CHAPTER

BIG DATA: A PRACTITIONERS PERSPECTIVE

10

Darshan Lopes, Kevin Palmer, Fiona O'Sullivan
Capgemini UK, UK

Big data has only recently come into common parlance and been accepted as a recognizable IT discipline over the past few years.

It could be argued that it represents the next major paradigmatic shift in Information Systems thinking. Big data solutions represent a significant challenge for some organizations. There are a huge variety of software products, deployment patterns and solution options that need to be considered to ensure a successful outcome for an organization trying to implement a big data solution.

With that in mind, the chapter will focus on four key areas associated with big data that require consideration from a practical and implementation perspective:

(i) Big Data is a new Paradigm – Differences with Traditional Data Warehouse, Pitfalls and Considerations.

(ii) Product considerations for Big Data – Use of Open Source products for Big Data, Pitfalls and Considerations.

(iii) Use of Cloud for hosting Big Data – Why use Cloud, Pitfalls and Considerations

(iv) Big Data Implementation – Architecture definition, processing framework and migration patterns from Data Warehouse to Big Data

10.1 BIG DATA IS A NEW PARADIGM – DIFFERENCES WITH TRADITIONAL DATA WAREHOUSE, PITFALLS AND CONSIDERATION

Wikipedia defines big data as:

Big data is a broad term for data sets so large or complex that traditional data processing applications are inadequate. [1]

It is generally described by the following characteristics known as the 3 V's – Volume (quantity of data), Velocity (speed at which it is generated), Variety (different types of data).

10.1.1 DIFFERENCES WITH TRADITIONAL DATA WAREHOUSE

Big data is very different from processing within a traditional data warehouse. These differences are highlighted in Table 10.1.

Software Architecture for Big Data and the Cloud. DOI: 10.1016/B978-0-12-805467-3.00010-7

Table 10.1 Differences between traditional data warehouse and big data

	Traditional data warehouse	Big data
Data format	Structured	Combination of structured and unstructured
Data types	Fixed format	Fixed format, audio, video, PDF, XML, JSON, Binary files + flexible formats
Data size	Typically terabytes	Petabytes and beyond
Storage	Relational data stores	Distributed file system
Operations	Known operations using SQL	Flexible queries using SQL + NoSQL
Repositories	Often fragmented multiple warehouses	Single repository using the concept of a data lake which is constantly gathering and adding data
Schema	Static	Unstructured data, nontransactional, dynamic schemas Metadata-driven design
Processing scalability	Scales vertically	Massively Parallel Processing capability

Whilst traditional data warehouses provide insight into the past to answer “What has happened?”, whereas big data, using advanced analytics and variety of data, tries to understand the future and to answer not only “What is happening now?” but also “What could happen?”.

Big data allows the data consumers to adapt their knowledge of traditional data combined with other data sources using the following capabilities:

- Large scale analytical processing to find previously unknown trends and learnings;
- The capability for flexible queries, allowing unrestricted exploration and experimentation of existing and new data sets;
- Use of SQL (for structured and standardized) or NoSQL; or even a combination if appropriate;
- Use of commodity hardware and open source software.

10.1.2 PITFALLS

10.1.2.1 Insufficient volume of data

It is important not to think of big data just in the context of having a large data volume to manage. An organization may not have the massive volumes of raw data as, say, Twitter or LinkedIn, but it may still have large volumes of business transaction data because of the business problems it is trying to solve. Also, the organization that may not produce large amounts of data itself could benefit from external “ancillary” information to enrich its own understanding of customers, competitors, and industry-wide trends.

10.1.2.2 Not having a business challenge

An organization would benefit from focusing on its specific challenge, be it cost reduction, increasing efficiency, or uncovering new revenue streams. Once the challenge has been established, the question becomes how big data technologies can be utilized. What data sets do I need? What technologies can help? For example, a furniture company wishes to enter Far Eastern markets. By using a combination of social networks, sentiment analysis, and translation capabilities, it can find a much cheaper and quicker alternative to traditional market research enabling its marketing campaign to be quicker online, have increased flexibility and reduced financial outlay. A general principle to keep in mind is to begin with your business needs and then perform a market scan to assess available tools and data appropriate to this need. This is no different to traditional systems' development, where there is no magic shortcut.

