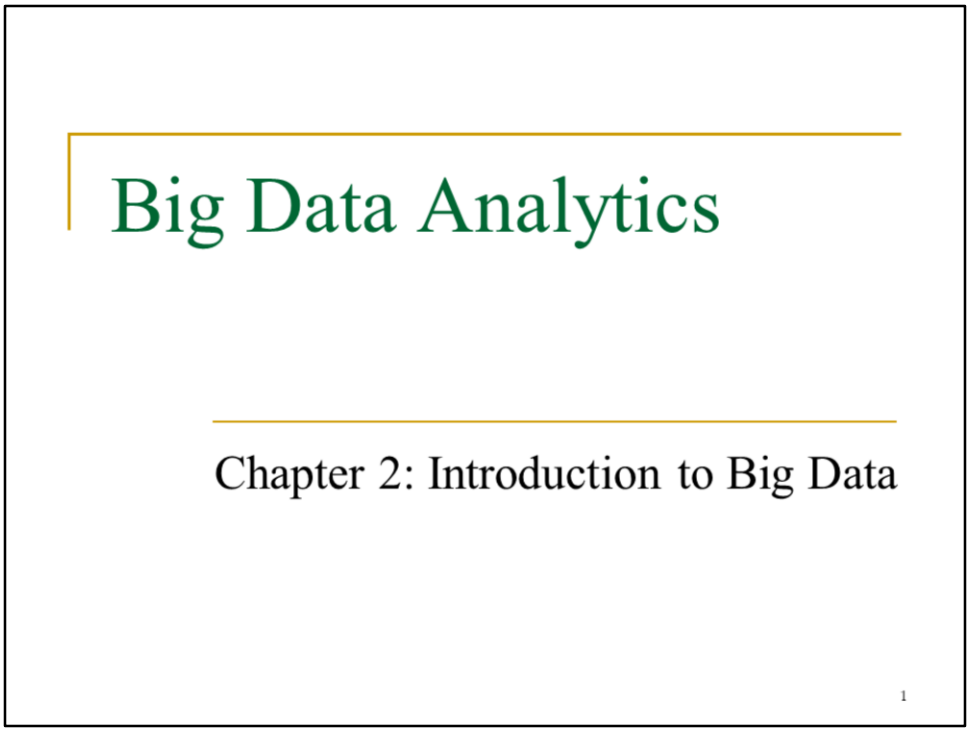
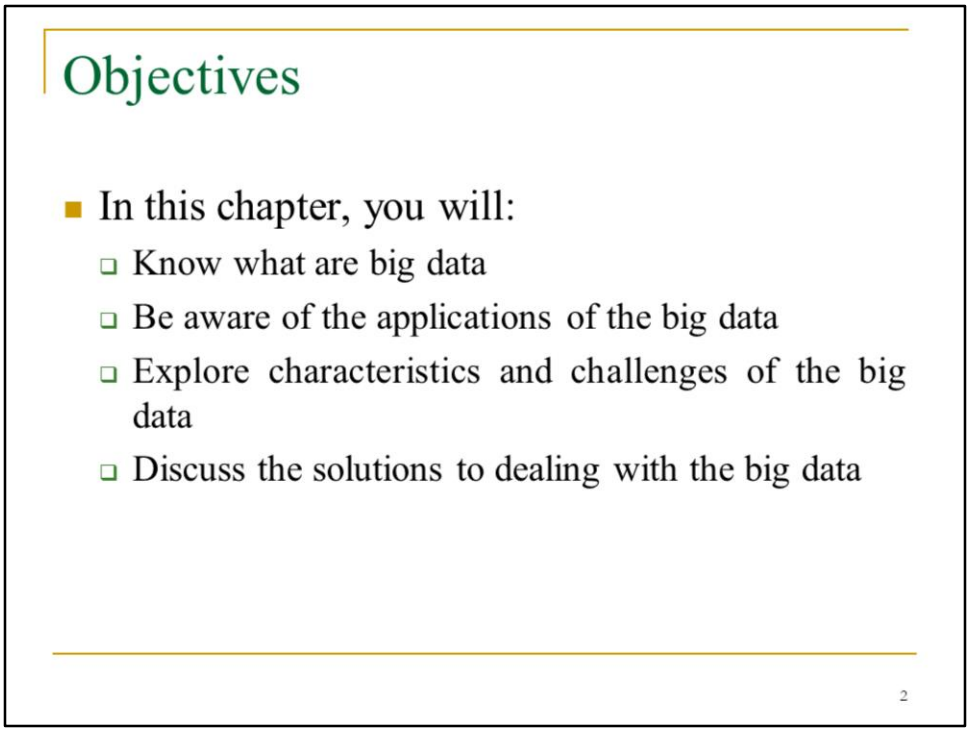
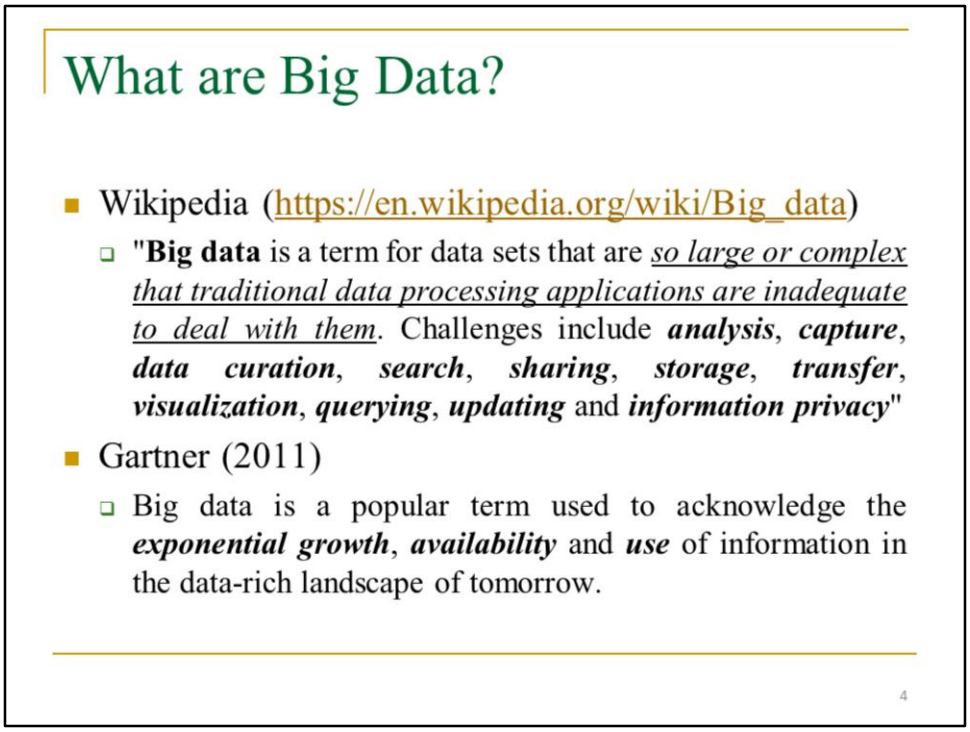


Chapter 2: Introduction to Big Data



In this chapter, we will learn what are big data, get familiar with real applications of big data, explore characteristics and challenges of big data, and discuss the solutions to dealing with big data.

Here is the outline of this chapter. We will first talk about the definition of big data.



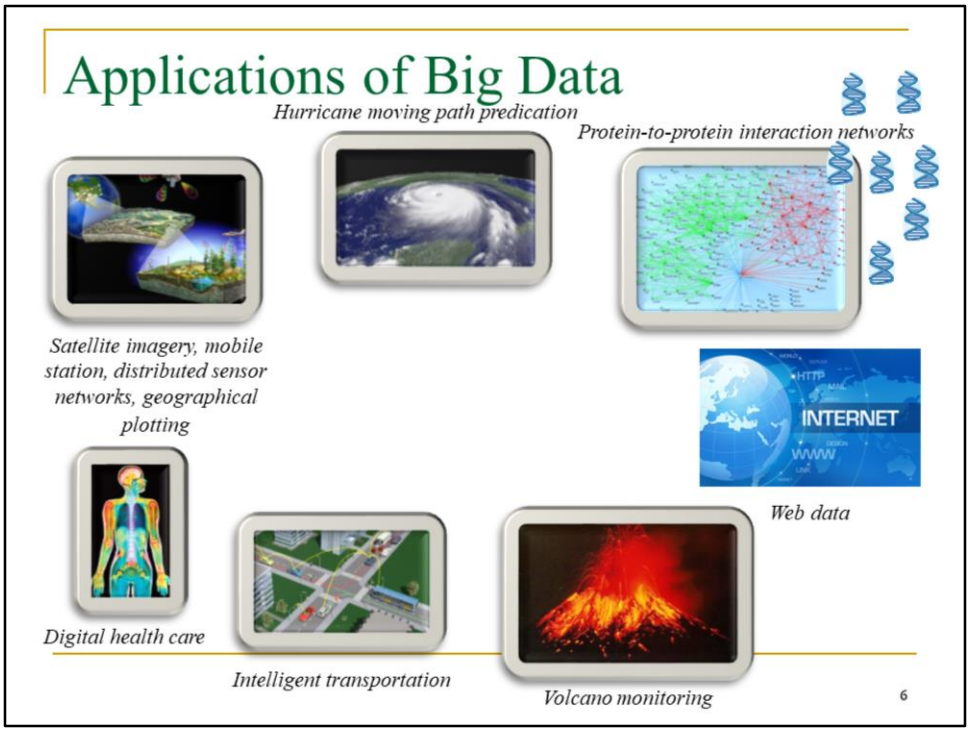
In fact, there is no unified formal definition about big data.

