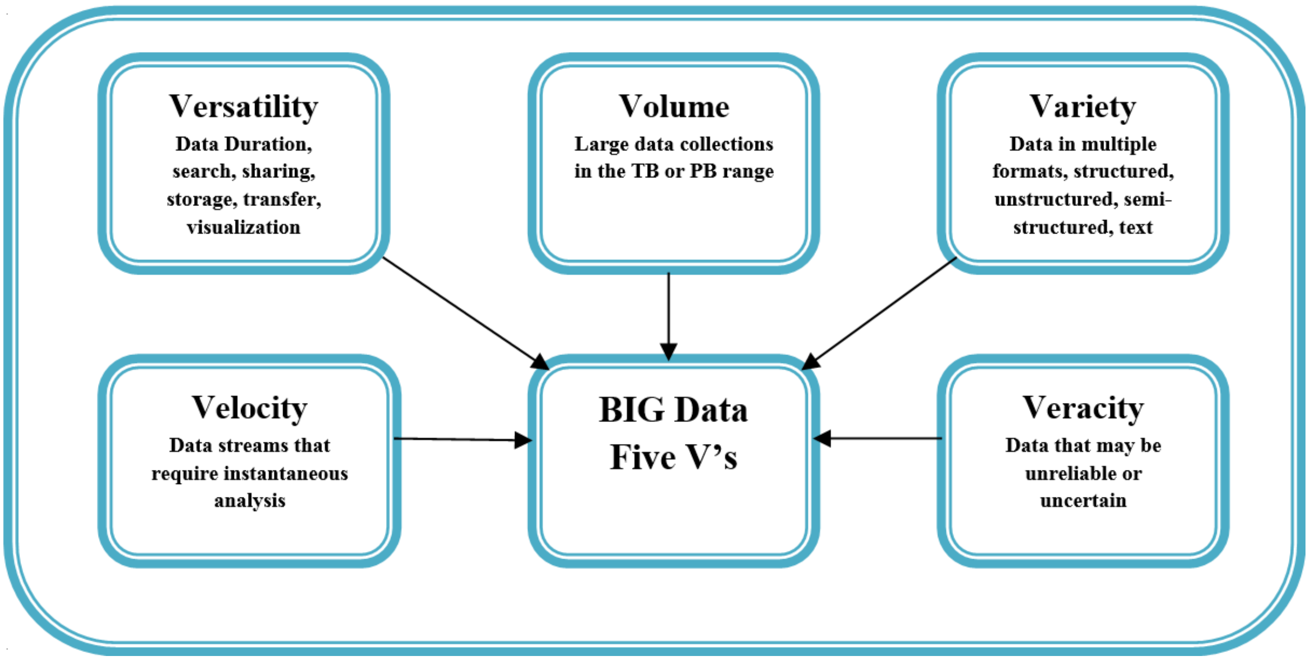

Fig 1: The defining "5 Vs" of big data

[29] We consider the transition from Web 1.0 to Web 2.0 as one of the major drivers that have led to big data. Indeed, as we have written in my book on Web 2.0 [26], this transition was determined by three parallel streams of development: the applications stream that has brought along a number of services anybody can nowadays use on the Internet and the Web; the technology stream which has provided the underlying infrastructure groundwork for all of this with fast moving and comprehensive advances in networking and hardware technology and quite a bit of progress regarding software; and finally the user participation and contribution stream (which we might also call the socialization stream) which has changed the way in which users, both private and professional ones, perceive the Web, interact with it, contribute to it, and in particular publish their own or their private information on it.

[29] These three streams have brought along a number of techniques, technologies, and usage patterns that at present converge, and the result is what has received the term "Web 2.0". While initially content was mostly read from the Web, content is nowadays constantly written to the Web; hence the term "read/write Web". An immediate consequence of the fact that more and more people publish on the Web through blogs, instant messaging, social networks, and otherwise is that increasing amounts of data arise. Additionally, data arises from commercial sites, where each and every user or customer transaction leaves a trace in a database. Several years back, this made companies start employing data warehouse technology for online analytical processing or the application of data mining tools to large data collections to generate new knowledge. Especially, these tools have reached a new maturity, so that besides stored data it is now possible to process, or to incorporate into processing, data streams which cannot or need not be stored. We indeed consider "big" data as a consequence of the Web 2.0 developments, and it remains to be seen how to exploit this data in a fruitful way.