10.1.2.3 Ignoring the data quality

The importance of data quality is directly proportionate to the type of analysis required and must be understood by the data owner. For example, data used for statutory reporting must be extremely accurate; however, data used for marketing segmentation provides a more general view and therefore would not require the same level of accuracy.

10.1.2.4 Big data can predict the future

Data alone cannot predict the future. A combination of understood data and well designed, considered analytical models can make reasonable predictions alongside defined assumptions. It is entirely within the data owner's responsibility to understand if and for how long these assumptions will be valid. In some cases very short term predictions might be more than sufficient, and new analytic models can be created continuously as requirements develop.

10.1.3 CONSIDERATIONS

- Focus on the business challenge first and then figure out the technology required to support this challenge.
- Look out for "ancillary" information from sources outside of the boundaries of your organization.
- Quality of data becomes important where the use case requires accurate outcomes from the analytics process.

10.2 PRODUCT CONSIDERATIONS FOR BIG DATA – USE OF OPEN SOURCE PRODUCTS FOR BIG DATA, PITFALLS AND CONSIDERATIONS

The number of products in the area of big data has been growing rapidly to meet the volume, speed, and complexity requirements.

10.2.1 THE USE OF OPEN SOURCE PRODUCT FOR BIG DATA

Whilst many vendor-driven products have evolved to handle analytics, it is important to consider the impact of the open source projects on the big data and analytics solution.

There are three key reasons for looking at the open source projects for the big data:

- Open source projects have been driving the innovation to meet the big data challenges by making the paradigm shift in processing of big data, i.e., taking processing to data, distributed file systems for data storage, and use of commodity or cloud hosting.
- There has been an explosion of new open source projects to meet analytics requirements.

- Cost effective compared to vendor products.

The best-known open source project related to big data processing is Apache Hadoop based on the technical paper written by Google [2]. Apache Hadoop has now spawned a number of related open source initiatives including Spark, HBase, Hive, Pig, and Avro. These Apache open source projects are driving the innovation within the big data world, processing at a faster rate than before, and producing a variety of feature rich platforms. The Apache Hadoop architecture and its ecosystem of related open source projects have become the industry standard for handling the current big data wave and the next wave, the Internet of Things.

10.2.2 PITFALLS

10.2.2.1 Not focusing on business needs and falling to the latest hype

Choosing an open source project based on hype with no business outcome in mind could end up as a costly venture for an organization. The scenario could be that open source project is not ready for enterprise scale production environment or the product does not add any value, only integration complexity. It is therefore important to understand the business problem the open source product is going solve. An organization whilst selecting the open source product needs to ask these questions: How ubiquitous is the software usage? Is it just short-term hype? Is software being used by multiple industries to solve their business problems? What is the maturity of the project? Has it gone beyond its 0.1 version and mature into 1.x or 2.x versions? Does it provide a stable version of bug-free software?

10.2.2.2 Not focusing on operational & nonfunctional requirements

It's not just the functional features that are important, operational and nonfunctional requirement assessment is crucial to successful big data delivery. Does it enable quick installation and deployment? Open source software generally is not written with ease of use in mind in terms of deployment and installation. Does it cover the required security features? Certain security features may not be available in the open source version of the software. For example, MySQL Community Edition does not provide database encryption. What is the upgrade path when new version is released? How are version upgrades managed? Does it provide backward compatibility?

10.2.2.3 Lack of sufficient document and or community base support

An organization should ask these questions: Who can provide support when it fails? What is the SLA for the fix to a bug? Are support contracts available? Is there sufficient documentation available in the open source community for this software product? How large is the community of contributors?

10.2.2.4 Not planning for separate environment to prove version compatibility

Generally, organizations tend to have development, test, and production environments. This is good if the products are relatively stable and unlikely to require frequent product upgrades. But given the rate and space of innovation in the big data products, it is more than likely that new versions with richer features will need to be installed on a frequent basis. To cater for this, organizations need to have a provision for a separate environment to prove any version compatibility proving before deploying the newer version to the development, test, and production environments.

