UNIT – I
INTRODUCTION TO DATA SCIENCE AND BIG DATA

Minal Apsangi

Unit I

- Introduction to Data science and Big Data, Defining Data science and Big Data, Big Data examples, Data explosion, Data volume, Data Velocity, Big data infrastructure and challenges, Big Data Processing Architectures, Data Warehouse, Re-Engineering the Data Warehouse, Shared everything and shared nothing architecture, Big data learning approaches.

- Objective
  - To introduce basic need of Big Data and Data science to handle huge amount of data.

Introduction

- Data : Lowest raw format of information or knowledge
- From where data comes ?
  - Every second tweets on Twitter
  - Every minute comments ,status updates, photos etc posted of Facebook
  - Customer transactions handling by various online shopping sites
  - Payments by using PayPal etc
  - And many more……

Big data

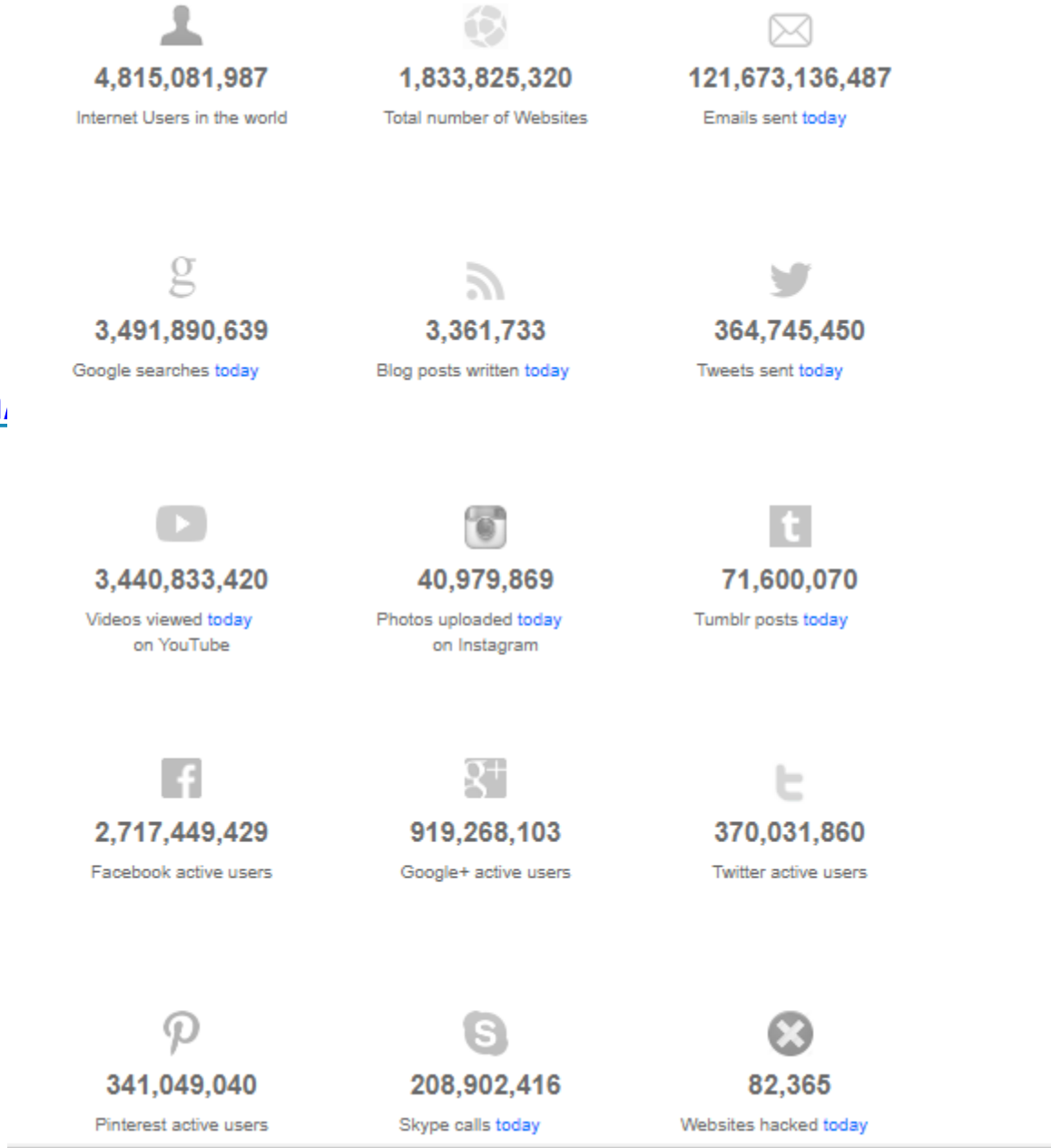

big data
information
search
management
analysis
science
based
petabytes
storage
internet
large
software
future
billion
research
sets
new
size
process
technology
volume
exabytes
years
real
business
processing
decision
online institute
approaches
development
use
million time


BIG
DATA
STORAGE
ANALYTICS
COMPLEX
TECHNOLOGIES
INFORMATION
DATABASES
RESEARCH
EXAMPLES
PETABYTES
SHARED
DISTRIBUTED
GENOMICS APPLIED
PROCESS
DEFINITION
WORKING
LARGE
PROCESSING
PRACTITIONERS
BIOLOGICAL
ONE
STORE
COST
SET
MOVING
CAPTURE
SOCIAL
LARGER
SAN
NEEDED
HUNDREDS
TERABYTES
TIME
INCLUDE
PARALLEL
SYSTEMS
CASE
INTERNET
RECORDS
SEARCH
LOGS
USING
DIFFICULTY
EVERY
MANAGE
CURRENT
CONTINUES
WITHIN
SIZE
ALSO
BUSINESS
MASSIVELY
ABILITY
SETS
MANAGEMENT
SOFTWARE
NETWORKS
DISK
CITATION
MPP
TARGET
USE

Data on the Internet...

Internet livestats

http://www.internetlivestats.com/

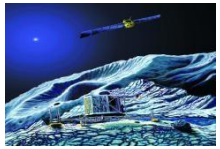


Who generates big data?

User Generated Content (Web & Mobile)

- E.g., Facebook, Instagram, Yelp, TripAdvisor, Twitter, YouTube

Health and scientific computing



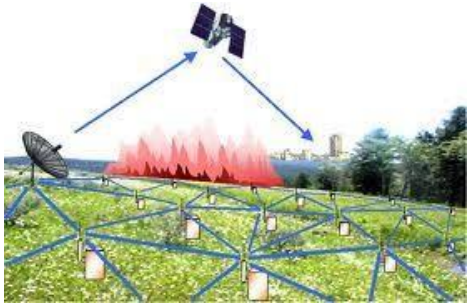
Who generates big data?

Log files

- Web server log files, machine system log files

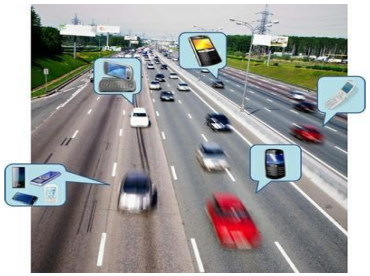
Internet Of Things (IoT)

- Sensor networks, RFIDs, smart meters



An example of Big data at work

Crowdsourcing



Sensing

Map data

Computing

Real time traffic info

Travel time forecast/nowcast

Data Explosion

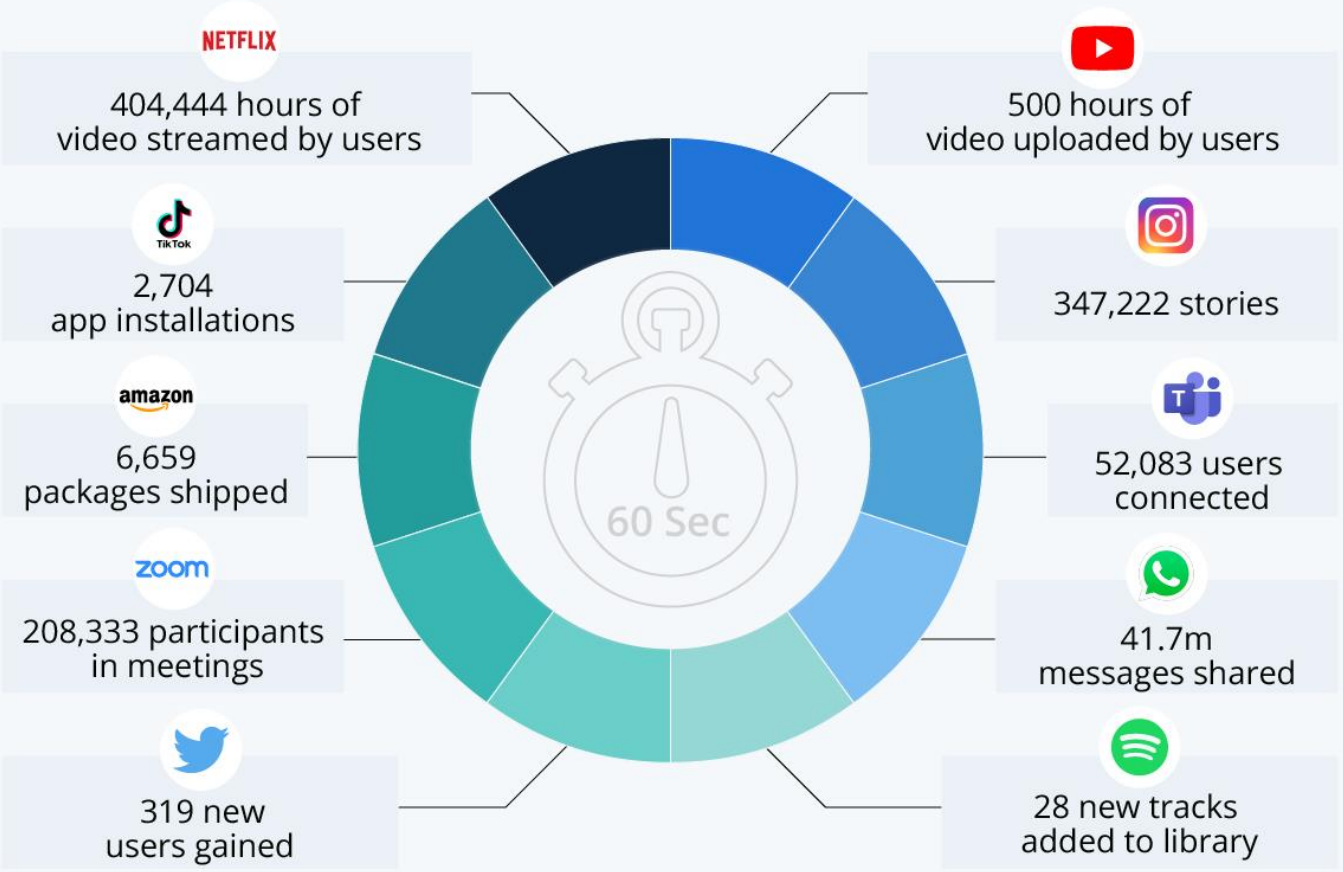
- Continuous increase in the volume of published information or data and the effects of this abundant information or Data

