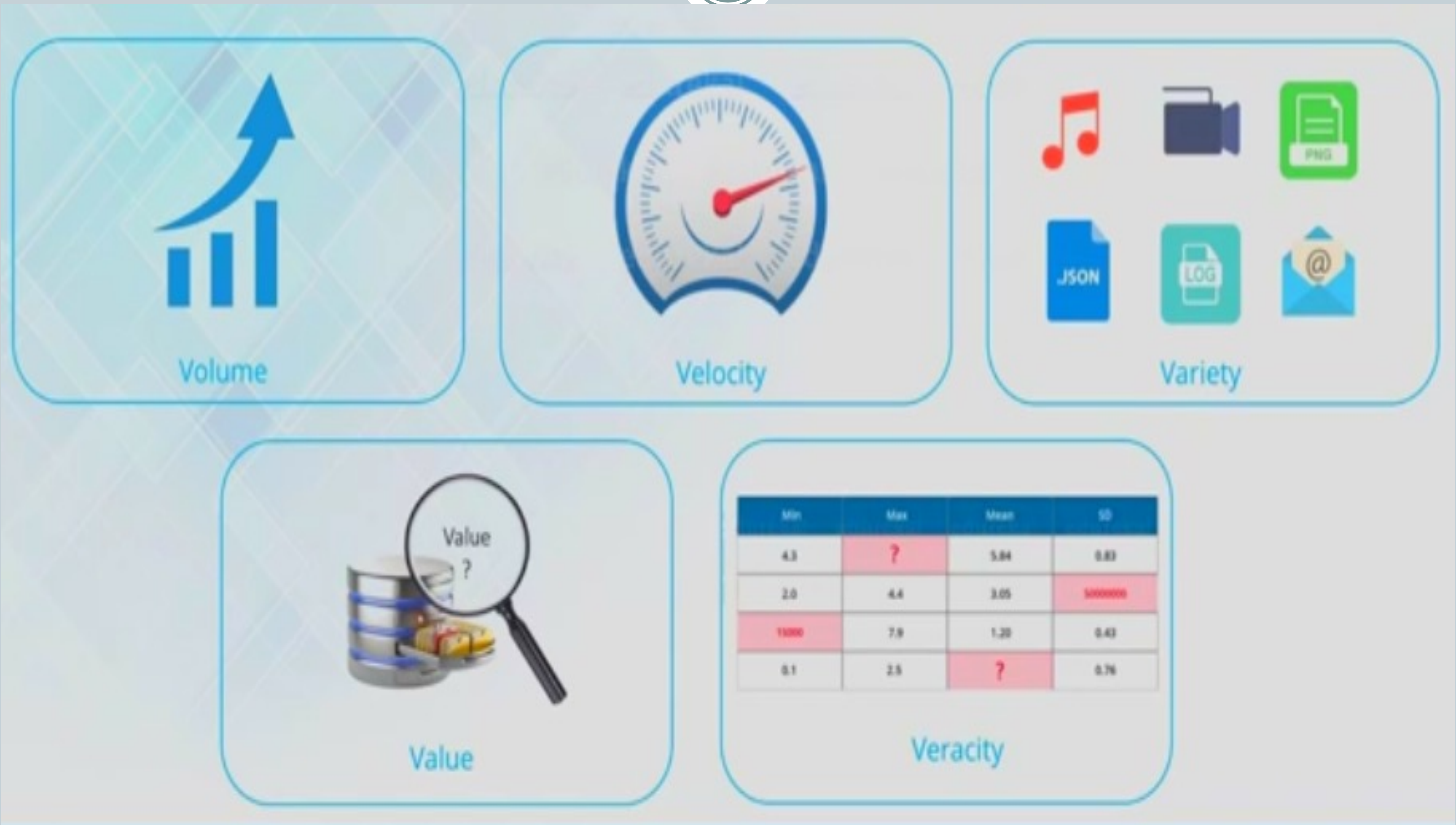
# Introduction to Big Data

# Characteristics of Big Data

# Characteristics of Big Data( 5 Vs of Big data )



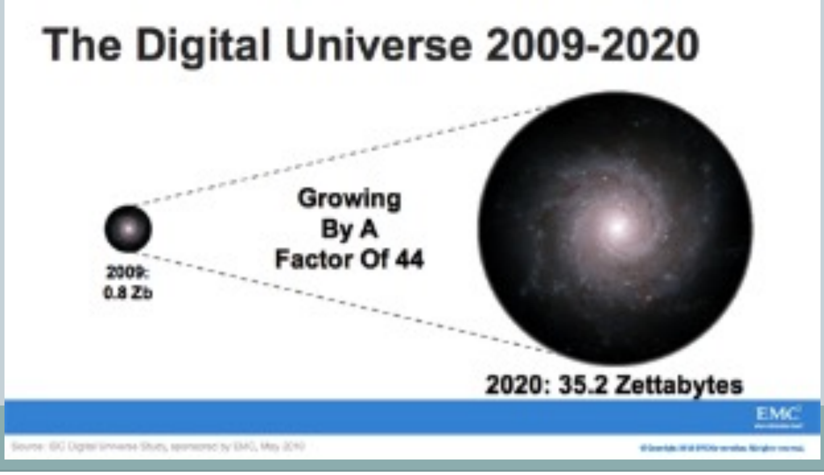
| Min | Max | Mean | SD |
|---|---|---|---|
| 4.3 | ? | 5.84 | 0.83 |
| 2.0 | 4.4 | 3.05 | 50000000 |
| 15000 | 7.9 | 1.20 | 0.43 |
| 0.1 | 2.5 | ? | 0.76 |

# Characteristics of Big Data( 5 Vs of Big data )

- 1st V-volume

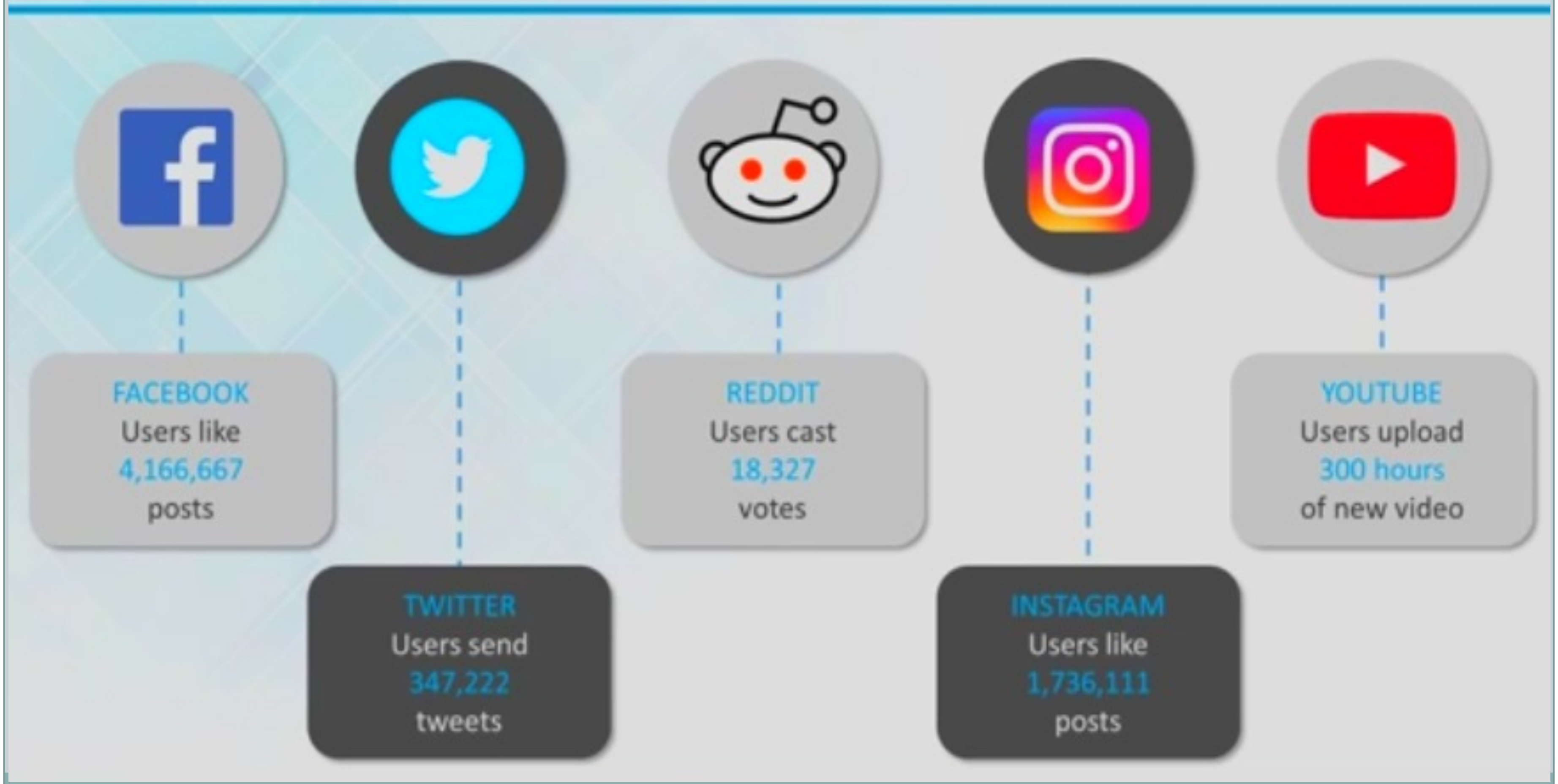
  - Data Volume

- 44x increase from 2009 to 2020 From 0.8 zettabytes to 35zb
- Data volume is increasing exponentially


The Digital Universe 2009-2020
2009:
0.8 Zb
Growing
By A
Factor Of 44
2020: 35.2 Zettabytes
EMC
Source: IDC Digital Universe Study, sponsored by EMC, May 2010

# Characteristics of Big Data( 5 Vs of Big data )

- 2nd V-velocity: **Data is being generated at every minute**



FACEBOOK
Users like
4,166,667
posts

TWITTER
Users send
347,222
tweets

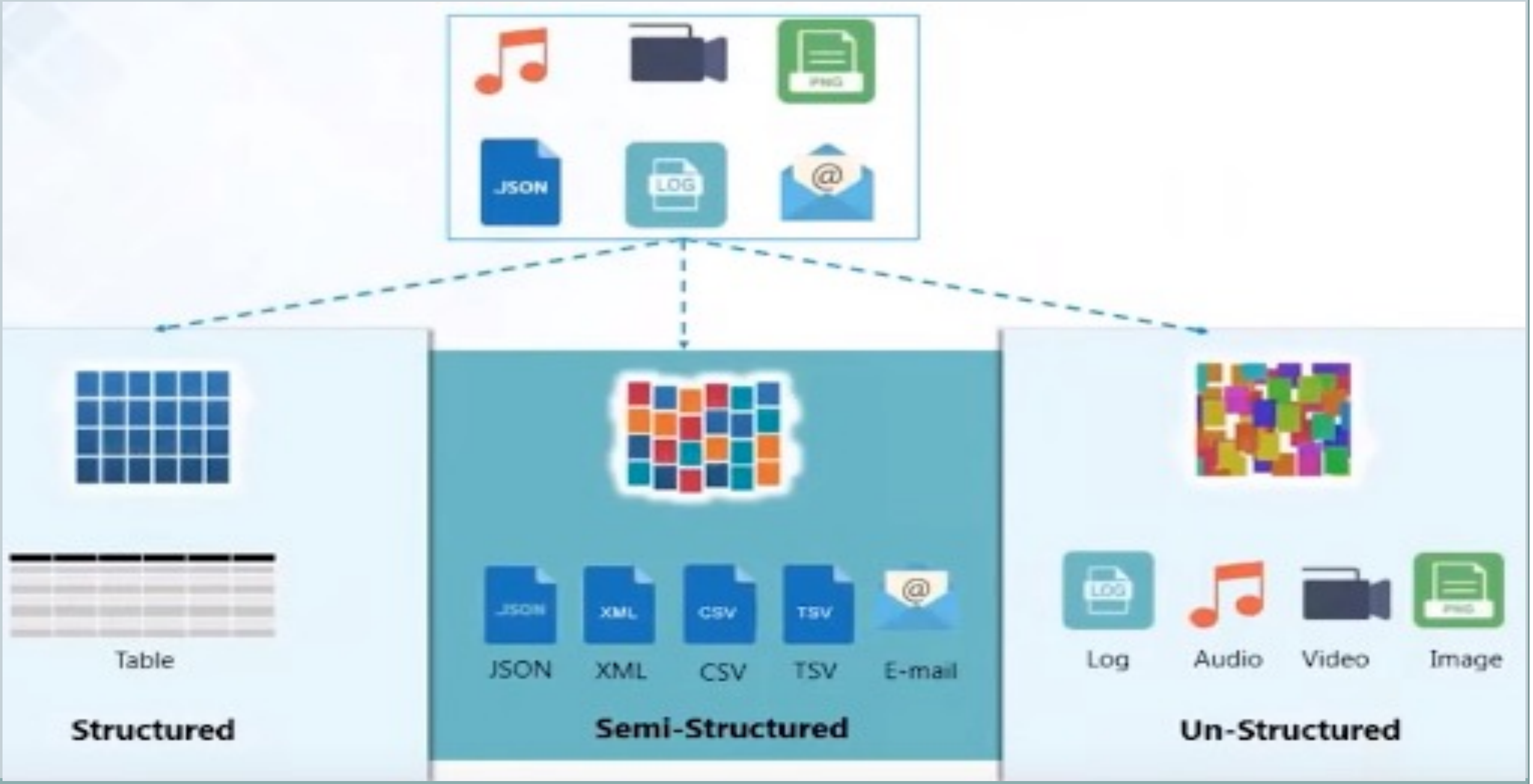
REDDIT
Users cast
18,327
votes

INSTAGRAM
Users like
1,736,111
posts

YOUTUBE
Users upload
300 hours
of new video

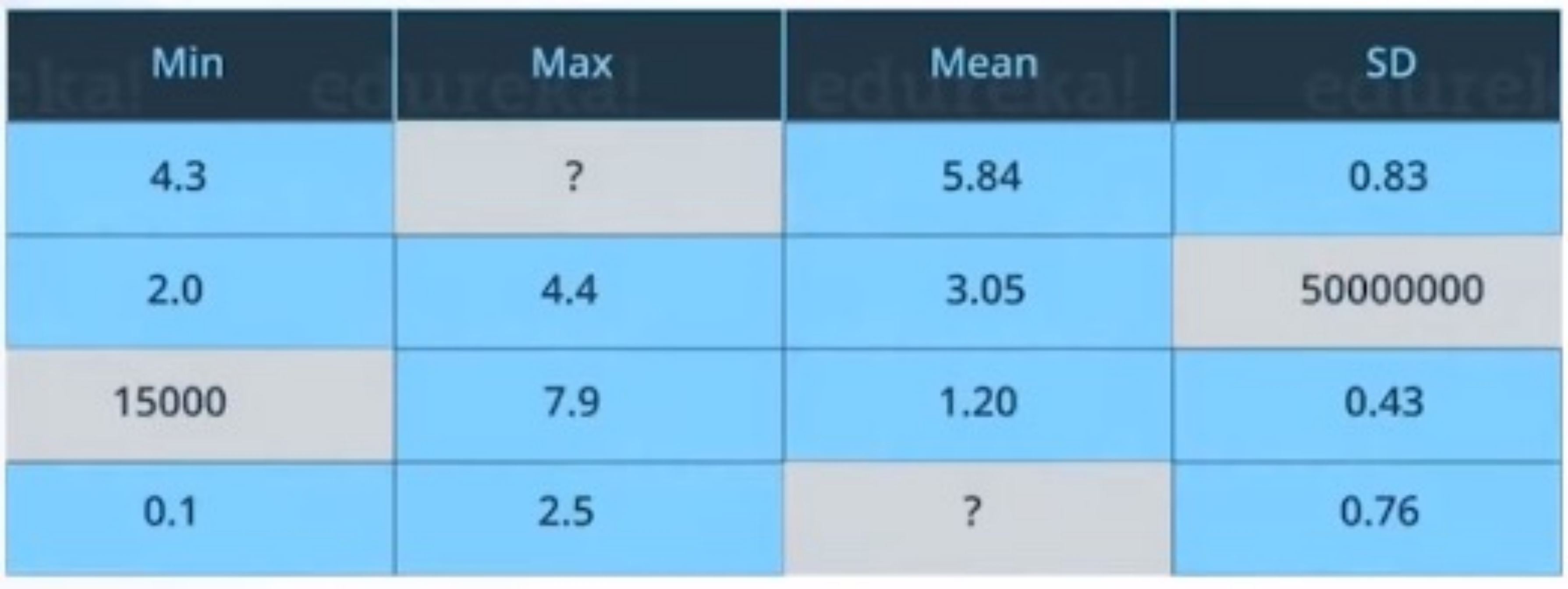
# Characteristics of Big Data( 5 Vs of Big data )