[29] As can be done for other developments in computer science, big data can be viewed from various perspectives and in various dimensions; these are summarized in Fig. 2. As my goal in this paper is to give a brief survey of the current state of the big data area, we will first look at several use cases in Section 2 which indicate the enormous potential that can be seen in big data processing through a variety of examples and use cases; this touches the economical dimension. Section 3 covers the technological dimension and hence the technology available for handling big data, in particular technology that has made it to the center of attention recently. Section 4 takes an organizational perspective and describes how to exploit big data in an enterprise environment where a data warehouse has been the tool of choice until now; as it will turn out, a data warehouse architecture can straightforwardly be augmented to allow for big data. Section 5 concludes the paper and tries to give an outlook into what will happen next. Due to a lack of expertise of the author, the legal dimension will not be dealt with in this paper.

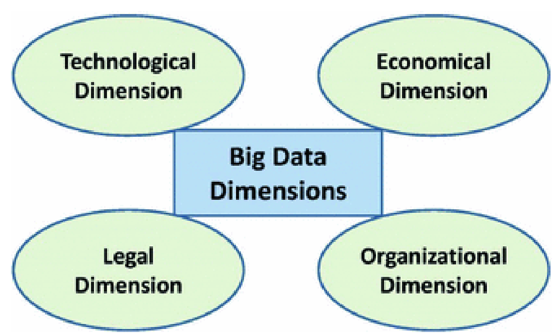

Fig 2: Big data dimensions

2. Big data use cases

In this section, we describe several use cases for big data which are intended to indicate that this is indeed a development that is different from what we have seen in the past. As will be seen, they stem from vastly distinct areas, and it has to be kept in mind that these examples do not represent an exhaustive list.

One of the examples of big data, there's two parts to the oil industry - "upstream" which is broadly finding oil and gas (exploration and production) and "downstream" which is refining and retail down to the consumer. The downstream business is much the same as any other manufacturing-to-customer supply chain organization in terms of its data use, and big data has impacted there in much the same way as in other sectors. Upstream, on the other hand, has always been about "big data", before the term "big data" was every used. A typical 3D seismic survey might involve tens of the bytes of raw data, and need to be processed on thousands of computing "cores" in parallel to form an image. As other industries start to develop ideas and approaches for managing "big data" these are starting to cross into the "upstream" side of the business, however at the moment I'd say they tend to be making existing systems/processes more efficient rather that radically changing how exploration and production is performed. In part, this is because oil exploration "big data" tends to be a modest number of very large pieces of digital information, as opposed to a very, very large number of small pieces. To take your example in mind, Traffic optimization is not a local problem. It is done on the basis of information from neighboring/adjacent/sequentially-routed traffic lights. Each traffic light cannot take an independent decision. Eventually that makes a stronger case for centralization. If we were to generalize your question, it's possible that some problems can be solved better by localization. But even in that case if the number of local units is high, the cost of individual processing units may be prohibitively more expensive than a centralized architecture.

Let's look at it from a futuristic perspective. Centralization and decentralization. In the future there will be intelligent independent units interacting with other intelligent unites. Coordinated by intelligent unit which would be smart enough to enforce policies and have override rules. There will be lot of data on the end unites as they will become smarter. Look at phones today how smarter and compact they are as compared to may be 10-15 years ago.

A commercial rather than an academic context, a big data intervention is successful when it produces value for an organization. Until then, it is merely an additional cost, a very real cost consisting of people and technology, that the organization has to carry and that ultimately serves to *reduce* the profitability of the organization. Some of the above cost could be attributed to R&D, but from the perspective of the Board of Directors, R&D is only useful if it is expected to produce a return one day, compared to being an "infinite" cost sink if the focus is not on what big data could do, but rather on what it is and the novelty of it all. In the above context, I would therefore propose that some of the top (organizational) cultural attributes could be:

- Innovation, including an understanding of the role of failure. In turn, experimentation, experience and a willingness to learn are significant contributors to innovation
- Persistence in the context of often needing to blaze your own trail to find how big data could add value in your particular organization. This in turn involves courage
- Discipline, meaning that the process is not haphazard, but diligently structured according to a particular objective to ensure that one doesn't run around in circles and produce spurious outcomes

On the point about having a (business) objective, it's non-negotiable in the context of measuring whether the big data intervention was successful.

3. Technology for handling big data

Big data analytics is often associated with cloud computing because the analysis of large data sets in real-time requires a platform like Hadoop to store large data sets across a distributed cluster and Map Reduce to coordinate, combine and process data from multiple sources.