From Wikipedia, it says “Big data is a term for data sets that are so large or complex that traditional data processing applications are inadequate to deal with them. Challenges include analysis, capture, data curation, search, sharing, storage, transfer, visualization, querying, updating and information privacy”.

From Gartner, it says “Big data is a popular term used to acknowledge the ***exponential growth***, ***availability*** and ***use*** of information in the data-rich landscape of tomorrow."

Outline

- The Definition of Big Data
- Applications of Big Data
- Characteristics/Challenges of Big Data
- Topics of Big Data



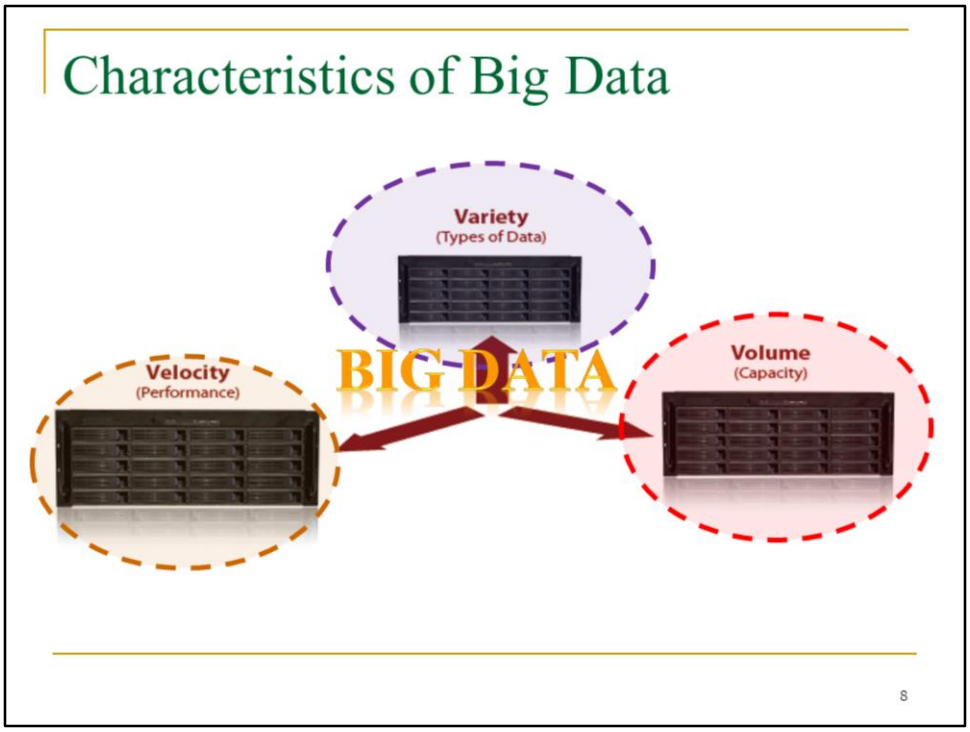
Big data have many real-life applications such as sensor networks, scientific databases, smart city, smart healthcare, Semantic Web, and so on.

Outline

- The Definition of Big Data
- Characteristics/Challenges of Big Data
- Topics of Big Data
- Applications of Big Data

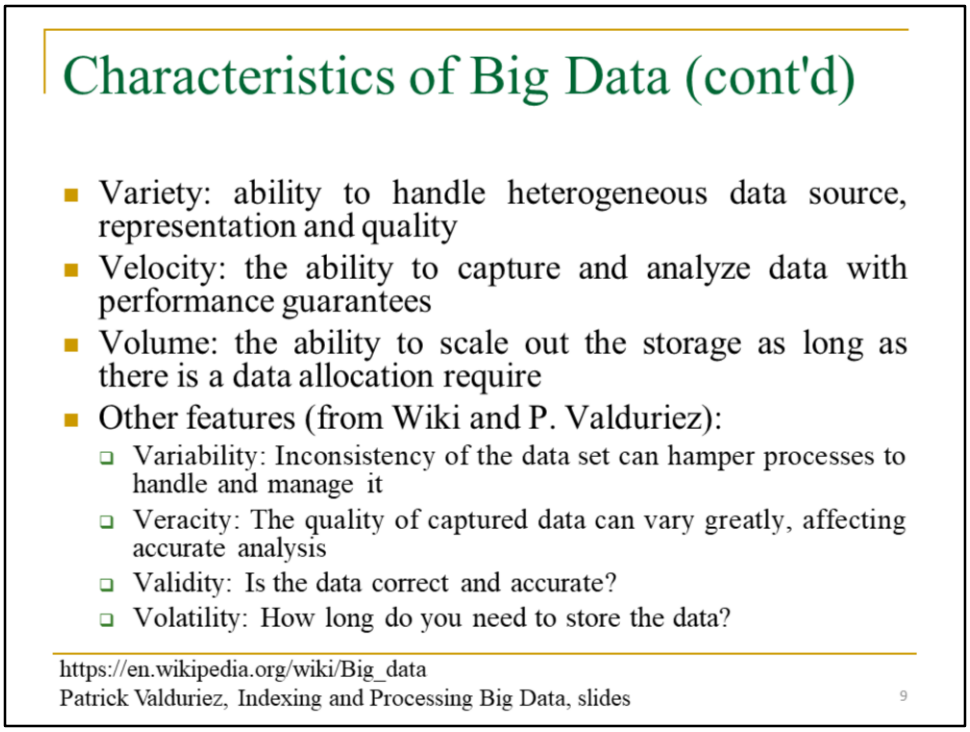
7

Big data have their own characteristics and challenges.

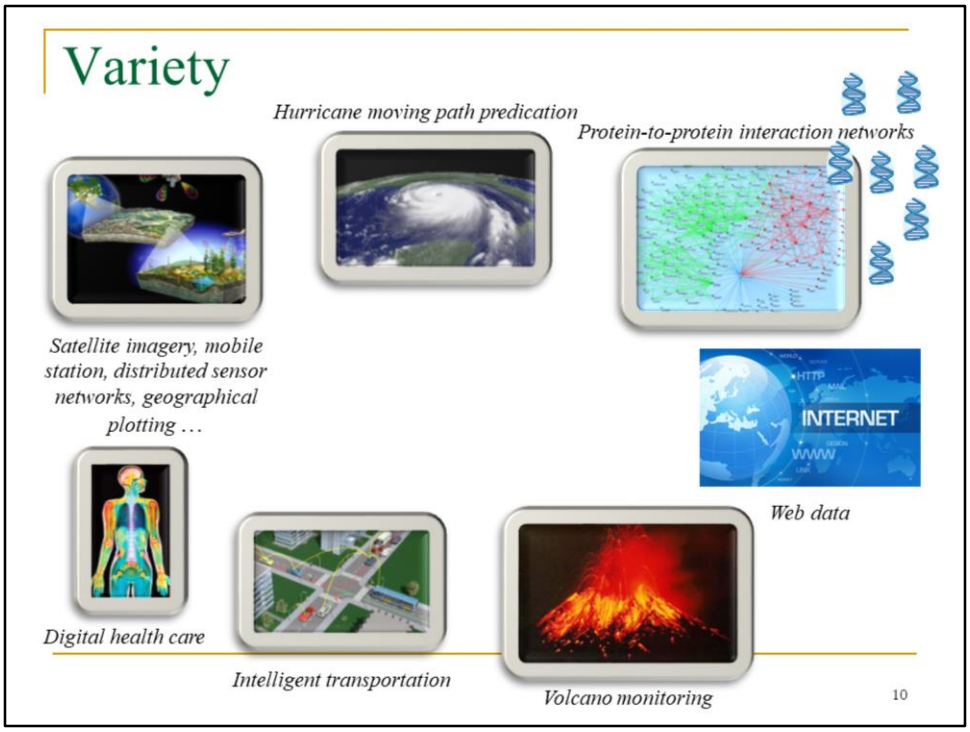


Gartner thinks the essential features of Big data can be summarized in 3V features.

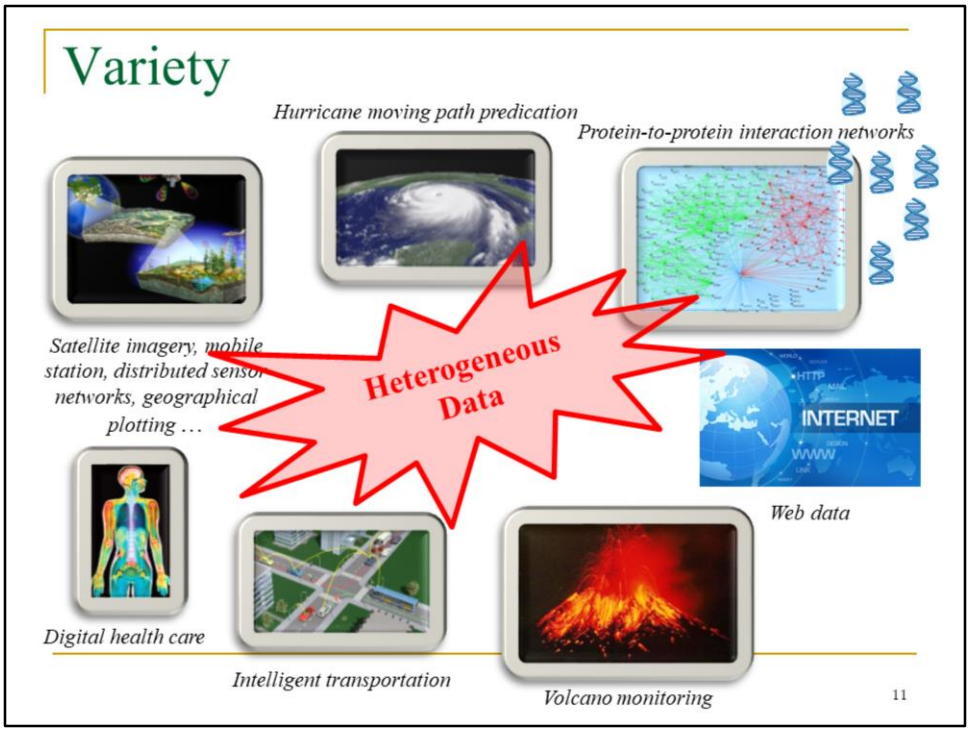
Variety: the ability to handle heterogeneous data source, representation and quality.
Volume: the ability to scale out the storage
Velocity: the ability to capture and analyze data with performance guarantees