- 3rd V-Variety: **different kinds of data generated from various sources**



# Characteristics of Big Data( 5 Vs of Big data )

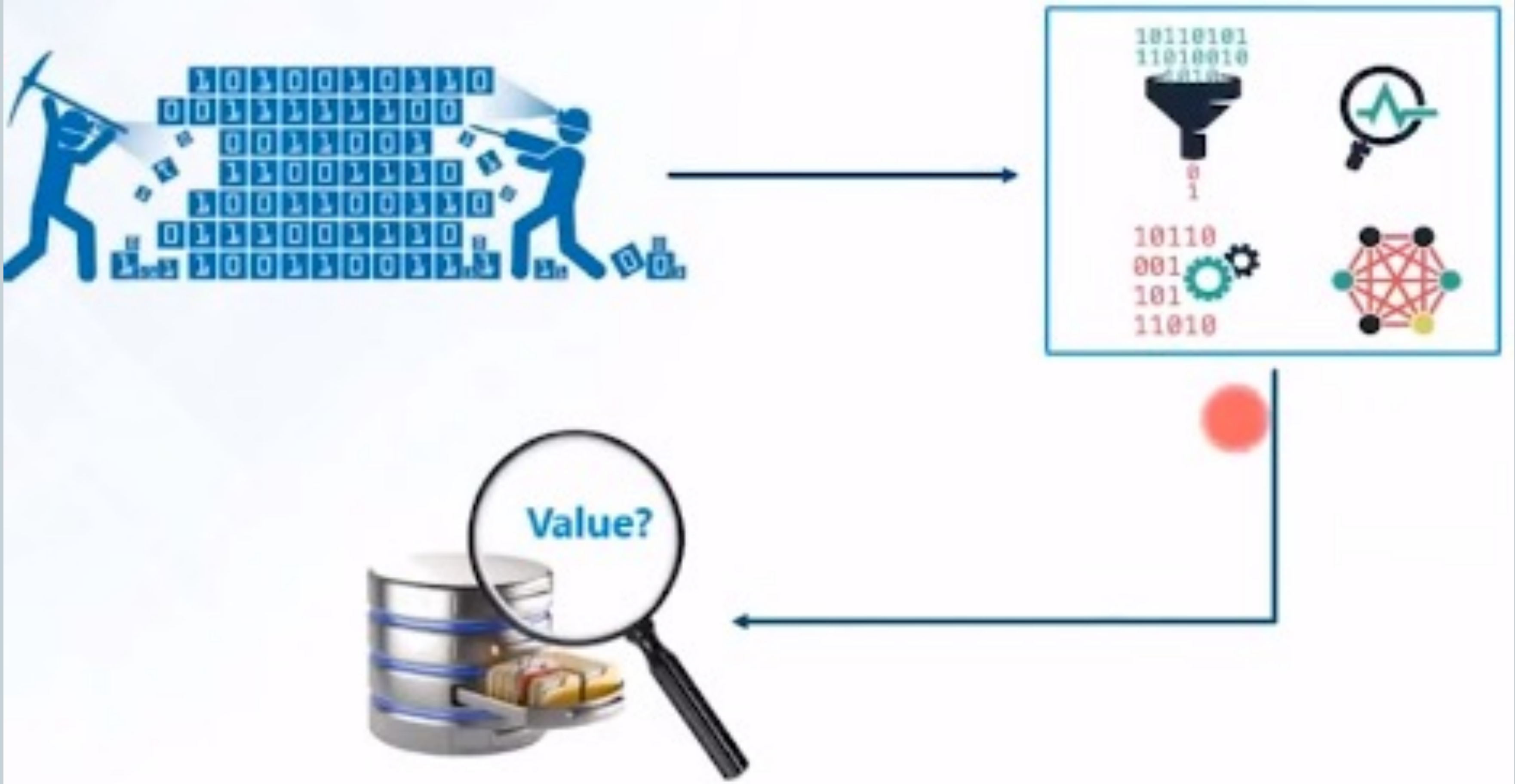
- 4th V - Veracity: **uncertainties and inconsistencies in big data**



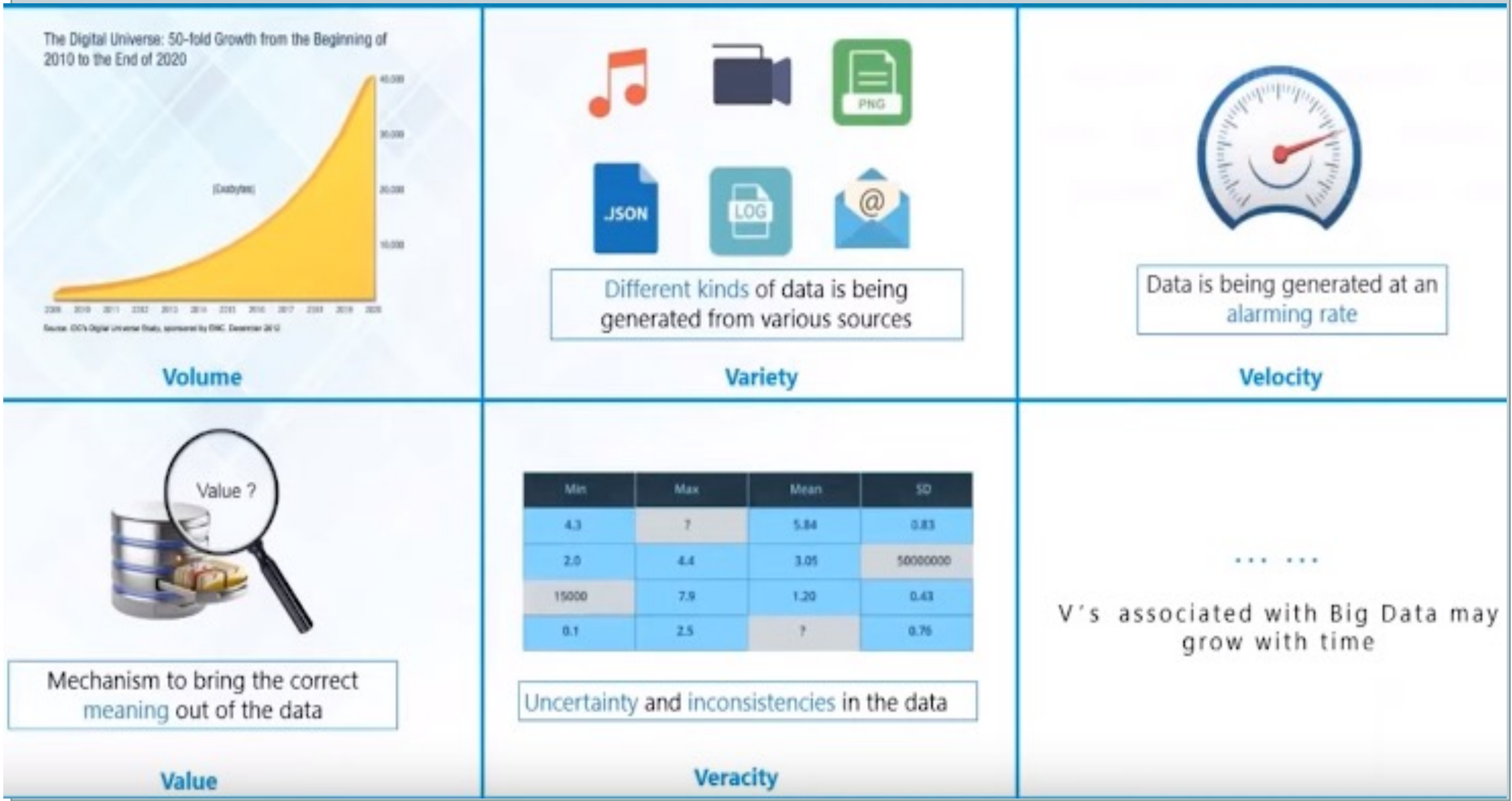
| Min | Max | Mean | SD |
|---|---|---|---|
| 4.3 | ? | 5.84 | 0.83 |
| 2.0 | 4.4 | 3.05 | 50000000 |
| 15000 | 7.9 | 1.20 | 0.43 |
| 0.1 | 2.5 | ? | 0.76 |

# Characteristics of Big Data( 5 Vs of Big data )

- $5^{th}$ V - Value: **Mechanism to bring correct meaning out of the data**



# Characteristics of Big Data( 5 Vs of Big data )



The Digital Universe: 50-fold Growth from the Beginning of 2010 to the End of 2020

Source: IDC's Digital Universe Study, sponsored by EMC, December 2012

**Volume**

Different kinds of data is being generated from various sources

**Variety**

Data is being generated at an alarming rate

**Velocity**

Value ?

Mechanism to bring the correct meaning out of the data

**Value**

| Min | Max | Mean | SD |
|---|---|---|---|
| 4.3 | ? | 5.84 | 0.83 |
| 2.0 | 4.4 | 3.05 | 50000000 |
| 15000 | 7.9 | 1.20 | 0.43 |
| 0.1 | 2.5 | ? | 0.76 |

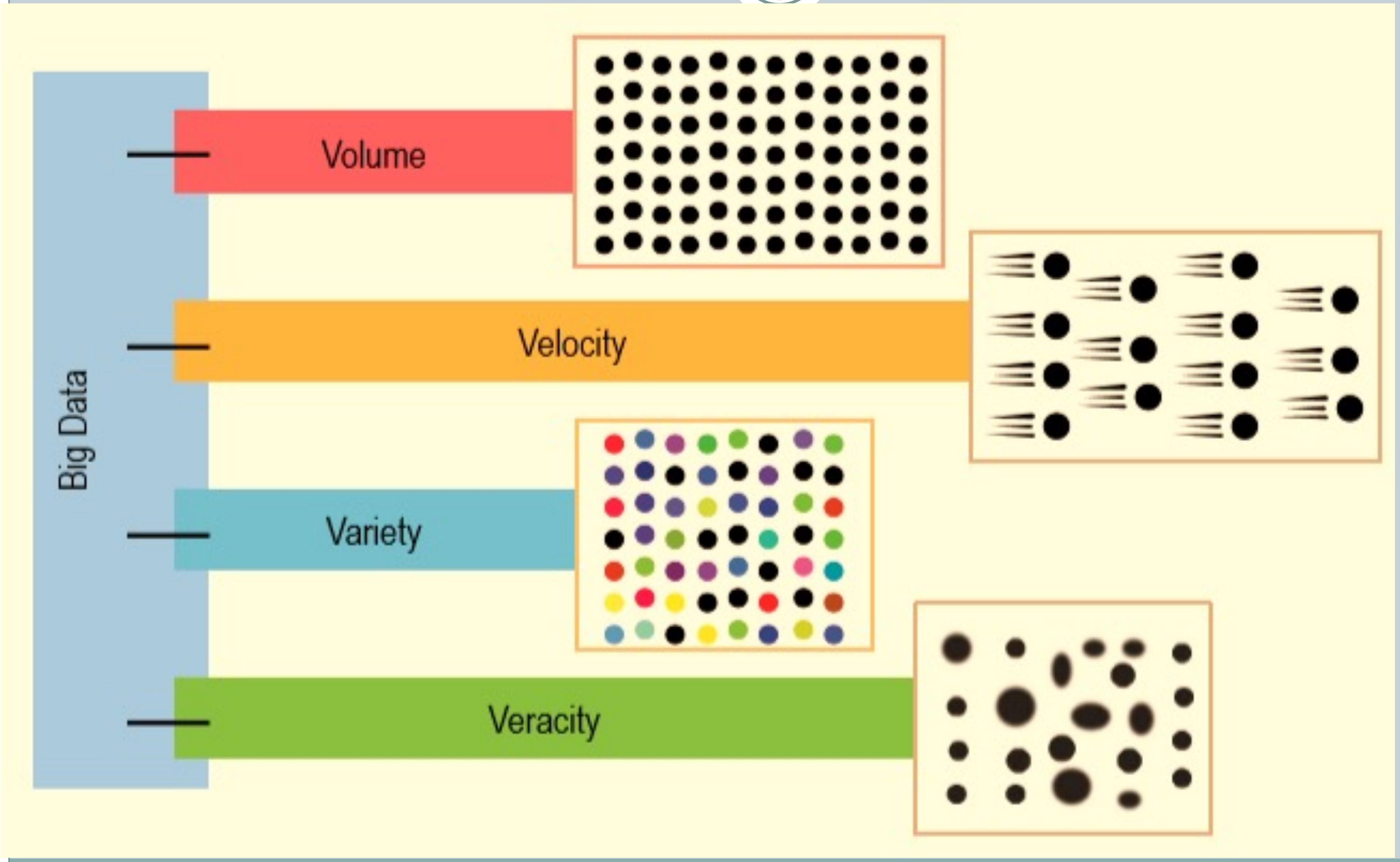
Uncertainty and inconsistencies in the data

**Veracity**

... ...