To cope with big data, a variety of techniques, methods, and technology have been developed in recent years, which are surveyed next. In particular, when data comes in such large quantities that local or in-house storage and processing is not an option anymore, it is not a surprise that "traditional" technology focusing around a central database is no longer apt. To determine what is needed and what fits in well, we first look at requirements for big data processing and then review technologies satisfying these requirements.

In a nutshell, these requirements can be characterized as follows:

- considerable processing power for complex computations;
- scalable, distributed and fault-tolerant data processing capabilities, including temporary or even permanent storage;
- parallel programming and processing paradigms suitable for handling large collections of data;
- Appropriate implementations and execution environments for these programming models and paradigms.

Column-oriented databases

Traditional, row-oriented databases are excellent for online transaction processing with high update speeds, but they fall short on query performance as the data volumes grow and as data becomes more unstructured. Column-oriented databases store data with a focus on columns, instead of rows, allowing for huge data compression and very fast query times. The downside to these databases is that they will generally only allow batch updates, having a much slower update time than traditional models.

Schema-less databases, or No SQL databases

There are several database types that fit into this category, such as key-value stores and document stores, which focus on the storage and retrieval of large volumes of unstructured, semi-structured, or even structured data. They achieve performance gains by doing away with some (or all) of the restrictions traditionally associated with conventional databases, such as read-write consistency, in exchange for scalability and distributed processing.

Map Reduce

This is a programming paradigm that allows for massive job execution scalability against thousands of servers or clusters of servers. Any Map Reduce implementation consists of two tasks:

- The "Map" task, where an input dataset is converted into a different set of key/value pairs, or tuples.

- The "Reduce" task, where several of the outputs of the "Map" task are combined to form a reduced set of tuples (hence the name).

Hadoop
Hadoop is by far the most popular implementation of Map Reduce, being an entirely open source platform for handling Big Data. It is flexible enough to be able to work with multiple data sources, either aggregating multiple sources of data in order to do large scale processing, or even reading data from a database in order to run processor-intensive machine learning jobs. It has several different applications, but one of the top use cases is for large volumes of constantly changing data, such as location-based data from weather or traffic sensors, web-based or social media data, or machine-to-machine transactional data.

Hive
Hive is a "SQL-like" bridge that allows conventional BI applications to run queries against a Hadoop cluster. It was developed originally by Facebook, but has been made open source for some time now, and it's a higher-level abstraction of the Hadoop framework that allows anyone to make queries against data stored in a Hadoop cluster just as if they were manipulating a conventional data store. It amplifies the reach of Hadoop, making it more familiar for BI users.

Pig
PIG is another bridge that tries to bring Hadoop closer to the realities of developers and business users, similar to Hive. Unlike Hive, however, PIG consists of a "Perl-like" language that allows for query execution over data stored on a Hadoop cluster, instead of a "SQL-like" language. PIG was developed by Yahoo!, and, just like Hive, has also been made fully open source.

Wibi Data
WibiData is a combination of web analytics with Hadoop, being built on top of H Base, which is itself a database layer on top of Hadoop. It allows web sites to better explore and work with their user data, enabling real-time responses to user behavior, such as serving personalized content, recommendations and decisions.

Platfora
Perhaps the greatest limitation of Hadoop is that it is a very low-level implementation of Map Reduce, requiring extensive developer knowledge to operate. Between preparing, testing and running jobs, a full cycle can take hours, eliminating the interactivity that users enjoyed with conventional databases. PLATFORA is a platform that turns user's queries into Hadoop jobs automatically, thus creating an abstraction layer that anyone can exploit to simplify and organize datasets stored in Hadoop.

Storage Technologies
As the data volumes grow, so does the need for efficient and Effective storage techniques. The main evolutions in this space are related to data compression and storage virtualization.

Sky Tree
Sky Tree is a high-performance machine learning and data analytics platform focused specifically on handling Big Data. Machine learning, in turn, is an essential part of Big Data, since the massive data volumes make manual exploration, or even conventional automated exploration methods unfeasible or too expensive.
If data can no longer be exclusively stored locally, it is near at hand to refer to cloud storage as an extension of local or in-house capabilities, or to stream processing systems that can vastly do without considerable local storage. For the sake of completeness, the difference between a database system and a data stream system is illustrated in Fig. 3 for the aspects of querying: A database query can be sent to a database system in an ad hoc manner and each query will be processed and produce a result individually (Fig. 3a), due to the fact that data is loaded and then permanently stored. In a data stream system, on the other hand, the data is streamed to a query processor continuously and without the option of being available for long periods of time; so the query processor can only respond to queries that have previously been registered with it and can produce result for the data stream by looking at a portion of the stream available within a certain window (Fig. 3b). The figure is, however, incomplete in that a stream processing system is often complemented by local storage or even part of a regular database system.