Various Sources of Big Data

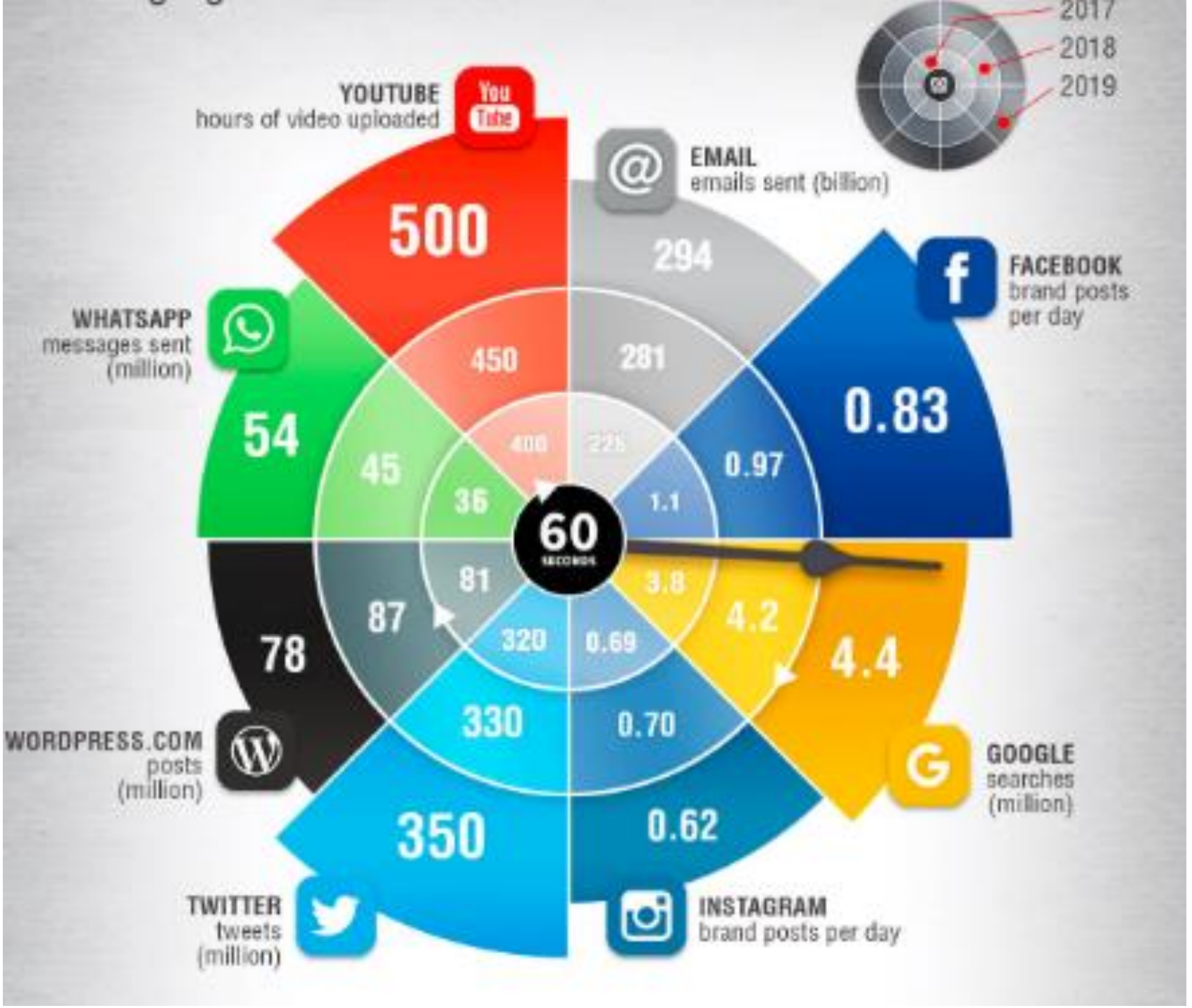
- Social Data
- Machine Data
- Transactional Data

A Minute on the Internet in 2020

Estimated amount of data created on the internet in one minute



Source: Visual Capitalist


2017
2018
2019
YOUTUBE
hours of video uploaded
500
EMAIL
emails sent (billion)
294
FACEBOOK
brand posts per day
0.83
WHATSAPP
messages sent (million)
54
450
281
45
36
0.97
1.1
60
81
87
3.8
4.2
4.4
78
320
0.69
330
0.70
WORDPRESS.COM
posts (million)
GOOGLE
searches (million)
350
0.62
TWITTER
tweets (million)
INSTAGRAM
brand posts per day

Defining Big Data

- **Big data** is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tools or traditional data processing applications.
- "***Big Data***" is data whose scale, diversity, and complexity require new architecture, techniques, algorithms, and analytics to manage it and extract value and hidden knowledge from it…

Challenges

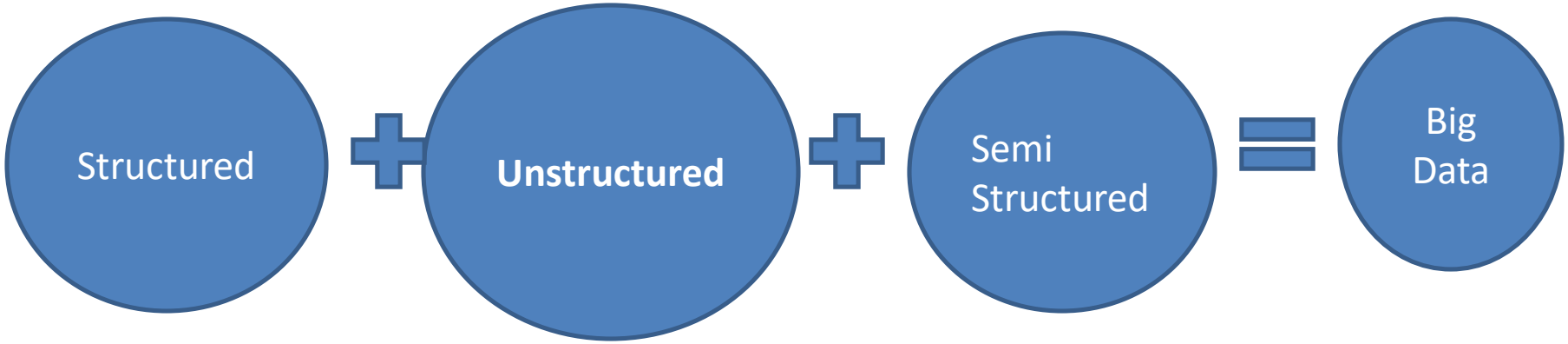
- The challenges include
  - capture
  - storage
  - search
  - sharing
  - transfer
  - analysis
  - visualization.

- Data Acquired from multiple sources can be categorized into
  - Internal Sources
  - External Sources

Data Source	Definition	Examples of Sources	Applications
Internal	Provides organized Data that originates from the enterprise that helps run business	•CRM •ERP •Customer Details •Products and Sales Data	Data is used to support daily business operations of an Organization
External	Provides unorganized Data that originates from external environment of an Organization	•Business Partners •Data Suppliers •Internet •Government •Market Research Organizations	This data is often analyzed to understand competitors , market ,environment and technology

Types of Big Data

- Structured
- Unstructured
- Semi Structured



Structured Data

- Organized in Pre- defined format
- Data that resides in fixed fields within a record
- Formatted data that has entities and their attributes mapped
- Used to query and report against predetermined data types

Sources of Structured Data

- Relational Databases
- Flat files in record format
- Multidimensional Databases
- Legacy Databases

Unstructured Data

- Consists of typically metadata
- Comprises inconsistent Data
- Consists of data in different formats like e-mails, text, audio, video or image files

Sources of Unstructured Data

- Text Internal and external to an Organization
  - Documents, logs, survey,feedbacks ,emails etc
- Data From Social Media
- Mobile Data

Semi-structured Data

- Also known as schema-less or self describing structure
- Part of structured data that contains tags

- Sources :
  - File systems like Web data in the form of cookies
  - Data exchange formats like JSON objects

What Leads to Data explosion

- Business Model transformations
- Globalization
- Personalization of services

The Vs of big data

The 3Vs of bigdata

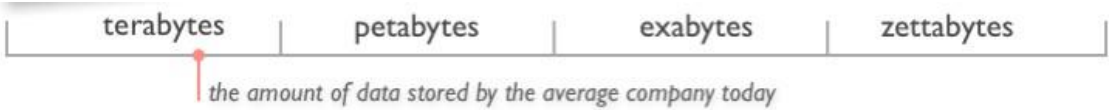
- Volume: scale of data
- Variety: different forms of data
- Velocity: analysis of streaming data

…but also

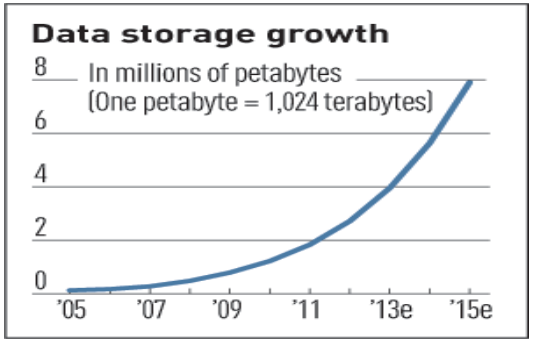
- Veracity: uncertainty of data
- Value: exploit information provided by data