V's associated with Big Data may grow with time

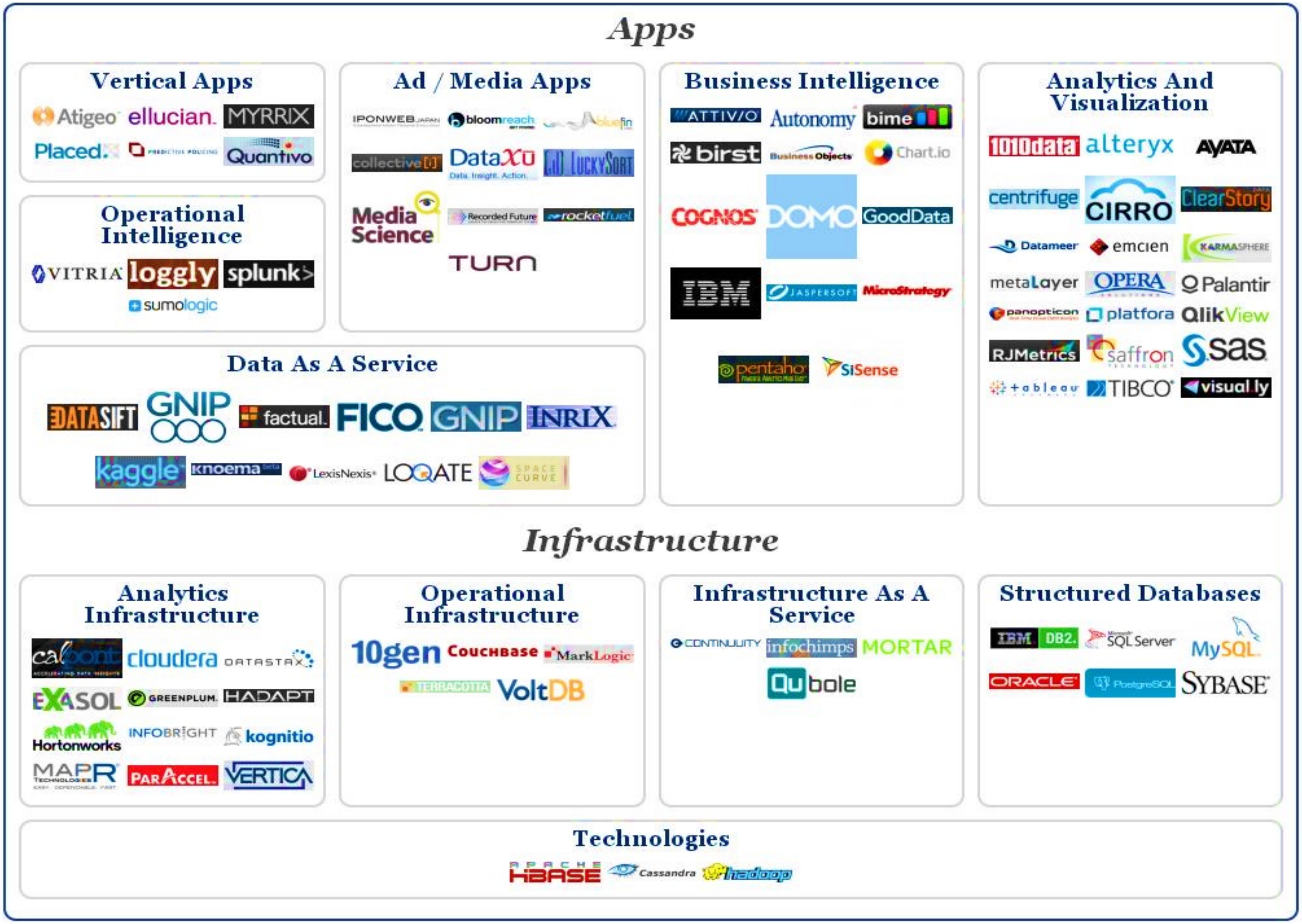
# Characteristics of Big Data( 5 Vs of Big data )



# Traditional DB vs Big Data

| Traditional data base/ data warehouse | Big Data |
|---|---|
| **Data** | **Data** |
| TB to PB | PB to ZB |
| Only structured | structured and unstructured |
| **Hardware** | **Hardware** |
| big central servers | computer clusters |
| Expensive | Cost effective |
| Hardware reliability | Unreliable HW |
| Limited scalability | Scales further |
| **Software** | **Software** |
| Centralized | Distributed |
| Schema based | Not schema based |
| Oracle/mysql/sql server | Hadoop |

# Big data tools



## Apps

### Vertical Apps

### Ad / Media Apps

### Business Intelligence

### Analytics And Visualization

### Operational Intelligence

### Data As A Service

## Infrastructure

### Analytics Infrastructure

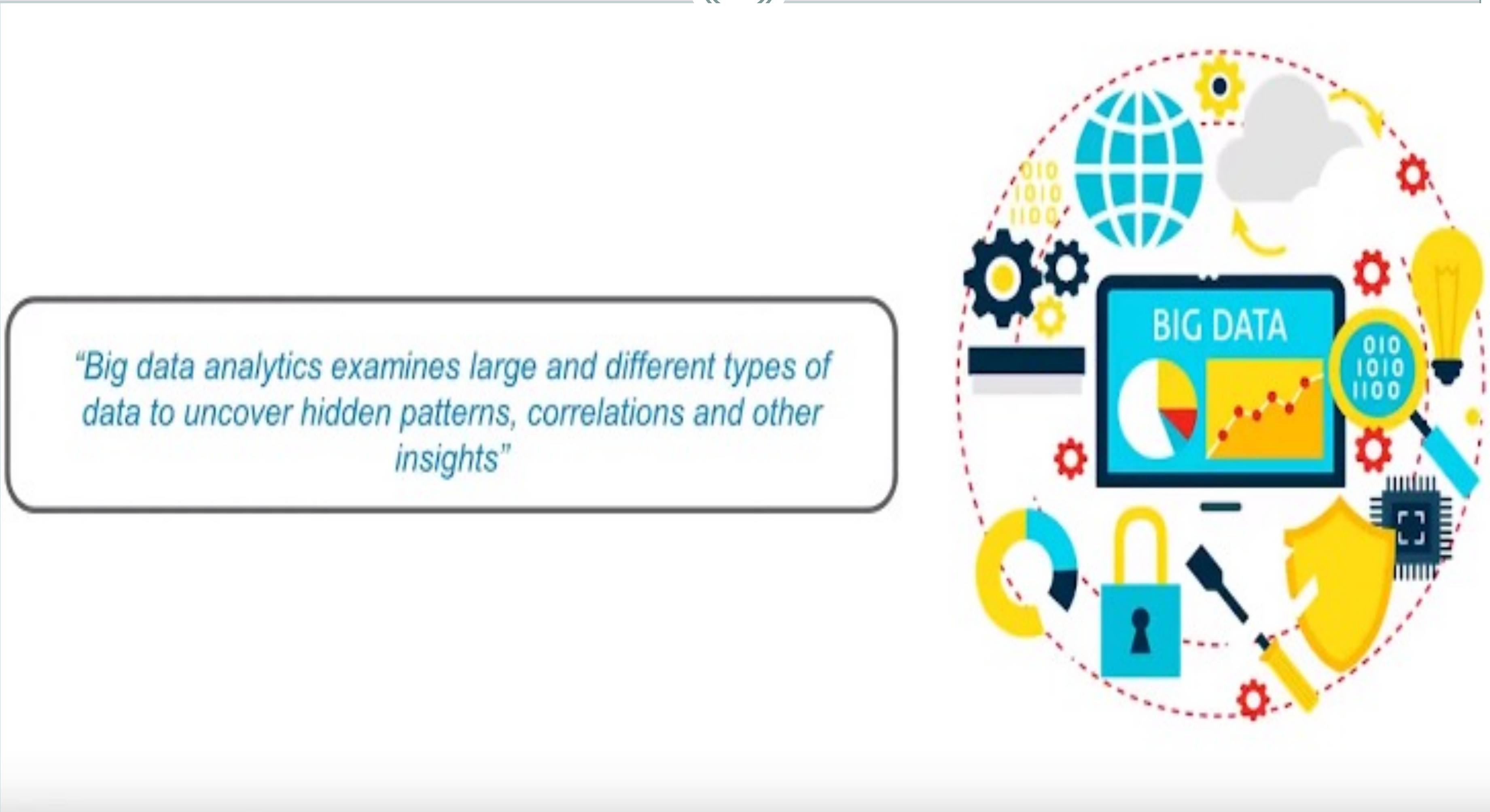
### Operational Infrastructure

### Infrastructure As A Service

### Structured Databases

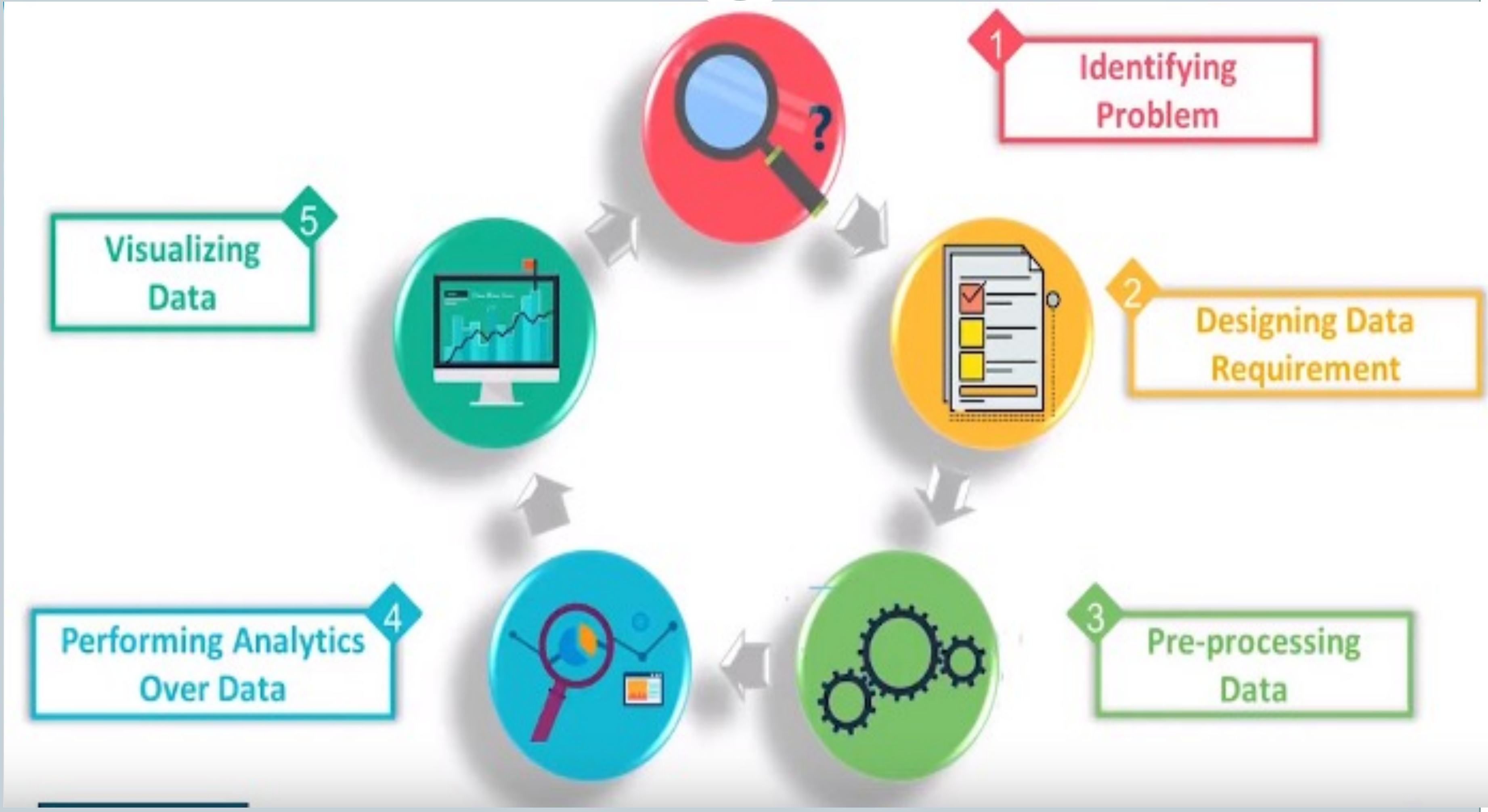
### Technologies

# What is Big data analytics

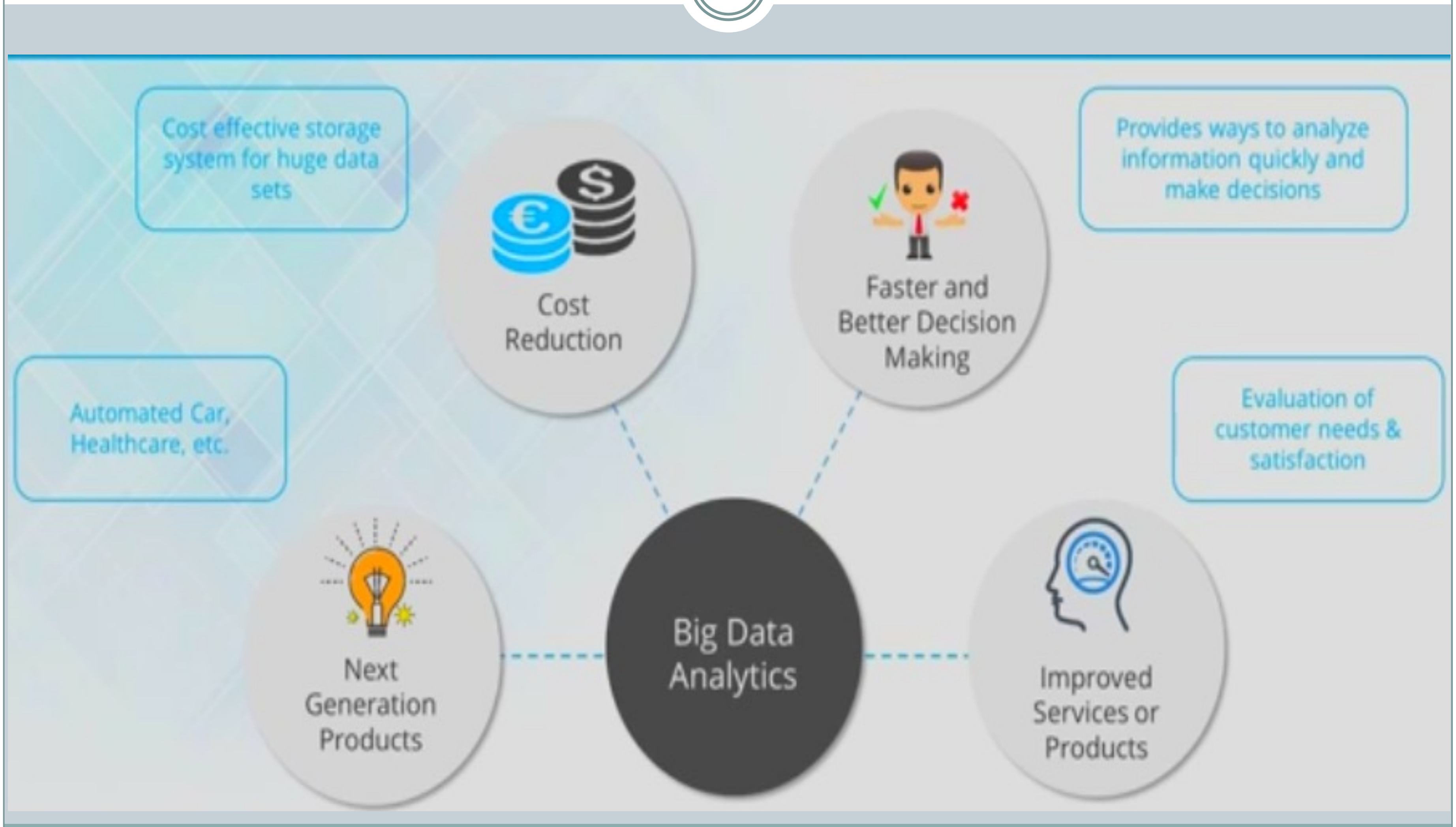


“Big data analytics examines large and different types of data to uncover hidden patterns, correlations and other insights”

# Stages in Big data analytics



# Big data analytics goals



Big Data Analytics

Cost Reduction

Cost effective storage system for huge data sets

Faster and Better Decision Making

Provides ways to analyze information quickly and make decisions

Next Generation Products

Automated Car, Healthcare, etc.

