TEACHING BIG DATA WITH LIMITED RESOURCES: PRACTICAL LESSONS FROM A SCALED-DOWN LAB

Assoc. Prof. Dr. Azlan Ismail

- Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA

- Institute for Big Data Analytics and Artificial Intelligence, Universiti Teknologi MARA

WHAT THIS TALK IS ABOUT

- Sharing experience from teaching a data science technology course
  - Students are a mix of background, not all from computer science.
  - Course focuses on real big data tasks using Cloudera Hadoop tools.

- Sharing experience of using a scaled-down lab
  - Lab refers to: **a virtualized teaching environment that simulates real big data systems.**
  - The lab setup is modest, but designed to reflect real-world system behavior and performance patterns.

WHY WE BUILT THIS MODEST INFRASTRUCTURE

1. Concept-heavy → students see theory, not systems
2. Abstract infra → Real infrastructure remains invisible
3. Simplified tools → Systems thinking & troubleshooting hidden
4. Cloud platforms → Greate for production, but abstract away the details we want students to see

Taken from: https://www.linkedin.com/pulse/understanding-types-scalability-storage-bq6zf/

Taken from: https://www.istockphoto.com/photos/distributed-computing

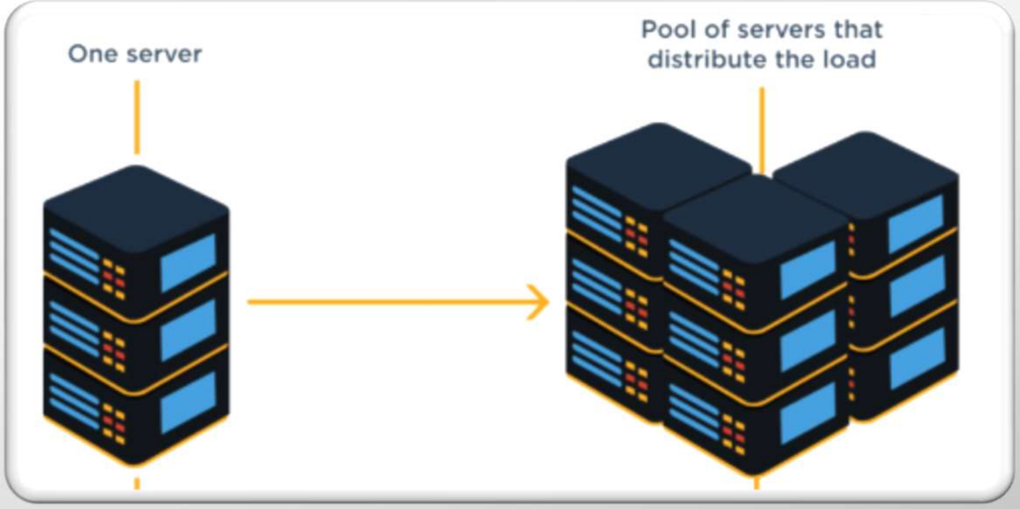
A MODEST LAB THAT MIRRORS REAL-WORLD SYSTEMS

What we set up?

- Designed **a teaching lab environment** to simulate real big data analytic workflows.
- Deployed a computing cluster with vendor support, which allowed us to focus on designing for student learning.
- **Platform:**
  - Cloudera hadoop distribution (CDH)
  - Tools: hive, spark, impala, sqoop, HDFS, YARN, etc
- **Infrastructure:**
  - 3 physical servers with 5 virtual machines
  - Master node + data nodes configured for cluster-based processing
- **Student access modes:**
  - Web UI (for hive, impala, sparksql, cloudera manager)
  - SSH terminal (for system-level exploration)
  - FTP client (for file exchange)

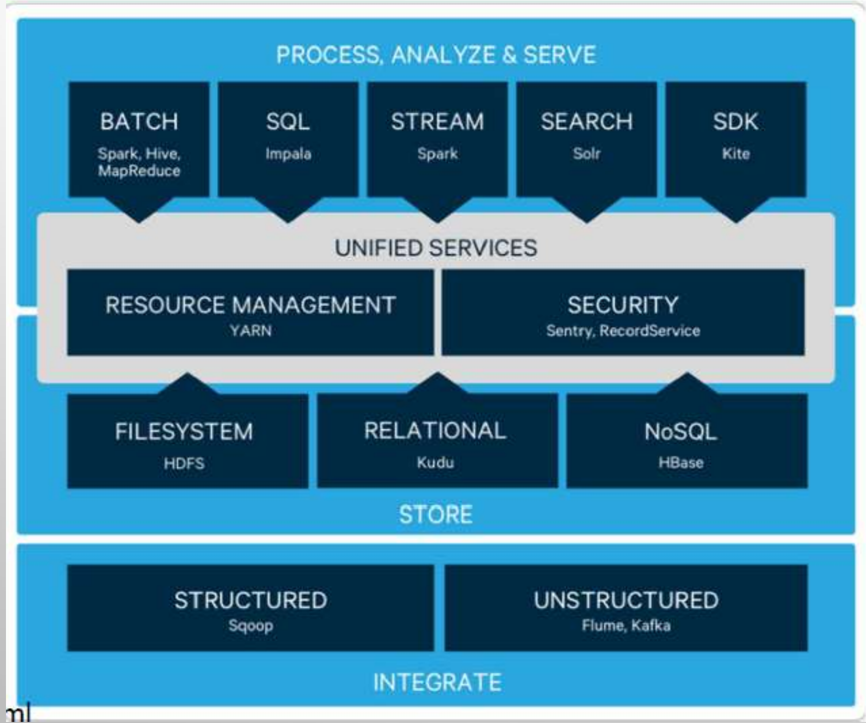
A MODEST LAB: SOFTWARE PLATFORM - HADOOP

- **What is Hadoop?**
- One of the first widely adopted open-source frameworks for big data.
- Provides the foundation for modern big data platforms.
- Makes it possible to store and process massive datasets across many machines.
- Key idea: scale out by adding nodes, not scale up with one big server.



A MODEST LAB: SOFTWARE PLATFORM

Cloudera Hadoop distribution (CDH) Platform



What ?

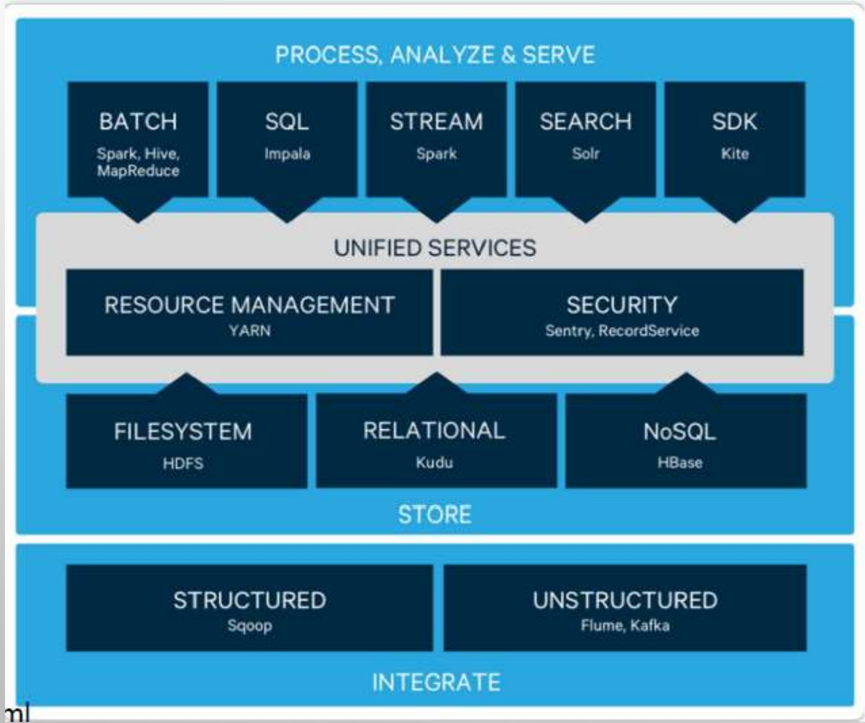
- ❑ An integrated Big Data ecosystem (Hive, Spark, Impala, Sqoop, HDFS, YARN).
- ❑ Supports the full workflow: ingest → store → process → analyze.
- ❑ Provides students with a realistic environment, not just single tools.

