Optimizing Apache Spark

Cluster Configurations Scenarios

Cluster Configurations Scenarios

Getting Started...

Taking into consideration everything we know now...

- Who will be using the cluster?
- What will the cluster be used for?
- Where will the cluster and/or data reside?
- When are the results needed?
- How do I control/predict the costs?

Can we predict, for a given scenario, which cluster configuration and set of features will best meet the needs of each specific scenario?

Cluster Configurations Scenarios
“It Depends”

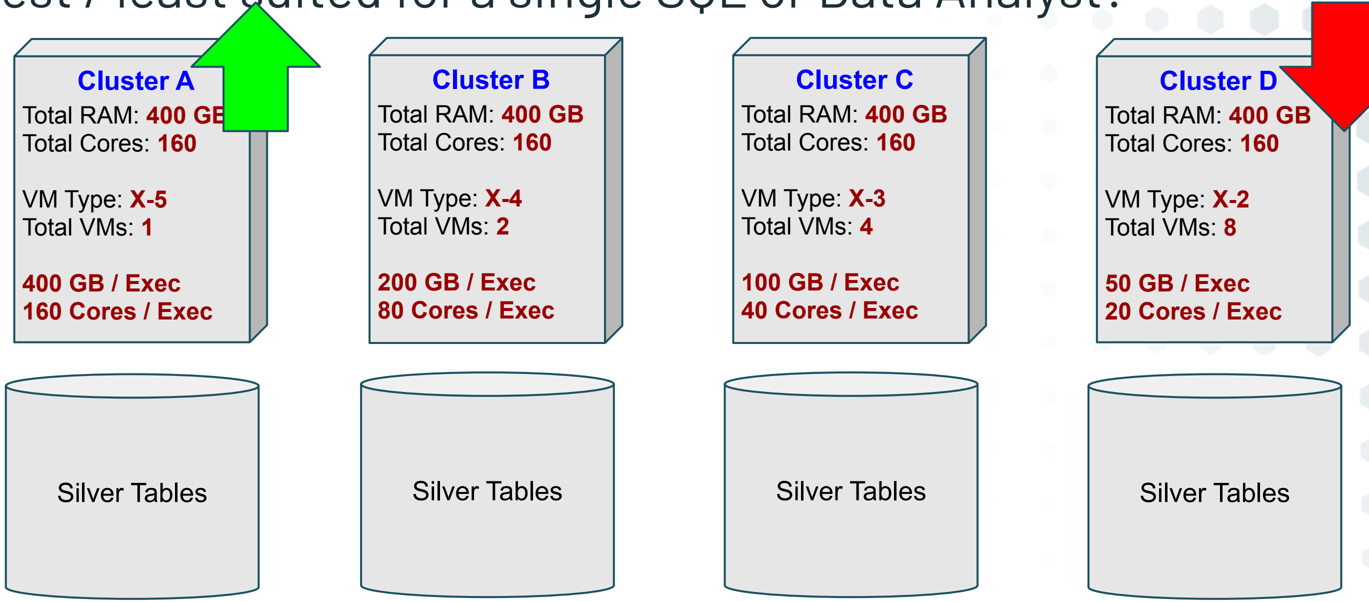
Really… it does depend…

- There is rarely a black or white, right or wrong, answer
- There are many different factors that could justify various decisions
- The conclusions presented here are generalizations only

Cluster Configurations Scenarios

Typical Analyst

Which of the following cluster configurations is best / least suited for a single SQL or Data Analyst?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

Silver Tables

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

Silver Tables

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

Silver Tables

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

Silver Tables

Category

- ❑ Memory Optimized
- ❑ Compute Optimized
- 🟡 Storage Optimized
- ❑ GPU Optimized
- ❑ General Purpose