10.2.3 CONSIDERATIONS

- There is no magic here, and good, well-founded technology selection principles still apply.
- Use open source version for quick proving but move to supported versions for product (e.g., Cloudera or Hortonworks System distribution of Hadoop) once selected.
- Conduct a capabilities gap analysis to understand the current state of products involved and gap in capability as a result of the business requirement.
- Maintain a product roadmap of the products involved in the big data solution, including features, current version and future upgrades, and any future compatibility issues.
- Maintain a separate environment to test new versions of the products and compatibility.

10.3 USE OF CLOUD FOR HOSTING BIG DATA – WHY TO USE CLOUD, PITFALLS AND CONSIDERATION

Cloud computing provides a shared multitenancy pool of processing resource to computers and other devices as the demand is required. Cloud computing can be rapidly provisioned and released to with minimal management effort and potentially reduced cost of ownership [3].

10.3.1 WHY TO USE CLOUD?

Adding “Complexity” alongside “Volume”, “Velocity”, and “Variety” of data, it is easy to follow the principle that to process large volumes of data requires high volume of computer processing power. There are cases where, due to the types of data being queried, the time to execute the query may be significant, and the enterprise has the choice to either accept the time to execute the query, or temporarily increase the processing capacity to reduce the time taken to execute. The cloud model enables such an increase in the processing capacity.

Here are some of the reasons why the cloud is important to big data:

Scalability – The cloud platform is highly scalable and provides the computing power required for the big data. This provides a fast capability to scale up and down as the business demands.

Pay as you go – The cloud model allows for pay-as-you-go model, which means organization only pays for the amount of resources it uses.

Low upfront cost – As a result of the data center and infrastructure already in place and the pay-as-you-go model, there is a low upfront cost, thus reducing operating costs and any high investment failure risk.

Self Service – Organizations are able to acquire the resources as needed via portal interface from the cloud provider.

It makes perfect sense to use the combination of the open source product and cloud model for the big data projects to deliver a solution, which has both cutting edge innovation and can be delivered efficiently with reduced time to market and at lower cost.

10.3.2 PITFALLS

10.3.2.1 Not knowing where the data will be stored

The data centers for the cloud will be spread across different countries and even continents. Organizations need to confirm where their data will end up, as regulatory laws may prevent data from being stored in a different country/continent.

10.3.2.2 Not understanding the SLA with the cloud provider

There should be an explicit service level agreement signed with the cloud provider describing performance, backup and recovery, availability and support.

10.3.2.3 Not understanding how to transfer data to cloud

The organization needs to ensure that the cloud provider has provided data management capability to import/export the data in a secure way.

10.3.2.4 Not knowing the processing profile required as cloud can scale

Whilst the cloud model is pay-as-you-go, it is easy to fall into a trap of paying for computing resources that the enterprise doesn't require all the time, based on lack of knowledge of the required processing resources, processing profile, and especially the design and implementation of poorly performing applications and queries. Designing with performance in mind is far easier (and in the majority of times far cheaper) than trying to add it in or apply it at a later stage. The implementation of a significant number application systems are delayed due to poor performance – it does not meet client expectations – which could have been avoided by taking the approach "design for performance."

10.3.3 CONSIDERATION

- Understand the usage and performance profile of the services being run in the cloud; understand the peaks and troughs of the processing requirements.
- When designing and building applications, performance requirements should be established and considered as early as possible in the development lifecycle.

10.4 BIG DATA IMPLEMENTATION – ARCHITECTURE DEFINITION, PROCESSING FRAMEWORK AND MIGRATION PATTERN FROM DATA WAREHOUSE TO BIG DATA

To understand what is required to implement a big data solution, it is important to understand:

- What is the typical architecture that underpins a big data solution?
- What is the processing framework involved in the big data solution?
- What are the patterns for transitioning from a data warehouse solution to a big data solution?

The diagram (Fig. 10.1) shows a typical big data architecture based on the Apache Hadoop framework.