Three key software layers:

- ❑ **Integrate –** software to transfer data into ecosystem
- ❑ **Store –** software to store and manage data
- ❑ **Process, Analyze & Serve –** software to process and analyze data

A MODEST LAB: WHY CLOUDERA HADOOP

Cloudera Hadoop distribution (CDH) Platform



ml

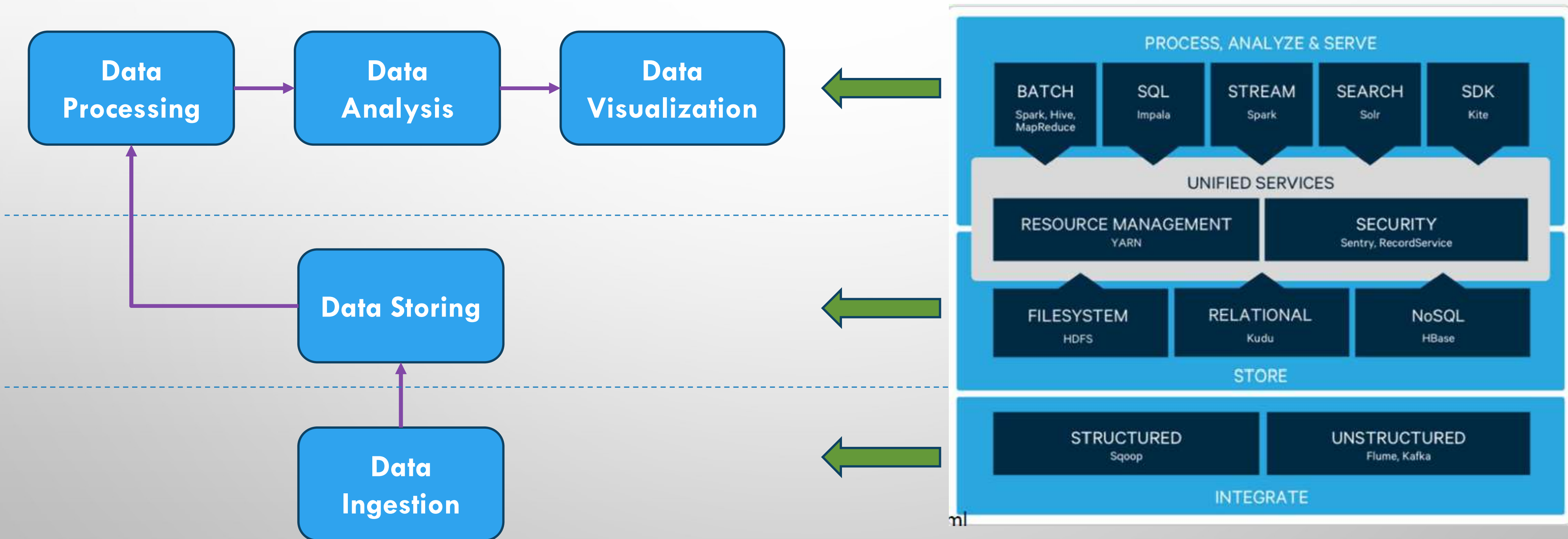
Why ?

- ❑ Shows ecosystem integration, not isolated skills.
- ❑ Reflects industry practice but scaled down for education.
- ❑ Cloudera Manager made system health and job tracking visible.
- ❑ Reusable and repeatable across courses and student cohorts.

Other reasons:

1. manageable for teaching.
2. no messy installations on student laptops.

A MODEST LAB: CLOUDERA HADOOP SUPPORTS BIG DATA ANALYTICS PIPELINE



Cloudera Hadoop distribution (CDH)

A MODEST LAB: MAPPING CLOUDER HADOOP WITH BIG DATA ANALYTICS PIPELINE

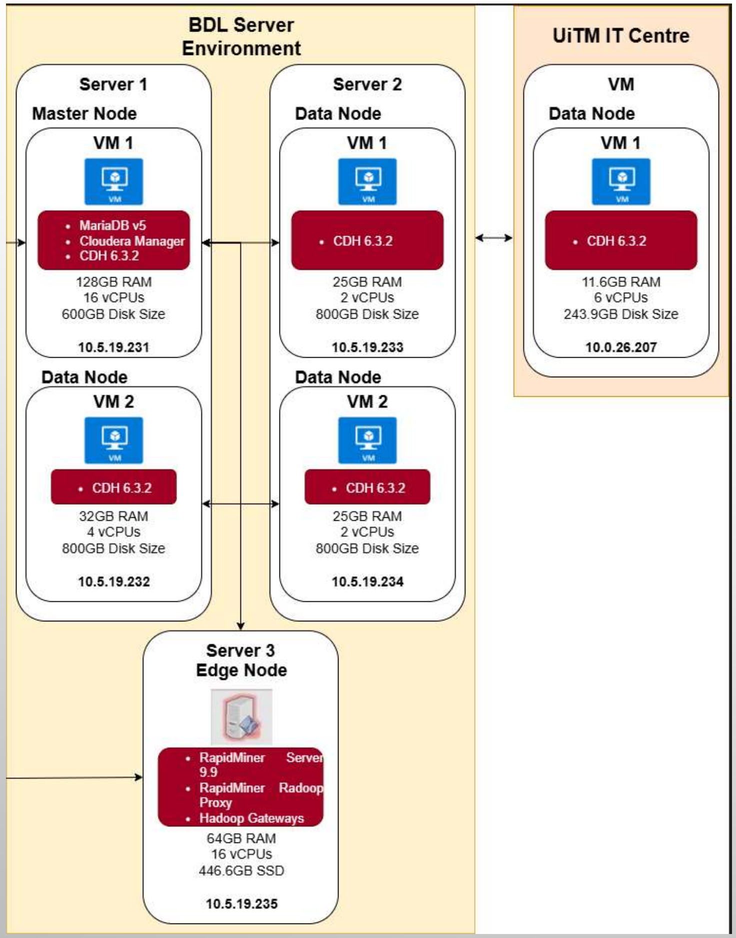
Pipeline Stage	Purpose	Cloudera Tool(s)
1. Data Ingestion	Ingest data from databases, logs, APIs	Sqoop (RDBMS → HDFS) Flume (logs → HDFS) Kafka (real-time streams) Hue Upload
2. Data Storing	Store raw or semi-structured data	HDFS (distributed file system) HBase (NoSQL, columnar)
3. Data Processing	Clean, transform, join, filter data	Hive (HiveQL) Spark (batch or structured streaming)
4. Data Analysis	Run SQL, ML, or statistical analysis	Impala (interactive SQL) Spark MLlib Hive (batch analytics)
5. Data Visualization	Present insights to users/admins	Hue (query interface + visualizations)

A MODEST LAB THAT MIRRORS REAL-WORLD SYSTEMS

What we set up?

- Designed **a teaching lab environment** to simulate real big data analytic workflows.
- Deployed a computing cluster with vendor support, which allowed us to focus on designing for student learning.
- **Platform:**
  - Cloudera hadoop distribution (CDH)
  - Tools: hive, spark, impala, sqoop, HDFS, YARN, etc
- **Infrastructure**:
  - 3 physical servers with 5 virtual machines
  - Master node + data nodes configured for cluster-based processing
- **Student access modes:**
  - Web UI (for hive, impala, sparksql, cloudera manager)
  - SSH terminal (for system-level exploration)
  - FTP client (for file exchange)

A MODEST LAB: THE INFRASTRUCTURE



3 physical servers with 5 virtual machines

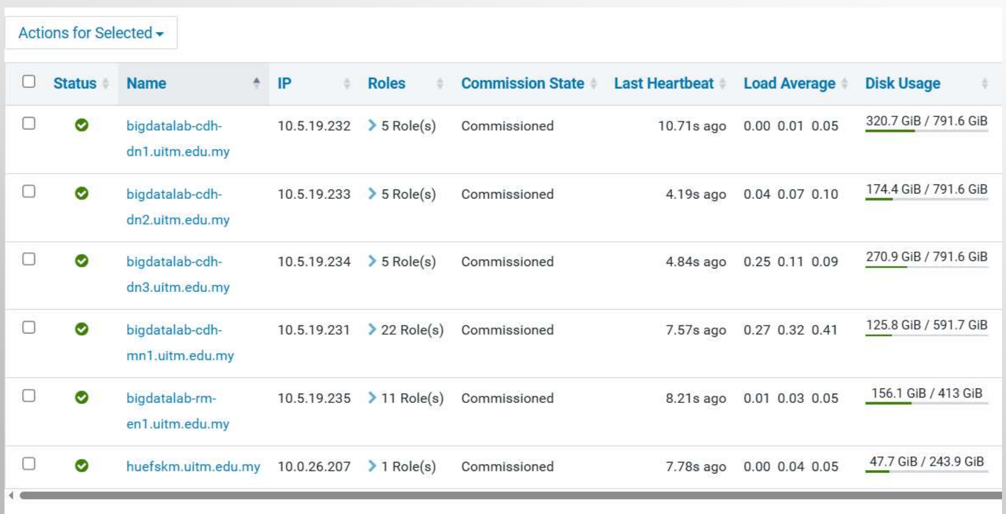
- **Master Node –** the *coordinator*, keeps track of what job goes where.
- **Data Nodes –** the *workers*, actually store and process pieces of the data.
- **Edge Node –** the *entry point*, where students accessed tools and submitted jobs.