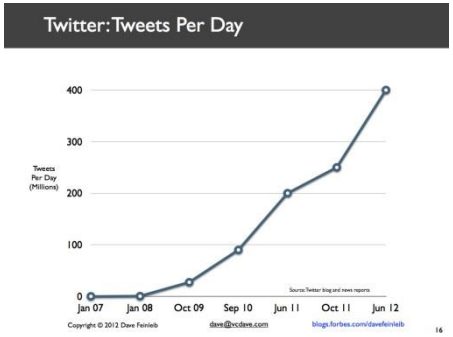
The Vs of big data

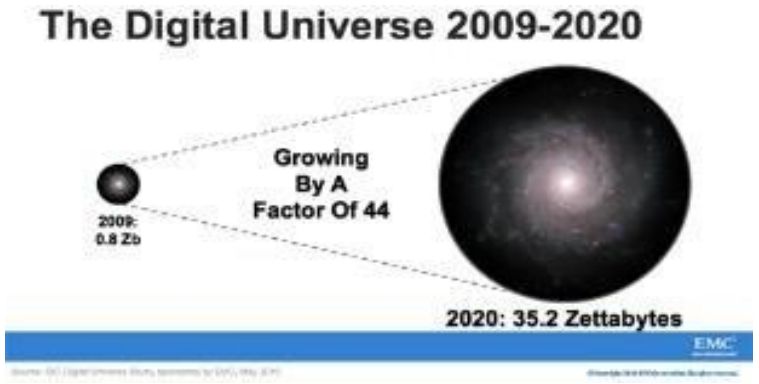
Volume



- Data volume increases exponentially over time
- 44x increase from 2009 to 2020
  - Digital data 35 ZB in 2020

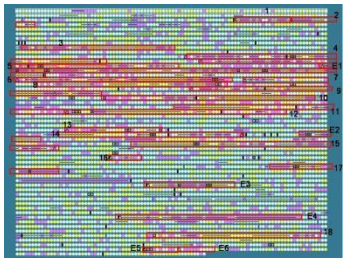


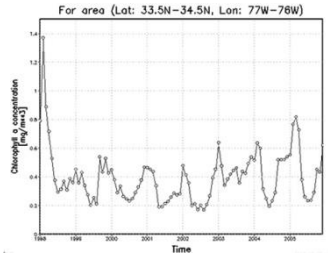


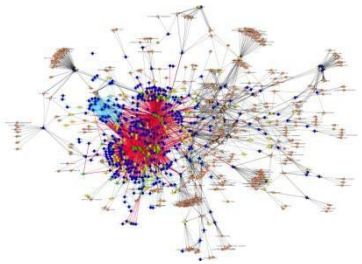


The Vs of big data

- Variety
  - Various formats, types and structures
    - Numerical data, image data, audio, video, text, time series







  - A single application may generate many different formats
    - Heterogeneous data
    - Complex data integration problem

Characteristics of Big Data: 3-Speed (Velocity)

- Data is being generated fast and need to be processed fast
- Online Data Analytics
- Late decisions ➔ missing opportunities

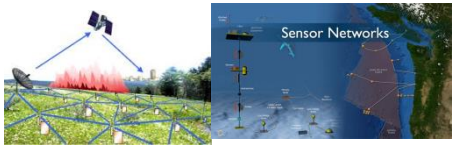
- **Examples**
  - **E-Promotions:** Based on your current location, your purchase history, what you like ➔ send promotions right now for store next to you
  - **Healthcare monitoring:** sensors monitoring your activities and body ➔ any abnormal measurements require immediate reaction

Real-time/Fast Data

Social media and networks
(all of us are generating data)

Scientific instruments
(collecting all sorts of data)

Mobile devices
(tracking all objects all the time)



Sensor technology and networks
(measuring all kinds of data)

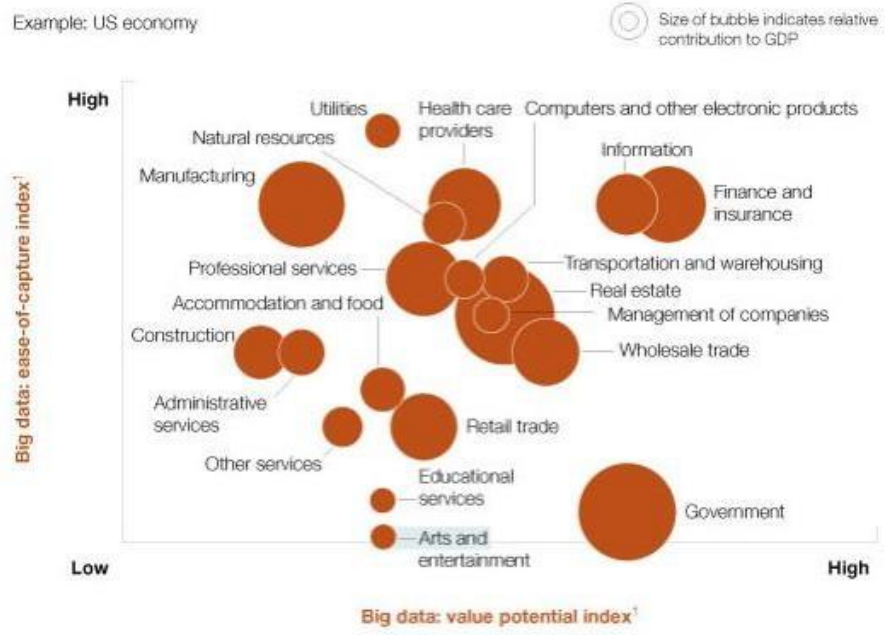
- The progress and innovation is no longer hindered by the ability to collect data
- But, by the ability to manage, analyze, summarize, visualize, and discover knowledge from the collected data in a timely manner and in a scalable fashion

The Vs of big data

- Veracity
  - Data quality

The Vs of big data

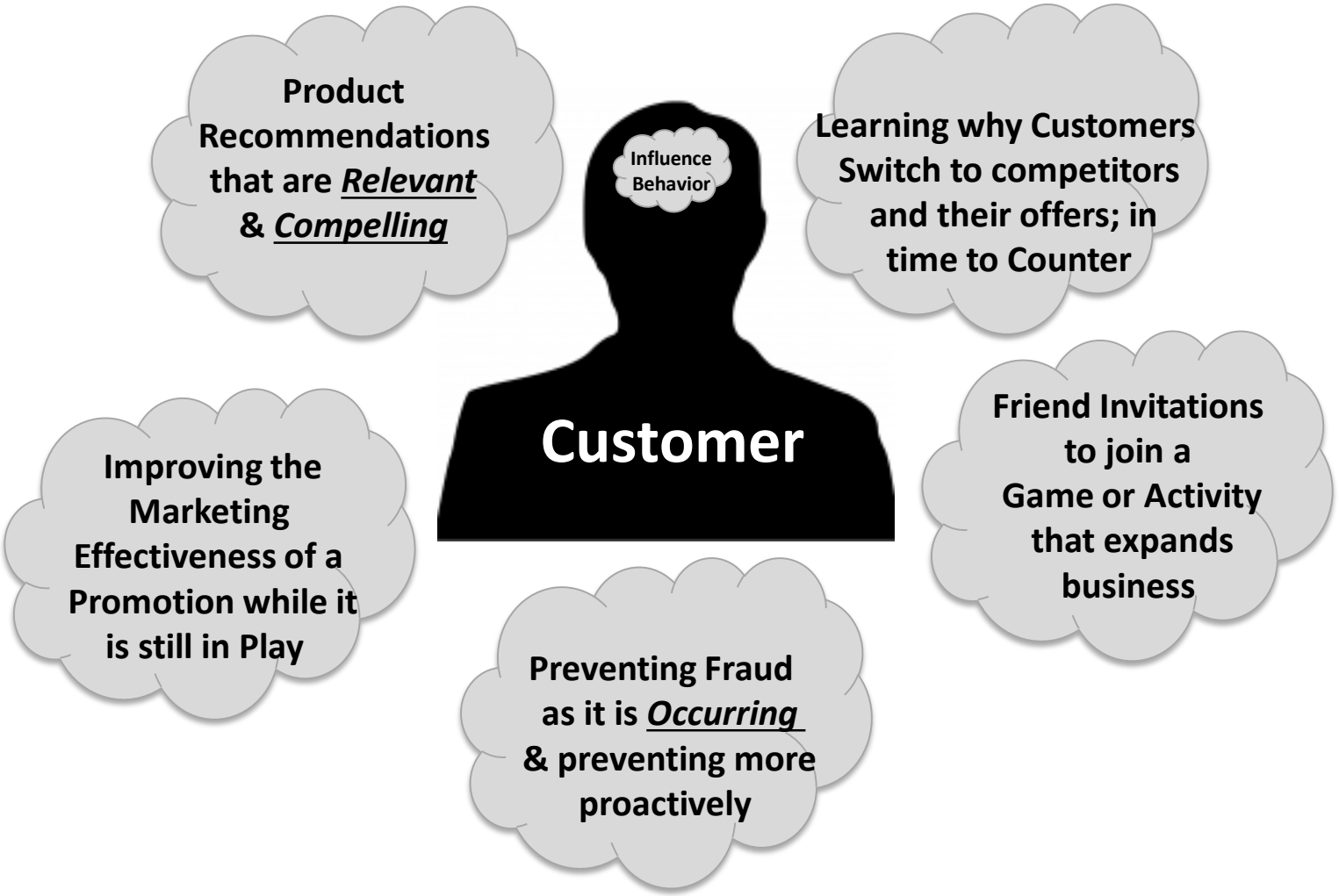
- Value
  - Translate data into business advantage