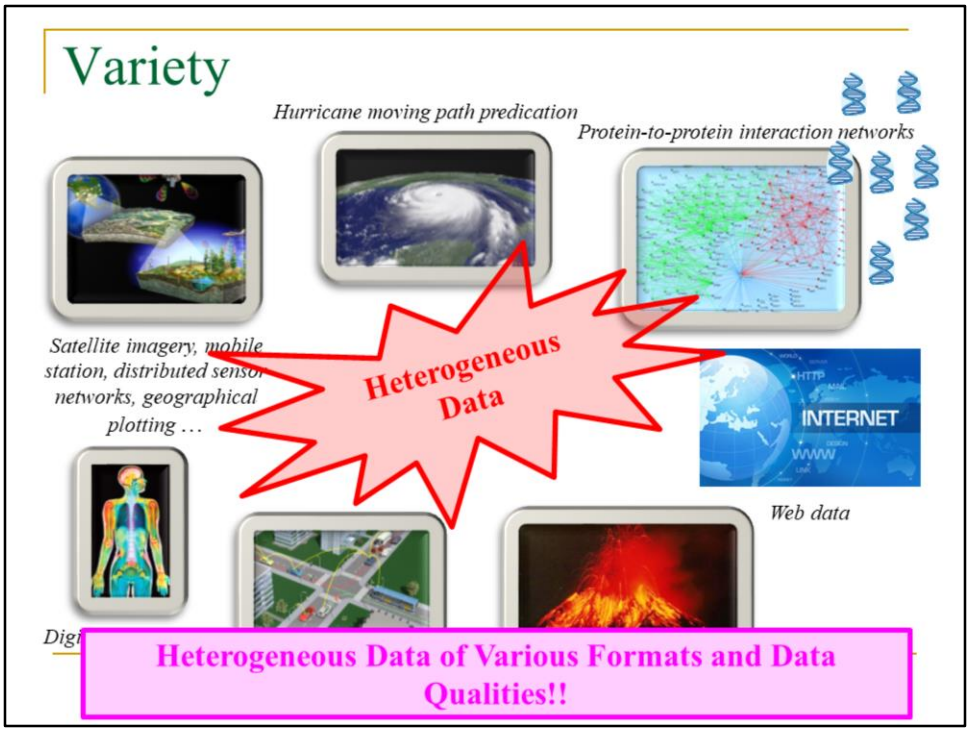
There are many other Vs, for example, variability, veracity, validity, and volatility. For the variability, there may be inconsistencies contained in the data set which can affect how we handle and manage the data. For the veracity, the quality of captured data can vary greatly, affecting accurate analysis. For the validity, the data may or may not be correct and accurate. For the volatility, the data may have their lifetimes, and you may not need some data long time ago (e.g., ten years ago). Therefore, some data may be expired and can be updated by removing them.



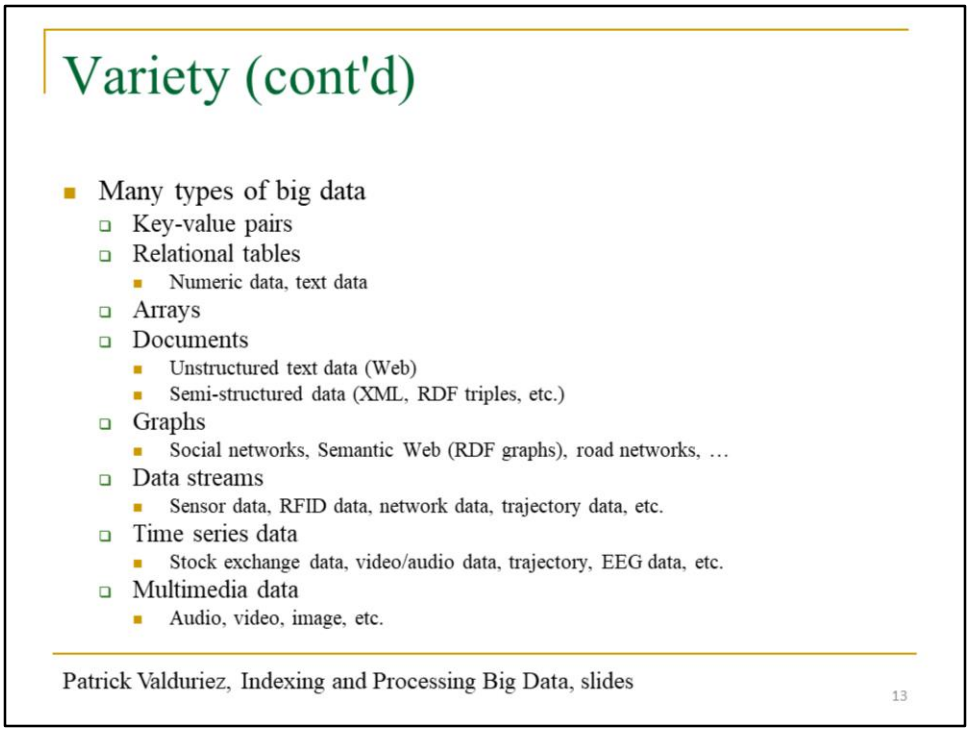
Let us briefly talk about each of the 3V features. For the variety, in real-world applications such as satellite images, sensor networks, biological database, Web data analysis, scientific databases, smart city/transportation systems, or smart healthcare, many real-world data are involved of different data types. For example, data can be of numeric values, text data, graphs, images, audio, videos, and so on.



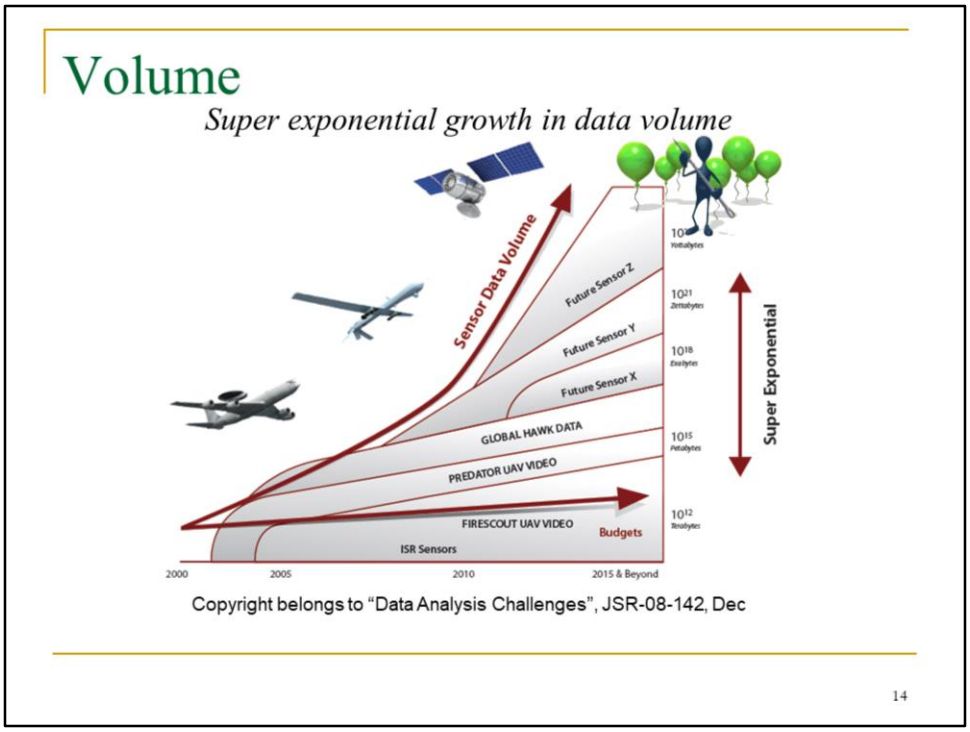
These data are quite heterogeneous.