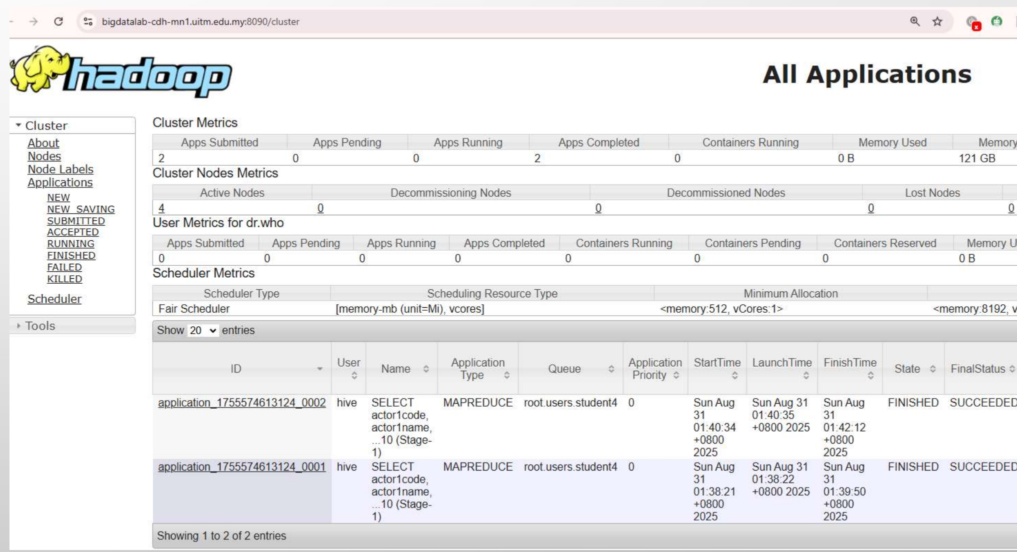
A MODEST LAB: THE INFRASTRUCTURE (WHY)

Why Set Up This Infrastructure (3 servers, 5 VMs)?

1.Pedagogical Need

1. Concept-heavy courses and notebooks hide the system layer.
2. Students rarely see how distributed systems *actually* work.
3. The infra made the *invisible visible*: nodes, job scheduling, monitoring.





https://bigdatalab-cdh-mn1.uitm.edu.my:7183/cmf/login

https://bigdatalab-cdh-mn1.uitm.edu.my:8090/cluster

A MODEST LAB: THE INFRASTRUCTURE (WHY)

Why Set Up This Infrastructure (3 servers, 5 VMs)?

2. Alignment with Hadoop's Core Features

1. **Distributed**: Even a small cluster showed data split across nodes.
2. **Reliable**: Students observed fault tolerance *and* the reality of service restarts.
3. **Commodity Hardware**: Demonstrated that Big Data principles can be learned without enterprise infrastructure.

A MODEST LAB: THE INFRASTRUCTURE (WHY)

Why Set Up This Infrastructure (3 servers, 5 VMs)?

3. Teaching Practicality

1. Small enough to manage with limited budget and support.
2. Big enough to mimic industry workflows (HDFS + YARN + Hive/Spark).

4. Scalable for Education

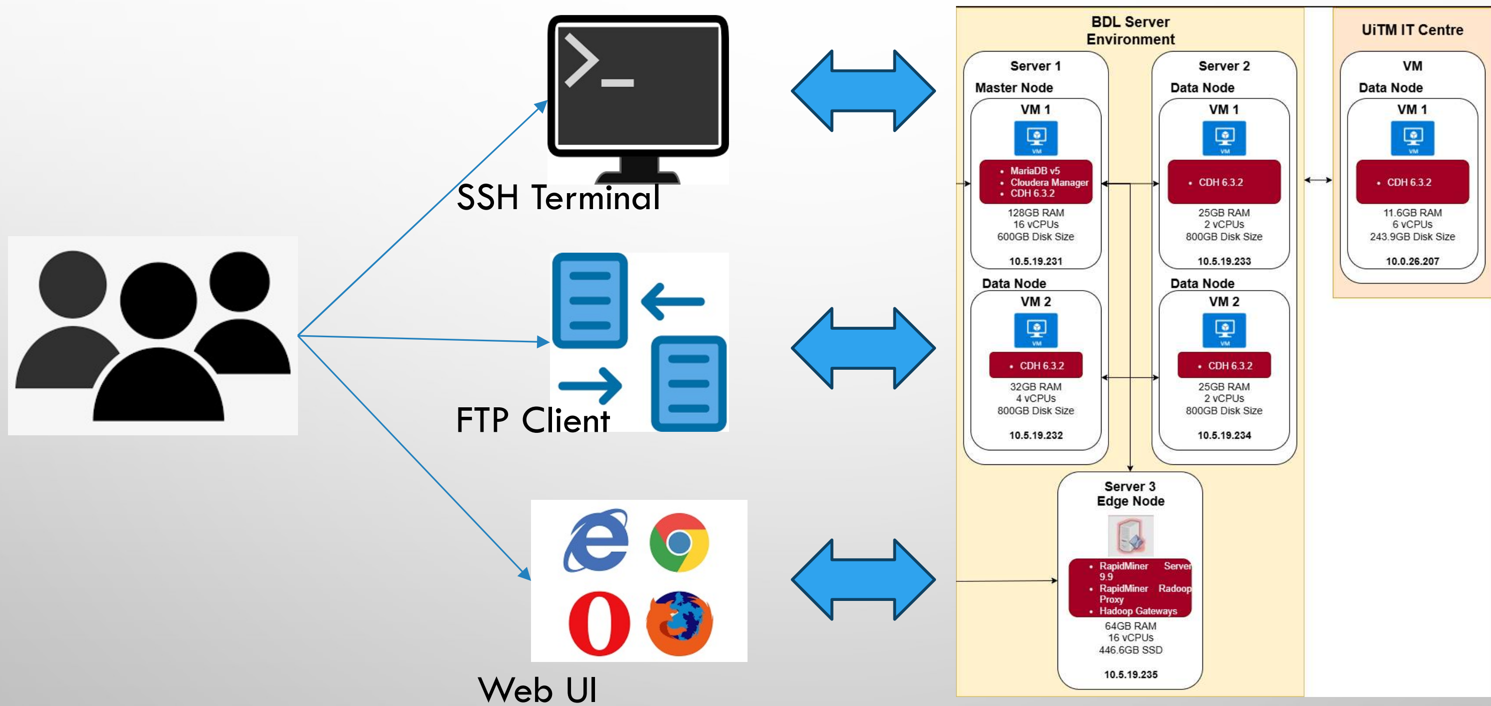
1. Reusable across courses, levels, and cohorts.
2. Flexible for both in-class and remote teaching.
3. Supports not just one-off labs, but assignments, projects, and FYPs.

A MODEST LAB THAT MIRRORS REAL-WORLD SYSTEMS

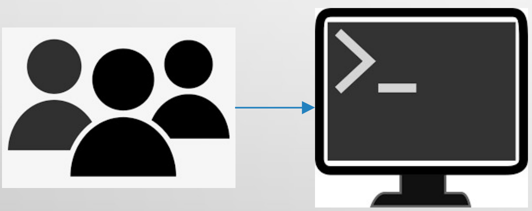
What we set up?

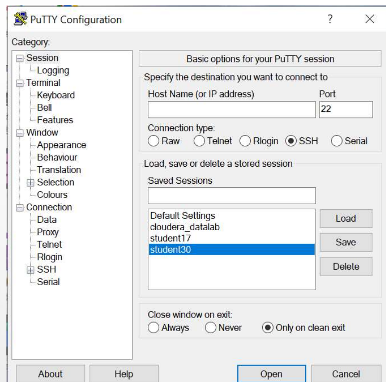
- Designed **a teaching lab environment** to simulate real big data analytic workflows.
- Deployed a computing cluster with vendor support, which allowed us to focus on designing for student learning.
- **Platform:**
  - Cloudera hadoop distribution (CDH)
  - Tools: hive, spark, impala, sqoop, HDFS, YARN, etc
- **Infrastructure**:
  - 3 physical servers with 5 virtual machines
  - Master node + data nodes configured for cluster-based processing
- **Student access modes:**
  - Web UI (for hive, impala, sparksql, cloudera manager)
  - SSH terminal (for system-level exploration)
  - FTP client (for file exchange)