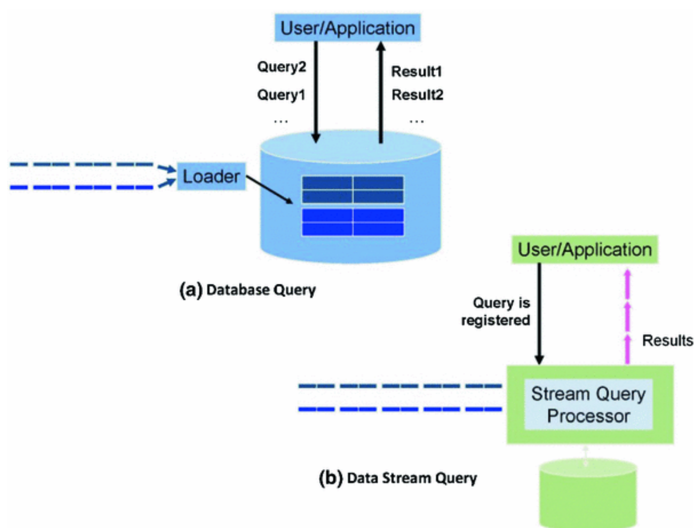

Fig 3: Database query vs. data stream query

[29] So for both computing and storage, cloud sourcing has become a typical scenario, which according to the US National Institute for Standards and Technology (NIST) is defined as follows—cloud sourcing is the utilization of IT capabilities from a cloud service provider based on the cloud paradigm with the following five characteristics: resource pooling, rapid elasticity, on-demand self-service, broad network access, and measured service. NIST defines three service models: Software-, Platform- and Infrastructure-as-a-Service, abbreviated as SaaS, PaaS and IaaS, respectively, which represent different types of services and, in a sense, different levels of abstraction from the underlying physical IT infrastructure. All three service models are used when it comes to big data: often IaaS for simple access to “unlimited” computing and/or storage capabilities, PaaS to establish one’s

own linguistic or algorithmic paradigm for processing big data, and SaaS when it comes to simply using a service or a combination of services for big data business analytics.

[29] Cloud providers in this area typically base their processing power on large collections of commodity hardware, including conventional processors ("compute nodes") connected via Ethernet or inexpensive switches, which are arranged in clusters and which are replicated within as well as across data centers. Replication as a form of redundancy is the key to hardware reliability and fault-tolerant processing, and in just the same way data is protected against losses via replication. The result is either a distributed file system such as the Hadoop distributed file system (HDFS, see below) or a globally distributed database such as Google's Spanner [6]. Besides fault tolerance and availability, distribution can enhance parallel processing of the given data, in particular when computing tasks can be executed independently on distinct subsets of the data. In such a case, data is often partitioned over several clusters or even data centers; Fig. 4 illustrates the difference between partitioning and replication.

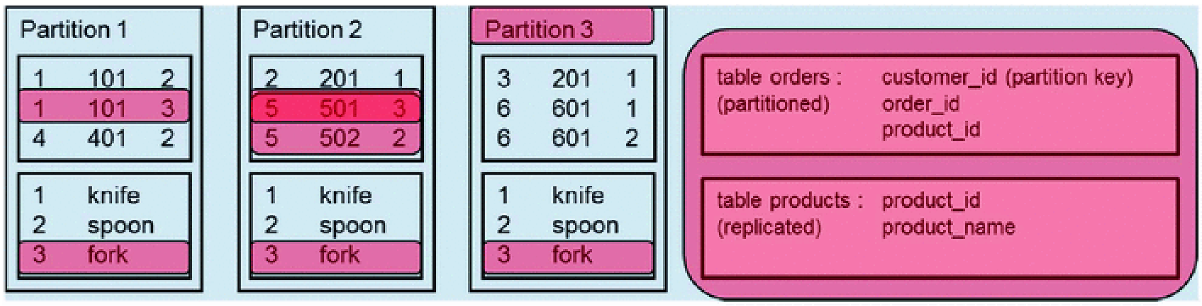

Fig 4: Partitioning vs. replication

In the example shown in Fig. 4, data from a relational database about customer orders is partitioned over three different sites in such a way that each site is assigned distinct customer numbers, while products data is replicated over the sites (i.e., identically copied). Queries and updates can now go to a particular partition or to multiple partitions at the same time. If an organization like the one shown is run by a cloud provider underneath a SaaS product, a user does not need to care about proper data handling.