Improved Services or Products

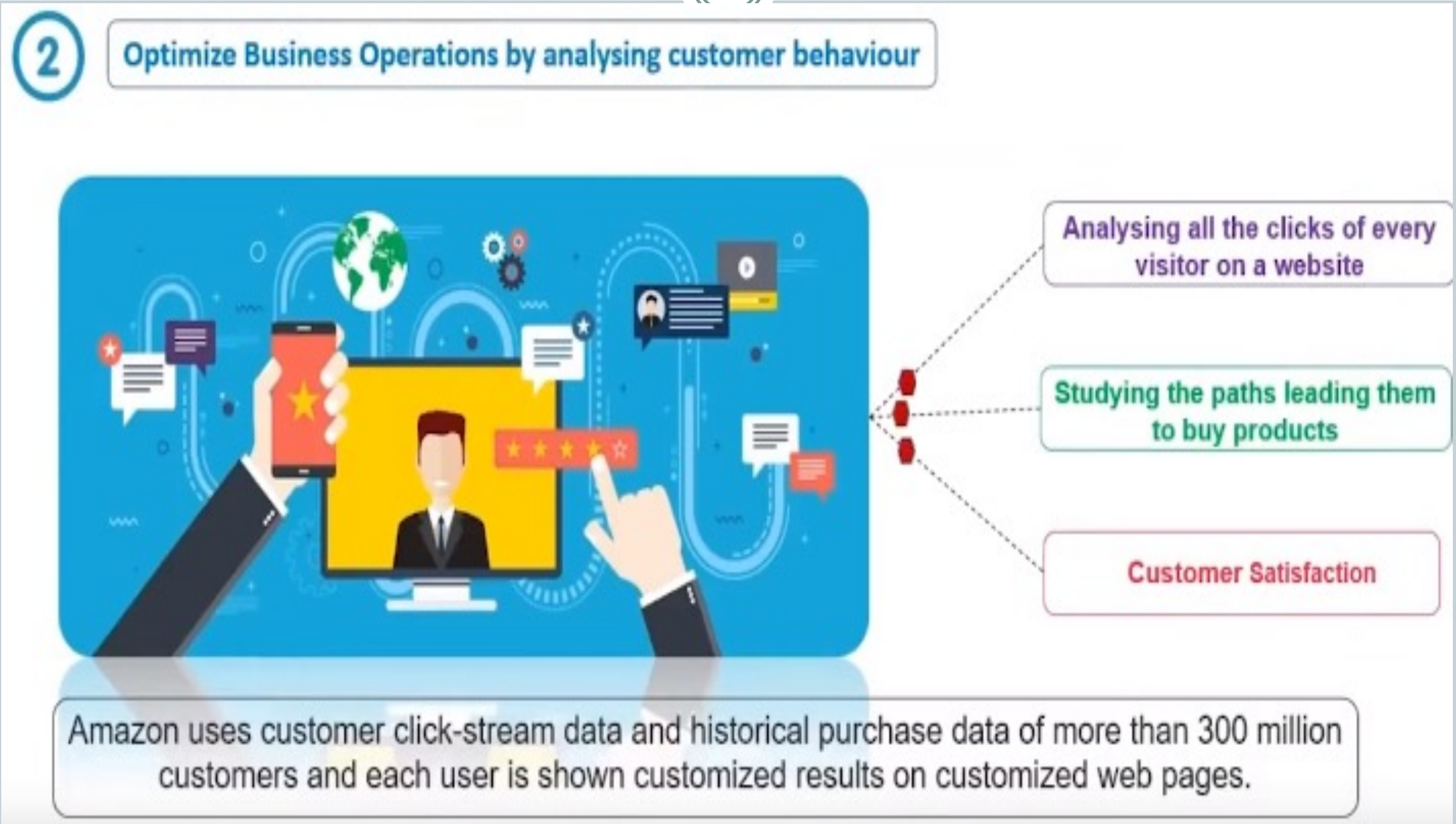
Evaluation of customer needs & satisfaction

# Big data analytics goals

1.Making organizations more smarter and efficient

New York Police Department is utilizing data patterns, scientific analysis, and technological tools to prevent the occurrence of crime

# Big data analytics goals



2 Optimize Business Operations by analysing customer behaviour

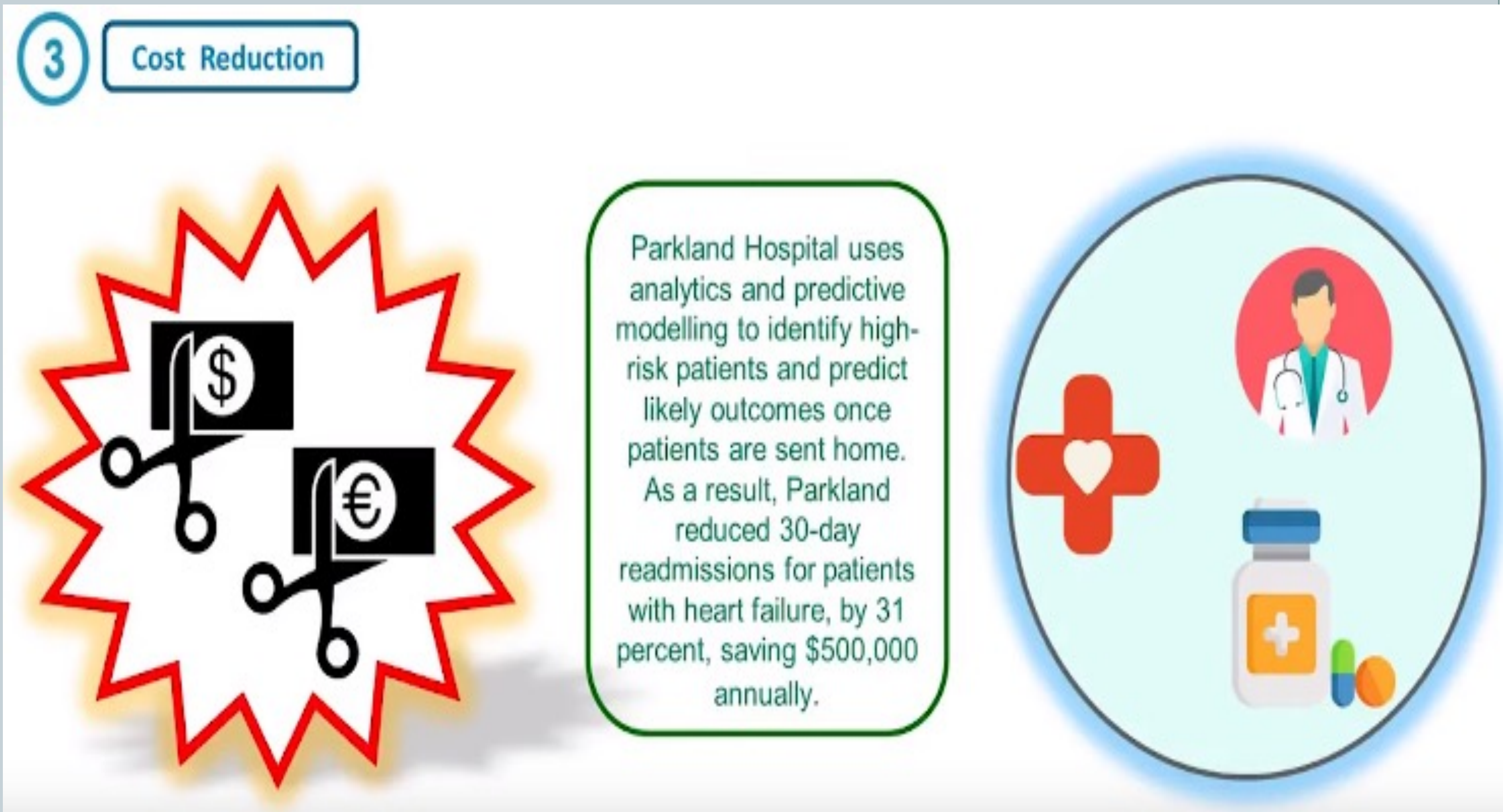
Analysing all the clicks of every visitor on a website

Studying the paths leading them to buy products

Customer Satisfaction

Amazon uses customer click-stream data and historical purchase data of more than 300 million customers and each user is shown customized results on customized web pages.

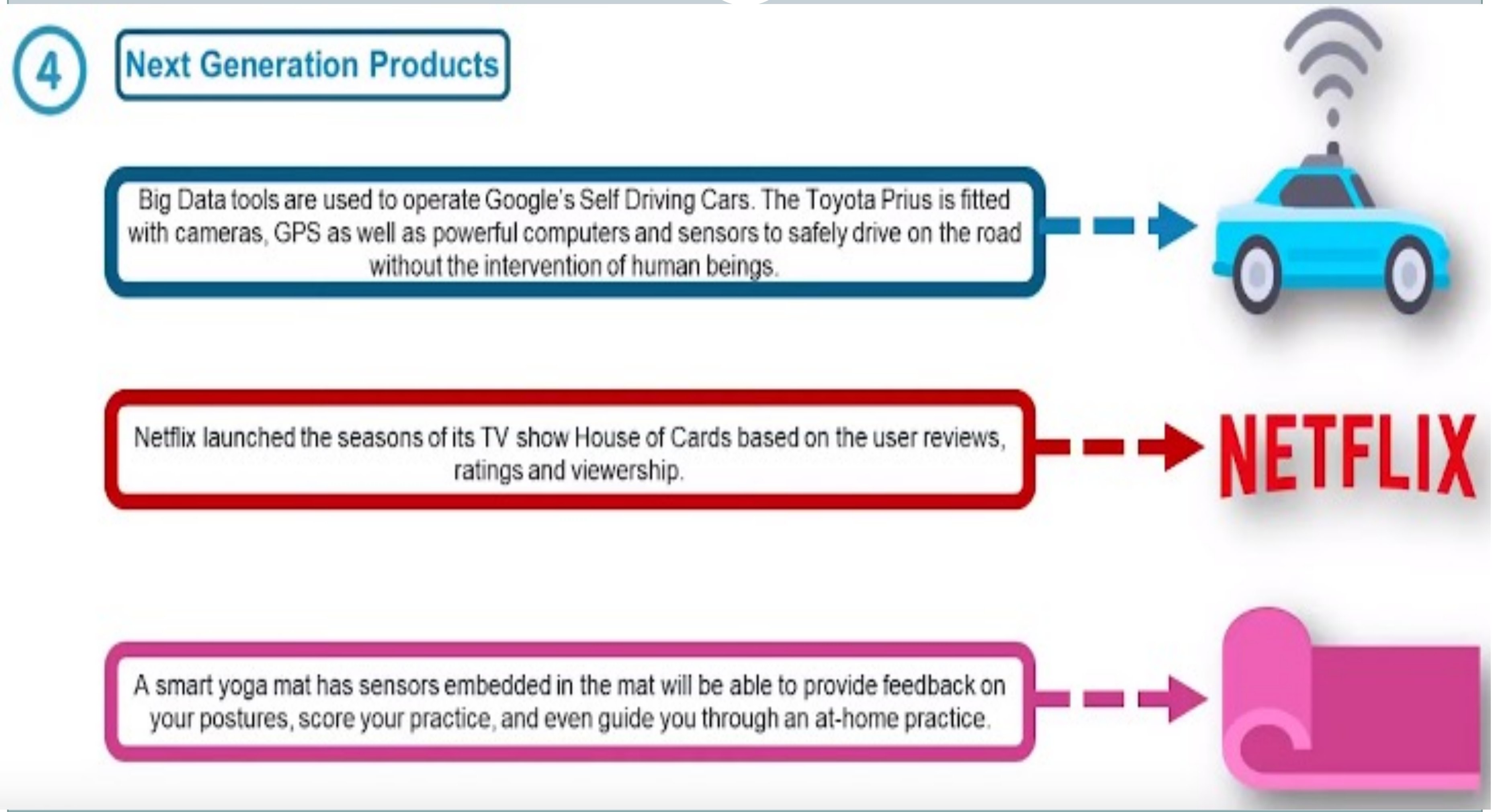
# Big data analytics goals



3 Cost Reduction

Parkland Hospital uses analytics and predictive modelling to identify high-risk patients and predict likely outcomes once patients are sent home. As a result, Parkland reduced 30-day readmissions for patients with heart failure, by 31 percent, saving $500,000 annually.

# Big data analytics goals



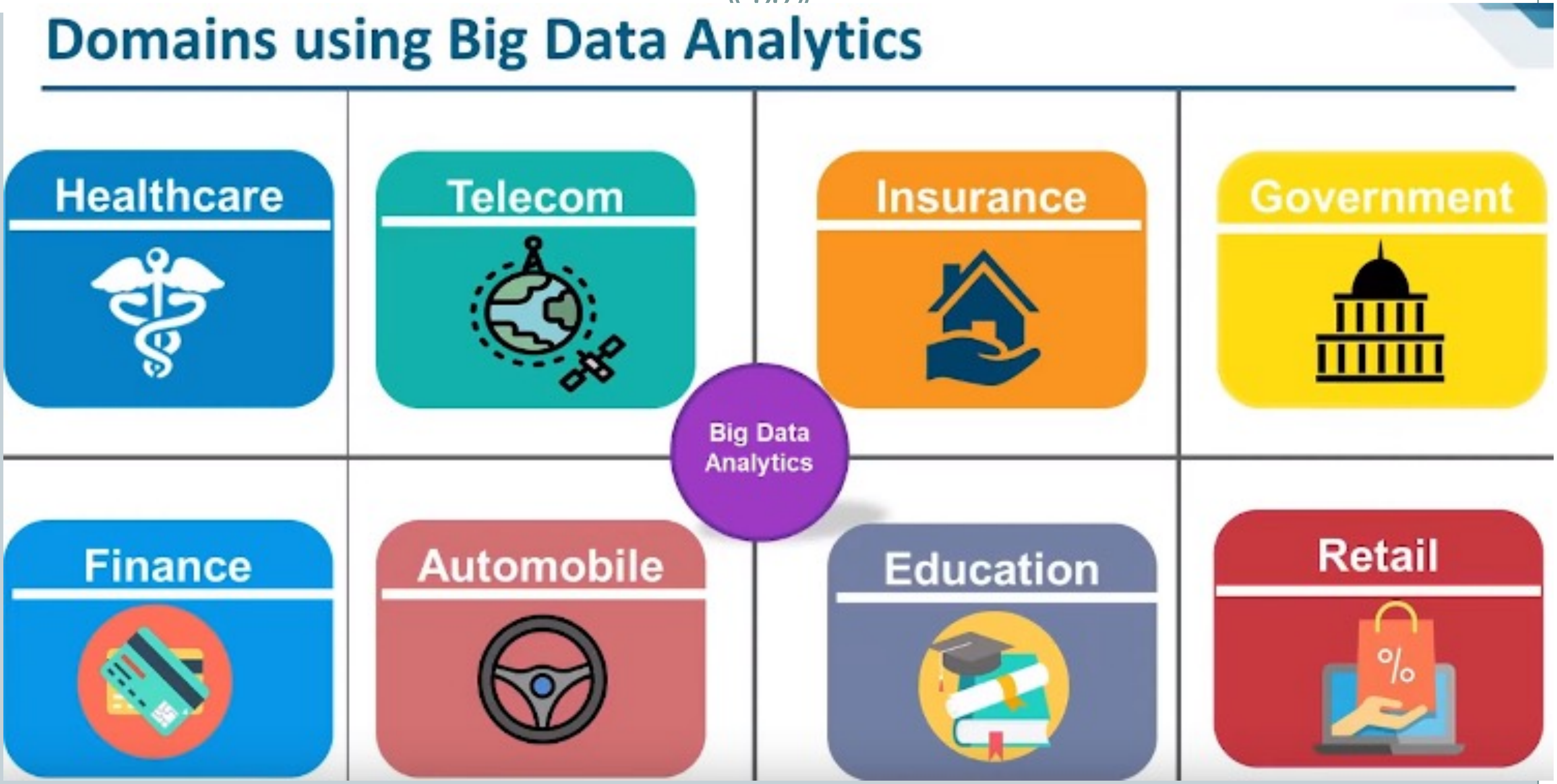
4 Next Generation Products

Big Data tools are used to operate Google's Self Driving Cars. The Toyota Prius is fitted with cameras, GPS as well as powerful computers and sensors to safely drive on the road without the intervention of human beings.