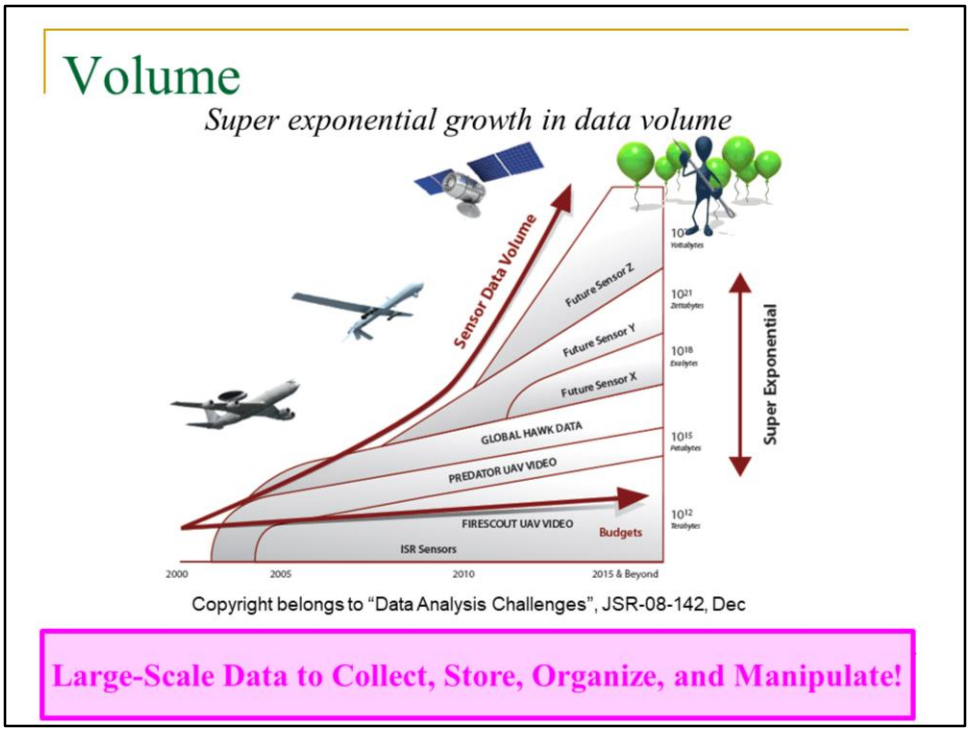
What is more, these data may contain noises and come from different data sources of various data qualities. Therefore, in practice, we need to handle heterogeneous big data of different data formats and different data qualities, which requires different data models/techniques and is rather challenging.



For types of big data, they can be key-value pairs which are used for distributed computing. They can be classical relational tables in relational databases (containing numeric, text or complex data attributes). We also have array data in biological databases, textual documents (containing unstructured, semi-structured XML or RDF data)), graphs in social networks, Semantic Web, or road networks, and data streams for sensor, RFID, Internet networks, trajectories, and so on. We may also encounter time series data (i.e., ordered data sequences over time), including stock exchange data, video/audio data, trajectory, EEG data (signal data collected from the scalp), and so on.



For the volume feature, the collected big data are now increasing dramatically over time. In the figure, the sensory data are collected with more and more sensors, and the volume of the sensory data increases exponentially over years.



With such large-scale data, it is very challenging to collect, store, organize and manipulate these data.

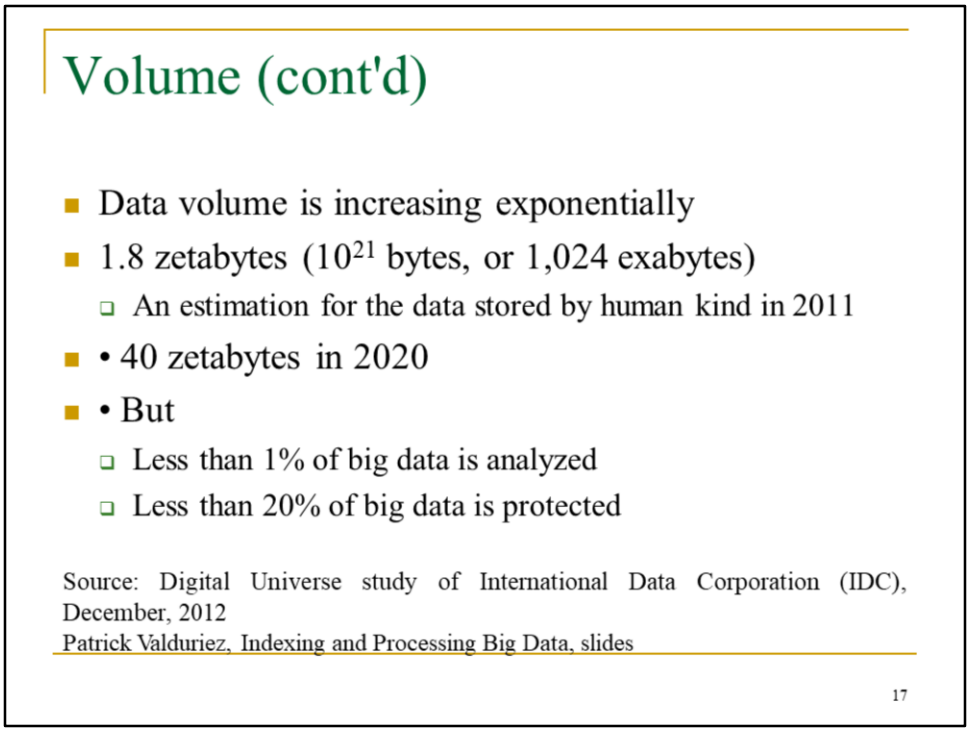
Volume (cont'd)

Old days: Only a few companies are generating data, all others are consuming data

Nowadays: All of us are generating data, and all of us are consuming data

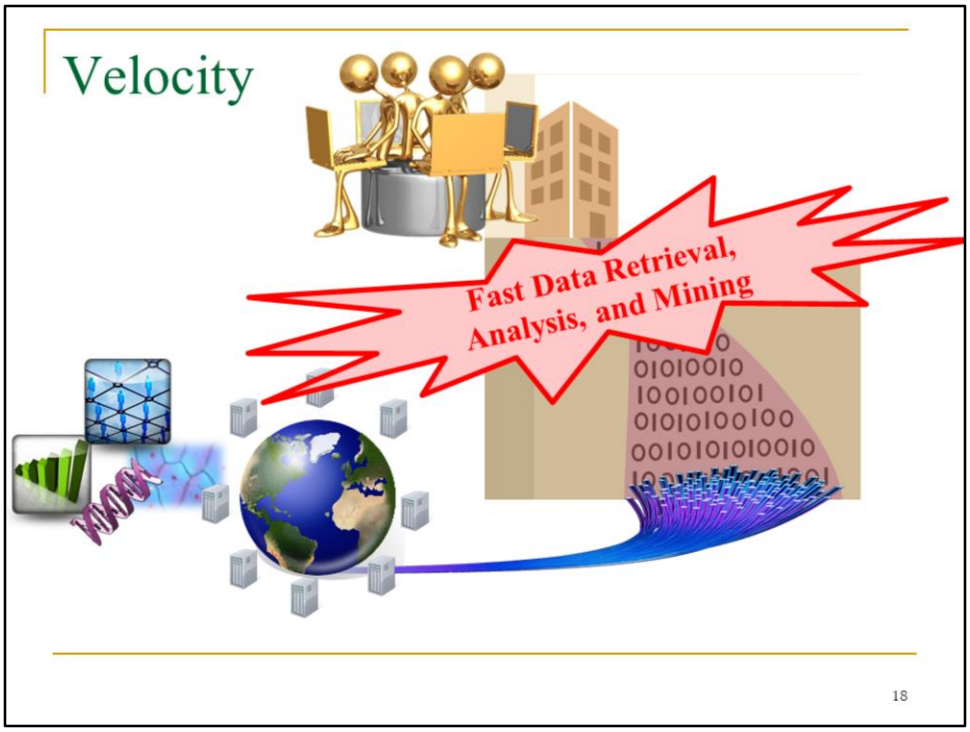
16

In the old days, we do not have many data. So as long as we have the data for analysis, people can do better than others. However, nowadays, we have too many data, sometimes overwhelmed by the data. Everyone is generating data. For example, on social networks, users (maybe you) often post messages, images, or videos, generating data with smartphones. Therefore, now we have too many data to process and analyze.

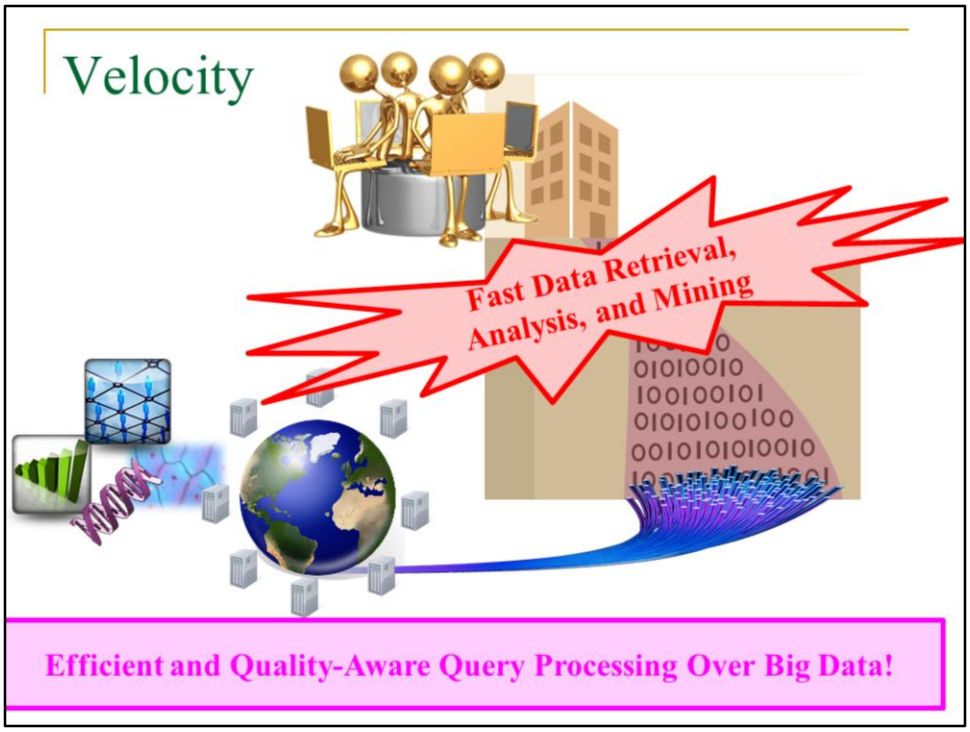


In 2011, there are about 1.8 zetabytes for data stored by human kind. It is expected that 40 zetabytes will be produced in 2020. It is not trivial how to efficiently process them.