[29] While replication is a measure to enhance data availability, since if one copy fails another might still be available, partitioning turns out to be the key to tackling many large data problems algorithmically. Partitioning essentially follows the "old" principle of divide and conquer, which has a long tradition in computer science and its algorithms. If data can be split into various independent partitions (as in the example in Fig. 4 above), processing of that data can exploit parallelism, for example by keeping multiple cores of a processor or multiple CPUs in a cluster busy at the same time. The results obtained by these cores or CPUs may need to be combined to form a final processing result. This is the basic idea of Google's map-reduce [7] (US Patent 7,650,331, granted in January 2010) which employs higher-order functions (Well known from the functional programming paradigm) for specifying distributed computations on massive amounts of data.

[29] Map-reduce are a combination of two functions, map and reduce, which work on key–value pairs. A map-reduce computation essentially works as shown in Fig. 5: input data is made available in a number of data chunks, which typically come from a distributed file system. These chunks are fed into map tasks executed by components called mappers. Mappers turn their given chunk into a sequence of key–value pairs; exactly how these key–value pairs are generated from the input data depends on the particular computing task and is determined by the code written by the user for the map function. Next, mapper intermediate outputs are collected by a master controller and grouped by their key values. The keys and their associated value groups are then given to reduce tasks in such a way that all key–value pairs with the same key end up at the same reducer component. Finally, reducers work on one key at a time, and combine all the values associated with that key in a task-dependent way again specified by the code written by the user for the reduce function.

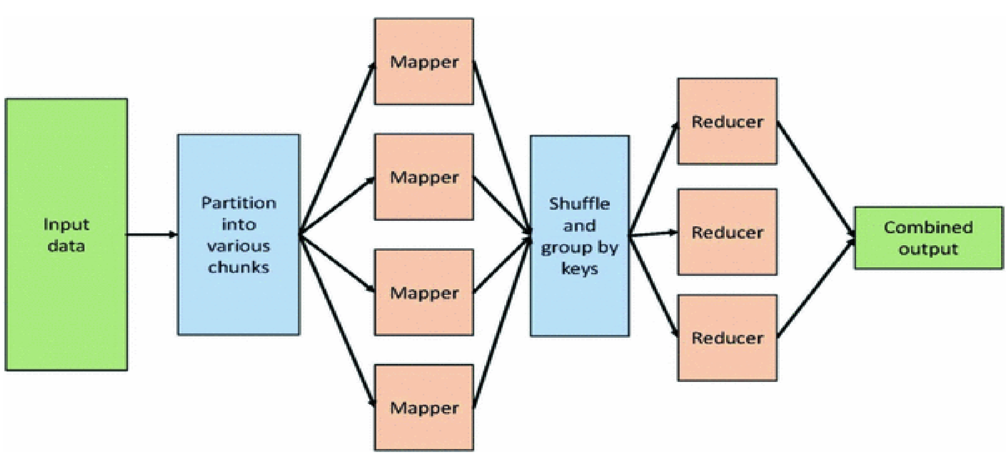

Fig 5: Principle of a map-reduce computation

4. How do we exploit big data?

We now look at the organizational dimension of big data and consider the situation where a company or institution wants to make use of it. What does it take to do so, and what needs to change if the company has previously set up a data warehouse for its data analytics purposes? In particular, we briefly look at strategy development and then present a modification of the "classical" data warehouse architecture that is intended to accommodate big data requirements [29].

As has been the case for many other IT adoption decisions that have arisen over the years, it makes sense to base a decision of whether to start a big data project or to adopt big data technology on well-grounded considerations. To this end, techniques such as a SWOT analysis can help, which may be able to reveal the strengths, weaknesses, opportunities, and threads of a particular technology or project. Another tool that could be used in decision making is context analysis, which looks at objectives, added values, and the general context and environment into which a project should fit. Both SWOT and context analysis are popular and have proven successful, for example, in business process modeling [23].

More comprehensive than specific analyses is the development of a strategy for big data, which may look like the one, shown in Fig. 8, [29].

Fig 8

The basic architecture of a data warehouse can also be recognized from the right half of Fig. 9, yet the figure also indicates how to extend a traditional data warehouse architecture for big data. Indeed, what is new in this figure is a wider selection of external data sources than typically considered and the extension by a map-reduce engine such as Hadoop on the left side. Various ways of communication need to be made available between these old and new building blocks, but in the end the setup might look as shown in the figure.

