

Topic 2: Big Data Fundamentals

1. What is Big Data?

Core Definition

- **Primary Definition:** Data so big it **cannot be stored on persistent storage devices attached to a single computer system**
- **Alternative Definition:** An infinite amount of data
- Requires **distributed storage** and **parallel processing**

2. Characteristics of Big Data - The 3Vs (and More)

The Original 3Vs

1. **Volume**

- Example: Billions of rows, millions of columns
- Data scale beyond single-machine capacity

2. **Variety**

- Complexity of data types and structures
- Structured, semi-structured, and unstructured data

3. **Velocity**

- Speed of new data creation and growth
- Real-time or near-real-time data streams

Additional Vs (Extended Characteristics)

4. **Veracity**

- Ability to represent and process uncertain and imprecise data
- Data quality and trustworthiness

5. **Value**

- Data as the driving force of next-generation business
- Extracting actionable insights from data

6. **Viability**

- Benefits we can potentially have from data analysis
- Business value proposition

7. **Vagueness**

- The meaning of found data is often very unclear
- Challenges in data interpretation

8. Validity

- Rigor in analysis is essential for valid predictions
- Ensuring data accuracy and reliability

9. Value

- Data science can aid decision making by pointing in the correct direction
- Directional guidance from analytics

Note: There are many other Vs (up to 42 documented variations!)

3. Sources of Big Data

Common Data Sources

1. **Clickstream data:** Web browsing patterns and user interactions
2. **Call centre data:** Customer service interactions and logs
3. **E-mail and instant-messaging:** Communication records
4. **Sensor data:** IoT devices, industrial sensors
5. **Unstructured data:** Text, documents, emails
6. **Geographic data:** Location-based information
7. **Satellite data:** Remote sensing, weather, mapping
8. **Image data:** Photos, videos, medical imaging
9. **Temporal data:** Time-series information
10. **And many more:** Social media, transaction logs, etc.

4. Traditional Data Architectures

Strengths of Traditional Architectures

1. **Centralized governance** of data repositories
2. **Light-fast inquiries** performed regularly in daily business
3. **Optimization** for OLTP (Online Transaction Processing) and OLAP (Online Analytical Processing)
4. **Security and access control:** Well-established mechanisms
5. **Fault-tolerance and backup:** Reliable data protection

Challenges for Traditional Architectures

1. New types of data

- Unstructured data
- Semi-structured data
- Cannot fit into traditional relational schema

2. Volume challenges

- Increasingly large amounts of data flowing into organizations
- Single-server capacity limitations

3. Computational paradigms

- Non-traditional NoSQL databases required
- Need to rapidly mine and analyze very large datasets

4. Cost implications

- Increasing cost of storing large amounts of data
- Increasing cost of analyzing massive datasets

5. Data analytics requirements

- Significant storage capabilities needed
- Significant processing capabilities needed

5. Data Lake Architecture

Key Components

- **Centralized repository** for storing all structured and unstructured data
- **Storage layer** for raw data in native format
- **Processing layer** for data transformation and analysis
- **Consumption layer** for analytics and reporting

Advantages

- Store data in raw format
- Schema-on-read approach (not schema-on-write)
- Flexibility in data analysis
- Cost-effective storage

6. Hardware Scalability for Big Data

Two Scalability Dimensions

1. Vertical Scaling (Scale Up)

- Add more power to existing machines
- Increase CPU, RAM, storage on single server
- **Limitations:** Hardware limits, cost increases exponentially

2. Horizontal Scaling (Scale Out)

- Add more machines to the cluster
- Distribute load across multiple servers
- **Advantages:**
 - Nearly linear scalability
 - Cost-effective
 - Better fault tolerance
- **Preferred approach** for big data systems

Why Horizontal Scaling for Big Data?

- Commodity hardware is cheaper
 - Linear or near-linear performance gains
 - Better fault tolerance (no single point of failure)
 - Easier to expand incrementally
-

Key Points for Exam

Critical Definitions

1. **Big Data:** Data too large for single-system storage
2. **3Vs:** Volume, Variety, Velocity (minimum to remember)
3. **Horizontal Scaling:** Adding more machines (preferred for big data)
4. **Vertical Scaling:** Adding more power to existing machines

Important Contrasts

- **Traditional vs Big Data Architectures**
 - Centralized vs Distributed
 - Structured vs Unstructured
 - OLTP/OLAP vs Batch/Stream Processing
- **Vertical vs Horizontal Scaling**
 - Scale Up vs Scale Out
 - Hardware limits vs Linear scalability
 - Expensive vs Cost-effective

Data Sources to Remember

- Structured: databases, spreadsheets
- Semi-structured: XML, JSON, logs
- Unstructured: text, images, video

Key Challenges

1. New data types (unstructured, semi-structured)
2. Increasing data volumes
3. Need for new computational paradigms (NoSQL, MapReduce)
4. Cost of storage and analysis
5. Analytics requirements