Netflix launched the seasons of its TV show House of Cards based on the user reviews, ratings and viewership.

A smart yoga mat has sensors embedded in the mat will be able to provide feedback on your postures, score your practice, and even guide you through an at-home practice.

# Big data analytics application domains

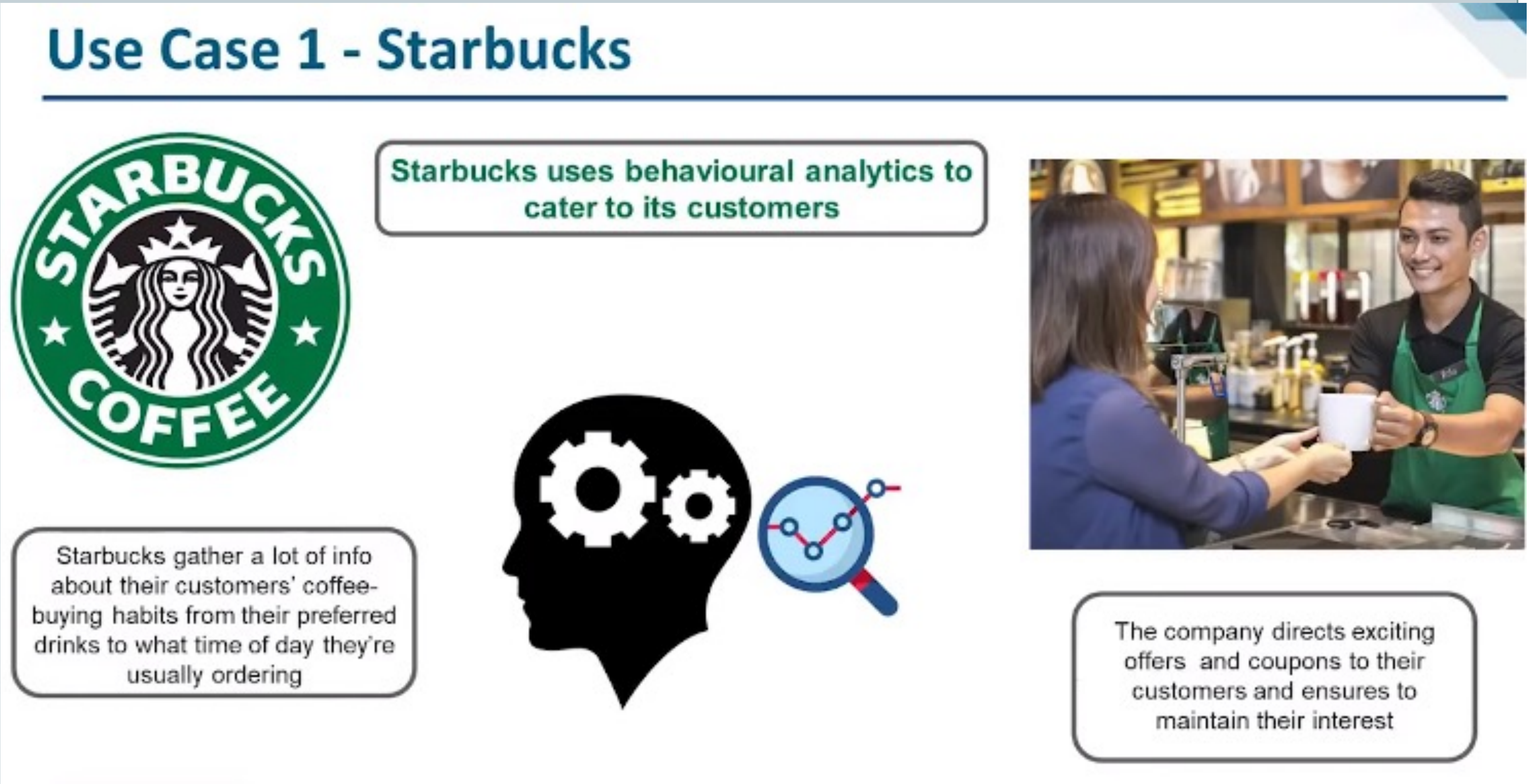


## Domains using Big Data Analytics

Big Data Analytics

- Healthcare
- Telecom
- Insurance
- Government
- Finance
- Automobile
- Education
- Retail

# Big data analytics use cases



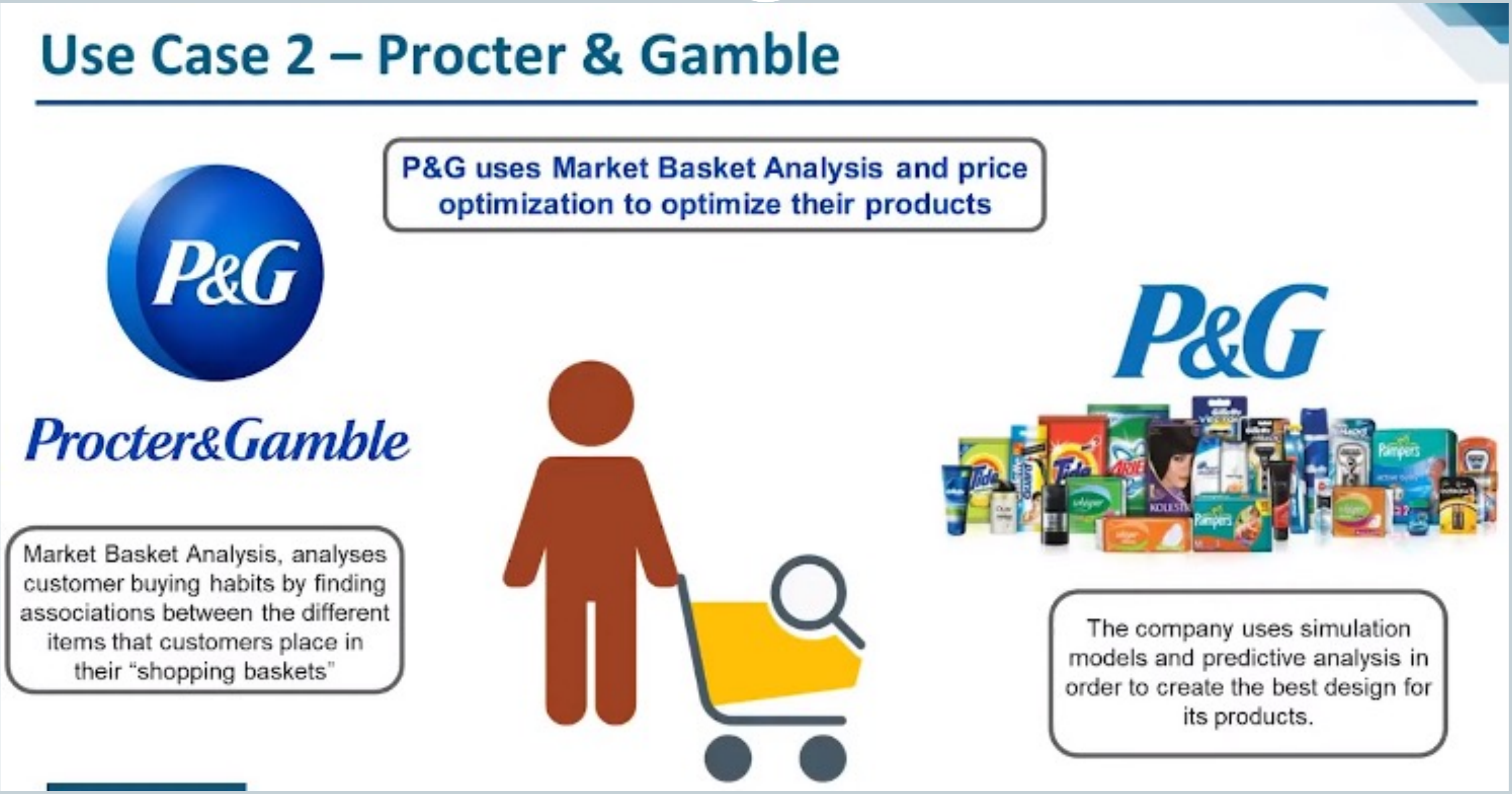
## Use Case 1 - Starbucks

**Starbucks uses behavioural analytics to cater to its customers**

Starbucks gather a lot of info about their customers' coffee-buying habits from their preferred drinks to what time of day they're usually ordering

The company directs exciting offers and coupons to their customers and ensures to maintain their interest

# Big data analytics use cases



## Use Case 2 – Procter & Gamble

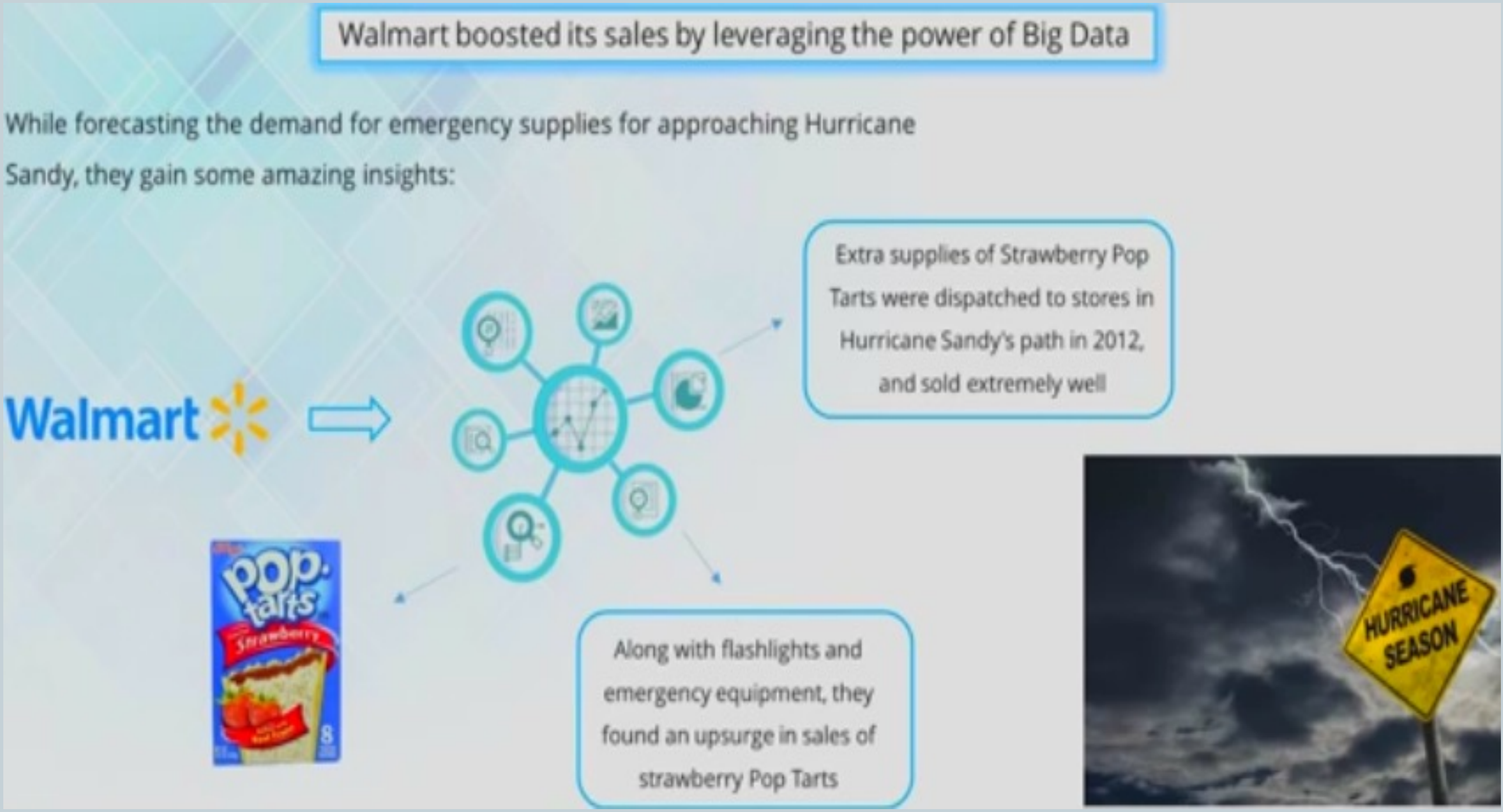
P&G uses Market Basket Analysis and price optimization to optimize their products

P&G
Procter&Gamble
P&G

Market Basket Analysis, analyses customer buying habits by finding associations between the different items that customers place in their "shopping baskets"

The company uses simulation models and predictive analysis in order to create the best design for its products.

# Big data analytics use cases



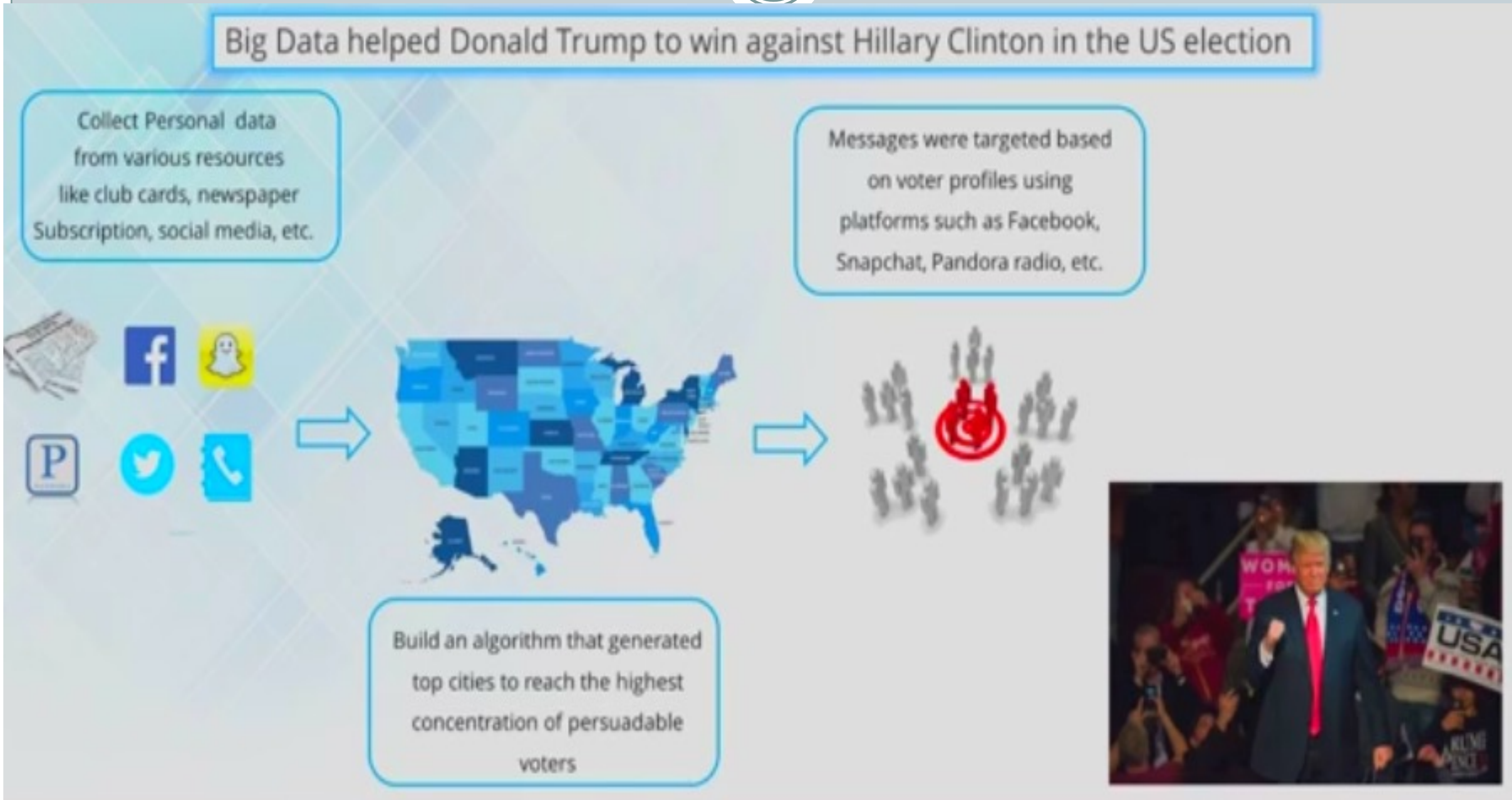
Walmart boosted its sales by leveraging the power of Big Data