Features

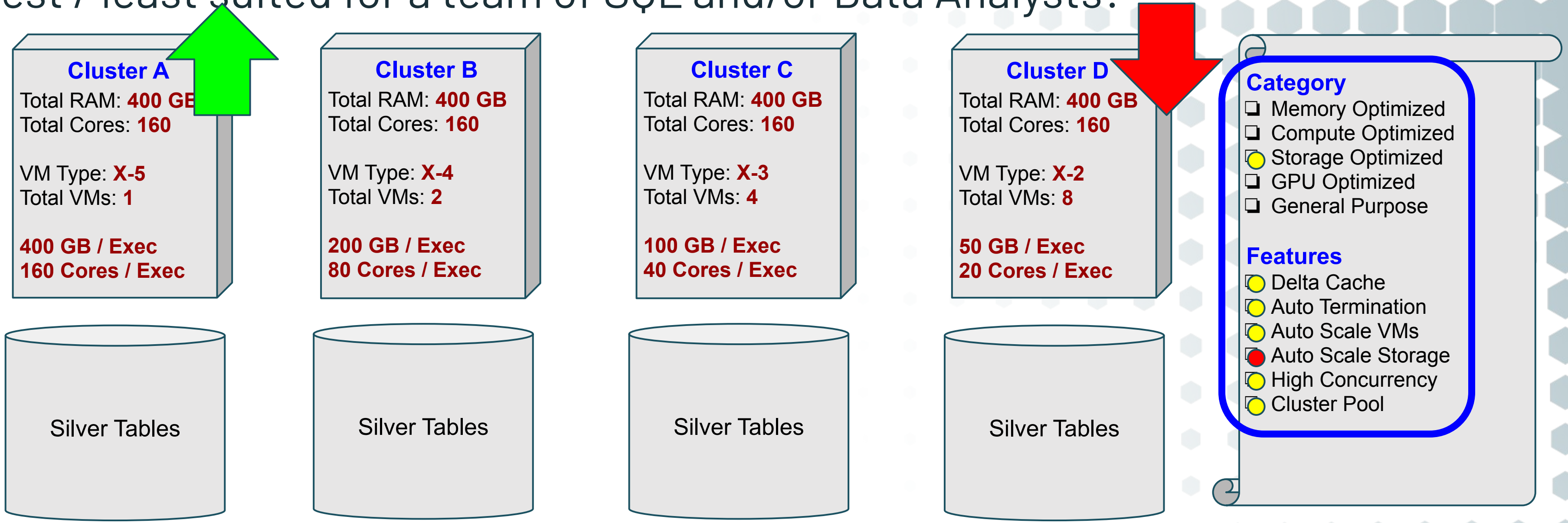
- 🟡 Delta Cache
- 🟡 Auto Termination
- 🟠 Auto Scale VMs
- 🔴 Auto Scale Storage
- 🔴 High Concurrency
- 🟡 Cluster Pool

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios

Team of Analyst

Which of the following cluster configurations is best / least suited for a team of SQL and/or Data Analysts?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

