

COMPUTER SCIENCE

Big data: the driver for innovation in databases

Bin Cui^{1,*}, Hong Mei¹ and Beng Chin Ooi²

INTRODUCTION

Advances in the technology frontier have resulted in major disruptions and transformations in the enterprise-wide information technology infrastructures. For the past three decades, classical database management systems have maintained a feverish pace in realizing significant efficiencies in dealing with the vast amount of information that needs to be maintained to model the operational characteristics of large-scale enterprises. Database research and development advances have primarily been focused in the areas of advanced data models, declarative query languages, high throughput transaction processing and database reliability, etc. In the intervening years especially in the 1990s, data warehousing and data analysis emerged as a major research and technology frontier. In particular, it was realized that transactional information at the enterprise level can be collated and analyzed to enable data-centric decision making.

Database management systems gained significant prominence in the era of Web-based services and E-commerce deployments. What is now considered classical, a typical web-service architecture encompasses database management systems (DBMSs) as the core tier to provision services and applications when coupled with two critical IT components referred to as the *Web servers* and *Application servers*. In the context of Web- and Internet-enabled database services, one of the major research challenges that

emerged was to deal with potentially unlimited number of users accessing the service over the Internet. It is now widely acknowledged that although the Web and Application server tiers in the three-tiered Web-based architecture can be scaled easily to handle a large number of users, the database tier becomes a scalability bottleneck since it cannot be easily scaled by deploying additional hardware or machines. Companies such as Google and Amazon have in fact abandoned the traditional DBMS technology in favor of proprietary data stores referred to as key-value stores [1,2].

While enterprises are struggling with the problem of poor database scalability, a new challenge has emerged that has further crippled the capability of modern IT infrastructures. This challenge has been labeled as the ‘big data’ problem. In principle, while earlier DBMSs focused on modeling operational characteristics of enterprises, big data systems are now expected to model vast amounts of heterogeneous and complex data. Classical approaches of data warehousing and data analysis are no longer viable to deal with both the scale of data and the sophisticated analysis that need to be conducted often in real time (e.g., online fraud detection). None of the commercial DBMS and Data Warehousing technologies provide an adequate solution in this regard which is evident from the efforts led by companies such as Facebook, Google and Baidu to build proprietary solutions. Clearly, scalable data management and complex data analytics

in the context of big data has emerged as a new research frontier in the foreseeable future. As an orthogonal challenge in the context of ‘big data’, since enterprises maintain vast amounts of sensitive user interaction data for its clients, it is imperative that adequate mechanisms are provided to ensure security and privacy of user data. Recently, an emerging technology referred to as cloud computing is being increasingly used in proprietary environments to manage large data centers. In particular, companies such as Amazon, Google, Yahoo and Microsoft have each developed their respective proprietary versions of cloud computing technology and have enjoyed unprecedented success. However, although cloud computing has demonstrated its superiority in the context of Web-based applications (e.g., Social Networking, Web Search, E-commerce), it is not mature enough to facilitate the enterprise processes that are based on DBMSs.

APPLICATIONS AS DEMAND DRIVERS

Due to the wide adoption of technologies, data from different sources and in different format are being collected at unprecedented scale. This gives rise to the so-called 3V characteristics of the big data: volume, velocity and variety. Although the massive data pose many challenges and invalidate earlier designs, they provide many great opportunities, and most of all, instead of making decisions based on small sets of data or calibration, decisions can now be made based on the data itself. Below, we briefly examine some of the big data applications.

Social networking

Online social networks such as Facebook, LinkedIn and Twitter provide new platforms for social interactions at a

worldwide scale. These new sources of data also allow for new kinds of data-analysis applications, e.g. understanding social behaviors at an aggregate scale. The enormous size of social networks allows unprecedented and new forms of analysis and brings new processing challenges. First, in many online social networks, data analysis makes use of the social graph. This requires new data-analysis algorithms that must be able to cope with massive graphs with $O(10^9)$ nodes. Existing graph algorithms are not designed to deal with graphs at such scale and are mainly tailored for graphs residing in main memory. Second, the scale of interactions in the social network and the difficulty of clustering the data make it especially challenging to be able to answer queries about the interactions in a timely manner.