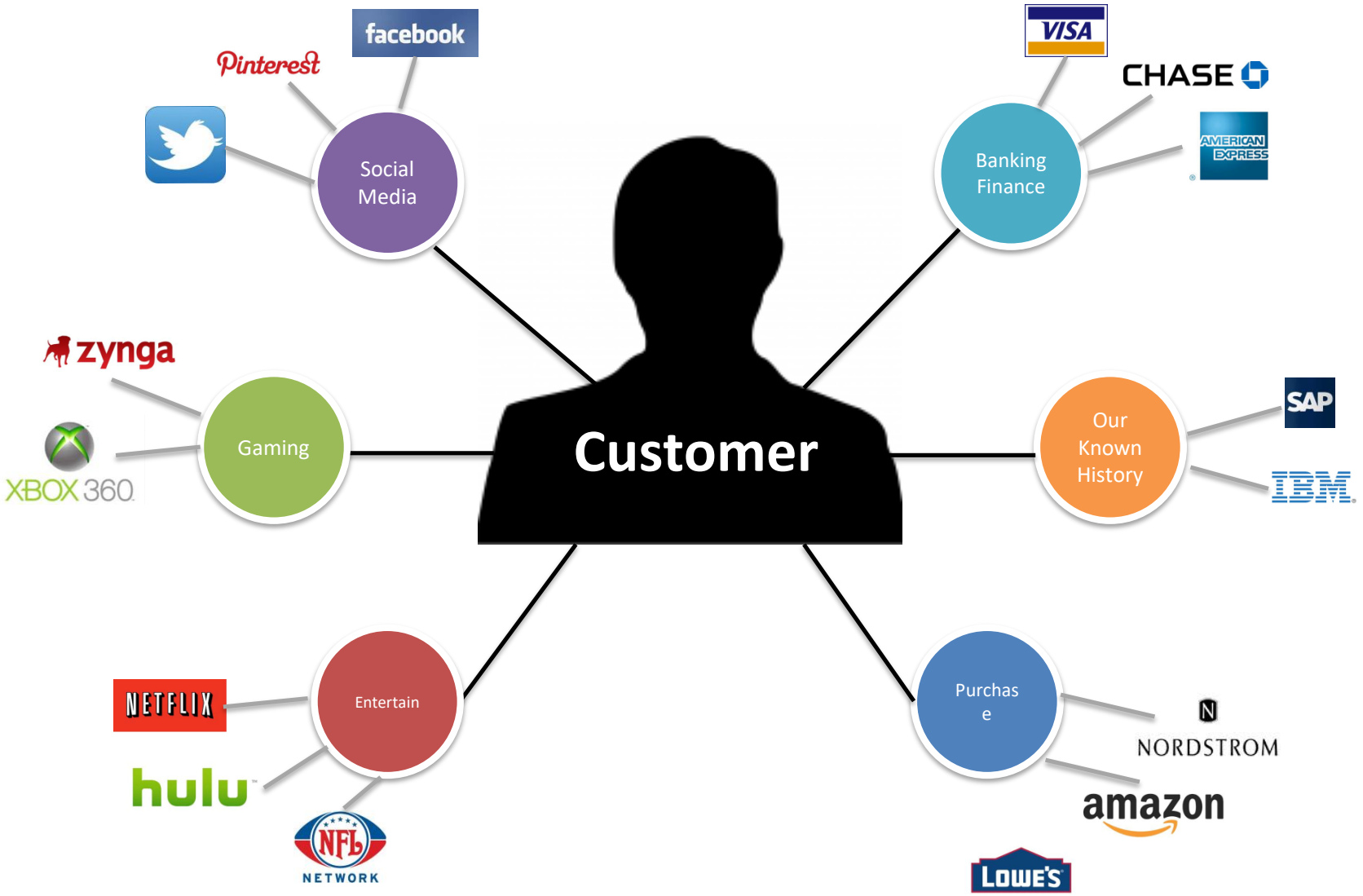
[1] For detailed explication of metrics, see appendix in McKinsey Global Institute full report *Big data: The next frontier for innovation, competition, and productivity*, available free of charge online at mckinsey.com/mgi.

Source: US Bureau of Labor Statistics; McKinsey Global Institute analysis

Real-Time Analytics/Decision Requirement



A Single View to the Customer



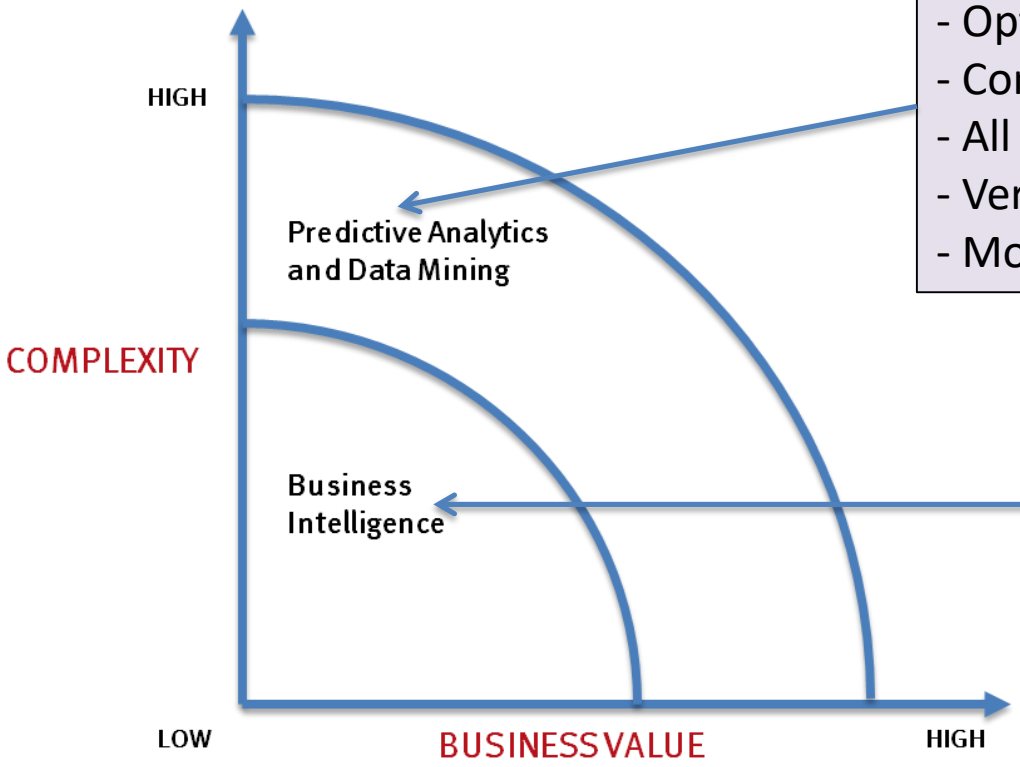
The Model Has Changed…

- **The Model of Generating/Consuming Data has Changed**

Old Model: Few companies are generating data, all others are consuming data

New Model: all of us are generating data, and all of us are consuming data

What's driving Big Data



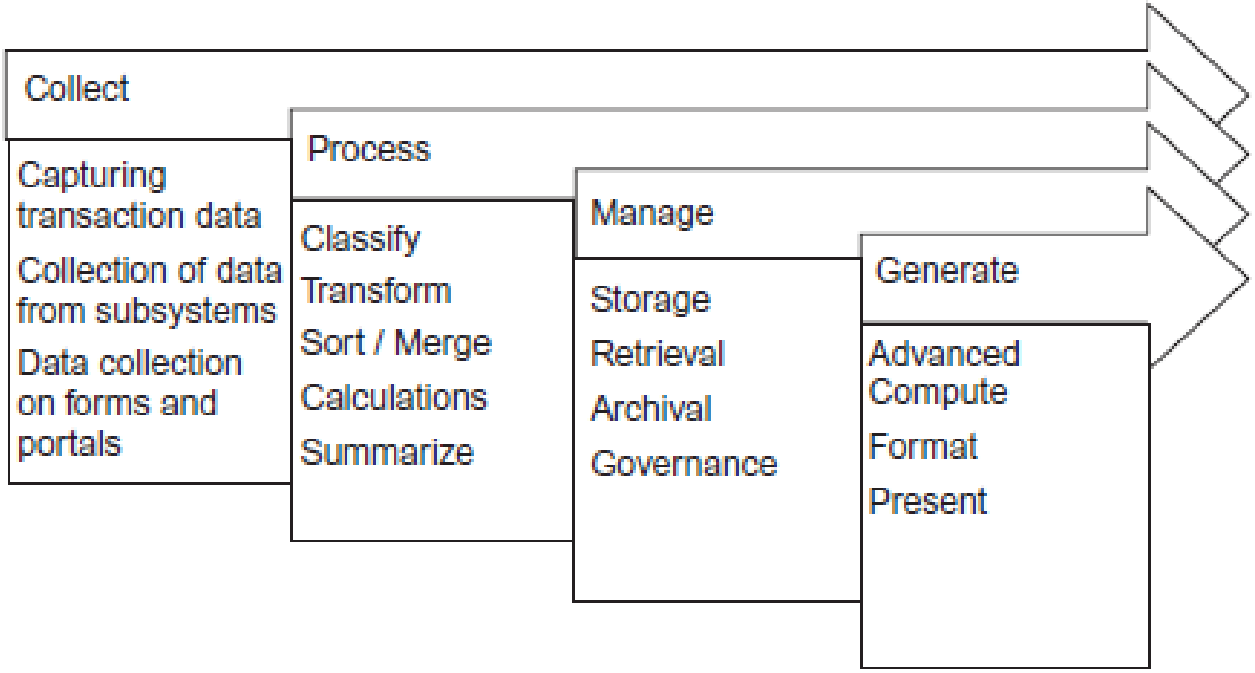
- Optimizations and predictive analytics
- Complex statistical analysis
- All types of data, and many sources
- Very large datasets
- More of a real-time

- Ad-hoc querying and reporting
- Data mining techniques
- Structured data, typical sources
- Small to mid-size datasets

Data Processing

- Data processing can be defined as the collection, processing, and management of data resulting in information generation to end consumers.

Data Processing Life Cycle



Data processing techniques

- There are two fundamental styles of data processing that have been accepted :
- *Centralized processing*.
- well suited for small organizations with one location of service.
- requires minimal resources
- very successful when the collection and consumption of data occurs at the same location.