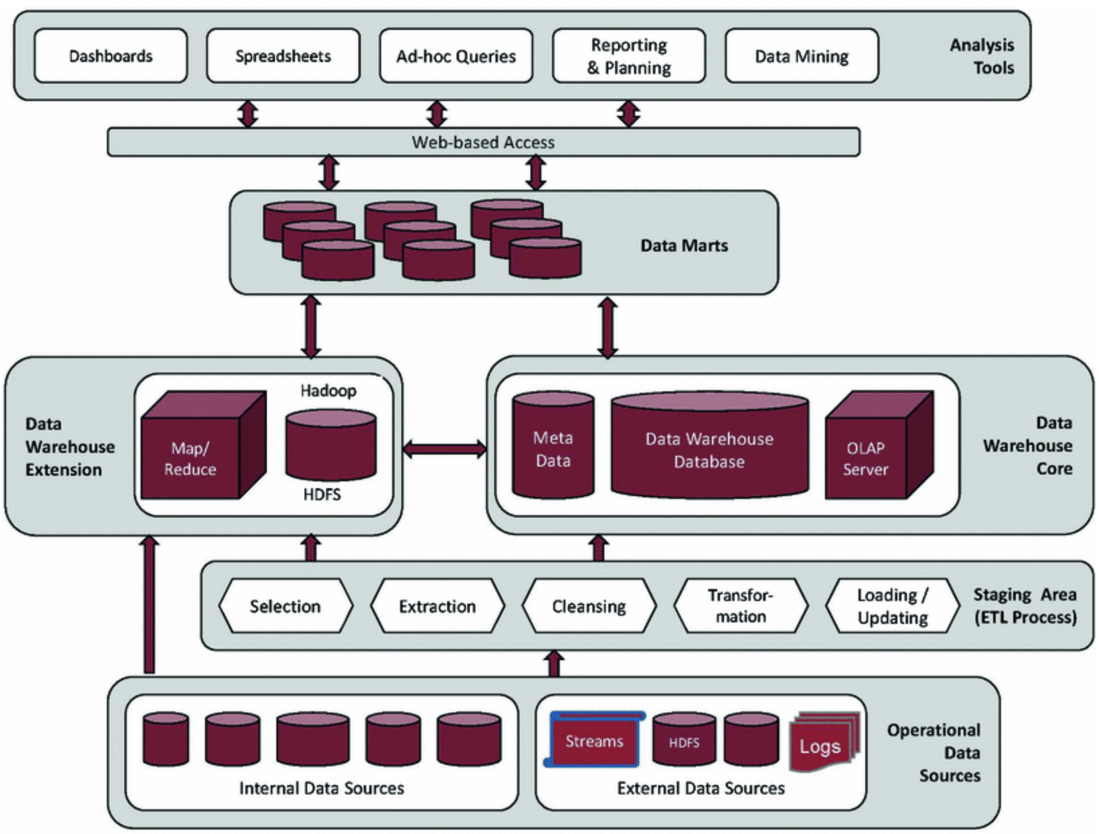

Fig 9: Data warehouse architecture enhanced for big data processing

5. Conclusions

[29] In this paper we have tried to survey various dimensions that are relevant to the field of big data that has emerged in recent years. Essentially, big data refers to the concept that data is nowadays available in an abundance that was never known before, that data-processing technology is capable of handling huge amounts of data efficiently, and that therefore there are large and primarily economic opportunities for exploiting this data. The notion of business intelligence that was "invented" in the context of (early) data mining as a circumscription of the fact that business can improve or enhance their "intelligence" regarding customers and revenues by analyzing and "massaging" their data to discover the unknown will now enter the next level. Indeed, a consequence of the fact that more and more data is made available in digital form not only allows businesses to gain new insights, but also

renders new discoveries possible in areas such as physics or health care which are not necessarily of primary type "business". So not only regarding business, big data can indeed be seen as the new intelligence enabler, since the broadness of data available today (not just its sheer size!) and the available technology enable us to perform analytics, to see connections, and to make predictions unthinkable only a short while ago.

[29] To conclude, we mention two developments that are foreseeable in the near future. The first is the fact that big data will have an impact on academic education. Indeed, a number of schools, so far primarily in the USA, have already launched programs for educating "data scientists". We expect this trend to continue, at the borderline of computer science, statistics, machine learning, and possibly other fields such as communication and social sciences or medicine.