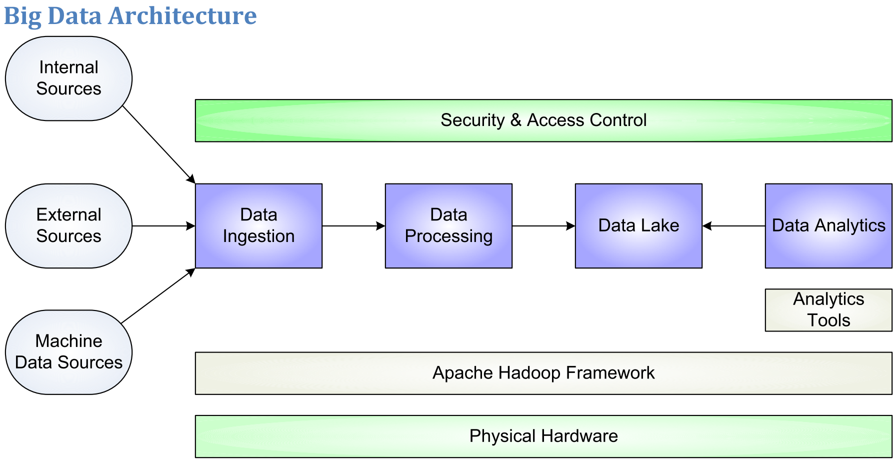

FIGURE 10.1

Big data architecture based on hadoop framework

Data Sources

The data can come from multiple sources:

- Internal – generated internally by the systems
- External – these could be social media generated or commercial packages of data or data from business partners
- Machine Data – data generated by various devices/sensors

Data Ingestion

Data Ingestion is a process of acquiring data from the sources and storing it for further processing.

Data Processing

This transforms the data to the required format or structure to make it effective for big data analytics. This is an optional step.

Data Lake

Data lake is a single repository to store all the data across the organization using a distributed file system.

Data Analytics

This is the part where analytic processing would be applied on the data stored in the data lake.

Apache Hadoop Framework

Apache Hadoop framework is one of the significant frameworks which is fast becoming a de-facto standard that provides the foundation to big data processing. It consists of the following: (a) core modules (HDFS, Hadoop YARN, Hadoop MapReduce) and (b) other Hadoop ecosystem components (Apache Hive, Apache Pig, Apache HBase, Apache Spark).

Physical Hardware

This could be a commodity hardware or cloud infrastructure such as a service platform.

Analytics Tools

These are the analytics products which enable data analytics by providing prepackaged functions or a development framework. Examples include Revolution R, MapR, Pentaho, Tableau, Platfora.

Security & Access Control

What data security should be applied? These could be encryption or tokenization or access control.

Organizations need to think about how to secure the data in-flight or data at rest in multitenancy shared data lake across multiple teams within the business. In addition, they will have to consider what type of analytic tool stack they want. Is a single analytics tool sufficient or is there a need for a stack of analytic tools to meet the multifaceted requirement of big data analytics? Also one has to consider the approach for introducing the new analytics tool in a way that minimizes risks and also reduces time to market.

The diagram in Fig. 10.2 shows the key attributes involved in big data processing. Understanding these attributes will help in the decision making for what big data technologies and design pattern are required to handle the business requirement.

Data Ingestion Type

Is the data being ingested in a batch process or in real time?

Data Ingestion Pattern

How is the data being ingested? Does it involve transferring files, using messages, connecting to RDBMS or via machine events?

Data Ingestion Frequency

How frequently is the data being ingested?

Data Format

What data formats are involved? These could be:

- Structured – predefined or dynamically created schema due to consistent structure, e.g., XML, Delimited files, JSON object.
- Unstructured – when it is not easy to define a schema, e.g., PDF, Audio files, Video files, picture, social media discussions.

Data Analytics Type

What type of analytics is required on the ingested data? Do we require text, visual, sentimental, or predictive analytics?

Data Types

What type of data is involved? These could be:

- Metadata – this is the contextual information of the ingested data.
- Full Record – this is the master data with full details.
- Full Cutover – full copy of the data.
- Delta – only changed attributes or records.
- Event – event generated by devices/sensors.

Data Analytics Type

What type of analytics is required on the ingested data?