- *Distributed processing. In this architecture data and its processing are distributed across* geographies or data centers, and processing of data is localized with the federation of the results into a centralized storage.
- Distributed architectures evolved to overcome the limitations of the

  There are several architectures of distributed processing:
  - *Client–server*
  - *Three-tier architecture*
  - *N-tier architecture*
  - *Cluster Architecture*
  - *Peer to peer architecture*

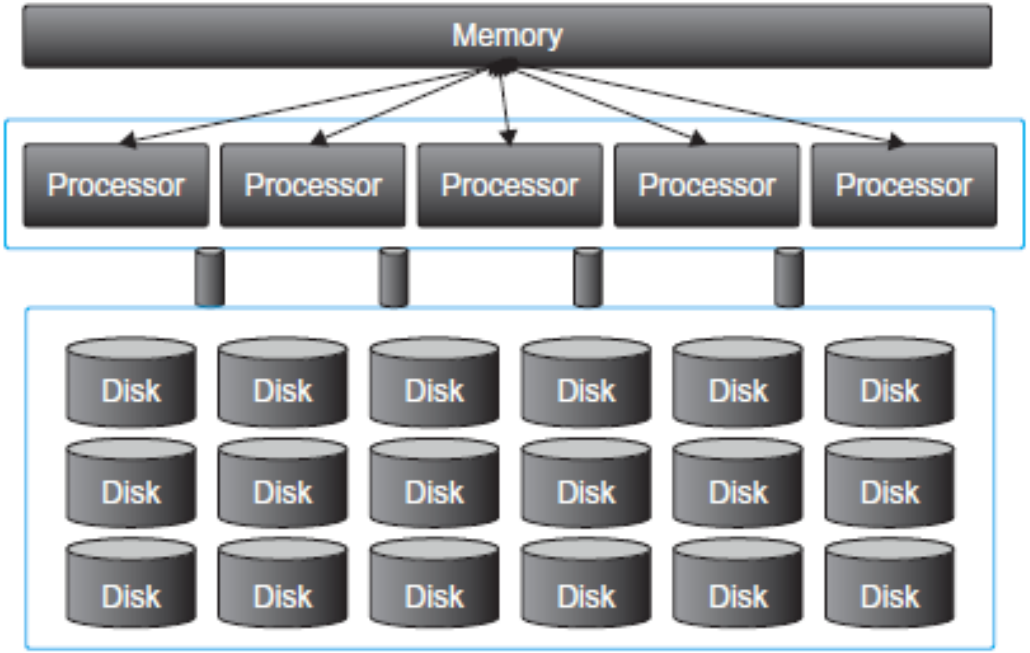
- Distributed processing has a lot of advantages and disadvantages.
- Advantages:
  - Scalability of systems and resources can be achieved based on isolated needs.
  - Processing and management of information can be architected based on desired unit of operation.
  - Parallel processing of data reducing time latencies.
- Disadvantages:
  - Data redundancy
  - Process redundancy
  - Resource overhead
  - Volumes

Initially Developed Architectures

- Shared Everything Architecture
  - Limited Scalability
- Shared nothing Architecture
  - Unlimited Scalability

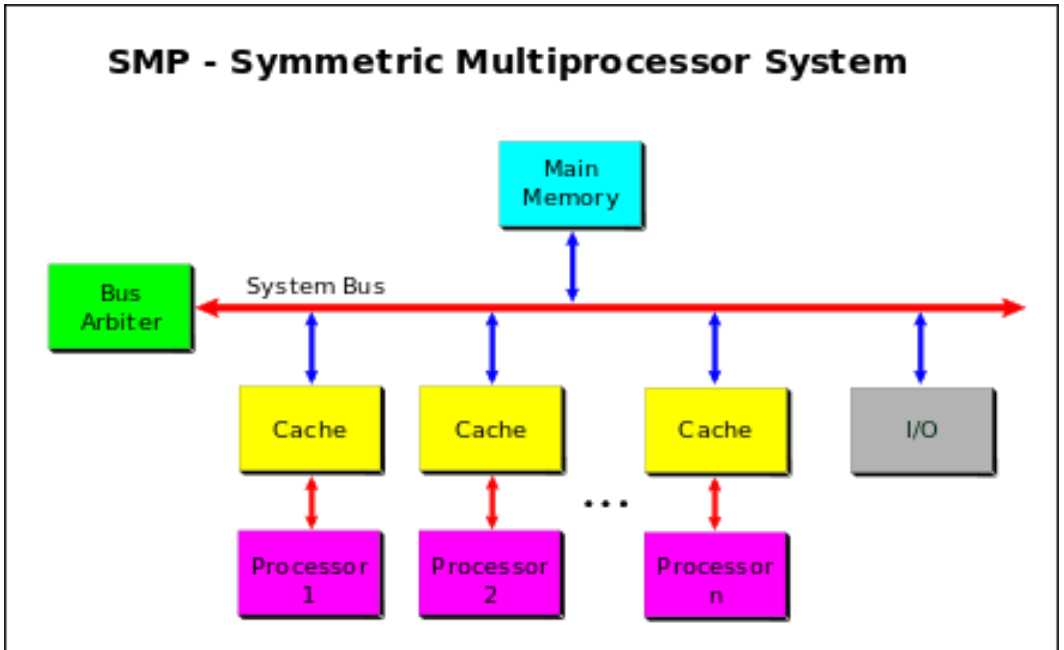
Shared everything architecture

- Shared-everything architecture refers to system architecture where all resources are shared including storage, memory, and the processer



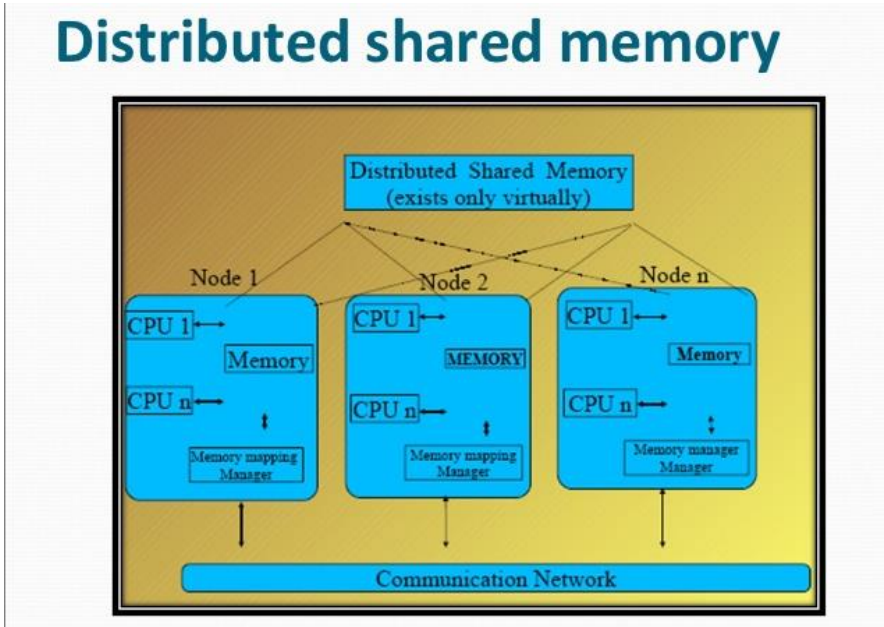
Shared everything architecture

- Limited Scalability
- Two variations
  - Symmetric multiprocessing (SMP)
  - Distributed Shared Memory(DSM)



Shared everything architecture

- Limited Scalability
- Two variations
  - Symmetric multiprocessing (SMP)
  - Distributed Shared Memory(DSM)



Shared everything architecture

- In the SMP architecture, all the processors share a single pool of memory for read–write access concurrently and uniformly without latency.
- Sometimes this is referred to as uniform memory access (UMA) architecture.
- This architecture focuses on maximizing resource utilization
- The drawback of SMP architecture is
  - when multiple processors are present and share a single system bus, which results in choking of the bandwidth for simultaneous memory access
  - therefore, the scalability of such system is very limited.

Shared everything architecture

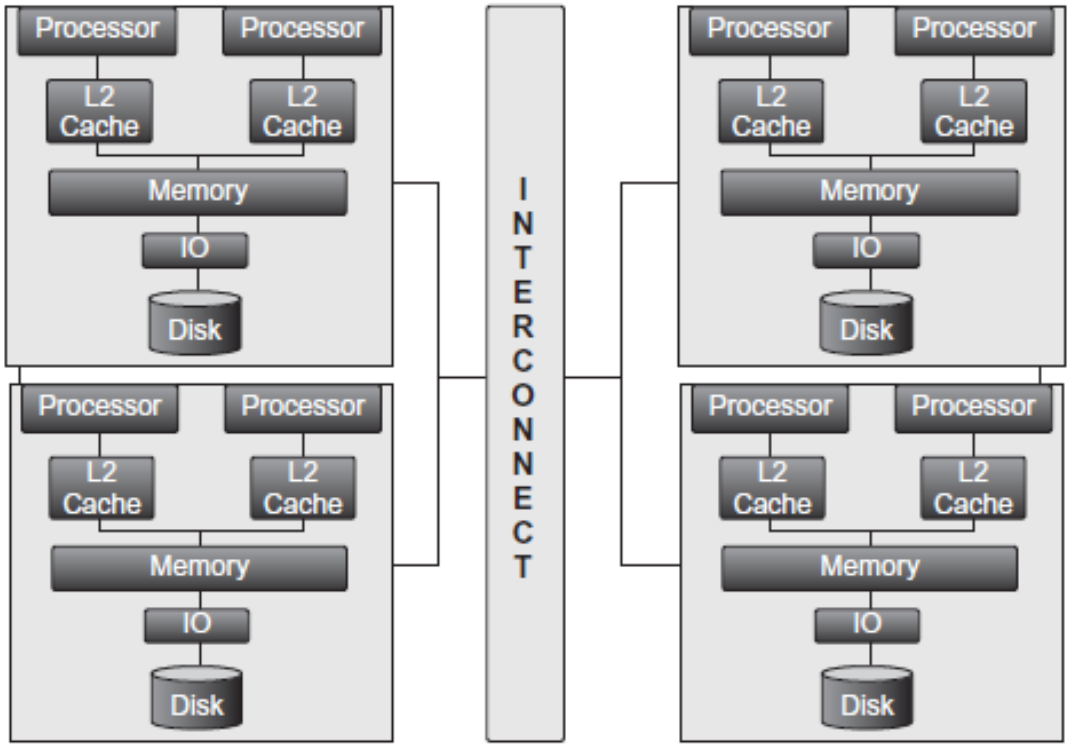
- The DSM architecture addresses the scalability problem by providing multiple pools of memory for processors to use.
- In the DSM architecture, the latency to access memory depends on the relative distances of the processors and their dedicated memory pools.
- This architecture focuses on maximizing performance.