Silver Tables

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

Silver Tables

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

Silver Tables

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

Silver Tables

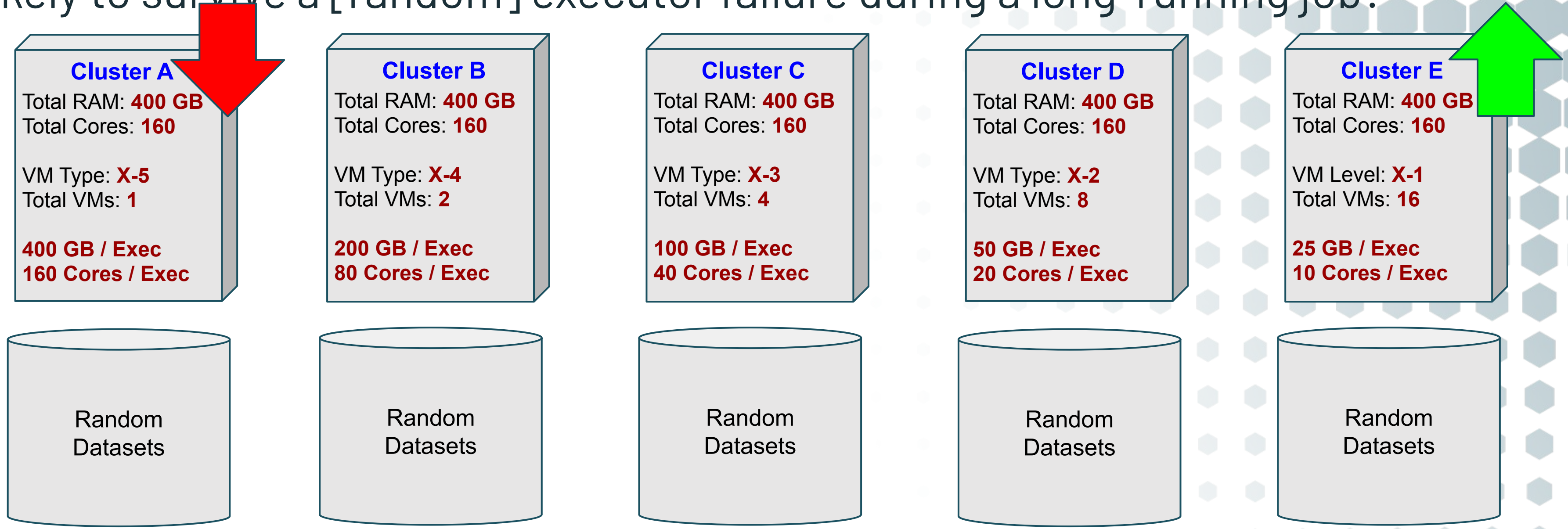
Category
- ❑ Memory Optimized
- ❑ Compute Optimized
- 🟡 Storage Optimized
- ❑ GPU Optimized
- ❑ General Purpose

Features
- 🟡 Delta Cache
- 🟡 Auto Termination
- 🟡 Auto Scale VMs
- 🔴 Auto Scale Storage
- 🟡 High Concurrency
- 🟡 Cluster Pool

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios
Cluster Stability

Which of the following cluster configurations is most / least likely to survive a [random] executor failure during a long-running job?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