Second, as has happened with other goods in the past, when data becomes a commodity, we will see the emergence of (virtual) marketplaces for data just as the past has seen the creation of marketplaces, say, for stock. The stock market is characterized by the fact that it not only sells shares in companies, but offers a variety of other products that may or may not be derived from basic stock. In a similar way, a data marketplace will offer raw data, say, on a certain topic, and will also offer a variety of ways in which this data can be processed prior to being sold. Different from the stock market, however, data marketplace may be open to anyone, i.e., users can act as sellers or buyers or both.

[29] Figure 10, which originally appeared in [15], shows the general schema of a data marketplace for integrating public Web data with other data sources. In analogy to a data warehouse architecture, the schema includes components for data extraction, transformation and loading, as well as Meta data repositories describing data and algorithms. In addition, the data marketplace offers interfaces for uploading data and methods for optimizing data, e.g., by employing operators with user-defined-functionality, as well as components for trading and billing the usage of these operators. In return, the provider of the user-defined function retrieves a monetary consumption (indicated by the euro symbol) from buyers. Moreover, in the case of large data volumes from the Web, the marketplace relies on a scalable infrastructure for processing and indexing data. A survey of the state of the-art in this field can be found in [24].

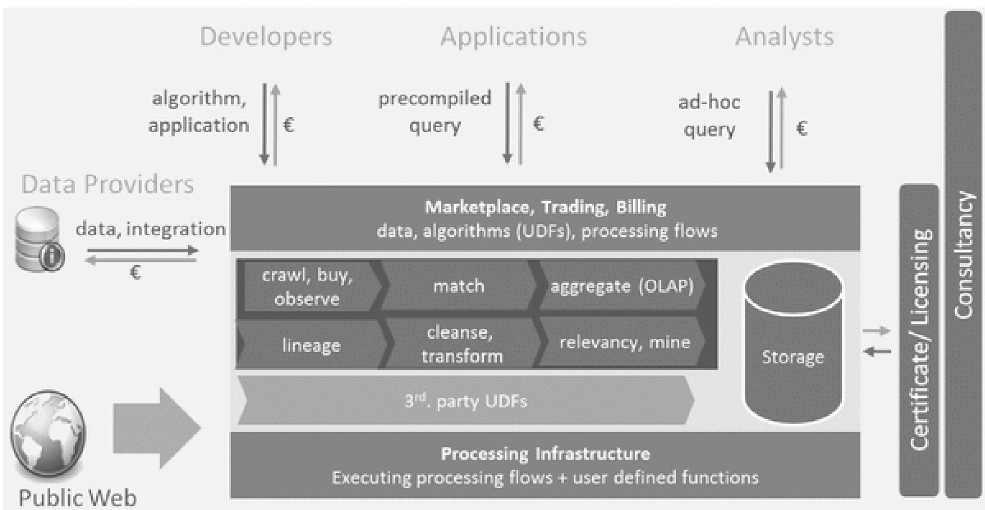

Fig 10: Concept of a data marketplace

6. Footnotes

1. http://www.intel.com/content/www/us/en/communications/internet-minute-infographic.html.
2. http://www.ibmbigdatahub.com/infographic/four-vs-big-data.
3. http://blogs.gartner.com/doug-laney/.
4. http://blogs.gartner.com/doug-laney/the-indy-500-big-race-bigger-data/.
5. http://www.quantumblack.com/formula-1-race-strategy-2/
6. http://www.directrelief.org/emergency/hurricane-sandy-relief-and-recovery/.
7. http://www.fitbit.com.
8. http://www.nike.com/cdp/fuelband/us/en_us/.
9. https://jawbone.com/up.
10. http://www.allthingsdistributed.com/2008/12/eventually_consistent.html.
11. http://hadoop.apache.org/.
12. http://hadoop.apache.org/docs/stable/hdfs_design.html.
13. http://wiki.apache.org/hadoop/PoweredBy.
14. https://www.eff.org/nsa-spying, http://www.theguardian.com/world/2013/sep/09/nsa-spying-brazil-oil-petrobras, http://www.bloomberg.com/news/2013-09-10/nsa-phone-records-spying-violated-court-rules-for-years.html.
15. http://io9.com/5877560/10-ways-big-data-is-creating-the-science-fiction-future.
16. http://www.techrepublic.com/blog/big-data-analytics/10-emerging-technologies-for-big-data/

6. References

1. Afrati F, Das Sarma A, Salihoglu S, Ullman JD. Vision paper: towards an understanding of the limits of map-reduce computation. CoRR abs/1204.1754, 2012.
2. Afrati F, Das Sarma A, Salihoglu S, Ullman JD. Upper and lower bounds on the cost of a map-reduce computation. PVLDB 2013; 6(4):277-288.
3. Agrawal D, Das S, El Abbadi A. Data management in the cloud: challenges and opportunities. Synth. Lect Data Manag. 2012; 4(6):1-138.