Enterprise data management

Currently, in many enterprises, users independently source, model, manage and store data to support their own area of responsibilities and functionalities. This is mainly due to two reasons. First, users have better domain knowledge and are thus able to customize the database to suit their own needs. Second, such a decentralized approach allows users to break the data requirement of the enterprise into smaller subsets and address these requirements by using smaller, independent databases. However, such an uncoordinated data management approach can result in data conflicts and quality inconsistencies within the enterprise, making it difficult for users to trust the data when it is used for operations and reporting at a higher level within the enterprise.

Scientific applications

Many parts of science now are about experiments which create a large amount of data and the subsequent analysis of the data. This is the challenge posed by Gray [3], how to support the data intensive scientific discovery paradigm. The scale of the data is significantly larger than that in most enterprise applications, e.g. science experiments can generate terabytes of data daily. This means that query pro-

cessing techniques that rely on indexing may not be feasible simply because there is insufficient time to build indexes on the data. Clearly, controls over the execution time are needed but there has been little work in this direction. The result of a query could range from the empty set to a significant fraction of the database. However, although a very large result is a complete and correct answer, it might not be a very useful answer and may also take too long to compute. This suggests that rather than large results, a summary of the data and applying sampling may be more appropriate.

Mobile computing

Smart phones and other connected mobile and embedded devices are becoming increasingly prevalent. This begets new conveniences for individuals to manage data, perform complex computations, as well as obtain real-time information that will aid their activities, thereby improving personal productivity. The key challenge will be how the mobile and cloud platform can be integrated holistically as a single computing experience. Another new dimension is that a new challenge arises which is the use of crowdsourcing and real-time data mining to further enhance the quality of the real-time, location-sensitive information that is available to the authorities, providers and end-users.

The sheer size of the data and rate that it is created provides many challenges in data management and processing. As has been mentioned, the existing database technologies are not able to handle the challenges presented by the big data. First, relational databases use well-defined schemas and require application data to fit into the relational paradigm of rows and columns. Unfortunately, a lot of data are unstructured. Data may be collected from various data sources, such as search logs, click streams and crawled pages. They may have various formats, and programmers must preprocess them and load into the database before performing any analysis. Second, it is hard for a relational database system to scale. Though expensive high-end servers are used to run parallel database

systems, they are not sufficient for handling the volumes of data today. Once a database's upper capacity is exceeded, database engineers must redistribute the data across multiple databases to break them up. Third, it is difficult for relational database systems to effectively deal with complicated queries. The increasing demand for data exploration and knowledge discovery needs more complicated *ad hoc* queries in data analysis, which makes it difficult to build indexes and views in the databases based on requirements and assumptions, even for those most agile systems. Below, we provide two immediate challenges and opportunities.

Scalable and elastic data management

Scalable and elastic data management has been a great challenge to the database research community for more than 20 years, and different distributed database systems were proposed to deal with large datasets. However, the scalability of these systems is still limited due to some common problems in distributed environment, such as synchronization costs and node failures. Therefore, most of the existing Cloud systems, such as BigTable and Cassandra, exploit different solutions to improve the system scalability. The techniques adopted by such systems include: (a) simple data model, (b) separated metadata and application data and (c) relaxed consistency requirements. However, they introduce different sets of problems, such as lack of support for real-time update, consistency and high-latency of selective retrieval. It is therefore important to design a system architecture that can dynamically support the elastic requirement of the applications for the storage and processing of multi-tenancy data over a distributed cluster of commodity compute nodes. Both horizontal and vertical partitioning strategies could be employed to partition and distribute the data that serve to achieve a high performance of query processing and updates. To facilitate an efficient location of selective data without having to scan the whole database over all the compute nodes, efficient and light-weight indexing

should be designed. Based on the access methods and data distribution, efficient query processing strategies could then be designed.

Scalable data analytics

The widespread adoption of computing technologies has resulted in a large amount of data in a wide variety of forms. Besides the traditional business data that are structured, social media applications generate graph data; location-based services and urban sensing applications produce spatial-temporal data; multimedia applications generate images and videos that are typically represented as high-dimensional features. There is therefore a need to study how best to represent and process these complex data structures efficiently, and to propose new paradigms of data analytics over different data types. Furthermore, user interaction and visualization are important for exploring large quantity of data of complex forms. Developing methods for interactive visual analytics is critical for successful scalable data analysis. To improve system usability, there is a need for a declarative programming model that is communication-centric (optimize inter-process communication) and data-centric (optimize data processing and support for large-scale concurrency).