A MODEST LAB: STUDENT ACCESS MODES

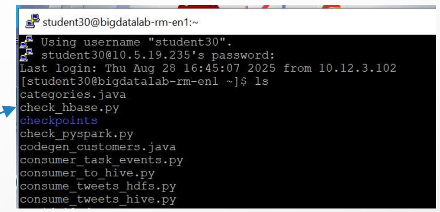


A MODEST LAB: STUDENT ACCESS MODES



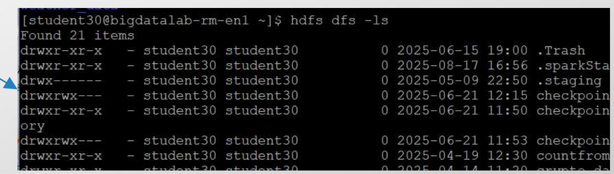


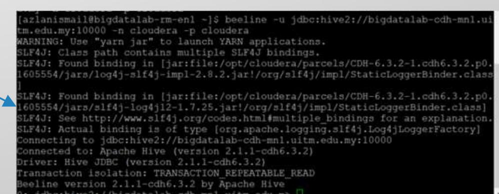
Using putty



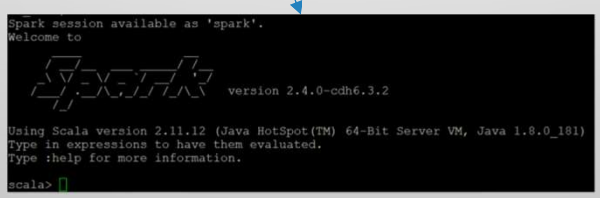
Accessing linux home directory

Accessing HDFS directory



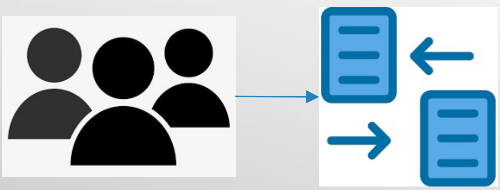


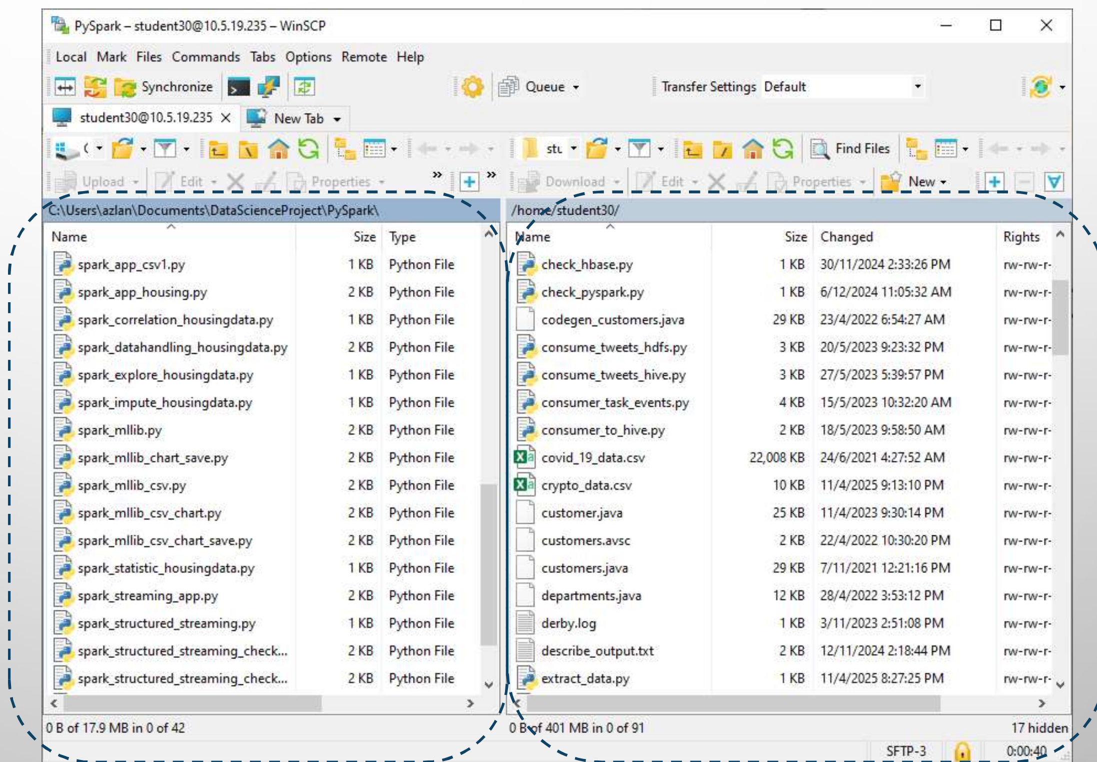
Accessing Hive Beeline



Accessing Spark

A MODEST LAB: STUDENT ACCESS MODES

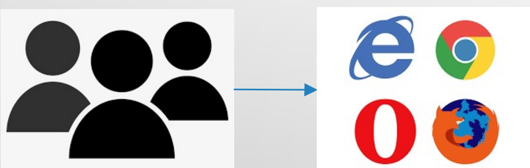


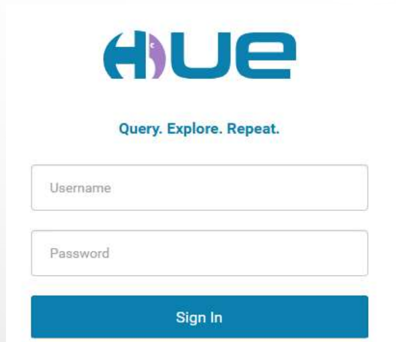


Local directory

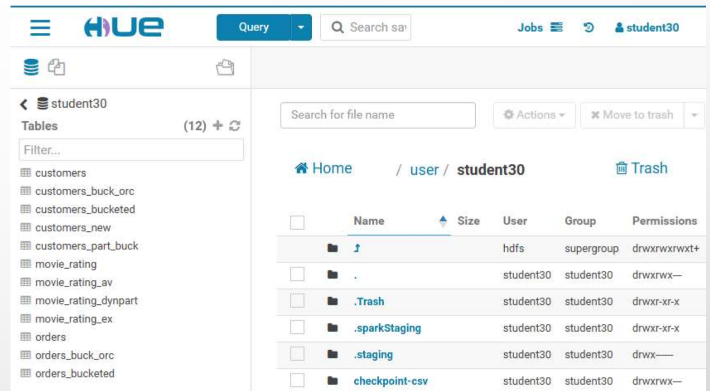
Remote directory

A MODEST LAB: STUDENT ACCESS MODES

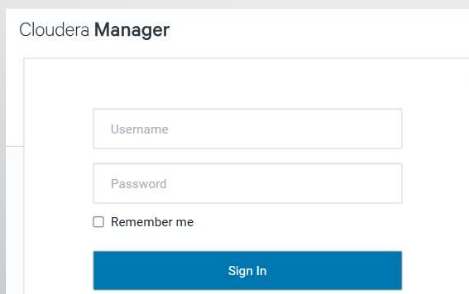




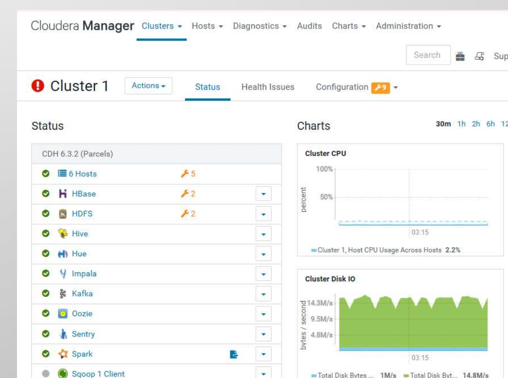
data engineer/analyst perspective



https://bigdatalab-rm-en1.uitm.edu.my:8889/hue/accounts/login?next=/



System administrator perspective

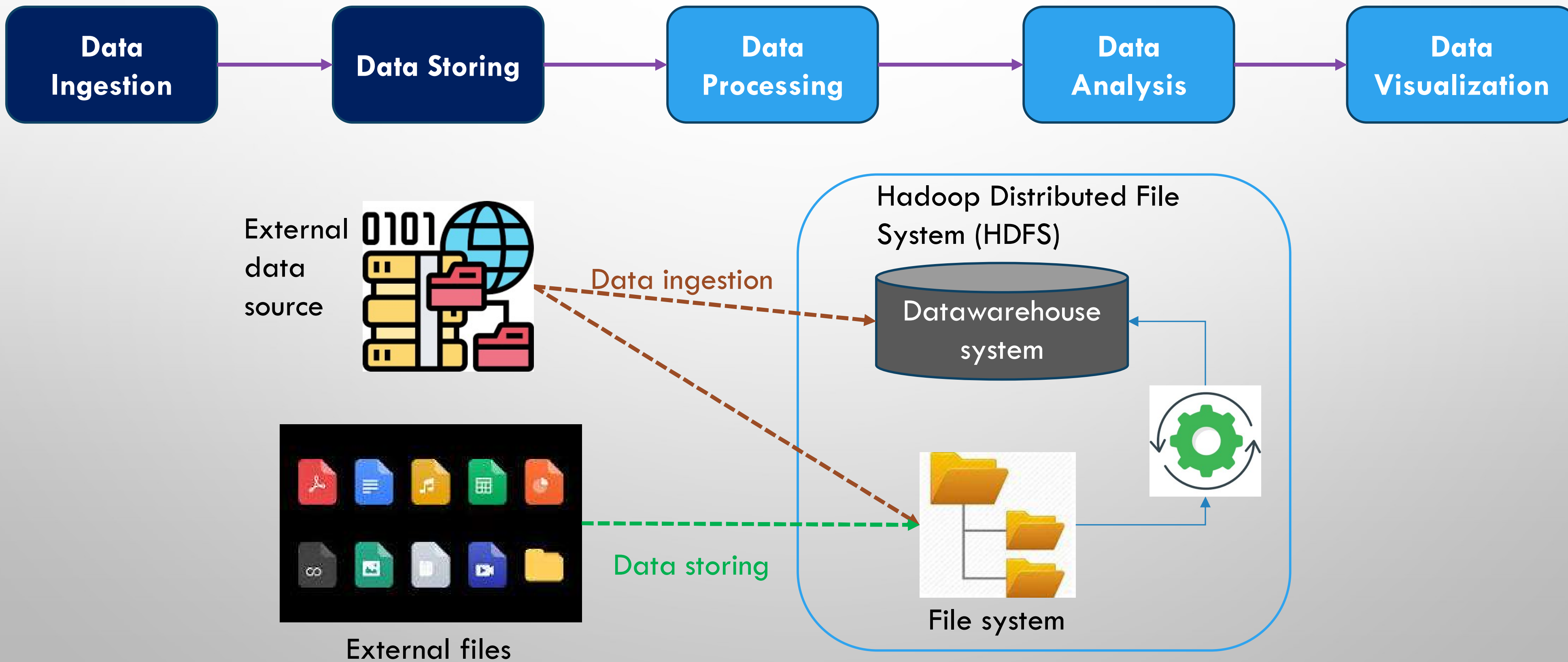


https://bigdatalab-cdh-mn1.uitm.edu.my:7183/cmf/login

FROM DOING TO UNDERSTANDING TO ASSIGNMENTS

- **From Theory to Action** – students performed end-to-end Big Data tasks (ingest, store, process, analyze).
- **From Execution to Insight** – they reflected on results and connected theory with system behavior.
- **Structured Assignments** – structured tasks captured these insights through benchmarking and reporting.