4. Battré D. Nephele/PACTs: a programming model and execution framework for web-scale analytical processing. Proc. 1st ACM Symp. Cloud. Comput. (SoCC) 2010, 119-130.
5. Chang F. Bigtable: a distributed storage system for structured data. ACM Trans. Comput. Syst. Cross Ref MATH 2008; 26(2):1-26.
6. Corbett JC. panner: Google's globally-distributed database. In: Proceedings of the 10th USENIX Symposium on Operating Systems Design and Implementation (OSDI), 2012.
7. Dean J, Ghemawat S. Map Reduce: simplified data processing on large clusters, proceedings of the 6th symposium on operating system design and implementation (OSDI) (2004) and communication. ACM Cross Ref 2008; 51(1):107-113.
8. Eich MH. Main memory database research directions. In: Proceedings of the International Workshop on Database Machines, 1989; 251-268.
9. Fedak G. Special issue of mapreduce and its applications. Concurr. Comput. Pract. Exp. 2013; 25(1).
10. Han J, Kamber M, Pei J. Data Mining: Concepts and Techniques, 3rd edn. Morgan Kaufmann Publishers, Burlington, 2011.
11. Inmon WH. Building the Data Warehouse, 4th edn. Wiley, New York, 2005.
12. Lewis M. Money ball: The Art of Winning an Unfair Game. Norton & Company, USA, 2004.
13. Loos P. In-memory databases in business information systems. Bus. Inf. Syst. Eng. Cross Ref 2011; 6:389-395.
14. Lynch N, Gilbert S. Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. ACM SIGACT News Cross Ref 2002; 33(2):51-59.
15. Muschalle A. Pricing approaches for data markets. In: Castellanos, M. (ed.) BIRTE 2012 (Proceedings of the 6th International Workshop on Business Intelligence for the Real Time Enterprise 2012, Istanbul), Springer LNBIP, New York, 2013, 129-144.
16. Pflanzl N. State-of-the-Art of Social Network Visualization. Master thesis, University of Münster, Department of Information Systems, 2012.
17. Plattner H, Zeier A. In-Memory Data Management-An Inflection Point for Enterprise Applications. Springer, Berlin, 2011.
18. Rajaraman A, Lescovec J, Ullman JD. Mining of Massive Datasets; downloadable from http://infolab.stanford.edu/ ullman/mmds. html, 2013.
19. Redmond E, Wilson JR. Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement. Pragmatic Programmers, Dallas, TX, USA, 2012.
20. Saecker S, Markl V. Big Data Analytics on Modern Hardware Architectures: A Technology Survey; European Business Intelligence Summer School (eBISS), Spinger LNBPI, New York, 2012, 125-149.
21. Sauer C, Härder T. Compilation of query languages into mapreduce. Datenbank-Spektrum Cross Ref 2013, 13(1):5-15
22. Shim K. Map Reduce algorithms for big data analysis. Spinger LNCS, 2013; 78(13):44-48.
23. Schönthaler F, Vossen G, Oberweis A, Karle T. Business Processes for Business Communities. Springer, Berlin Cross Ref 2012.
24. Schomm S, Stahl F, Vossen G. Marketplaces for data: an initial survey. ACM SIGMOD Rec. Cross Ref 2013; 42(1):15-26.
25. Shute J. F1: A Distributed SQL Database That Scales. Proc. VLDB Endowment, 6(11):1068-1079.
26. Vossen G, Hagemann St. Unleashing Web 2.0-From Concepts to Creativity. Morgan Kaufmann Publishers, Burlington, 2007.
27. Weikum G, Vossen G. Transactional Information Systems—Theory, Algorithms, and the Practice of Concurrency Control and Recovery. Morgan Kaufmann Publishers, San Francisco, 2002.
28. White T. Hadoop: The Definitive Guide, 3rd edn. O'Reilly Media, Sebastopol, 2012.
29. Gottfried Vossen: University of Münster, European Research Center for Information Systems (ERCIS)Department of Management Systems, The University of Waikato Management School, 2014.