Furthermore, a white paper [4] was recently created through a distributed conversation among more than 20 prominent researchers, which illustrated the challenges and opportunities with big data, and discussed the research agenda in this field as well. The analysis of big data involves multiple distinct phases as shown in Fig. 1. Major steps in big data analysis are shown in the flow on the top of Fig. 1, which include acquisition, extraction, integration, analysis and interpretation. Below it, the challenges introduced by big data are shown. The authors also discuss both what has already been done and what challenges remain to exploit big data.

EMERGING TECHNOLOGIES

DBMSs have become a ubiquitous operational platform in managing huge amount

of business data. DBMSs have evolved over the last four decades and are now functionally rich. However, as the arrival of the big data era, these database systems showed up the deficiencies in handling big data.

Recently, a new distributed data-processing framework called MapReduce was proposed [5], whose fundamental idea is to simplify the parallel processing using a distributed computing platform that offers only two interfaces: map and reduce. Programmers implement their own map and reduce functions, while the system is responsible for scheduling and synchronizing the map and reduce tasks. By defining the ‘Map and Reduce’ functions, MapReduce applications are able to deal with much more complicated tasks than SQL queries. Different from relational databases, MapReduce uses a much simpler data model and views data as key-value pairs. As a result, the programmers are free to structure their data and data parsing and loading before analysis can be conducted. MapReduce applications can handle data stored either in an unstructured file system as well as in a structured database. Meanwhile, the key-value data model used in MapReduce leads to the emergence and population of many key-value stores such as Bigtable [1]. As the key-value abstraction naturally allows horizontal partitioning, key-value stores can provide

better scalability and availability than relational databases. MapReduce is being used increasingly in applications such as data mining, data analytics and scientific computation. Its wide adoption and success lie in its distinguishing features, including flexibility, scalability, efficiency and fault tolerance.

Along with the praises brought by its simplicity and flexibility, MapReduce is also criticized for its reduced functionality. Already much effort has been devoted to address these problems; yet an active area of research remains to be explored, such as SQL-like declarative language enhancement for the ease of use, DBMS operator implementation for functionality enhancement, performance improvement for iterative computation (for detailed survey, please refer to [6]).

Generally speaking, MapReduce systems are good at complex analytics and extract-transform-load tasks at large scale, while parallel databases perform better in efficient querying of large data sets. Some researchers attempt to incorporate the best characteristics of both parallel databases and MapReduce systems. For example, by using the MapReduce framework as its middle layer and using distributed PostgreSQL as its bottom layer, HadoopDB [7] benefits from both the scalability of MapReduce and the efficiency of parallel databases. There also exist many other

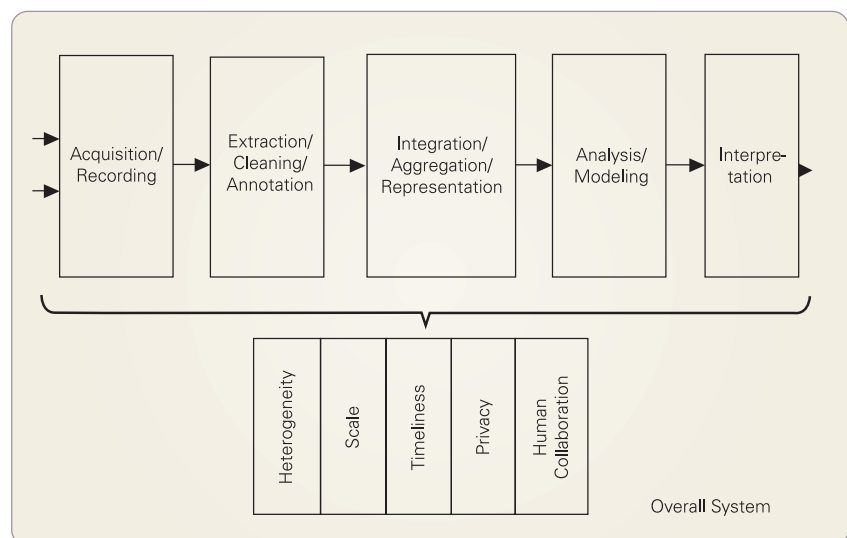


Figure 1. The big data analysis pipeline [4].

distributed data processing systems that go beyond the MapReduce framework. These systems have been designed to address various problems not well handled by MapReduce, which are listed as follows.

Interactive analysis

MapReduce is optimized for batch processing but not fast interactive analysis. Google tried to overcome the problem by building a completely different system named Dremel to support fast interactive analysis. Dremel [8] splits data into different fields and stores these fields in different files before execution, so that the time for data parsing and loading at runtime is reduced. Further, Dremel uses multi-level serving trees to execute queries so that intermediate aggregation can reduce the amount of data that needs to flow in the system. Popular Dremel-like systems include Apache's Drill, Cloudera's Impala and Metamarkets' Druid.