FROM THEORY TO ACTION

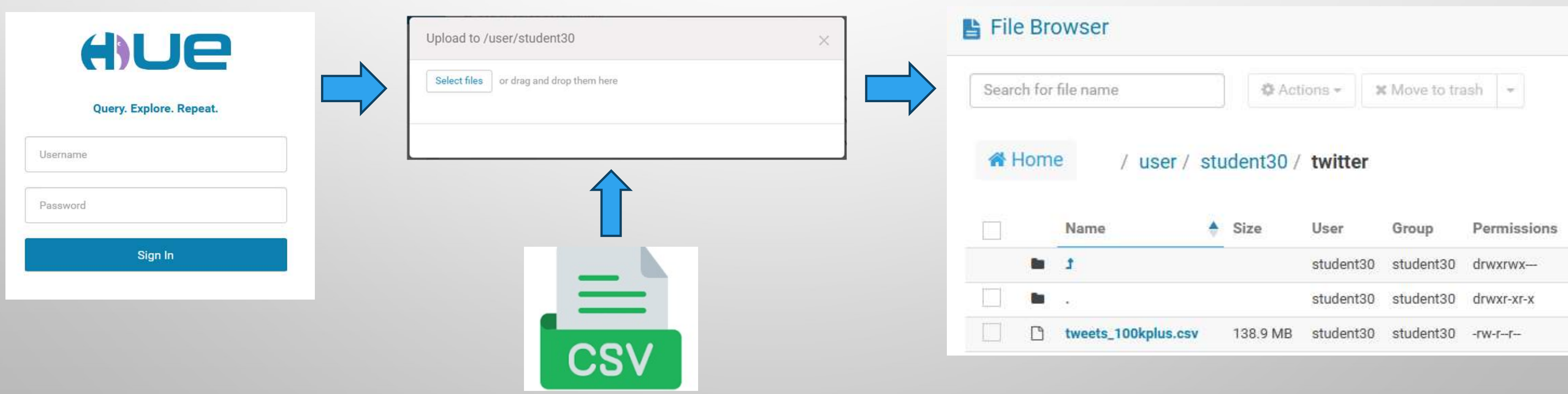


FROM THEORY TO ACTION

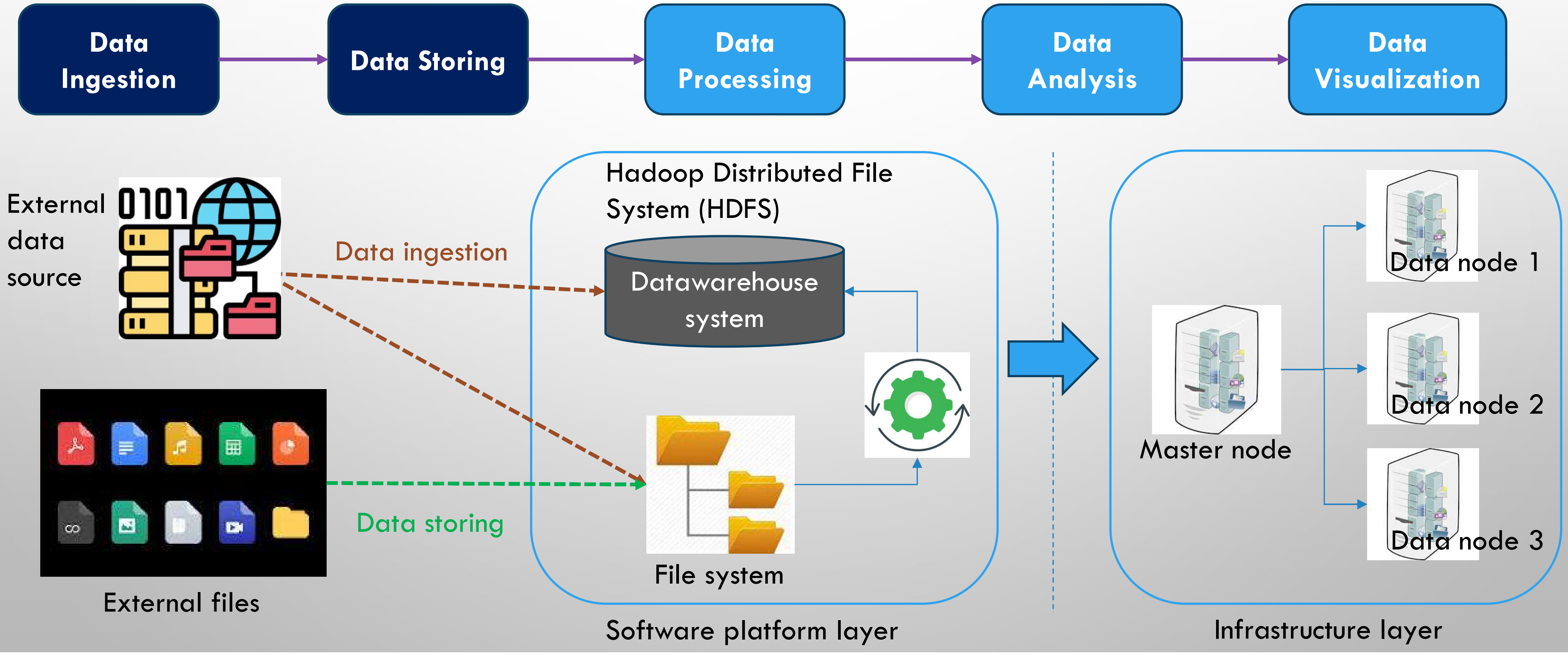

```
[student30@bigdatalab-rm-en1 ~]$ sqoop import --connect jdbc:mysql://bigdatalab-
rm-en1:3306/retail_db --username student --password p@ssw0rd  --table customers
--target-dir /user/student30/customers_comp  --compression-codec SnappyCodec  --
delete-target-dir
```

```
                Map input records=12435
                Map output records=12435
                Input split bytes=479
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=330
                CPU time spent (ms)=4870
                Physical memory (bytes) snapshot=1147584512
                Virtual memory (bytes) snapshot=13925732352
                Total committed heap usage (bytes)=1716518912
                Peak Map Physical memory (bytes)=298045440
                Peak Map Virtual memory (bytes)=3485007872
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=415825
22/04/22 22:19:58 INFO mapreduce.ImportJobBase: Transferred 406.0791 KB in
8 seconds (24.9925 KB/sec)
22/04/22 22:19:58 INFO mapreduce.ImportJobBase: Retrieved 12435 records.
```