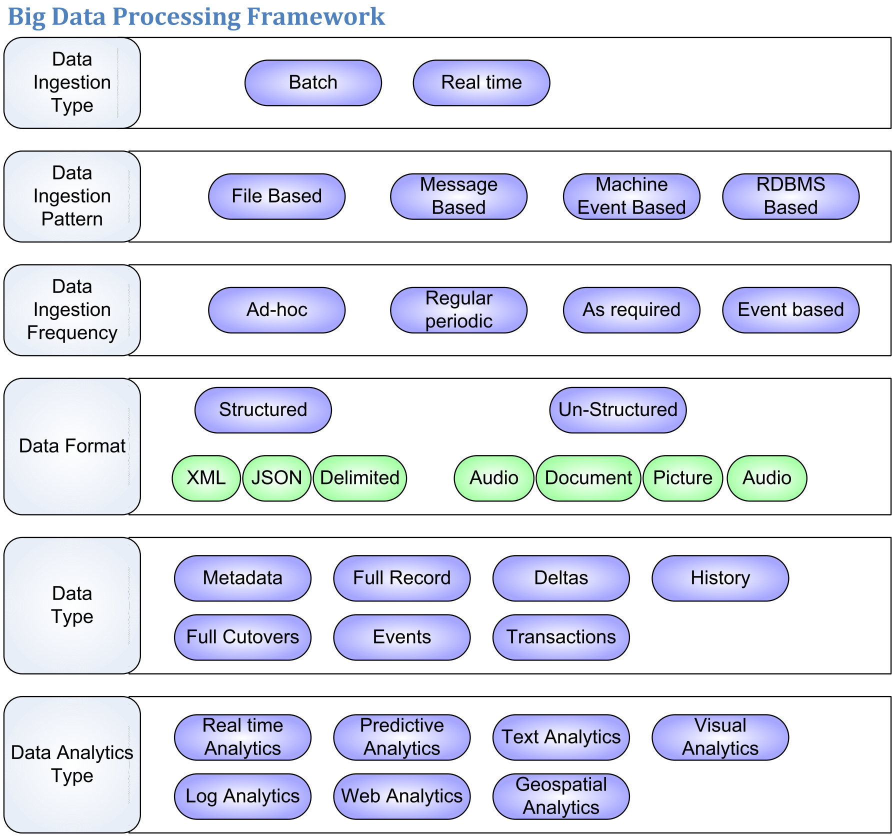

FIGURE 10.2

Big data processing framework

The above framework provides a way to understanding the impact on type of analytics tools required. Is there more focus on visual analytics, for example? It can also be used to understand the impact on the capacity of the infrastructure required by getting a view on the ingestion type and ingestion frequency. If the organization is using a system integrator or hosting on the cloud, this framework will help to think about the SLA required on the infrastructure.

10.4.1 PATTERNS FOR TRANSITIONING FROM DATA WAREHOUSE TO BIG DATA

The traditional data warehouse solution might look as depicted in Fig. 10.3.

The transition to a big data solution could be done via the following patterns:

1. Augmenting the big data solution within the data warehouse solution
2. Using only the big data solution
3. Adopting a hybrid model with coexisting data warehouse and big data solutions

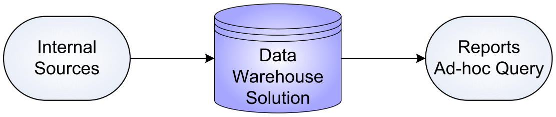

FIGURE 10.3

Traditional data warehouse solution

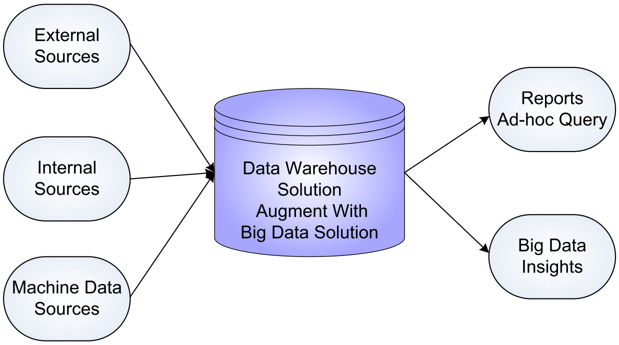

FIGURE 10.4

Big data solution added to a data warehouse solution

1. Augmenting the big data solution within the data warehouse solution

The existing data warehouse solution could be augmented with the big data solution to handle multiple data sources. (See Fig. 10.4.)