Random Datasets

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

Random Datasets

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

Random Datasets

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

Random Datasets

Cluster E
Total RAM: **400 GB**
Total Cores: **160**

VM Level: **X-1**
Total VMs: **16**

25 GB / Exec
10 Cores / Exec

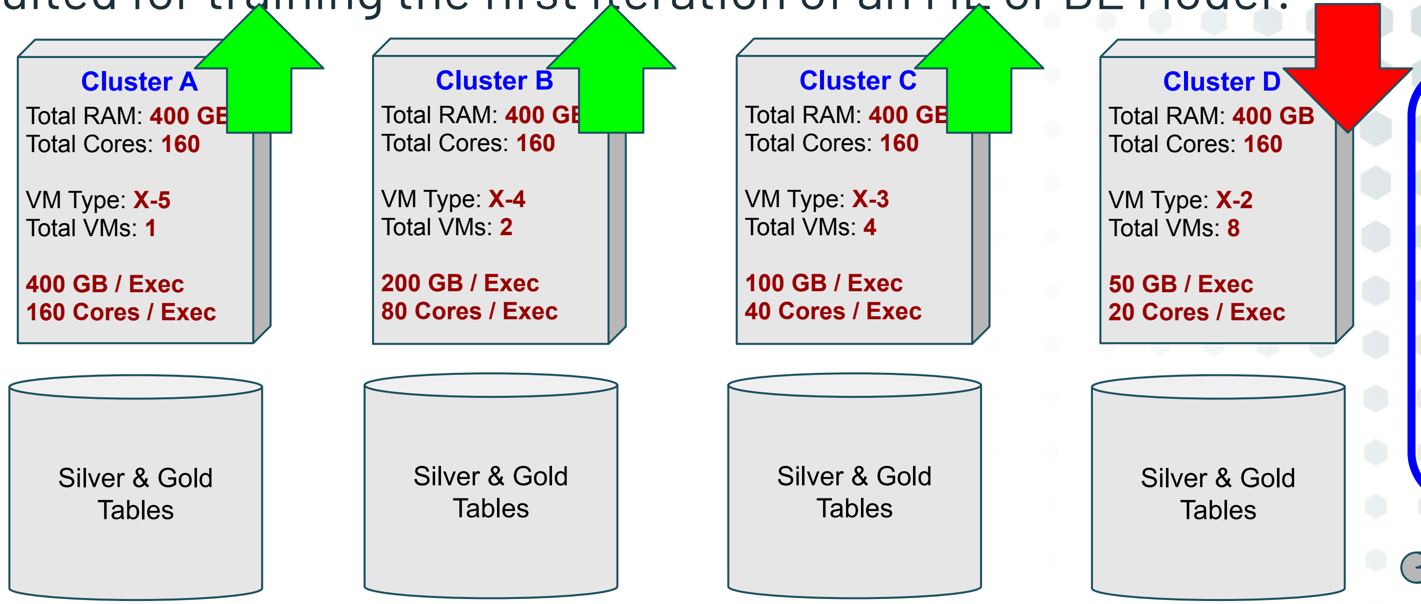
Random Datasets

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios

Training ML Models, 1st Iteration

Which of the following cluster configurations is best / least suited for training the first iteration of an ML or DL Model?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

Silver & Gold Tables

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

Silver & Gold Tables

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

Silver & Gold Tables

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

Silver & Gold Tables

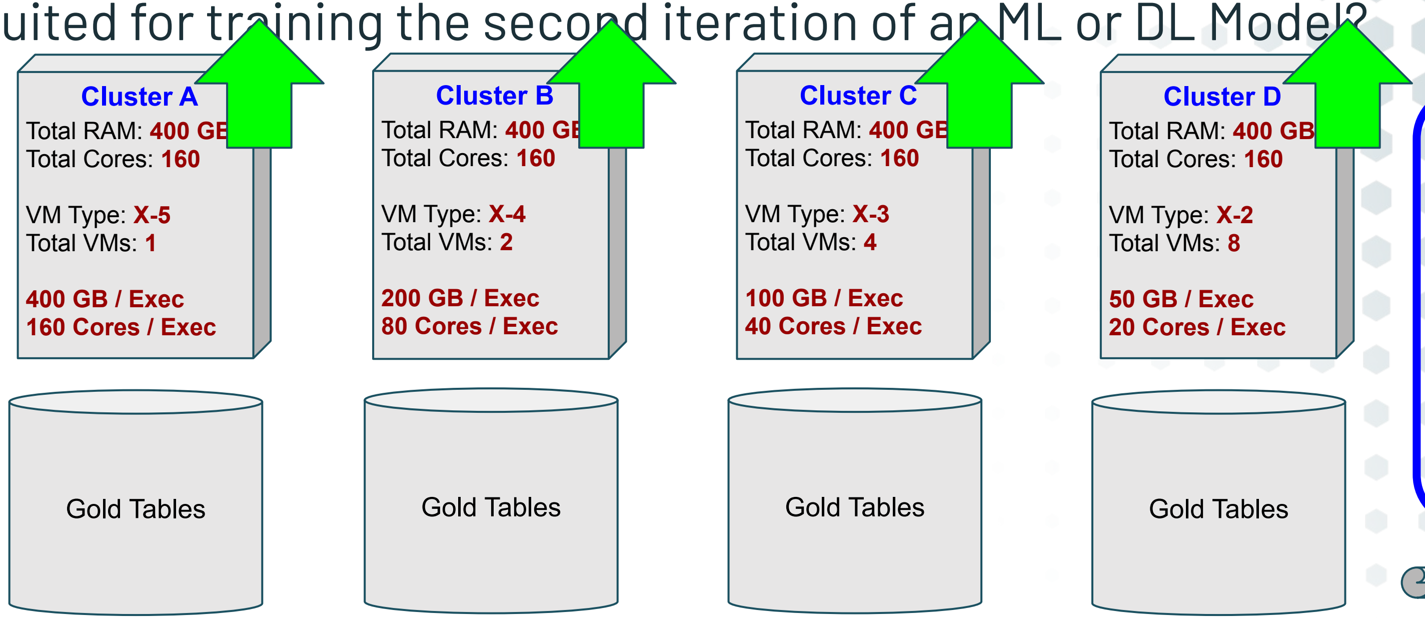
Category
- ❑ Memory Optimized
- ❑ Compute Optimized
- 🟡 Storage Optimized
- 🟠 GPU Optimized
- ❑ General Purpose