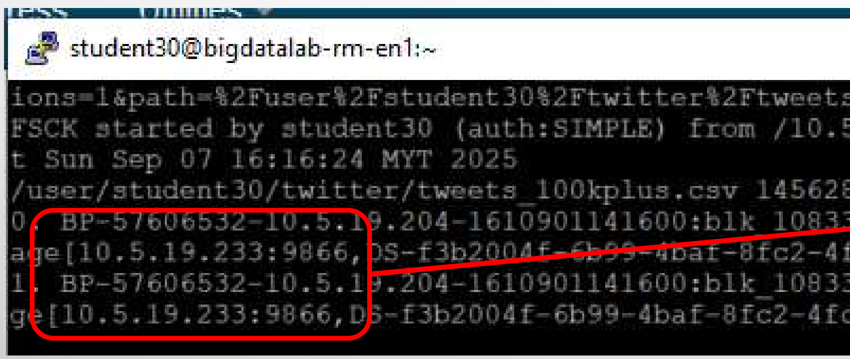
FROM THEORY TO ACTION

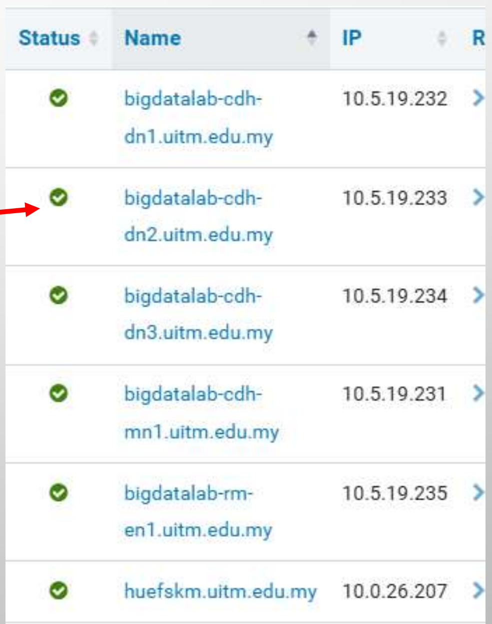


FROM THEORY TO ACTION



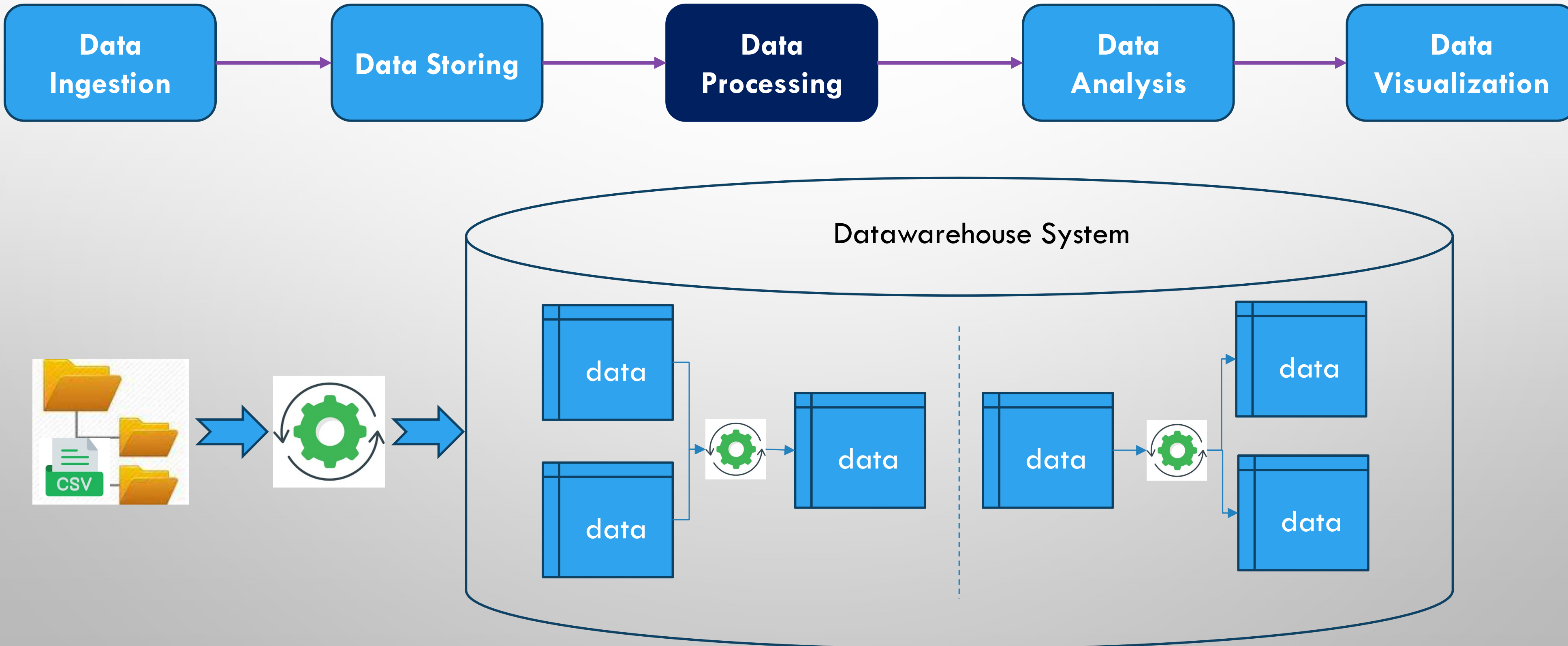
FROM THEORY TO ACTION





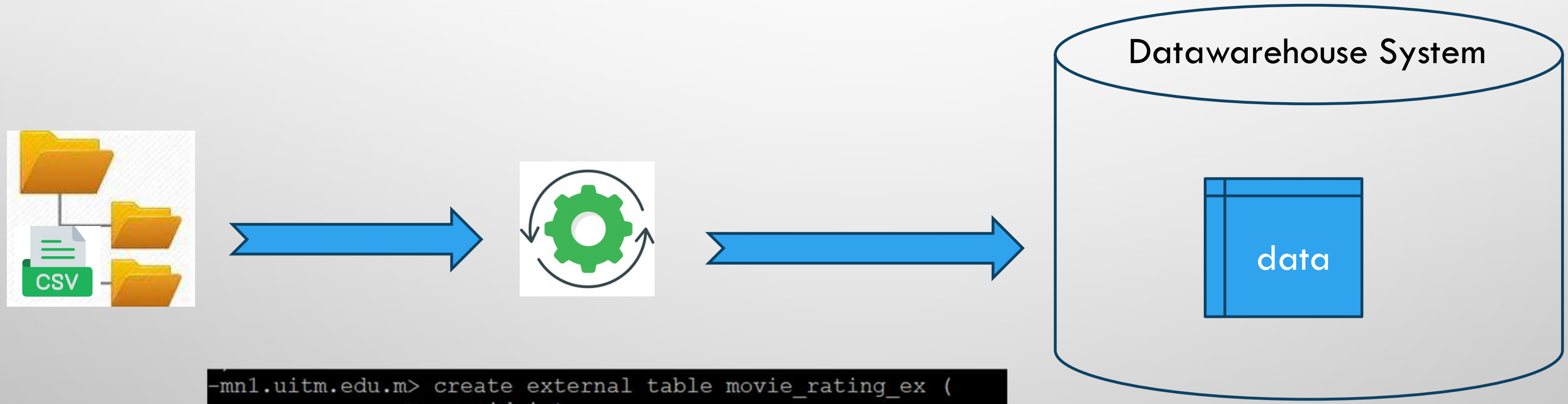
Status	Name	IP	R
	bigdatalab-cdh-dn1.uitm.edu.my	10.5.19.232	
	bigdatalab-cdh-dn2.uitm.edu.my	10.5.19.233	
	bigdatalab-cdh-dn3.uitm.edu.my	10.5.19.234	
	bigdatalab-cdh-mn1.uitm.edu.my	10.5.19.231	
	bigdatalab-rm-en1.uitm.edu.my	10.5.19.235	
	huefskm.uitm.edu.my	10.0.26.207	

FROM THEORY TO ACTION



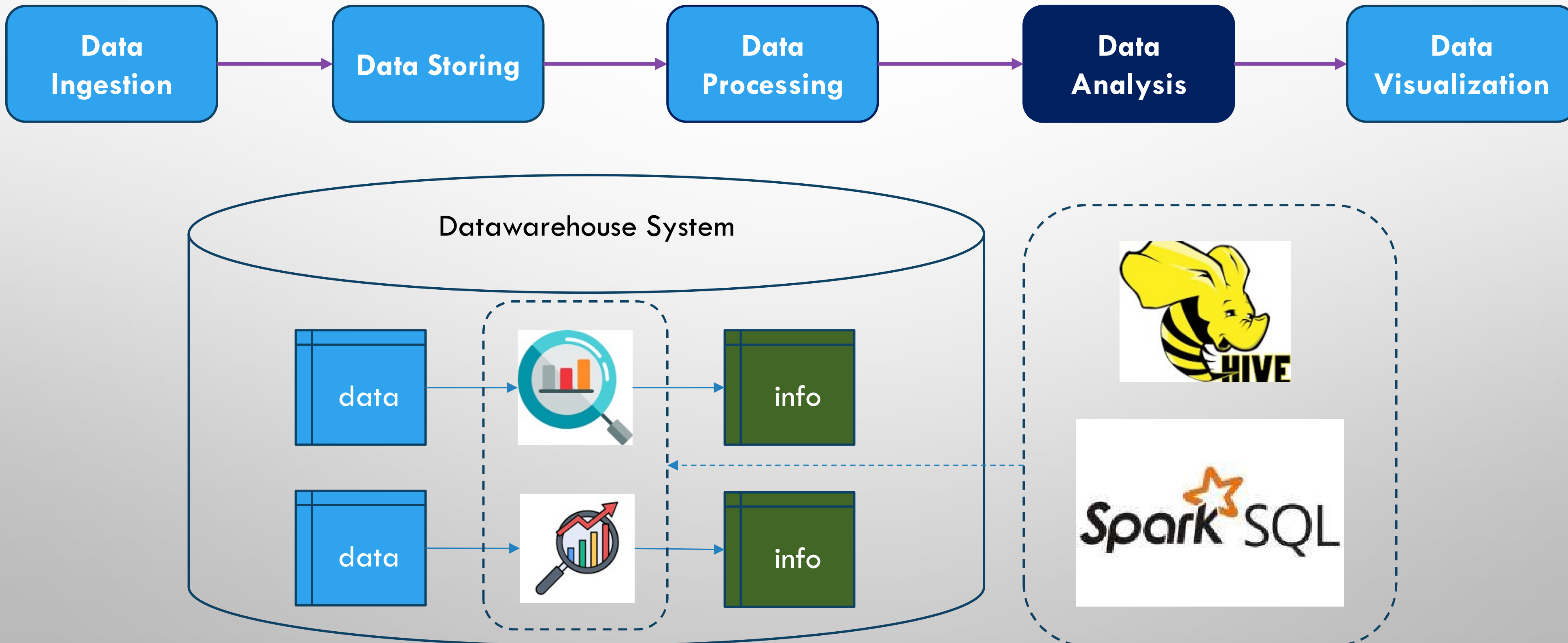
FROM THEORY TO ACTION

Data Ingestion → **Data Storing** → **Data Processing** → **Data Analysis** → **Data Visualization**