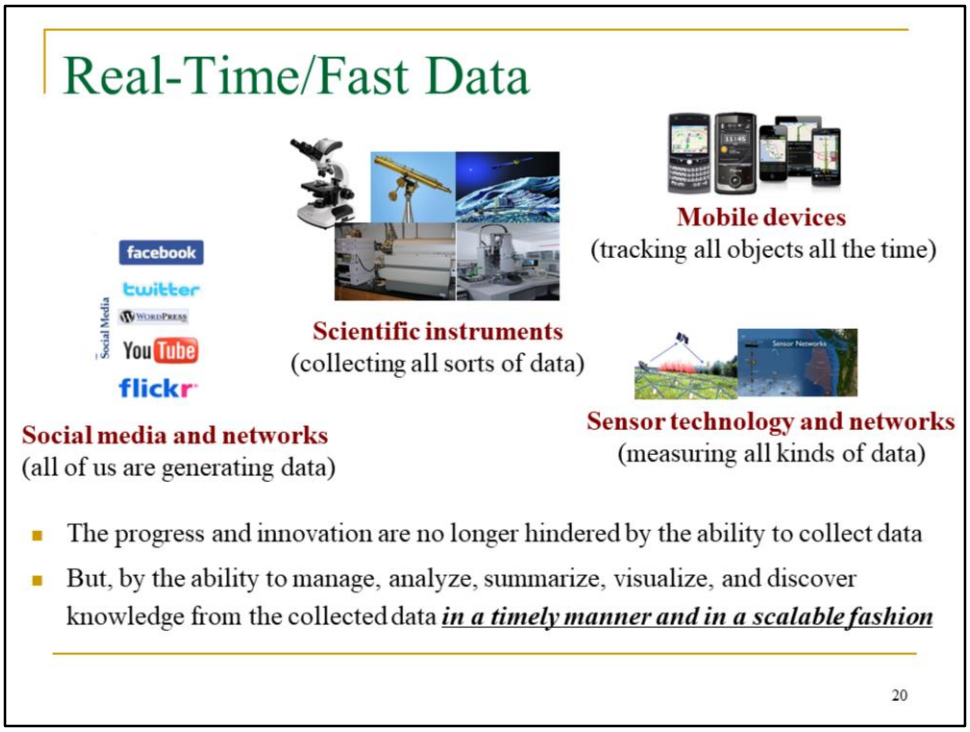
It is reported that only less than 1% of big data is analyzed, and others are just left/stored without any analysis. Moreover, less than 20% of big data is protected, and the remaining ones are under the risks of releasing people's privacy. Therefore, recently, the security and privacy preserving are vey hot for big data. This is however out of the scope of this course, and will not be covered by this course.



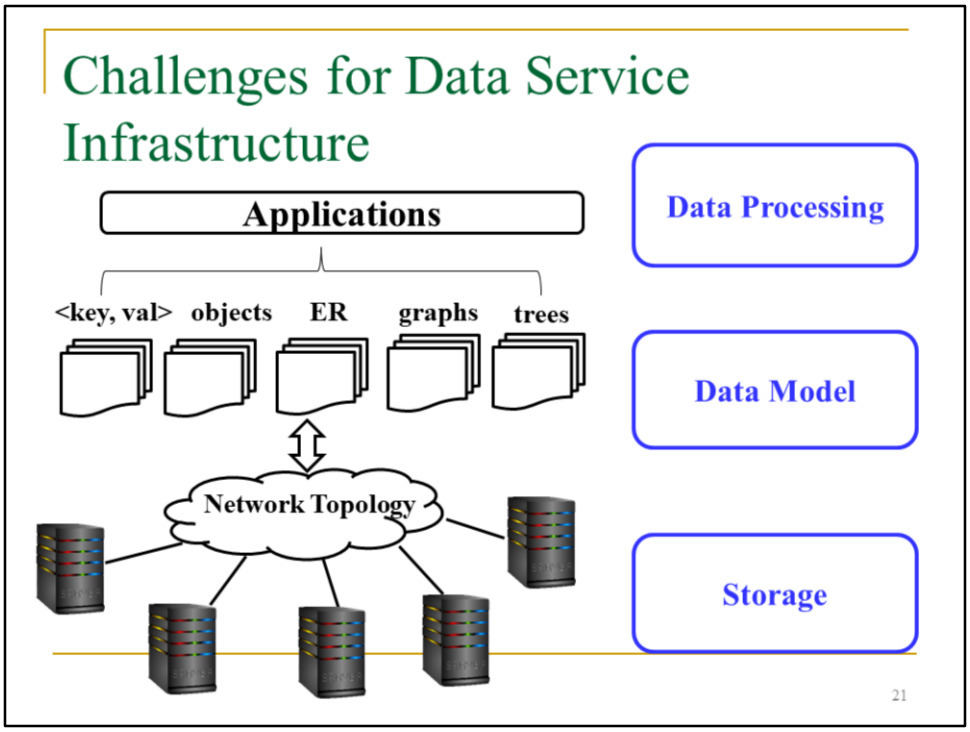
For the velocity, since we have large-scale data and some data are arriving or generated very fast in real applications, it is also challenging to fast analyze these data, retrieve useful information from these data, and even mine some new knowledge from the data.



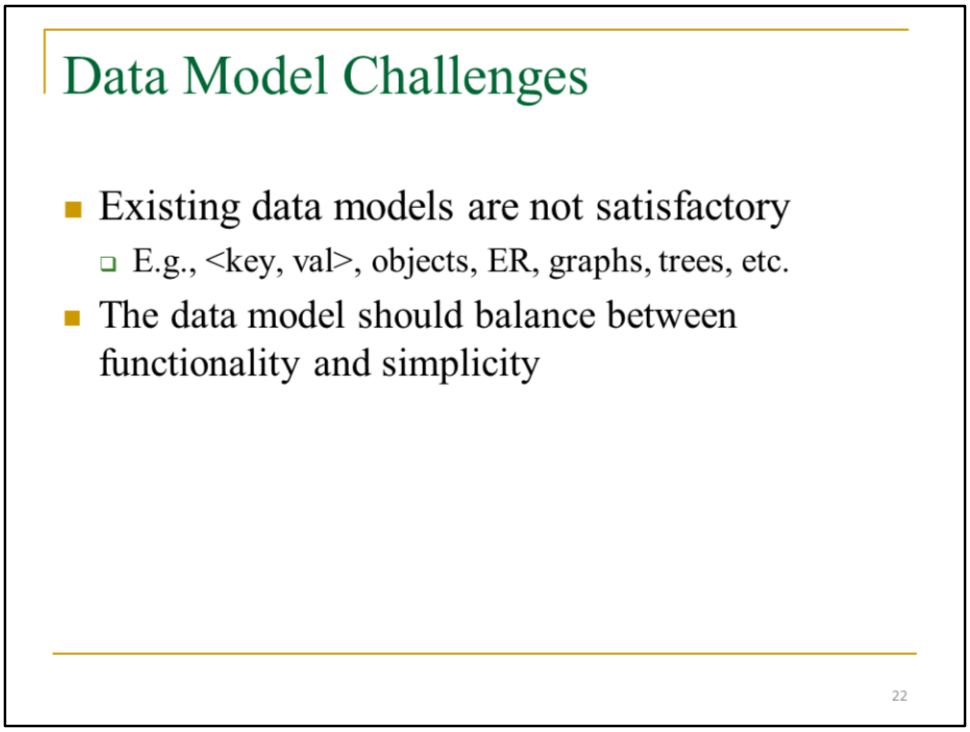
Therefore, we also need to design efficient and quality-aware approaches to process queries on such big data with small response time and low resource consumptions (such as the memory consumption).



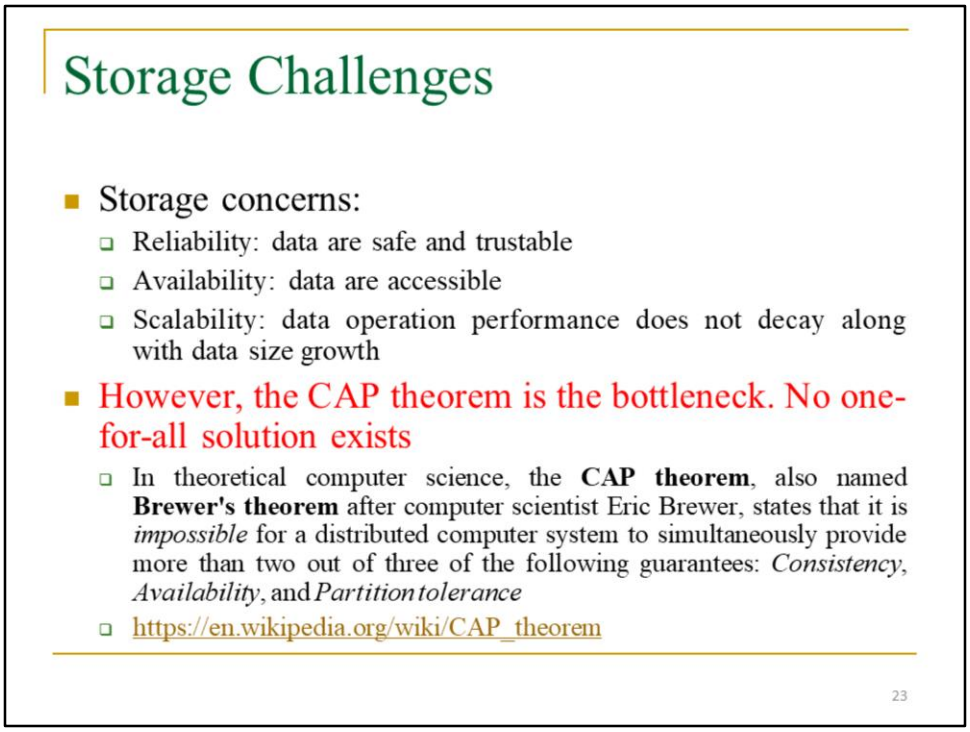
Nowadays, we have many real-time and fast arriving data such as social media data, scientific data, mobile data, and sensory data. The progress and innovations are no longer hindered by the ability to collect data, but by the ability to manipulate large-scale data in a timely manner and in a scalable fashion.