Graph analysis

Directly running graph analysis task in MapReduce will lead to massive data movement since it does not exploit the underlying graph structure. Recently, some graph computation frameworks have emerged to solve this challenge. Google introduced a vertex-centric graph computation system called Pregel [9], which stores the underlying graph in memory to speed up random access, and executes graph computation using a bulk synchronous parallel (BSP) model. There are several other implementations for graph analysis, including Apache Hama, GoldenOrb, Giraph, Phoebus, GPS and GraphLab.

Real-time analysis or stream processing

The MapReduce framework works well for offline-batched analytics, but it was not designed for real-time decision making. Recently, systems such as S4 (simple scalable streaming system) [10]

were proposed to handle real-time data processing. S4 is a distributed stream processing engine that allows programmers to develop applications for continuous stream processing. It combines the Actors model and the MapReduce model, and hence applications can be massively concurrent with a simple programming interface.

Generic data processing

Efforts have been made in developing alternative parallel processing platforms that have MapReduce flavor, but are more general. One example of this line of work is epiC [11]. EpiC was designed to handle variety of data (e.g., structured and unstructured), variety of storage (e.g., database and file systems) and variety of processing (e.g., SQL and proprietary APIs). The important characteristic of epiC, from a MapReduce or data management perspective, is that it simultaneously supports both data intensive OLAP and OLTP.

CONCLUSIONS

With the advancement and wide adoption of technologies, data have been created at an unprecedented rate. Coupled with the problems of size and heterogeneity, we have the 3V problems to handle and value to create out of the data. The value of data is unleashed when it can be integrated and made sense with other data. The big data presents us the challenges and opportunities in designing new processing platforms for integrating, managing and processing the massive data, and for providing contextual analysis by working with the domain and subject experts, and the visualization of the massive data. The potential research topics in this field lie in all phases of data management pipeline that includes data acquisition, data integration, data modeling, query processing, data analysis, etc. Besides, the big data also brings great challenges and opportunities to other computer science disciplines such as system architecture, storage system, system

software and software engineering, which are beyond the scope of the paper. Overall, data are indeed the root of the problems, and they drive the development of many new technologies and decision making.

Bin Cui, Hong Mei and Beng Chin Ooi

1. School of EECS and Key Laboratory of High Confidence Software Technologies, Peking University, China

2. School of Computing, National University of Singapore, Singapore

*Corresponding author.

E-mail: bin.cui@pku.edu.cn

REFERENCES

- Chang, F, Dean, J and Ghemawat, S *et al.* *Proceedings of the Seventh USENIX Symposium on Operating Systems Design and Implementation*. 2006; pp. 205–18.
- DeCandia, G, Hastorun, D and Jampani, M *et al.* *Proceedings of Twenty-first ACM SIGOPS symposium on Operating systems principles*. 2007; pp. 205–20.
- Hey, T, Tansley, S and Tolle, K, eds. *The Fourth Paradigm: Data-Intensive Scientific Discovery*, Microsoft Research, 2009.
- Jagadish, HV. *Challenges and opportunities with big data*, 2012. <http://www.cra.org/ccc/files/docs/init/bigdatawhitepaper.pdf>.
- Dean, J and Ghemawat, S. *Proceedings of the Fifth USENIX Symposium on Operating Systems Design and Implementation*. 2004; pp. 137–50.
- Li, F, Ooi, BC and Ozsu, T *et al.* *ACM Comput Surv* 2013; **46**.
- Abouzeid, A, Bajda-Pawlikowski, K and Abadi, D *et al.* *Proceedings of the VLDB Endowment*. 2009; **2**(1): 922–33.
- Hall, A, Bachmann, O and Büssow, R *et al.* *Proceedings of the VLDB Endowment*. 2012; **5**(11): 1436–46.
- Malewicz, G, Austern, M and Bik, A *et al.* *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*. 2010; pp. 135–46.
- Neumeyer, L, Robbins, B and Nair, A *et al.* *Proceedings of the 2010 IEEE International Conference on Data Mining Workshops*. 2010; pp. 170–77.
- Chen, C, Chen, G and Jiang, D *et al.* *Proceedings of the 11th International Conference on Web Information Systems Engineering*. 2010; pp. 1–19.