While forecasting the demand for emergency supplies for approaching Hurricane Sandy, they gain some amazing insights:

Extra supplies of Strawberry Pop Tarts were dispatched to stores in Hurricane Sandy's path in 2012, and sold extremely well

Along with flashlights and emergency equipment, they found an upsurge in sales of strawberry Pop Tarts

# Big data analytics use cases



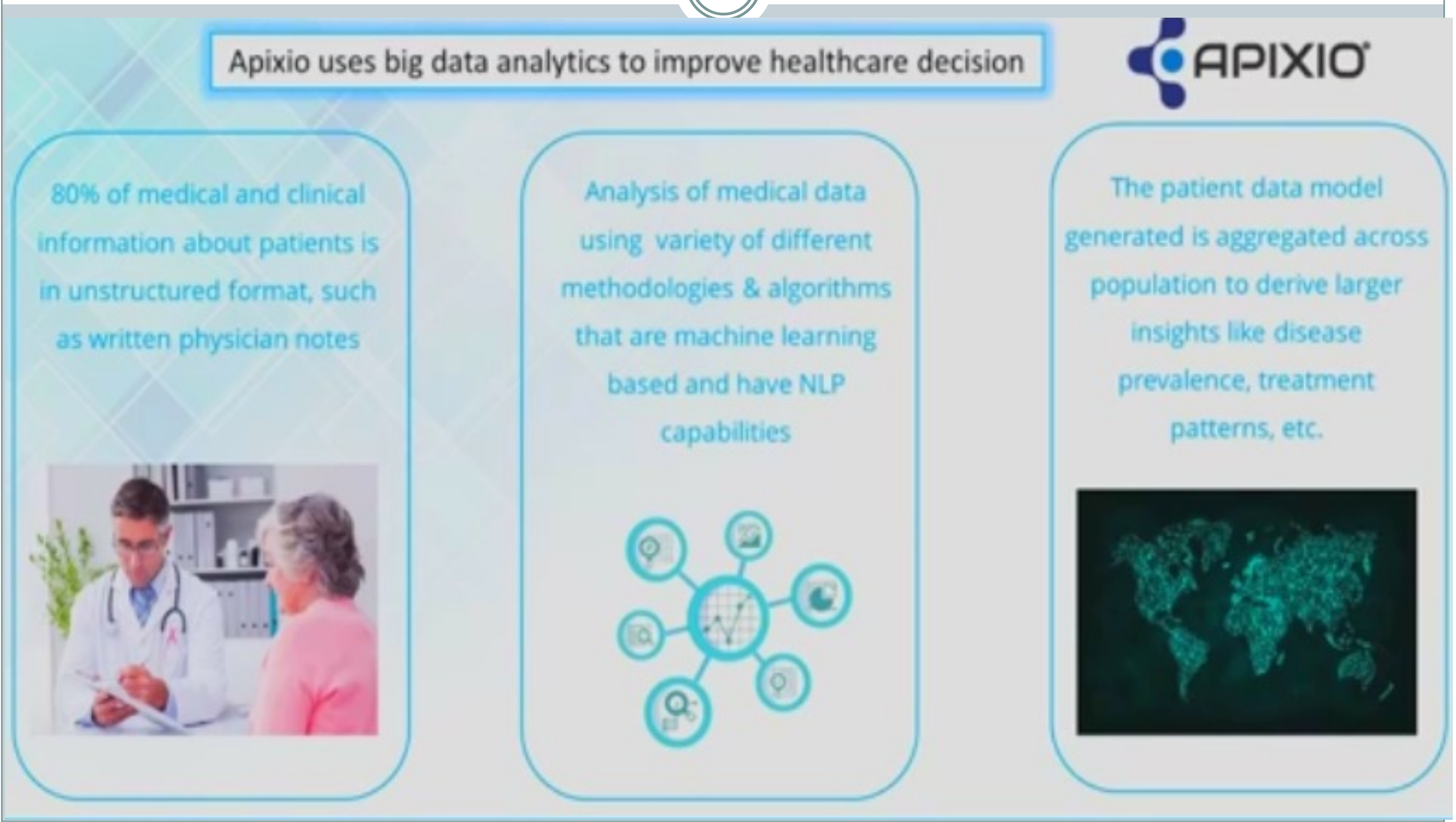
Big Data helped Donald Trump to win against Hillary Clinton in the US election

Collect Personal data from various resources like club cards, newspaper Subscription, social media, etc.

Build an algorithm that generated top cities to reach the highest concentration of persuadable voters

Messages were targeted based on voter profiles using platforms such as Facebook, Snapchat, Pandora radio, etc.

# Big data analytics use cases



Apixio uses big data analytics to improve healthcare decision

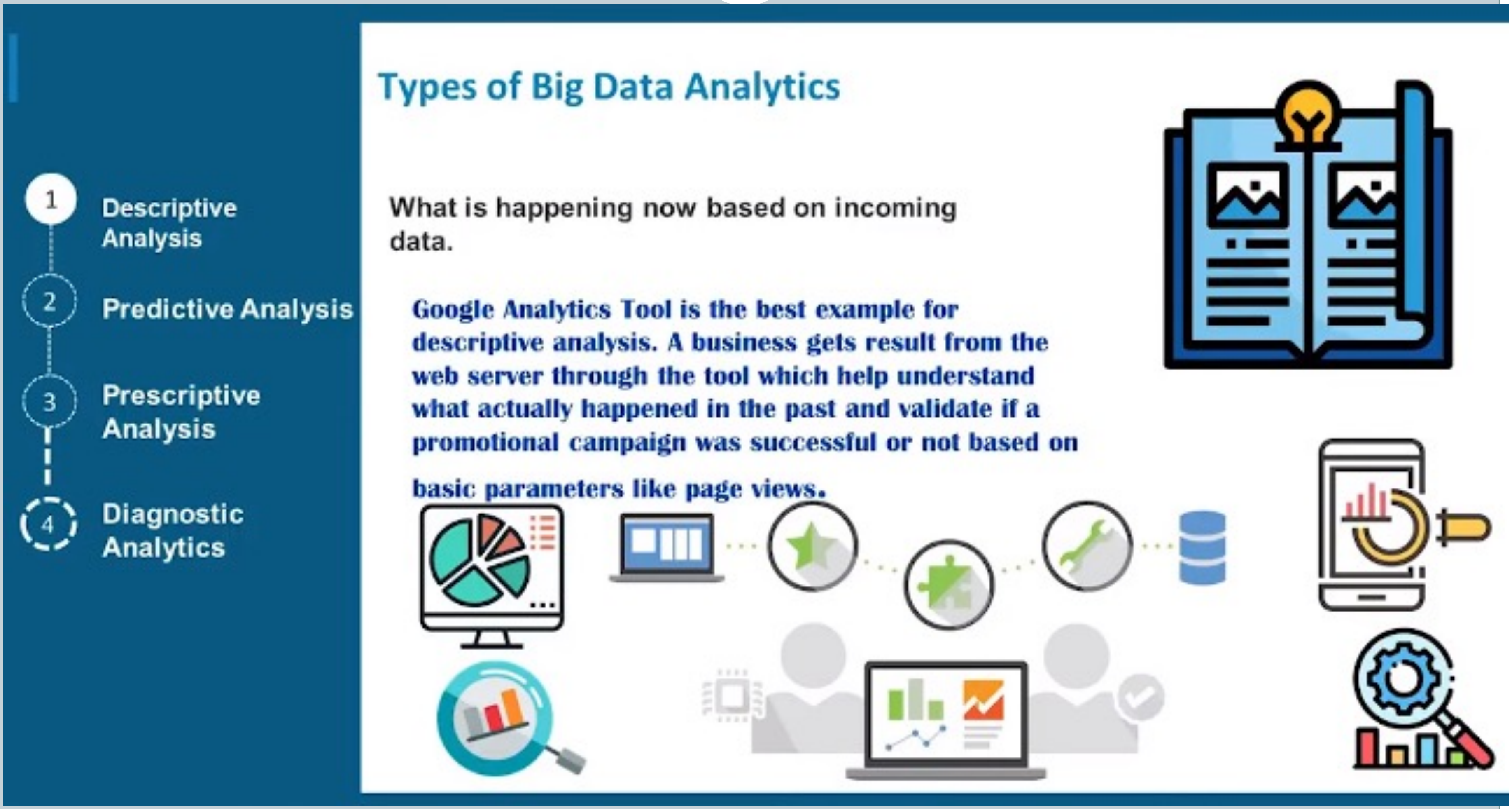
APIXIO

80% of medical and clinical information about patients is in unstructured format, such as written physician notes

Analysis of medical data using variety of different methodologies & algorithms that are machine learning based and have NLP capabilities

The patient data model generated is aggregated across population to derive larger insights like disease prevalence, treatment patterns, etc.

# Types of Big data analytics



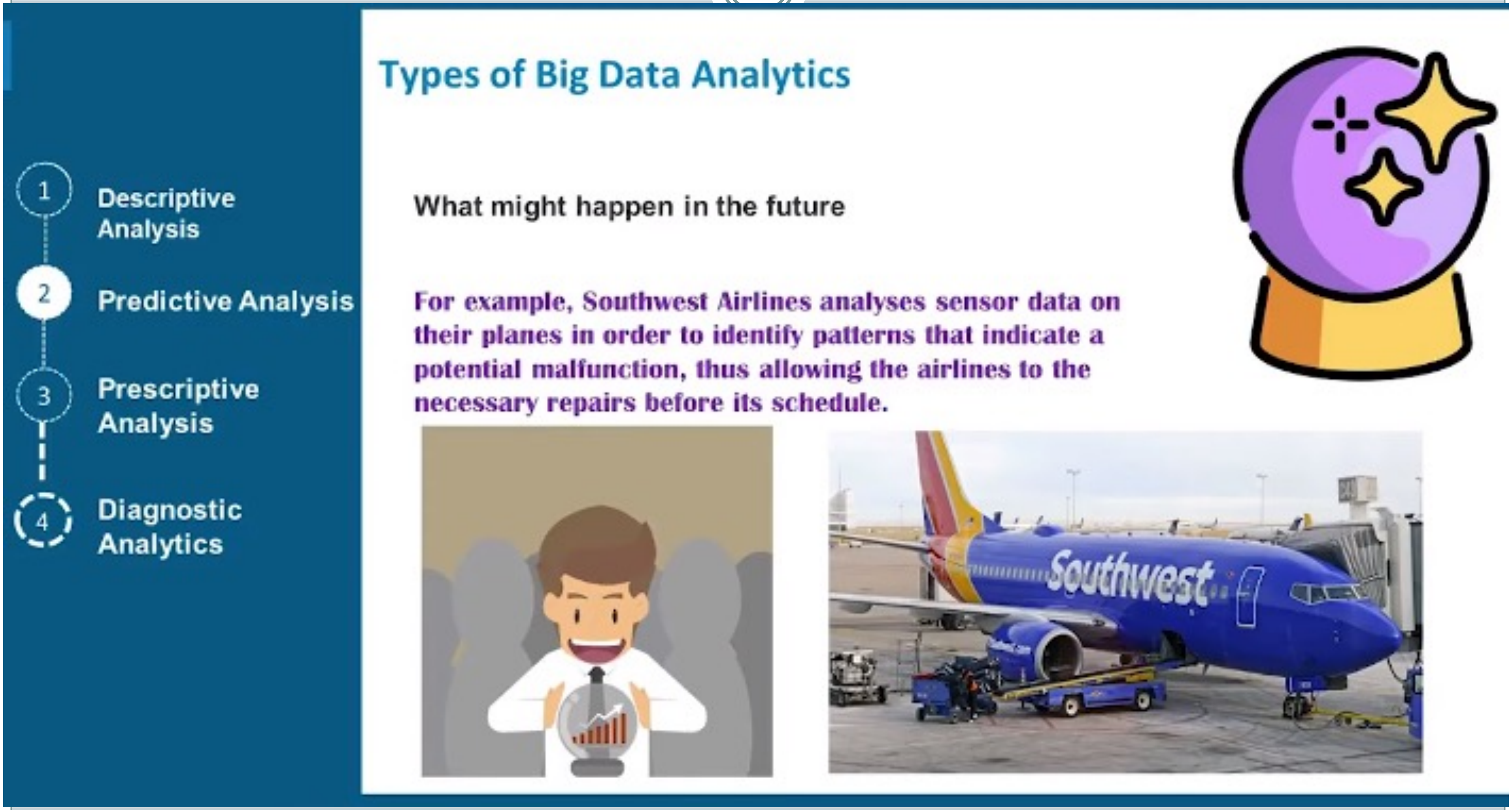
## Types of Big Data Analytics

1. Descriptive Analysis
2. Predictive Analysis
3. Prescriptive Analysis
4. Diagnostic Analytics

What is happening now based on incoming data.