Shared everything architecture

- Both SMP and DSM architectures have been deployed where transactional data is small.
- Data warehouses have been deployed on this architecture.

Shared –nothing Architecture

- Shared-nothing architecture is a distributed computing architecture where multiple systems (called nodes) are networked to form a scalable system



Data Warehouse

- Data warehouse is a subject-oriented, nonvolatile, integrated, time-variant collection of data in support of management's decision.
- It is a blend of technologies and components which allows the strategic use of data.
- There are two parts to the data warehouse
  - design
  - architecture.

- Design :
  - This part deals with the data architecture and processing
  - it answers the data encapsulation from the user.
- Architecture
  - This part deals with the database architecture, infrastructure, and system architecture

- Business requirements analysis
  - In this step the key business requirements are gathered from business users and sponsors.
  - outline the needs for data from an analysis perspective.
- Data analysis
  - analyzed for data types, business rules and quality

- Data modeling
  - Models are converted to relational Models
- Data movement
  - the process of extracting, loading, and transformation of data is designed,
  - developed, and implemented.
- Data Quality
  - Data quality issues are resolved
- Data transformation
  - Deals with Data Summarization, aggregation and encryption
- Data presentation
  - includes views and other semantic layers are created for user access.

- Gaining competitive advantage
- ● Reducing operational and financial risks
- ● Increasing revenue
- ● Optimizing core business efficiencies
- ● Analyzing and predicting trends and behaviors
- ● Managing brand presence, channels, and reputation
- ● Managing customer expectations proactively

Reengineering the DW

- Replatforming
  - Modify to a new platform including all hardware and infrastructure.
- Platform Engineering
  - Modify pieces and parts of infrastructure and gain the desired scalability and performance
- Data Engineering
  - Data Structures are reengineered
  - New additions are made to data Models

Reengineering the DW

- Replatforming
  - replatform the data warehouse to a new platform including all hardware and infrastructure.
  - depending on the requirement of the organization, data warehouse appliances, commodity platforms, tiered storage, private cloud, and in-memory technologies can be deployed.

Benefits

- provides an opportunity to move the data warehouse to a scalable and reliable platform.
- The underlying infrastructure and the associated application software layers can be architected to provide security, lower maintenance, and increase reliability.
- will provide us an opportunity to optimize the application and database code.
- will provide some additional opportunities to use new functionality.
- makes it possible to re-architect things in a different/better way, which is almost impossible to do in an existing setup.

Disadvantages

- takes a long cycle time to complete, leading to disruption of business activities
- not be feasible for certain aspects of data processing i.e. complex calculations that need to be rewritten if they cannot be directly supported by the functionality of the new platform.

Platform engineering

- modify pieces and parts of the infrastructure to get great gains in scalability and performance

- Reduce the cost of the data warehouse.
- Increase efficiencies of processing.
- Simplify the complexities in the acquisition, processing, and delivery of data.
- Reduce redundancies.
- Minimize customization.
- Isolate complexity into manageable modular environments

- Platform reengineering can be done at multiple layers
  - Storage Level
  - Server Reengineering
  - Network Reengineering
  - *Data warehouse appliances*
  - *Application server*

Data engineering

- new concept where the data structures are reengineered to create better performance
  - *Partitioning*
  - *Colocation*
  - *New data types*
  - *New database functions*

Big data learning approaches

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

Applications

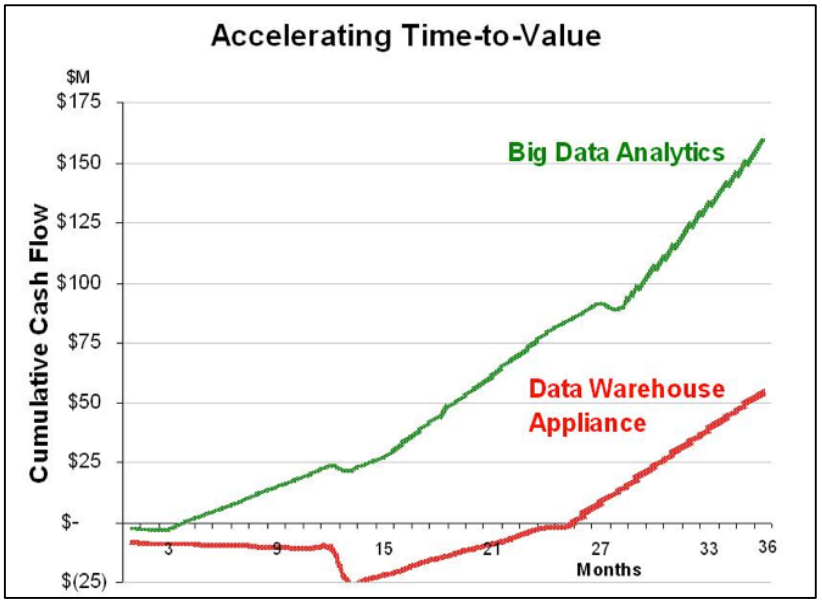
- Weather forecasting
- Healthcare
  - Medical Research
  - Personalized Treatment
  - Cost Reduction
  - Health Population
- Media and Entertainment
  - Targeted ads
  - Customer sentiment analysis
  - Recommendations
  - Customer Retentions

Applications

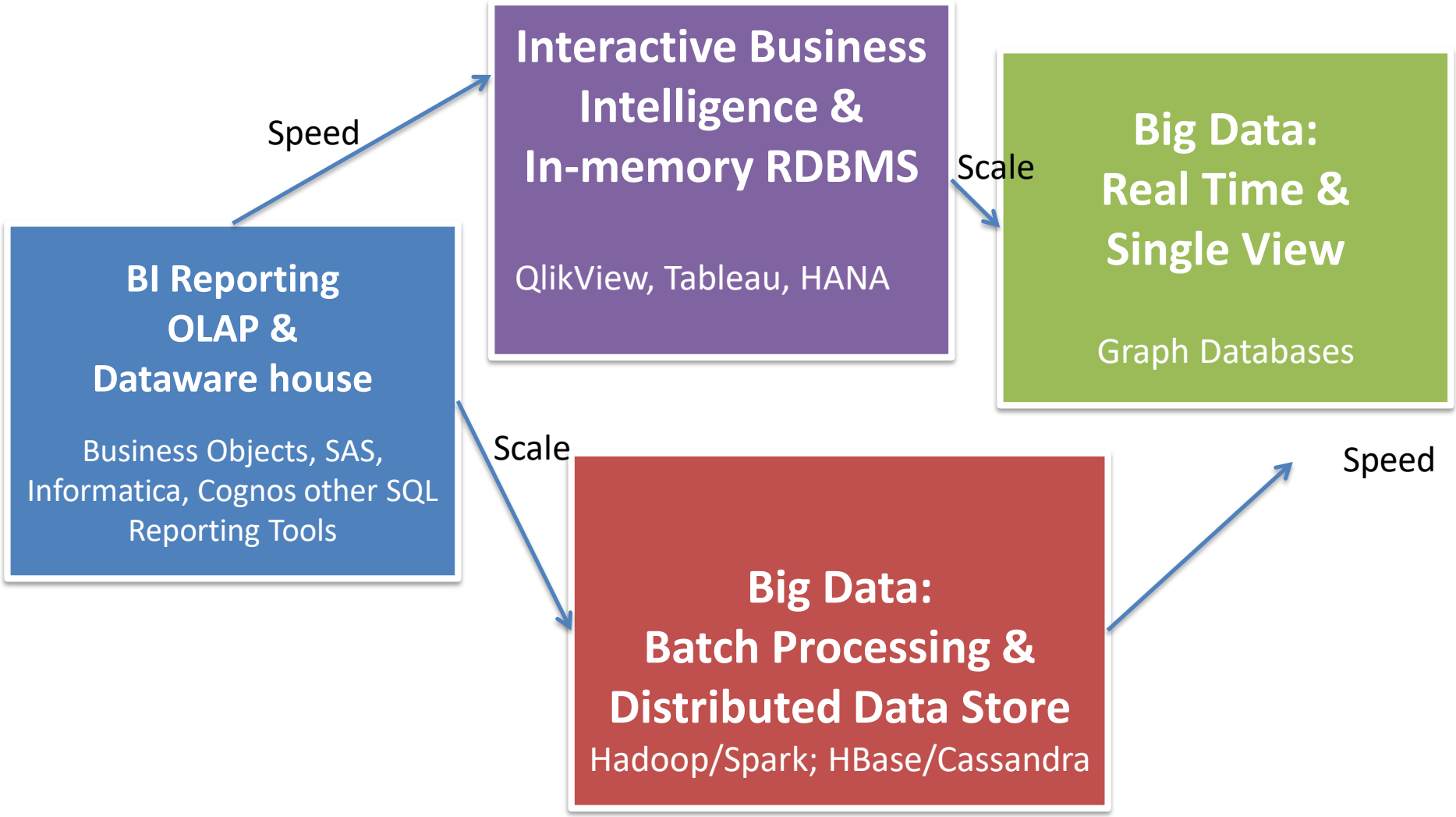
- Logistics
  - Flexible routing
  - Capacity Planning
  - Smart Warehouses
  - Customer Satisfaction
- Government and Law enforcement
- Travel and Tourism