The current storage networks are not adaptive to the large-scale data with fast updates, due to reasons such as network traffic constraints. Moreover, data are from heterogeneous data sources, which can be of different data types, modeled by different data models such as key-value pairs, complex objects, Entity-Relationship model, graphs, or trees. The infrastructure should provide data processing tools or APIs to handle different data models. Current data service infrastructure is not ready for processing big data, and new tools should be designed to adapt the 3V features of big data.

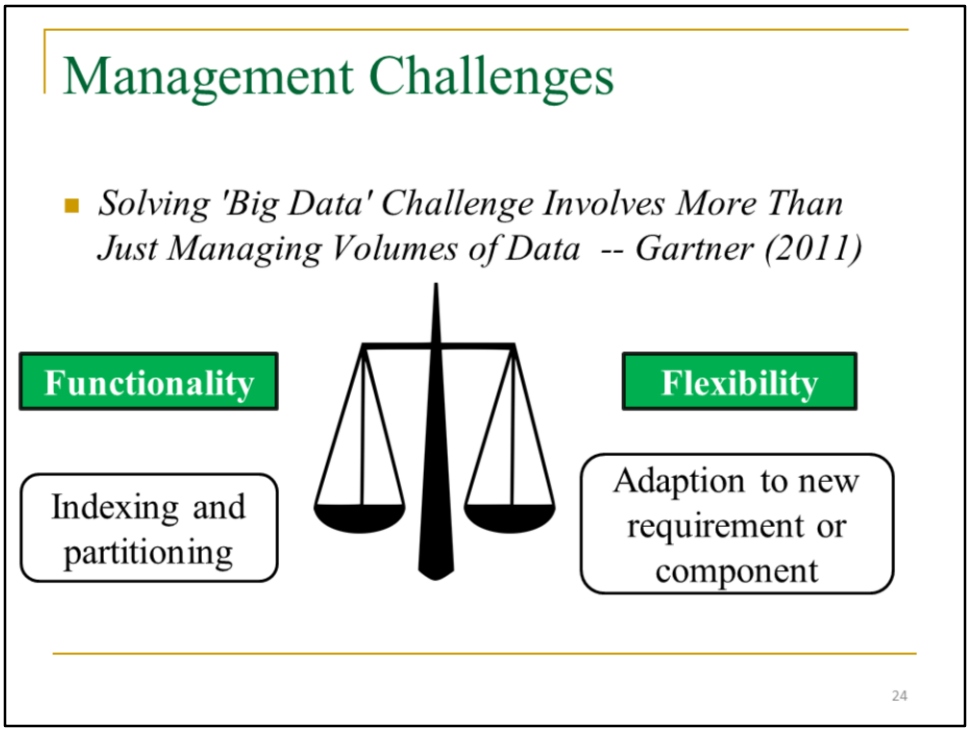


Existing data models such as key-value pairs, objects, ER, graphs, and trees are not adaptive to 3V features of big data (i.e., volume, velocity, and variety). Especially, for variety, the data model should be simple, but adaptive to heterogeneous data. For velocity, the data model should well capture features of big data, but at the same time be efficient for query processing. No previous data models can cover all different data models. Thus, we need to achieve a tradeoff between functionality and simplicity.



For the storage, it concerns about reliability, availability and scalability. Reliability mainly contains two parts: fault tolerance and consistency. Availability means data are always accessible (even when some servers have hardware failure). Scalability means the performance of the big data processing should grow smoothly with large data size.

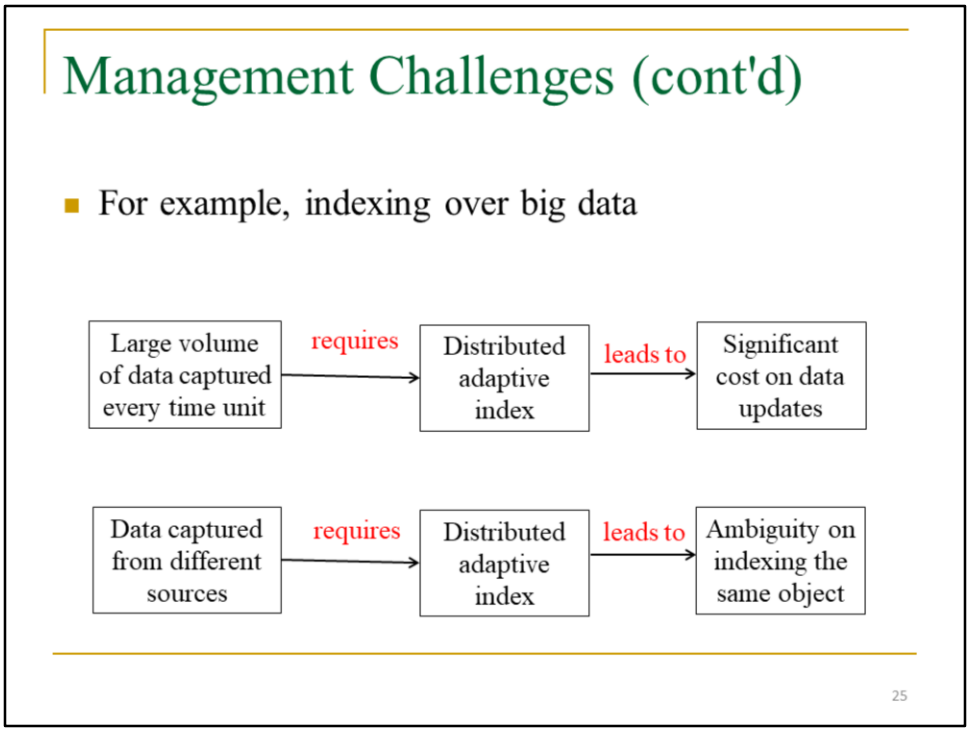
However, there is a CAP theorem for the storage problem. The CAP theorem states that it is *impossible* for a distributed computer system to simultaneously provide more than two out of three of the following guarantees: *Consistency*, *Availability*, and *Partition tolerance.* Similarly, here for the storage, we cannot guarantee all the 3 criteria, reliability, availability, and scalability, at the same time.



We not only need to store big data of large scale, but also want to effectively manage them.

The management of big data requires both functionality and flexibility. With good functionality, we can achieve good performance via indexing or data partitioning; with good flexibility, we can handle heterogeneous data that are adaptive to new requirements or component.

https://openclipart.org/detail/248664/scales



As an example, we consider indexing the big data. To tackle the volume challenge of big data, we need the distributed adaptive index, which however leads to high cost of data updates. To tackle the variety challenge of big data (from different data sources), we can also use the distributed adaptive index. However, it also leads to the ambiguity problem of identifying the same entity (or object) in the index.

Therefore, to manage big data, if we use some techniques, there must be some sacrifices or defects of these techniques. This is rather challenging to design good techniques to avoid such defects.

Challenges on Processing Big Data

- For example, new query language (algebra) for big data

Desired	Sacrifices & Overheads
Flexibility	Complexity in data modeling
Relational Supporting	Poor scalability
Uncertain Supporting	Poor scalability and significant computing overhead
Scalability	Less functionality
Efficiency & Effectiveness	Poor scalability

26

Similarly, for processing big data, we have our goals such as flexibility, relational supporting, uncertain supporting, scalability, and efficiency & effectiveness. However, existing processing tools may not be perfect. There are always some sacrifices and overheads. For example, to support the flexibility, we may need to model the heterogeneous and flexible data formats with higher complexity. With the scalability requirement, there must be less functionality. With high efficiency & effectiveness, the processing tool may be less scalable.

Therefore, we may need to find a balance or trade-off between the desired goal and sacrifices/overheads.

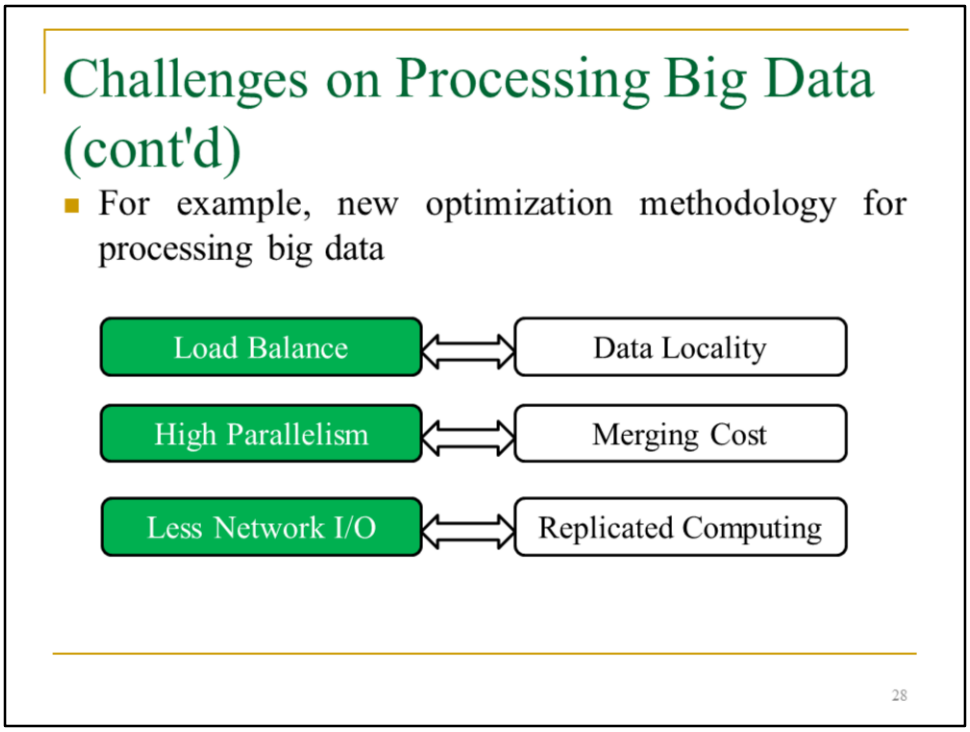
Challenges on Processing Big Data (cont'd)