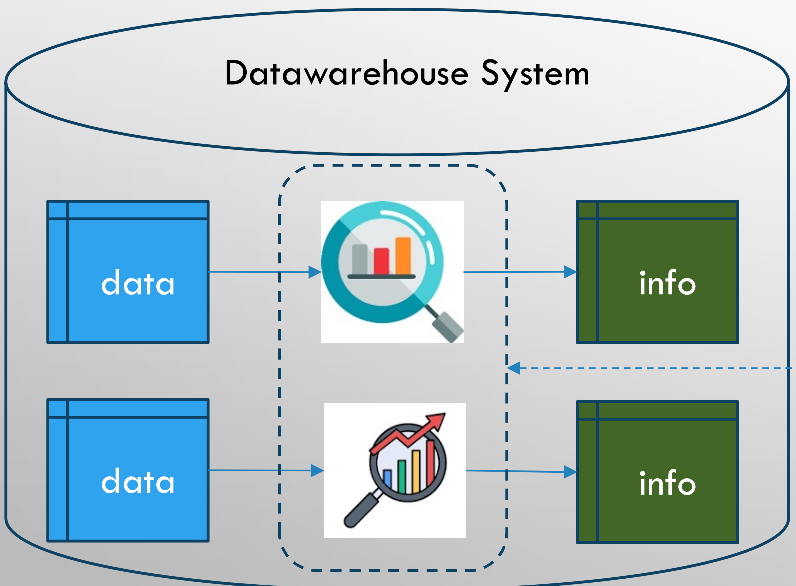



```
-mn1.uitm.edu.m> create external table movie_rating_ex (
. . . . . . . . .> userid int,
. . . . . . . . .> movieid int,
. . . . . . . . .> rating int,
. . . . . . . . .> unixtime string)
. . . . . . . . .> row format delimited
. . . . . . . . .> fields terminated by ','
. . . . . . . . .> location '/user/student30/movie_rating2';
```

FROM THEORY TO ACTION



FROM THEORY TO ACTION

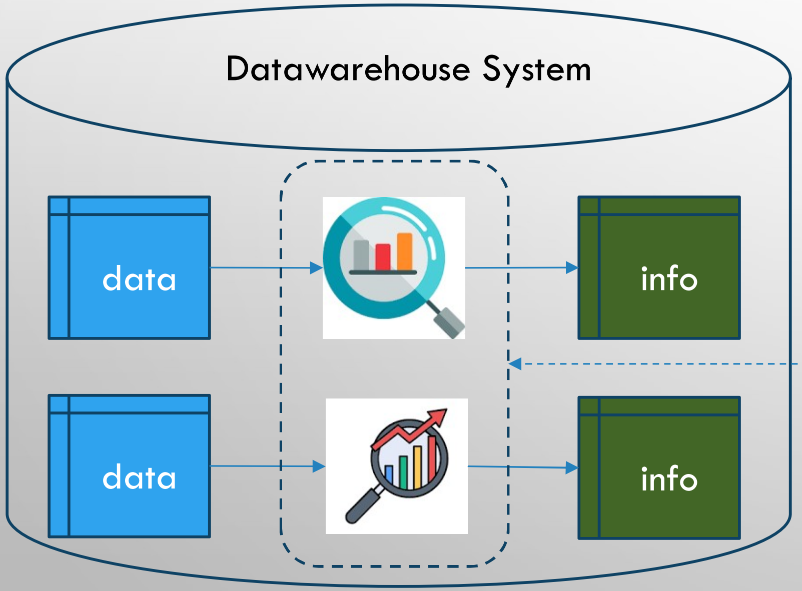


Spark SQL

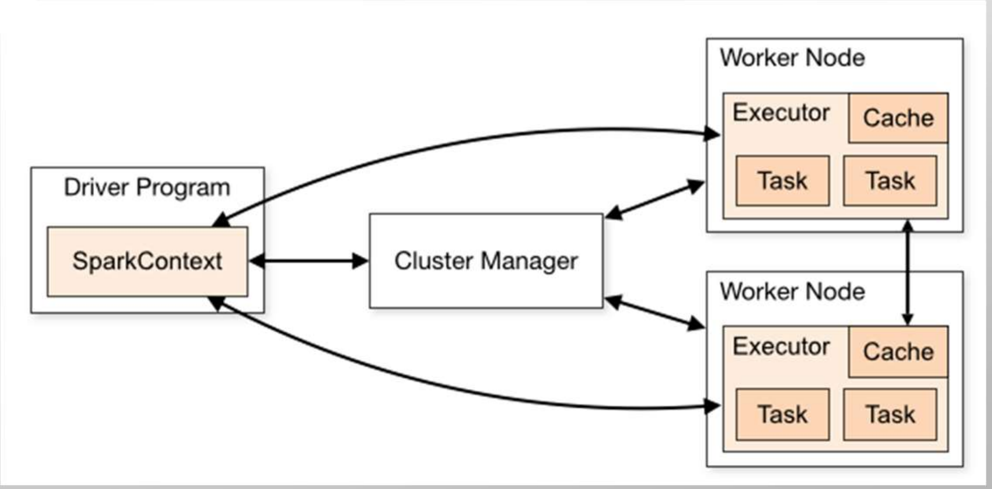
```python
# Create a new SparkSession with Hive support
hive_spark = SparkSession.builder \
    .appName("My Hive Spark Application") \
    .enableHiveSupport() \
    .getOrCreate()

# Set the database name
database_name = "student30"

# Convert the word frequency DataFrame to a temporary table
word_frequency_single_partition.createOrReplaceTempView("word_frequency_table")

# Save the temporary table as a Hive table in the specified database
hive_spark.sql(f"CREATE TABLE IF NOT EXISTS {database_name}.word_frequency_hive
                AS SELECT * FROM word_frequency_table")
```

FROM THEORY TO ACTION

Data Ingestion → Data Storing → Data Processing → **Data Analysis** → Data Visualization