Big Data Analytics

- Big data is more real-time in nature than traditional DW applications
- Traditional DW architectures are not well-suited for big data apps
- Shared nothing, massively parallel processing, scale out architectures are well-suited for big data apps



THE EVOLUTION OF BUSINESS INTELLIGENCE



Big Data Architecture

Big Data Architectures

A big data architecture is designed to handle the ingestion, processing, and analysis of data that is too large or complex for traditional database systems

Big data solutions typically involve one or more of the following types of workload:

- ***Batch processing*** *of big data sources at rest*
- ***Real-time processing*** *of big data in motion*
- ***Interactive exploration*** *of big data*
- ***Predictive analytics*** *and* ***machine learning***

Big data architectures when you need to:

- *Store and process **data in volumes too large** for a traditional database*
- *Transform **unstructured data** for analysis and reporting*
- *Capture, process, and analyze **unbounded streams of data** in real time, or **with low latency***"

The most frequently used big data architecture is the **Lambda Architecture**

The lambda architecture was proposed by Nathan Marz in 2011

***Hadoop**, for example, can **parallelize large-scale batch computations** on very large amounts of data, but the computations have **high latency**. You don't use Hadoop for anything where you need low-latency results.*

NoSQL databases *like Cassandra achieve their* ***scalability*** *by offering you a* ***much more limited data model*** *than you're used to with something like SQL. Squeezing your application into these limited data models can be very complex. And because the* ***databases are mutable****, they're* ***not human-fault tolerant****.*

Source: Databricks

> "***Lambda architecture*** *is a way of* ***processing*** *massive quantities of data (i.e. "****Big Data****") that provides access to* ***batch-processing*** *and* ***stream-processing*** *methods with a hybrid approach.*

Lambda architecture: Definition #3

Source: Wikipedia

"***Lambda architecture*** *is a* ***data-processing architecture*** *designed to* ***handle massive*** *quantities of* ***data*** *by taking advantage of both* ***batch*** *and* ***stream-processing*** *methods.*

Lambda architecture: Requirements

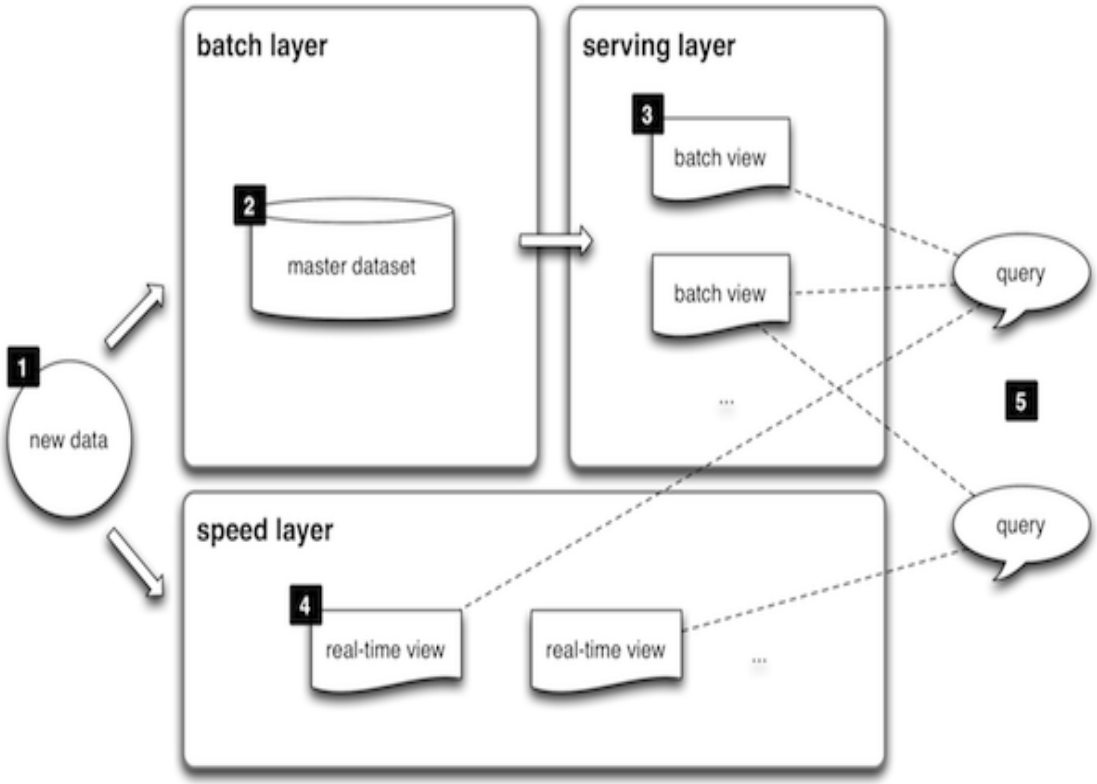
Fault-tolerant against both hardware failures and human errors

Support variety of use cases that include low latency querying as well as updates

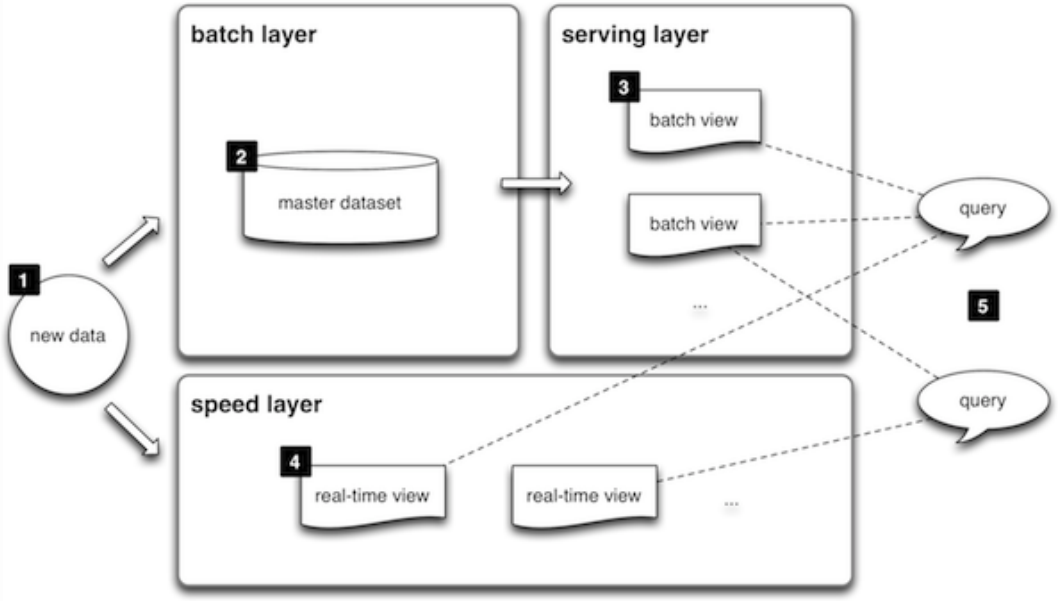
Linear **scale-out capabilities**

Extensible, so that the system is manageable and can accommodate newer features easily

Lambda architecture

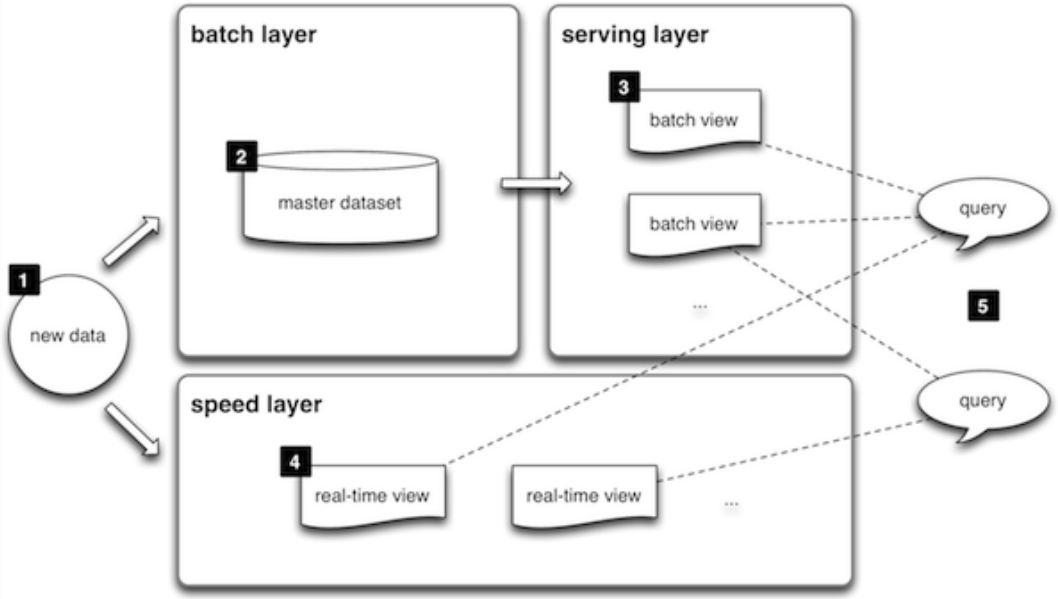


Lambda architecture



1. **All data** entering the system is **dispatched to both** the **batch layer** and the **speed layer** for processing