- For example, new computing paradigm for processing big data

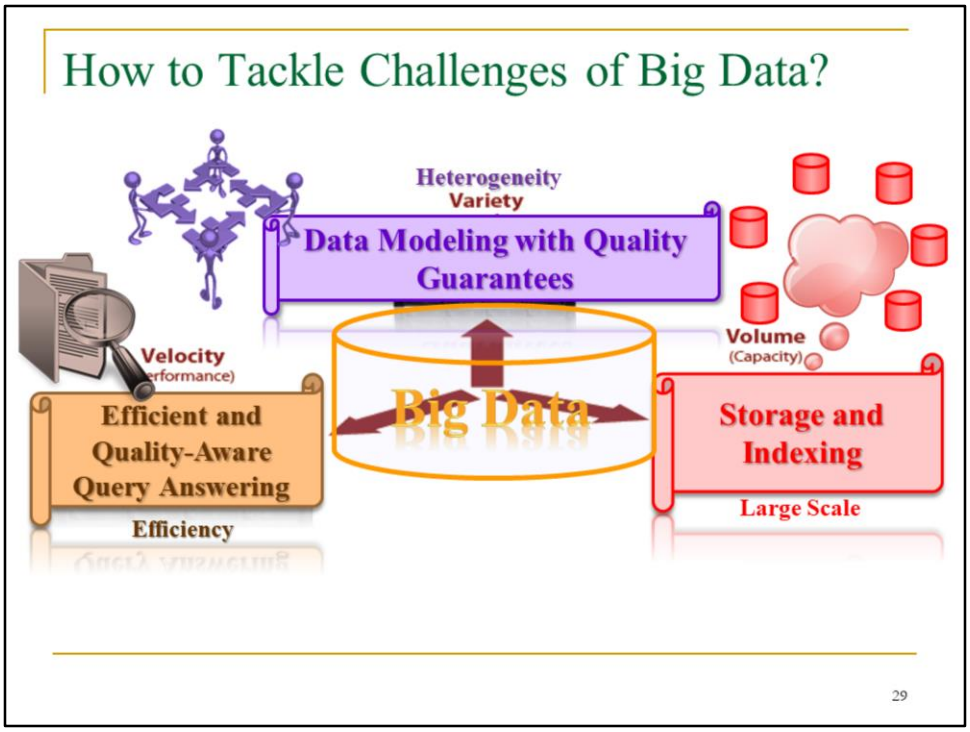
Distributed Computing Paradigm	Limitations
Message Passing	Poor scalability and fault tolerance
Unified Access	Invalidated efficiency over large computing nodes
MapReduce	Poor functionality

27

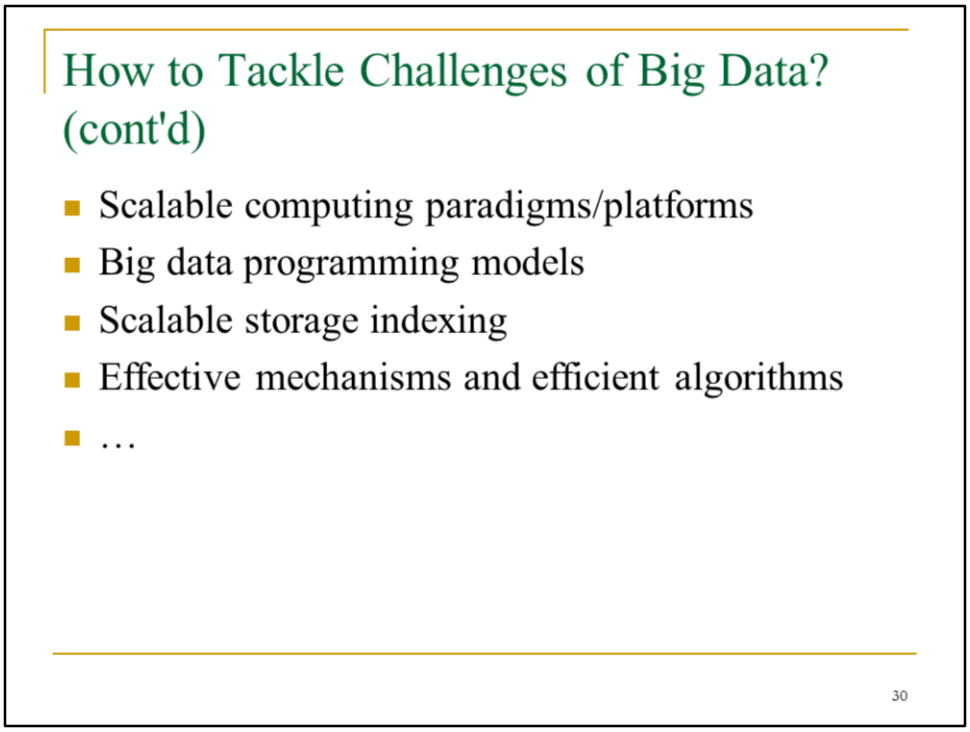
Moreover, distributed computing paradigms include message passing, unified access, and MapReduce. Although the message passing has the advantages in handling complex computing process and computing efficiency, they are not easy to maintain and not fault tolerant. MapReduce is popular for its great scalability and fault tolerant, but suffers from its framework that limits the functionalities.



In addition, for new optimization methodology, there are also many conflictions. For example, load balance is conflicting with data locality. High parallelism however may result in high merging cost. Less network I/Os may not work for replicated computing. We should consider optimization techniques for big data processing specific for real applications.



In order to tackle the challenges of big data, for 3V features, we will design different techniques. For example, for variety, we will model heterogeneous big data with quality guarantees. For volume, we will design effective storage and indexing mechanisms to deal with large-scale data, such as cloud-based storage. For velocity, we will develop effective and efficient query processing algorithms that consider data/query qualities.



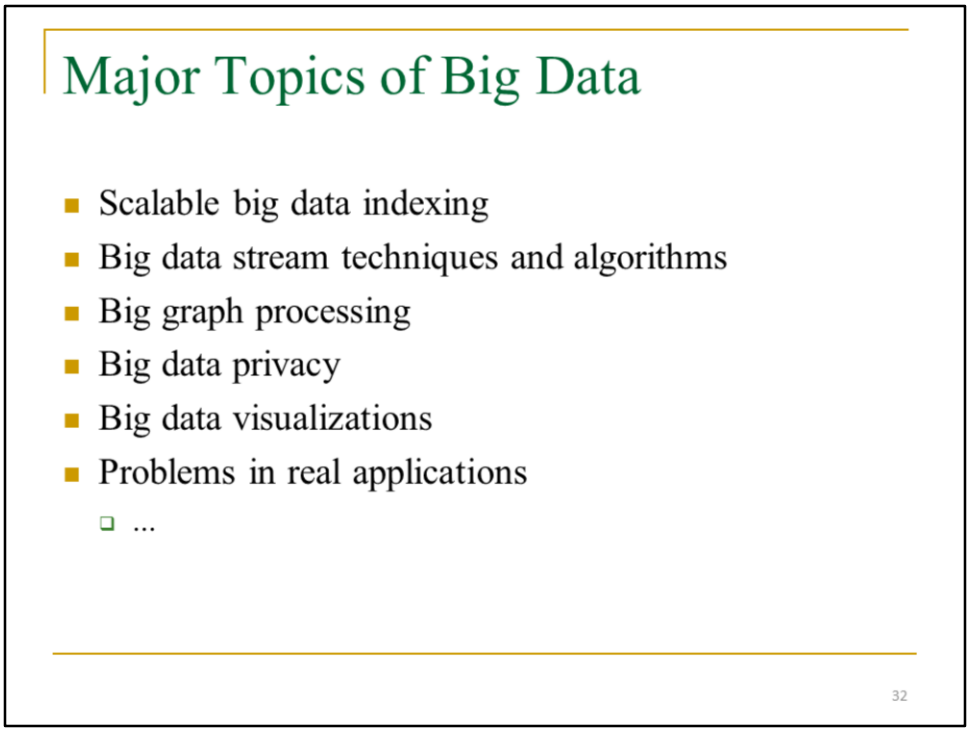
There are many detailed problems for big data processing such as scalable computing paradigms/platforms, big data programming models, scalable storage indexing, effective mechanisms and efficient algorithms, and so on.