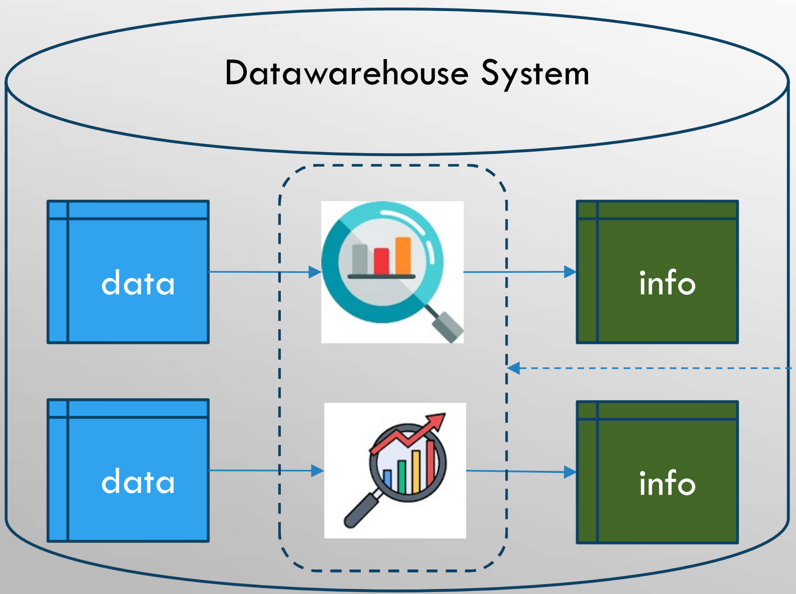


Spark SQL



FROM THEORY TO ACTION

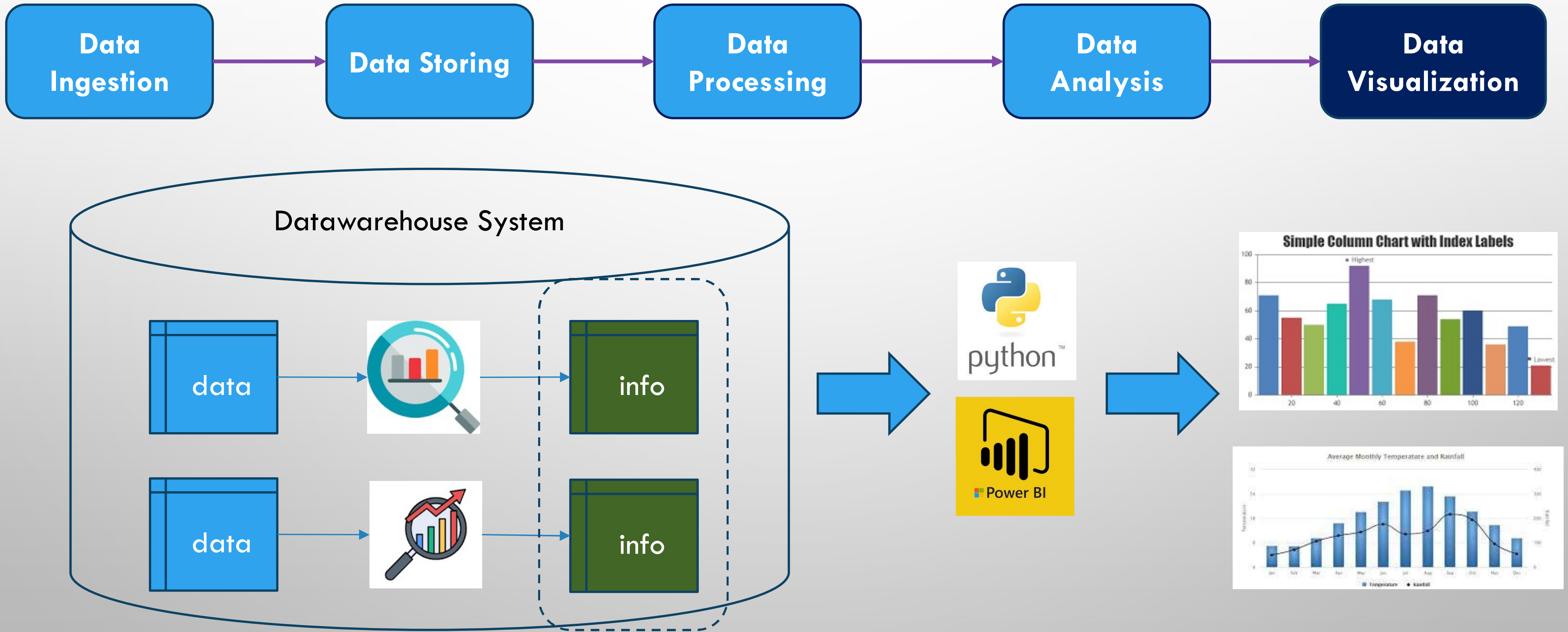
Data Ingestion → Data Storing → Data Processing → **Data Analysis** → Data Visualization



Spark SQL

Executor ID	Address	Status	RDD Blocks	Storage Memory	Disk Used	Cores	Active Tasks	Failed Tasks	Complete Tasks	Total Tasks	Task Time (GC Time)
driver	bigdatalab-rm-en1.uitm.edu.my:40946	Active	0	0.0 B / 384.1 MB	0.0 B	0	0	0	0	0	0 ms (0 ms)
1	bigdatalab-rm-en1.uitm.edu.my:38982	Active	0	0.0 B / 413 MB	0.0 B	1	0	0	295	295	27 s (0.6 s)
2	bigdatalab-rm-en1.uitm.edu.my:40435	Active	0	0.0 B / 413 MB	0.0 B	1	0	0	283	283	26 s (0.5 s)
3	bigdatalab-rm-en1.uitm.edu.my:37471	Active	0	0.0 B / 413 MB	0.0 B	1	0	0	47	47	10 s (0.3 s)
4	bigdatalab-cdh-dn2.uitm.edu.my:41871	Active	0	0.0 B / 384.1 MB	0.0 B	1	0	0	0	0	0 ms (0 ms)
5	bigdatalab-cdh-dn1.uitm.edu.my:33286	Active	0	0.0 B / 384.1 MB	0.0 B	1	0	0	1	1	9 s (0.2 s)
6	bigdatalab-cdh-dn1.uitm.edu.my:44715	Active	0	0.0 B / 384.1 MB	0.0 B	1	0	0	0	0	0 ms (0 ms)

FROM THEORY TO ACTION



FROM EXECUTION TO INSIGHT

What students learned from their actions

- ❑**File formats matter** → observed clear performance differences (text vs ORC).
  "I didn't expect file formats to matter so much, ORC was so much faster than Text."
- ❑**Design choices impact outcomes** → partitioning & bucketing changed query speed and efficiency.
  "Partitioning and bucketing really changed the way queries performed."
- ❑**Comparative mindset** → developed ability to analyze spark vs hive beyond syntax.
  "I could finally see why Spark behaves differently from Hive, even on the same data."
- ❑**System awareness** → monitoring with cloudera manager showed how jobs consumed resources.
  "Watching Cloudera Manager showed me how jobs actually use memory and CPU."
- ❑**Reflective learning** → moved from *running queries* to *explaining why results differ*.
  "It wasn't just about running queries, I learned to explain why the results differed."

ASSIGNMENT: FROM STRUCTURED TASKS TO INSIGHT

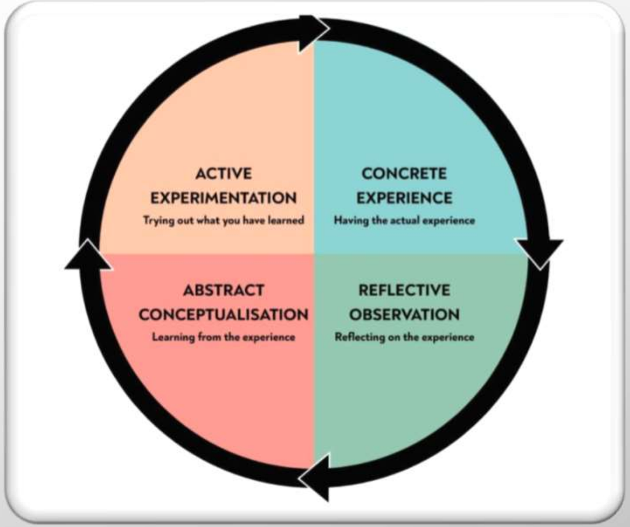
ASSIGNMENT: SPARK VS HIVE – COMPARATIVE BENCHMARKING

- **Objective:** evaluate Sparksql vs Hive across the same table setups to understand architectural and performance differences.
- **Student tasks:**
  - Get a dataset and upload it to HDFS.
  - Formulate the comparative benchmarking framework.
  - Create hive tables with different configurations (such Internal/external, Textfile/parquet, Partitioned, bucketed).
  - Run complex queries (with join & group-by) on each setup.
  - Collect measurements: such as execution time, Storage/file size.
  - Analyze trade-offs: query time vs file size vs table configuration.
  - Interpret findings using charts + discussion.
- **Learning focus:**
  - Storage design, hands-on benchmarking, memory-aware computing, interpreting system performance and understanding distributed processing behavior.

WHY KOLB'S EXPERIENTIAL CYCLE ANCHORED OUR COURSE DESIGN

Why kolb?

- Kolb's experiential learning theory supports **learning by doing**, followed by **reflection and experimentation** which is ideal for hands-on, technical courses.
- It provides a complete cycle.
- It is widely applied in computing education.
- "Many of our students didn't just run code, they reflected on why certain configurations worked better, formed new hypotheses, and tested again. That's kolb in action."



MAPPING KOLB'S CYCLE TO OUR ASSIGNMENTS

Kolb Stage	Student Activity
Concrete Experience	Students ran queries, loaded data, benchmarked Hive vs Spark
Reflective Observation	They reflected on why Parquet / ORC was faster, why bucketing / partitioning mattered
Abstract Conceptualization	They linked performance outcomes to data structure configurations / system design principles
Active Experimentation	They changed configs, reran jobs, tested new hypotheses

WHY THIS APPROACH IS SCALABLE AND REUSABLE

- **Modest infrastructure, real impact**
  - Deploying cluster environment: 3 servers, 5 VMs
  - Stable enough for multi-user access big data operations
- **Designed for teaching, not production**
  - No need for complex cloud services or autoscaling
  - Students accessed real systems, not emulators or simplified GUIs
- **Reusable across courses and levels**
  - Can be used for postgraduate and undergraduate related courses,
  - Python analytics and benchmarking projects, and FYPs
- **Remote and in-class friendly**
  - Easy access for students regardless of learning mode (Intranet, Internet)

WHY NOT USE AWS OR GOOGLE CLOUD?

Cloud platforms are powerful, but not always practical for teaching

- Free tiers are limited or time-limited
- Budget and billing control are difficult for classroom settings

Limited control and visibility

- Hard to expose students to system-level details (YARN memory, spark logs, file system behavior)

Repeatability and reset issues

- Labs are harder to reset or reuse consistently across semesters

Our setup prioritizes learning over cloud elasticity

- It's scalable for teaching, not for enterprise compute loads
- Students interact directly with tools *and* the environment

FUTURE DIRECTIONS

Scaling up the lab

- ❑ Expanding the current cluster to support larger datasets and multi-course usage.

Cloud integration

- ❑ Introducing AWS academy / AWS educate modules to complement hadoop fundamentals.

Student pathways

- ❑ Linking course outcomes to certifications (e.G. AWS data engineer certification).

Collaboration

- ❑ Inviting colleagues to utilize the scaled-down lab for teaching and research testbed (e.G. For pilot experiments, applied domain research).

CONCLUSION

1

You don't need a cloud budget or enterprise cluster to teach big data meaningfully.

2

A well-designed, scaled-down lab can expose students to:

- Distributed storage
- Query optimization
- Performance tuning
- System-level thinking

3

The learning was **not just technical,** it was **reflective, iterative, and empowering**


Home > Education and Information Technologies > Article

Data science technology course: The design, assessment and computing environment perspectives

Published: 24 January 2023

Volume 28, pages 10209–10234, (2023) Cite this article

Download PDF

Access provided by Universiti Teknologi MARA

Azlan Ismail, Sofianita Mutalib & Haryani Haron

Published article: https://link.springer.com/article/10.1007/s10639-022-11558-8