**Google Analytics Tool is the best example for descriptive analysis. A business gets result from the web server through the tool which help understand what actually happened in the past and validate if a promotional campaign was successful or not based on basic parameters like page views.**

# Types of Big data analytics



## Types of Big Data Analytics

1 Descriptive Analysis

2 Predictive Analysis

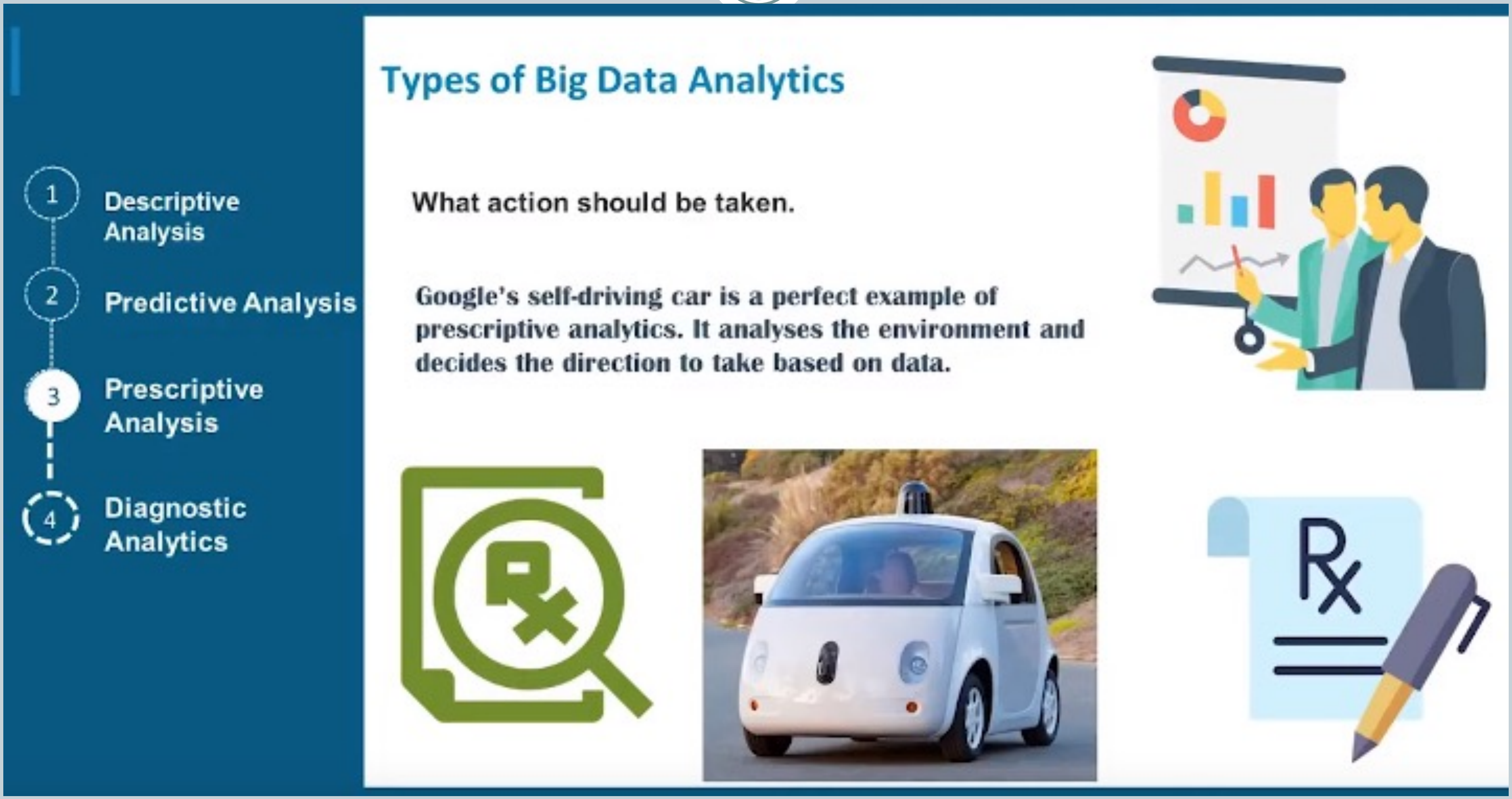
3 Prescriptive Analysis

4 Diagnostic Analytics

What might happen in the future

For example, Southwest Airlines analyses sensor data on their planes in order to identify patterns that indicate a potential malfunction, thus allowing the airlines to the necessary repairs before its schedule.

# Types of Big data analytics



1 Descriptive Analysis

2 Predictive Analysis

3 Prescriptive Analysis

4 Diagnostic Analytics

## Types of Big Data Analytics

What action should be taken.

Google's self-driving car is a perfect example of prescriptive analytics. It analyses the environment and decides the direction to take based on data.

# Types of Big data analytics

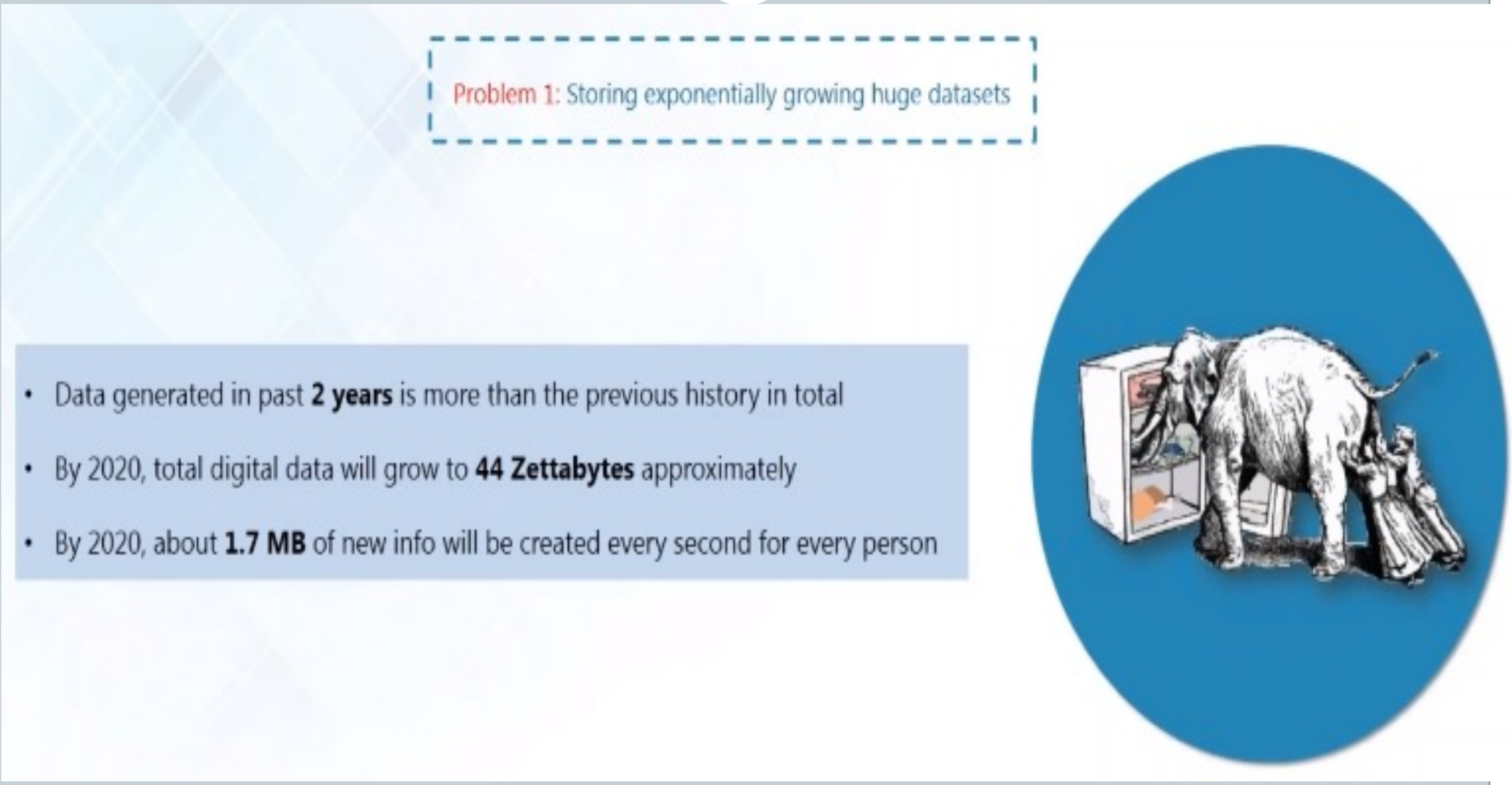
## Types of Big Data Analytics

1. Descriptive Analysis
2. Predictive Analysis
3. Prescriptive Analysis
4. Diagnostic Analytics

Why did it happen

For a Social Media marketing campaign, you can use diagnostic analytics to assess the number of posts, mentions, followers, fans, page views, reviews, pins, etc. and analyse the failure and success rate of the campaign at a fundamental level.

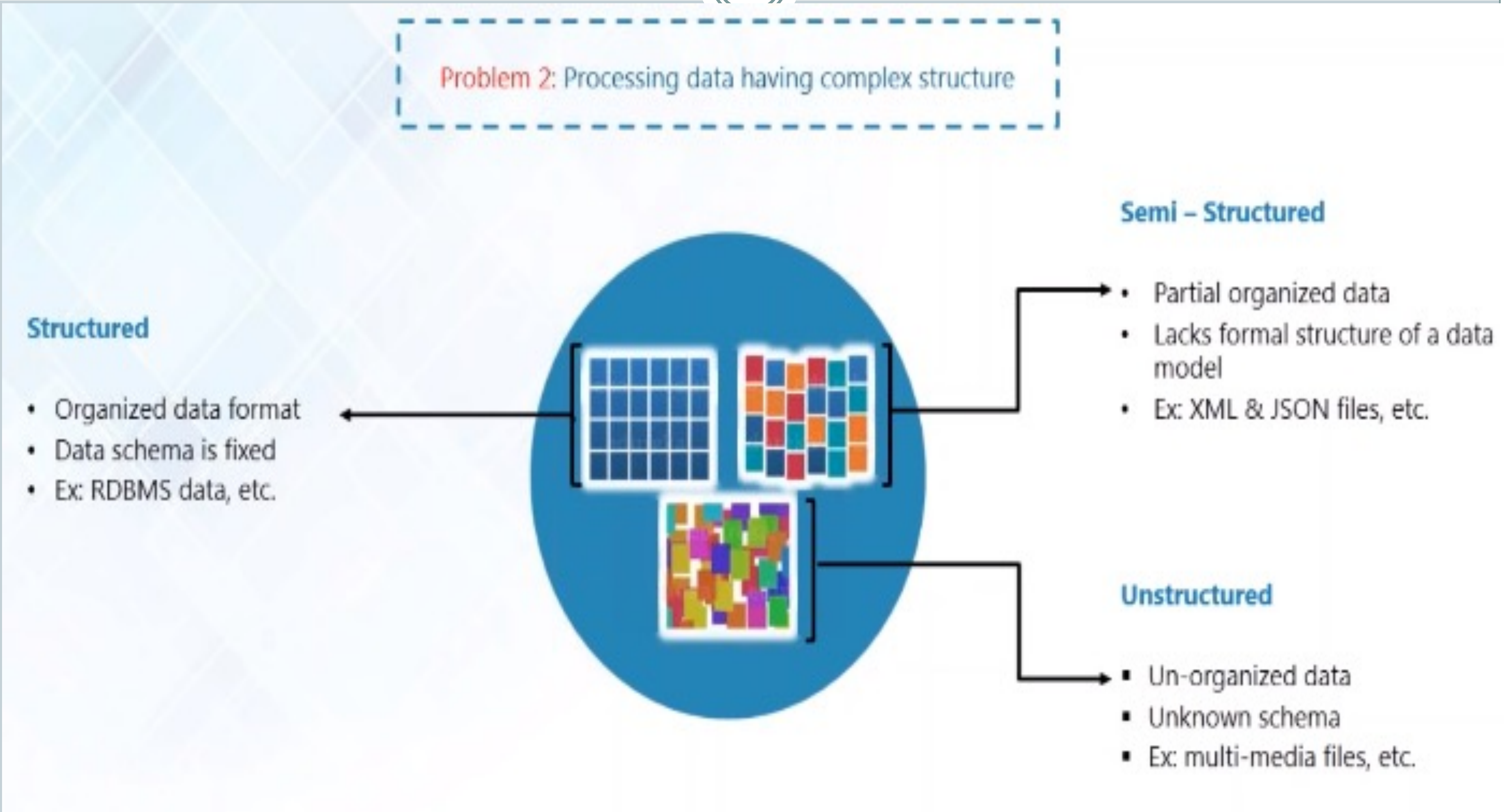
# Challenges/problems  with Big data



Problem 1: Storing exponentially growing huge datasets

- Data generated in past **2 years** is more than the previous history in total
- By 2020, total digital data will grow to **44 Zettabytes** approximately
- By 2020, about **1.7 MB** of new info will be created every second for every person

# Challenges/problems with Big data



Problem 2: Processing data having complex structure

**Structured**

- Organized data format
- Data schema is fixed
- Ex: RDBMS data, etc.

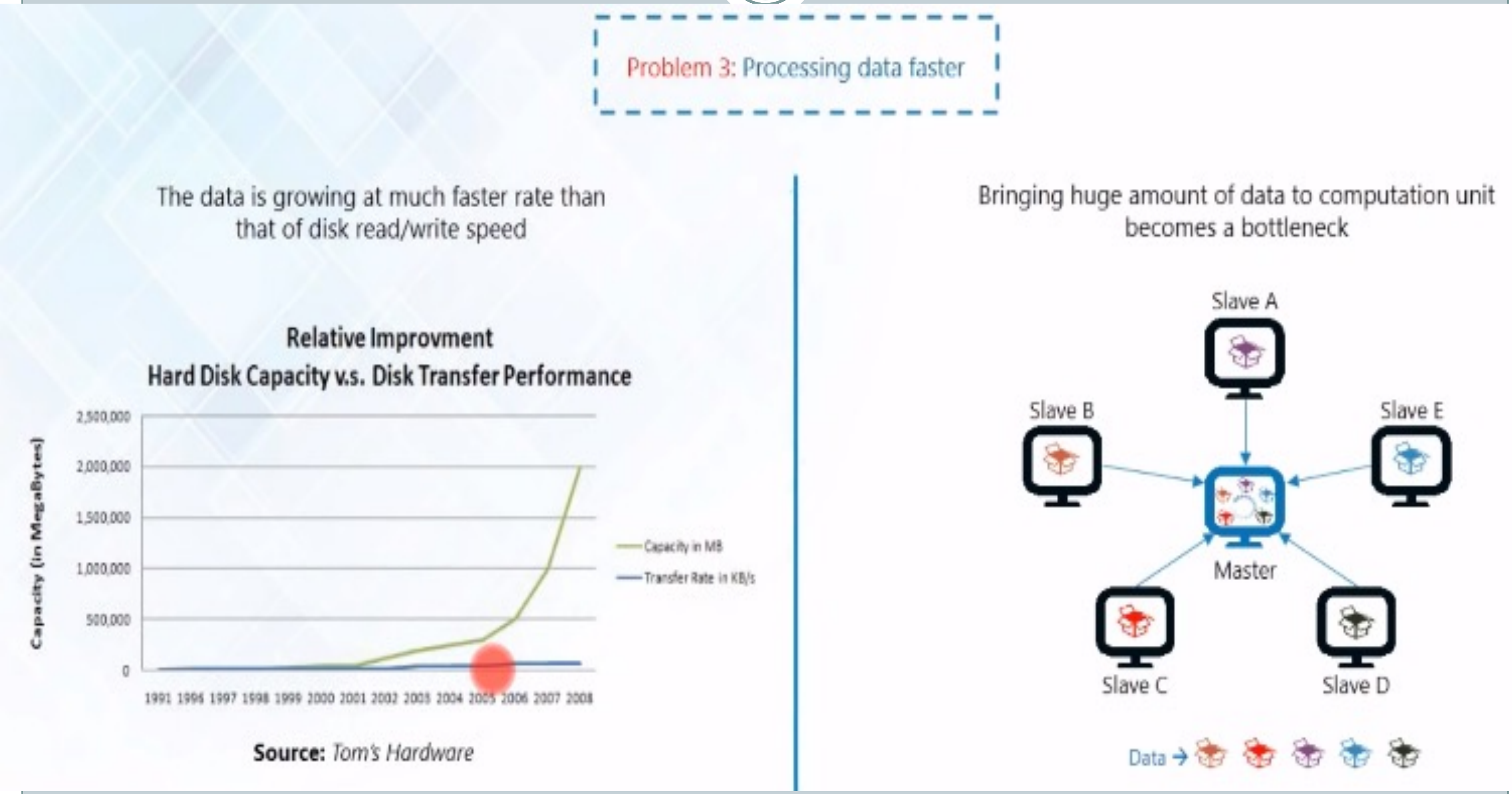
**Semi – Structured**

- Partial organized data
- Lacks formal structure of a data model
- Ex: XML & JSON files, etc.