Outline

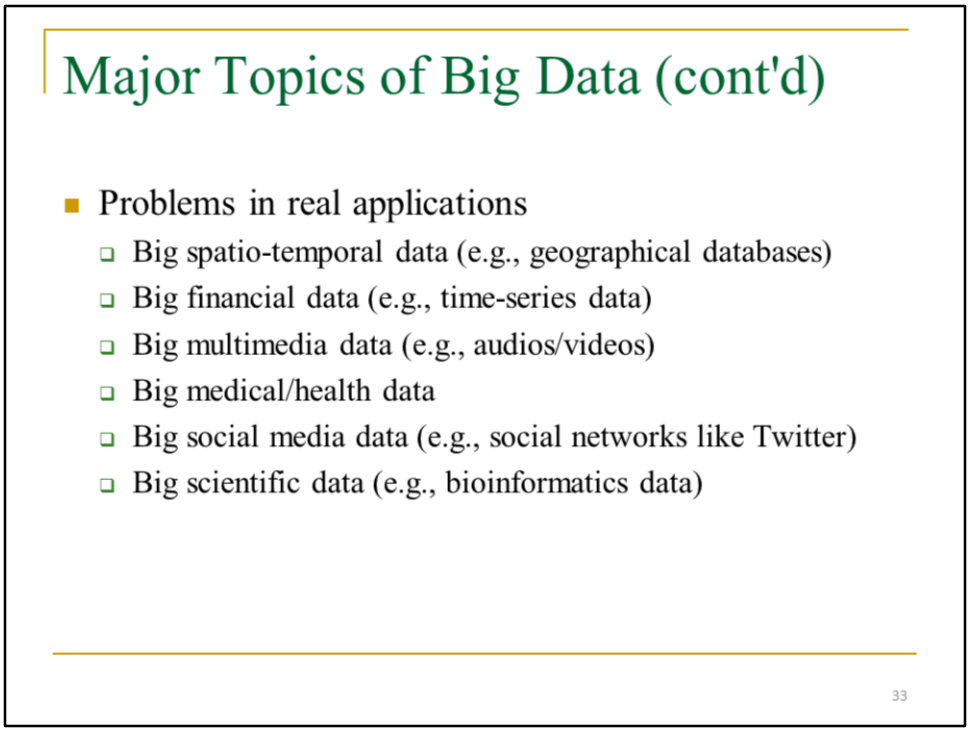
- The Definition of Big Data
- Applications of Big Data
- Characteristics/Challenges of Big Data
- Topics of Big Data

31

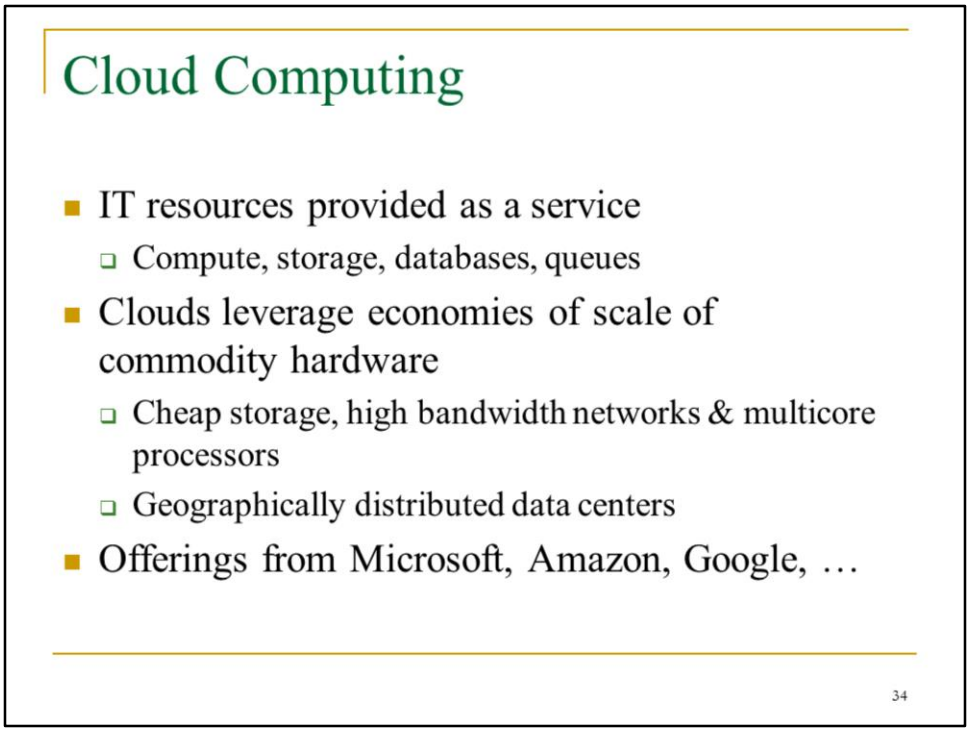
Next, we will talk about some topics of big data.



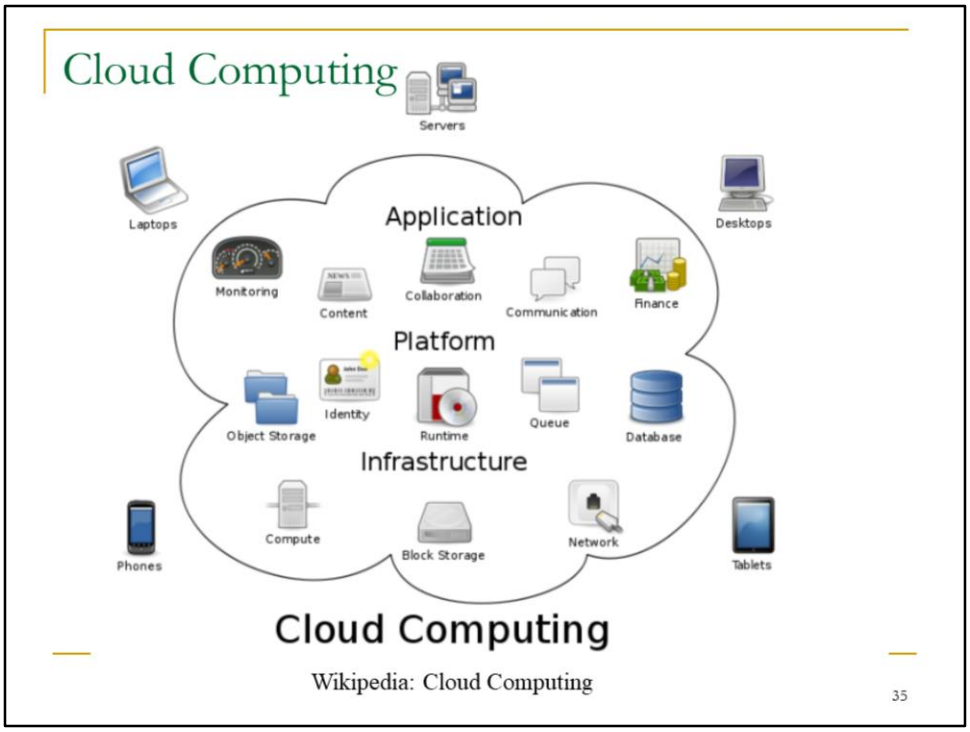
Here is a list of big data topics such as scalable big data indexing, big data stream techniques and algorithms, big graph processing, big data privacy, big data visualizations, and big data problems in real applications.



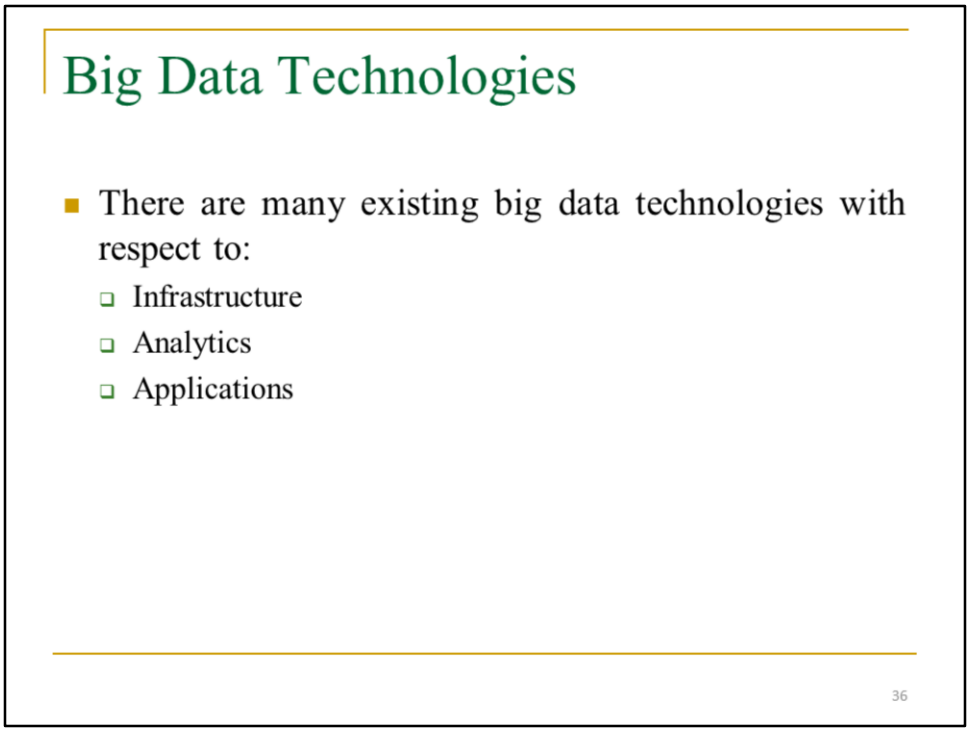
For big data problems in real applications, we can consider spatio-temporal data, financial data, multimedia data, medical/health data, social-media data, or scientific data.



One possible solution to dealing with big data is to consider cloud computing. There are many commercial products for cloud computing, such as the ones from Microsoft, Amazon, and Google. The companies maintain the cloud with cheap storage, high bandwidth networks, and multicore processors, distributed at different data centers. The cloud provides cloud-users with computing services such as storage, computation, data stores, and so on.



Here is a figure illustrating the cloud computing from Wikipedia.



There are many existing big data technologies with respect to infrastructure, analytics platform, and applications. We will cover some of them in our later lectures.