Pros:

- Consistent solution
- Minimal impact to existing business service

Cons:

- Licence cost
- Limits on the scalability
- Retained dependency on the legacy solution

Risk Profile

- **Medium Risk** as a result of new technology infusion with old technology

2. Using only the big data solution

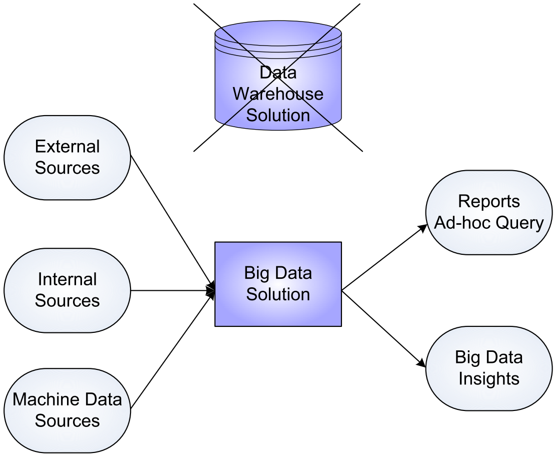

FIGURE 10.5

Single data by decommissioning of data warehouse solution

The existing data warehouse solution could be decommissioned and the new big data solution implemented to handle insights as well as reporting. (See Fig. 10.5.)

Pros:

- Simplifies the IT estate
- Highly scalable
- Commodity hardware

Cons:

- Significant impact to existing business service
- All eggs in one basket

Risk Profile

- **High Risk** as a result of dependency on a single solution for data queries/analytics

3. Adopting a hybrid model with coexisting data warehouse and big data solutions

The existing data warehouse solution could be retained alongside the new big data solution. (See Fig. 10.6.)

Pros:

- Minimal impact to existing business service
- Separation of responsibilities
- Commodity hardware and scalable

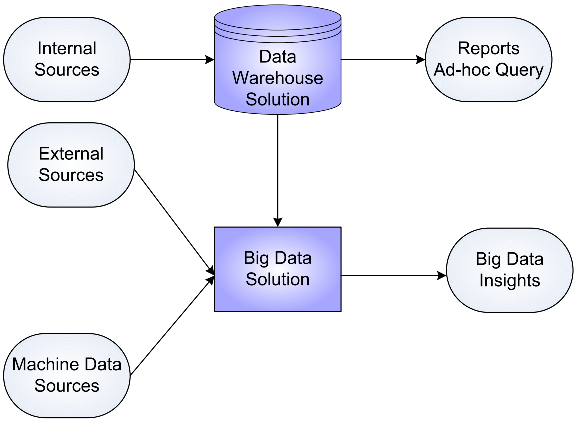

FIGURE 10.6

Hybrid model

Cons:

- Higher cost as a result of maintaining two solutions

Risk Profile

- **Low to Medium Risk** depending on how quickly the big data solution is implemented

Each big data transition pattern has pros and cons associated with it. An organization may want to take a full transformational approach or have high risk appetite and as a result choose the high risk pattern of the single big data solution to cover all use cases. Ultimately, the choice of the pattern selected by the organization depends on (a) risk appetite and (b) funding availability.

10.5 CONCLUSION

This is a significant growth area within the IT industry. As more data is captured and made available to be analyzed, more tools are being developed and released onto the market to exploit the data. Some of these tools are open source, some are open source with vendor-provided support, and others need specific software licence.

The most important “V” of the big data is finding the value in the data. There is no magic bullet for getting the right big data implementation. But using a combination of business problem focus, open source solution, power of the cloud, and understanding the transition to big data architecture can accelerate the journey and minimize the investment risk.

REFERENCES

[1] https://en.wikipedia.org/wiki/Big_data.
[2] http://research.google.com/archive/gfs.html.
[3] https://en.wikipedia.org/wiki/Cloud_computing.