**Unstructured**

- Un-organized data
- Unknown schema
- Ex: multi-media files, etc.

# Challenges/problems  with Big data



Problem 3: Processing data faster

The data is growing at much faster rate than that of disk read/write speed

Relative Improvment
Hard Disk Capacity v.s. Disk Transfer Performance

**Source:** *Tom's Hardware*

Bringing huge amount of data to computation unit becomes a bottleneck

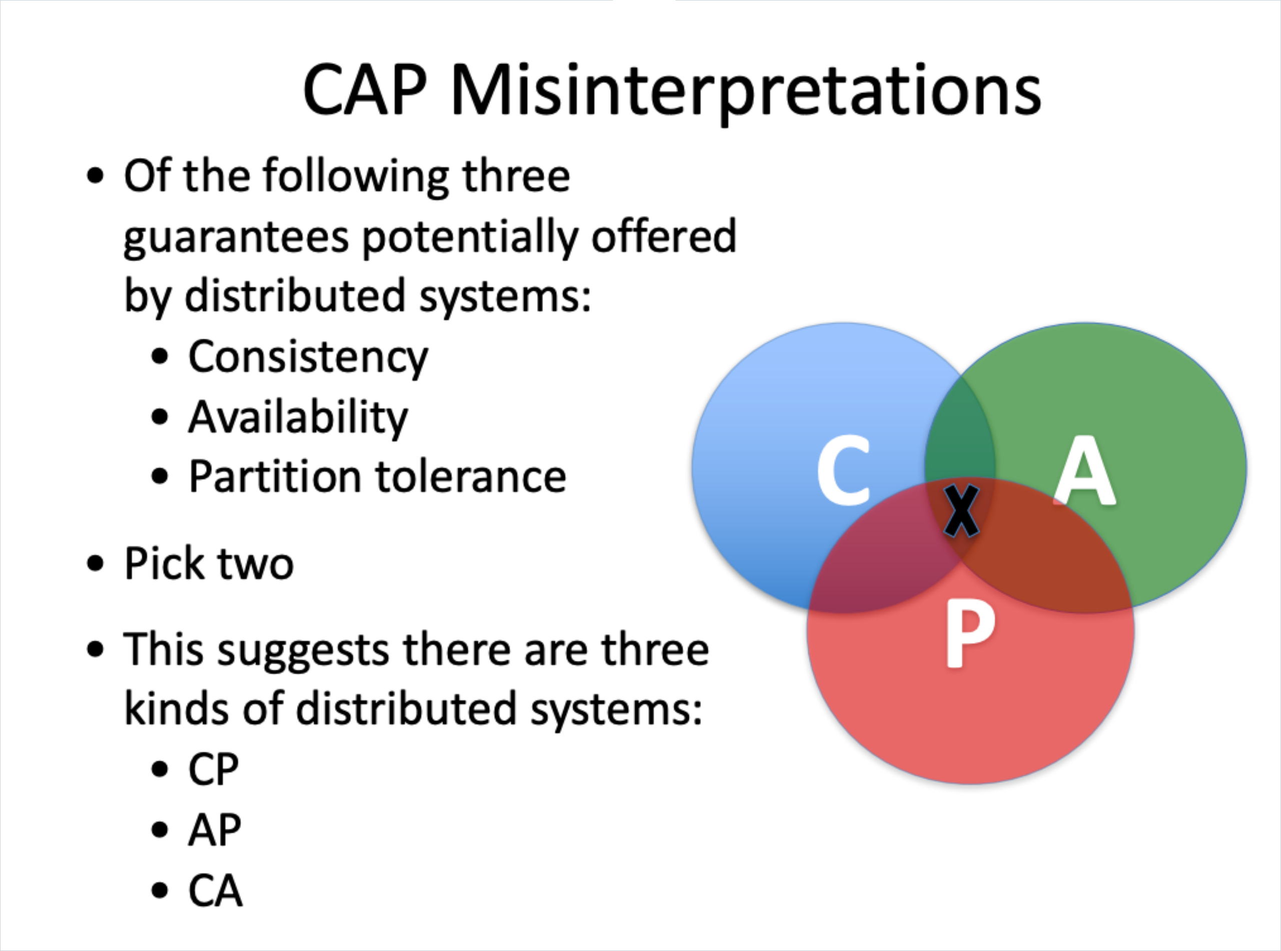
# CAP Theorem

# What is CAP Theorem?

- Consistency – Sequential consistency (a data item behaves as if there is one copy
- Availability: – Node failures do not prevent survivors from continuing to operate
- Partition-tolerance: – The system continues to operate despite network partitions
- CAP says that “A distributed system can satisfy any two of these guarantees at the same time **but not all three**

# Sequential consistency

- Makes it appear as if there is one copy of the object
- Strict ordering on ops from same client
- A single linear ordering across client ops
    - If client a executes operations {a1, a2, a3, ...}, client b executes operations {b1, b2, b3, ...}
    - Then, globally, clients observe some serialized version of the sequence
        - e.g., {a1, b1, b2, a2, ...} (or whatever)



# CAP Misinterpretations

- Of the following three guarantees potentially offered by distributed systems:
  - Consistency
  - Availability
  - Partition tolerance
- Pick two
- This suggests there are three kinds of distributed systems:
  - CP
  - AP
  - CA

# Issues with CAP

- What does it mean to choose or not choose partition tolerance?
  - – P is a property of the environment, C and A are goals
  - – In other words, what's the difference between a "CA" and "CP" system? both give up availability on a partition!
- Better phrasing: "*if the network can have partitions, do we give up on consistency or availability?*"

# Witnesses: P is unavoidable

- Coda Hale, Yammer (Microsoft?) software engineer:
  - *"Of the CAP theorem's Consistency, Availability, and Partition Tolerance, Partition Tolerance is mandatory in distributed systems. You cannot not choose it."*
- Werner Vogels, Amazon CTO
  - "An important observation is that in larger distributed-scale systems, network partitions are a given; therefore, consistency and availability cannot be achieved at the same time.
- Daneil Abadi (UMD), Co-founder of Hadapt; Vertica, VoltDB contributor
  - "So in reality, there are only two types of systems ... I.e., if there is a partition, does the system give up availability or consistency?

# Consistency or Availability?

- Consistency and Availability is not a “binary” decision
- AP systems relax consistency in favor of availability – but are not inconsistent
- CP systems sacrifice availability for consistency but are not unavailable
- This suggests both AP and CP systems can offer a degree of consistency, and availability, as well as partition tolerance

# AP: Best Effort Consistency

- Example:
  - CDNs / Web caches
  - DNS
  - BlockChain
  - CRDTs
- Trait:
  - Optimistic concurrency control
  - Expiration/Time-to-live
  - Conflict resolution

# CP: Best Effort Availability

- • Example:
  - Majority protocols (Paxos, Raft)
  - Distributed Locking (Google Chubby Lock service)
- Trait:
  - Pessimistic locking
  - Make minority partition unavailable

# Types of Consistency

- Strong Consistency
  - After the update completes, any subsequent access will return the same updated value.
- Weak Consistency
  - It is not guaranteed that subsequent accesses will return the updated value.
- Eventual Consistency
  - Specific form of weak consistency
  - It is guaranteed that if no new updates are made to object, eventually all accesses will return the last updated value (e.g., propagate updates to replicas in a lazy fashion)

# Eventual Consistency Variations

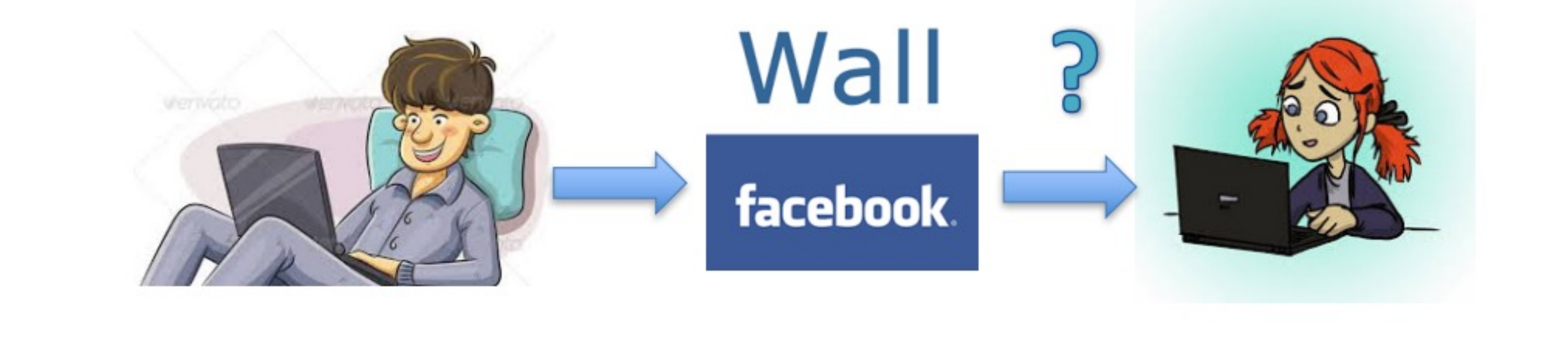
- Causal consistency
  - Processes that have causal relationship will see consistent data
- Read-your-write consistency
  - A process always accesses the data item after it's update operation and never sees an older value
- Session consistency
  - As long as session exists, system guarantees readyour-write consistency
  - Guarantees do not overlap sessions

# Eventual Consistency Variations

- Monotonic read consistency
  - If a process has seen a particular value of data item, any subsequent processes will never return any previous values
- Monotonic write consistency
  - The system guarantees to serialize the writes by the same process
- In practice
  - A number of these properties can be combined – Monotonic reads and read-your-writes are most desirable

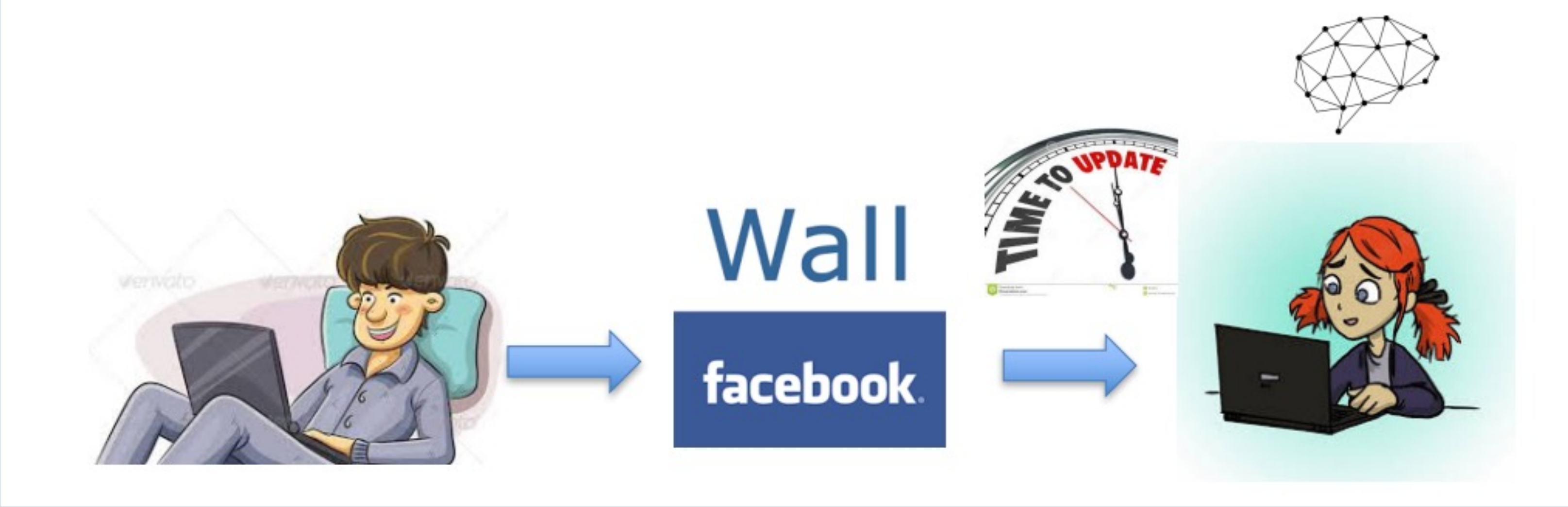
# Eventual Consistency - A Facebook Example

- Bob finds an interesting story and shares with Alice by posting on her Facebook wall
- Bob asks Alice to check it out
- Alice logs in her account, checks her Facebook wall but finds:
  - Nothing is there!



# Eventual Consistency - A Facebook Example

- Bob tells Alice to wait a bit and check out later
- Alice waits for a minute or so and checks back:
  - **She finds the Cambridge Analytica story Bob shared with her!**



# Eventual Consistency - A Facebook Example

- Reason: it is possible because Facebook uses an **eventual consistent model**
- Why would Facebook choose an eventual consistent model over the strong consistent one?
  - Facebook has more than 1 billion active users
  - It is non-trivial to efficiently and reliably store the huge amount of data generated at any given time
  - Eventual consistent model offers the option to reduce the load and improve availability

# Dynamic Tradeoff between C and A

- An airline reservation system:
  - When most of seats are available: it is ok to rely on somewhat out-of-date data, availability is more critical
  - When the plane is close to be filled: it needs more accurate data to ensure the plane is not overbooked, consistency is more critical
- Neither strong consistency nor guaranteed availability, but it may significantly increase the tolerance of network disruption

# Heterogeneity: Segmenting C and A

- No single uniform requirement
  - Some aspects require strong consistency
  - Others require high availability
- Segment the system into different components
  - Each provides different types of guarantees
- Overall guarantees neither consistency nor availability
  - Each part of the service gets exactly what it needs
- Can be partitioned along different dimensions

# Partitioning Strategies

- Data Partitioning
- Operational Partitioning
- Functional Partitioning
- User Partitioning
- Hierarchical Partitioning

- Idea: provide differentiated guarantees depending on X {data/op/func/user/component}

# Partitioning Examples

- Data Partitioning
  - Different data may require different consistency and availability
- Example:
  - Shopping cart: high availability, responsive, can sometimes suffer anomalies
  - Product information need to be available, slight variation in inventory is sufferable
  - Checkout, billing, shipping records must be consistent

# Partitioning Examples

- Operational Partitioning
  - Each operation may require different balance between consistency and availability
- Example:
  - Reads: high availability; e.g.., “query”
  - Writes: high consistency, lock when writing; e.g., “purchase”
- Functional Partitioning
  - System consists of sub-services
  - Different sub-services provide different balances
  - Example: A comprehensive distributed system
    - Distributed lock service (e.g., Chubby) : Strong consistency
  - DNS service:
    - High availability

# Partitioning Examples

- **User Partitioning**
  - Try to keep related data close together to assure better performance
  - Example: Craigslist
    - Might want to divide its service into several data centers, e.g., east coast and west coast
    - Users get high performance (e.g., high availability and good consistency) if they query servers close to them
    - Poorer performance if a New York user query Craglist in San Francisco

# Partitioning Examples

- Hierarchical (node) Partitioning
  - Large global service with local “extensions”
  - Different location in hierarchy may use different consistency
- Example:
  - Local servers (better connected) guarantee more consistency and availability
  - Global servers has more partition and relax one of the requirement

# Summary

- CAP is a tool for thinking about trade-offs in distributed systems
- Misinterpreted + contentious
- The devil (in designing distributed systems) is often in the details: real systems cannot be classified into one of CA/AP/CP
- Many eventual consistency variants, widely adopted by popular systems