Lambda architecture



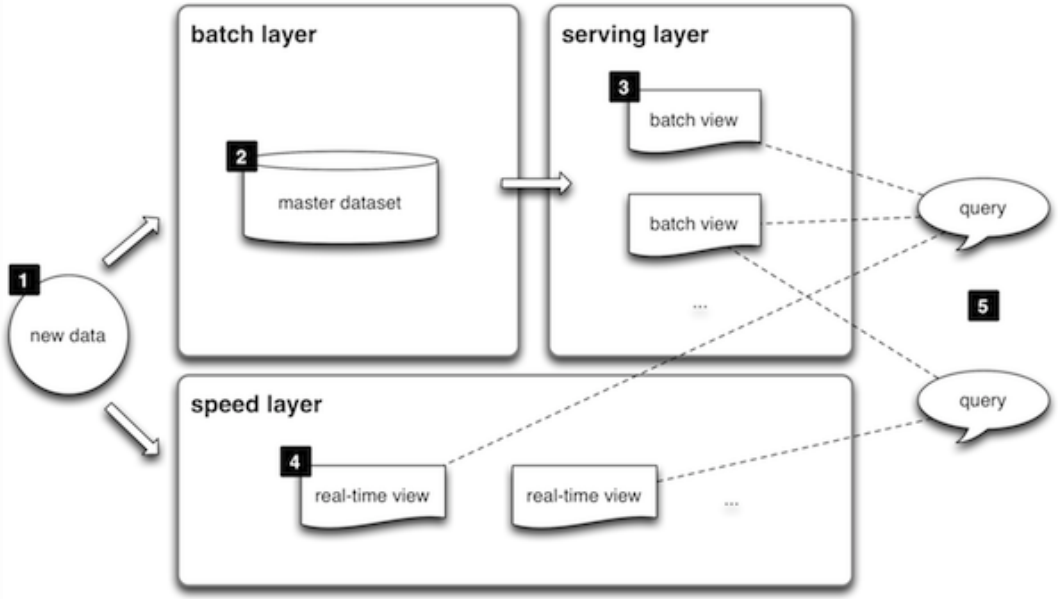
2. The **batch layer** has two functions:

   (i)managing the **master dataset** (an **immutable**, **append-only** set of **raw data**), and

   (ii)to **pre-compute** the **batch views**

From http://lambda-architecture.net/

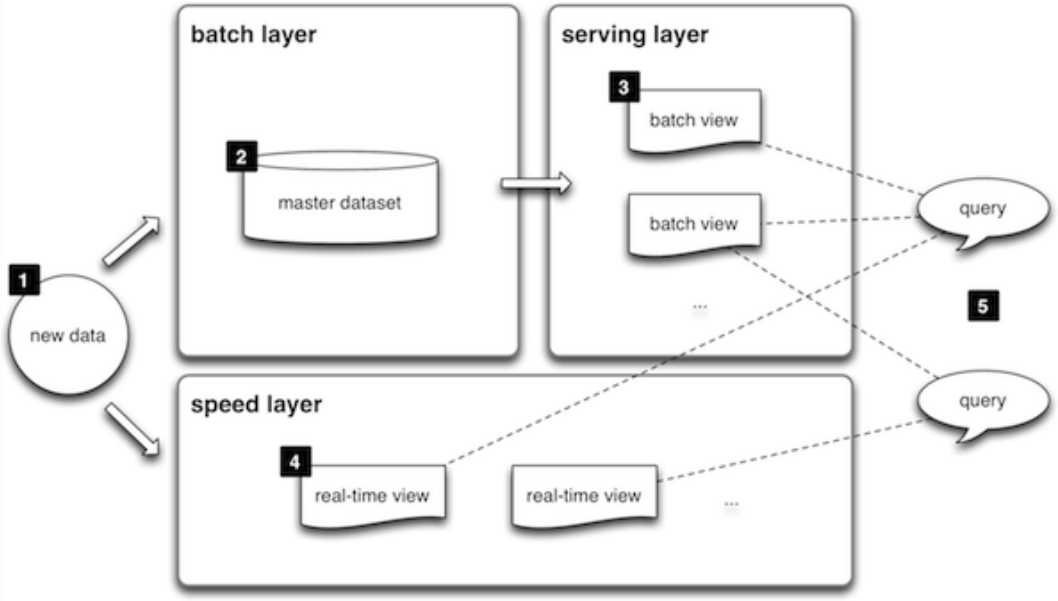
Lambda architecture



3. The **serving layer indexes** the **batch views** so that they can be **queried in low-latency**, ad-hoc way

From http://lambda-architecture.net/

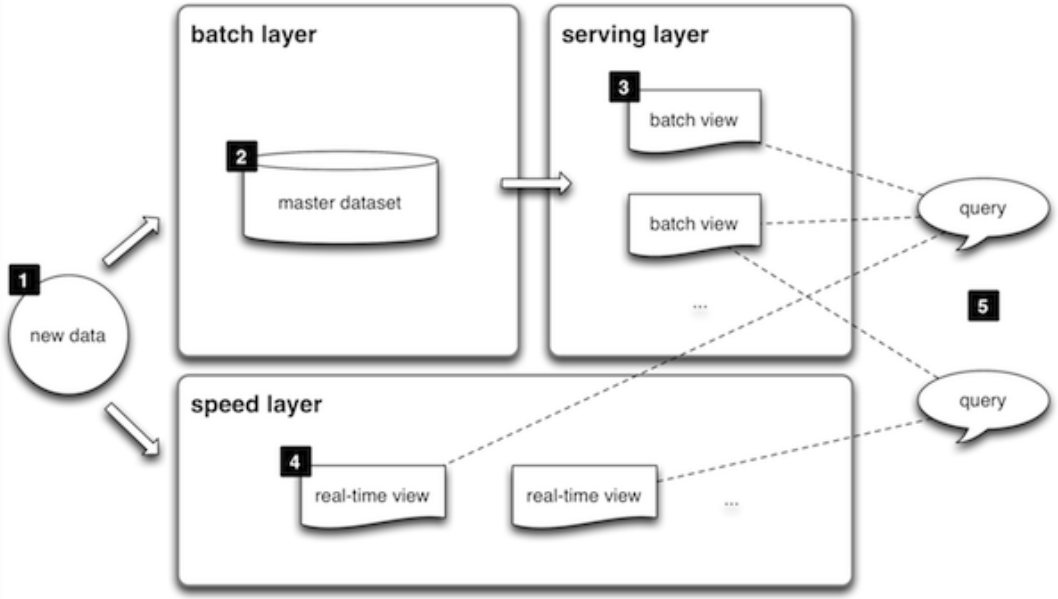
Lambda architecture



4. The **speed layer** compensates for the high latency of updates to the serving layer and **deals with recent data only**

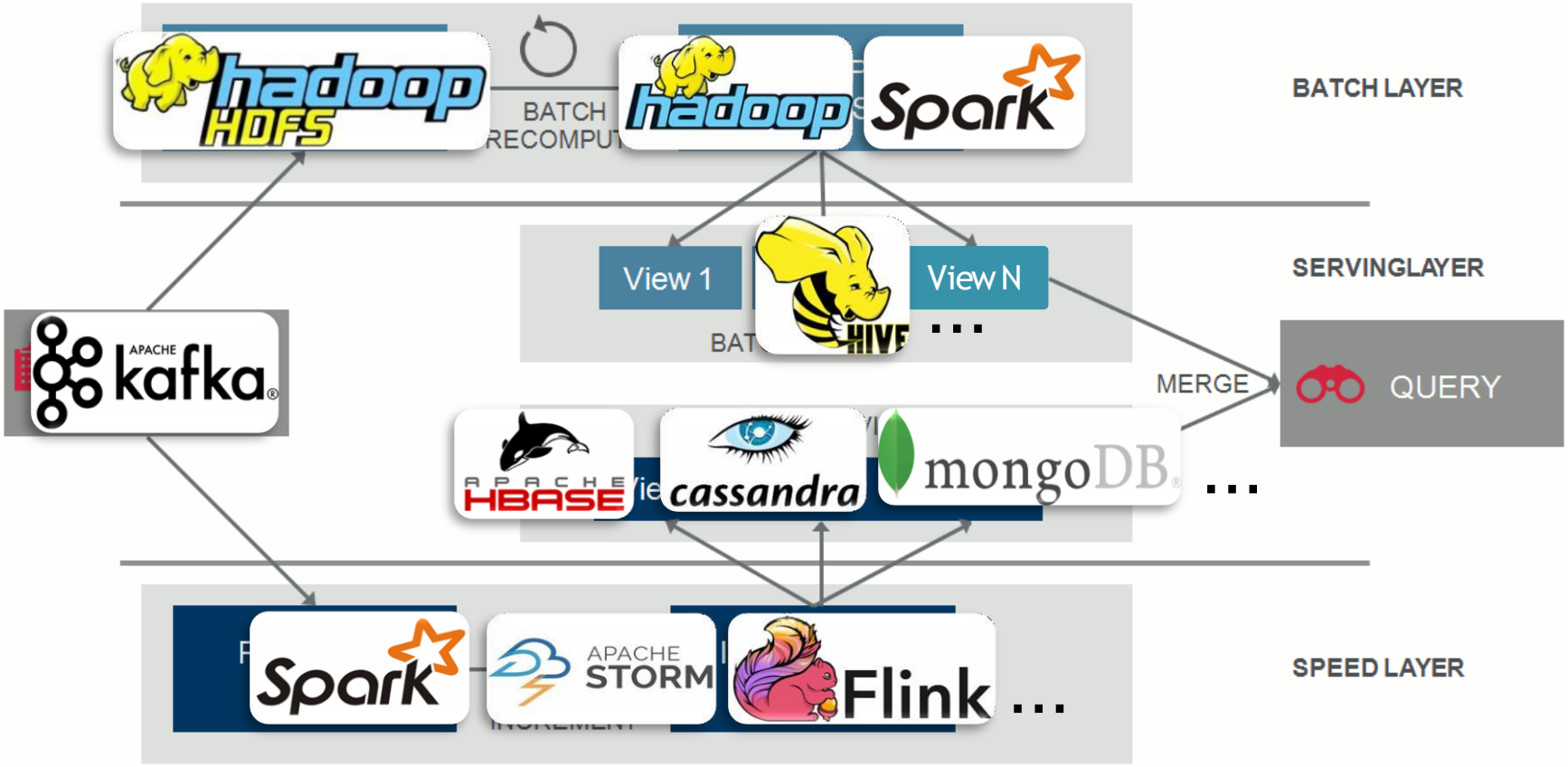
From http://lambda-architecture.net/

Lambda architecture



5. Any incoming **query** can be **answered by merging** results from **batch views** and **real-time views**

From http://lambda-architecture.net/


hadoop
HDFS
BATCH
RECOMPUT
hadoop
Spark
BATCH LAYER
View 1
View N
HIVE
...
SERVINGLAYER
APACHE
kafka
MERGE
QUERY
APACHE
HBASE
cassandra
mongoDB
...
Spark
APACHE
STORM
Flink
...
SPEED LAYER

Thank you