Features
- 🟡 Delta Cache
- 🟡 Auto Termination
- 🔴 Auto Scale VMs
- 🔴 Auto Scale Storage
- 🔴 High Concurrency
- 🟡 Cluster Pool

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios
Training ML Models, 2nd+ Iteration

Which of the following cluster configurations is best / least suited for training the second iteration of an ML or DL Model?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

Gold Tables

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

Gold Tables

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

Gold Tables

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

Gold Tables

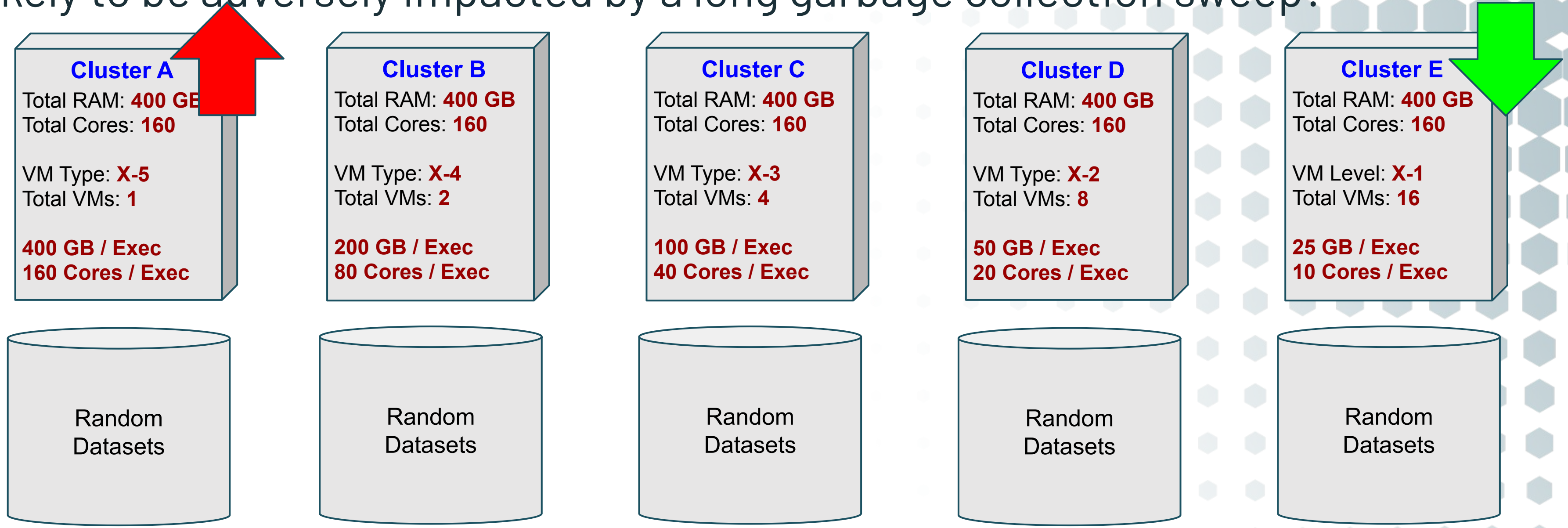
Category
- ❑ Memory Optimized
- ❑ Compute Optimized
- 🟡 Storage Optimized
- 🟡 GPU Optimized
- ❑ General Purpose

Features
- 🔴 Delta Cache
- 🟡 Auto Termination
- 🔴 Auto Scale VMs
- 🔴 Auto Scale Storage
- 🔴 High Concurrency
- 🔴 Cluster Pool

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios
Garbage Collection

Which of the following cluster configurations is most / least likely to be adversely impacted by a long garbage collection sweep?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

Cluster E
Total RAM: **400 GB**
Total Cores: **160**

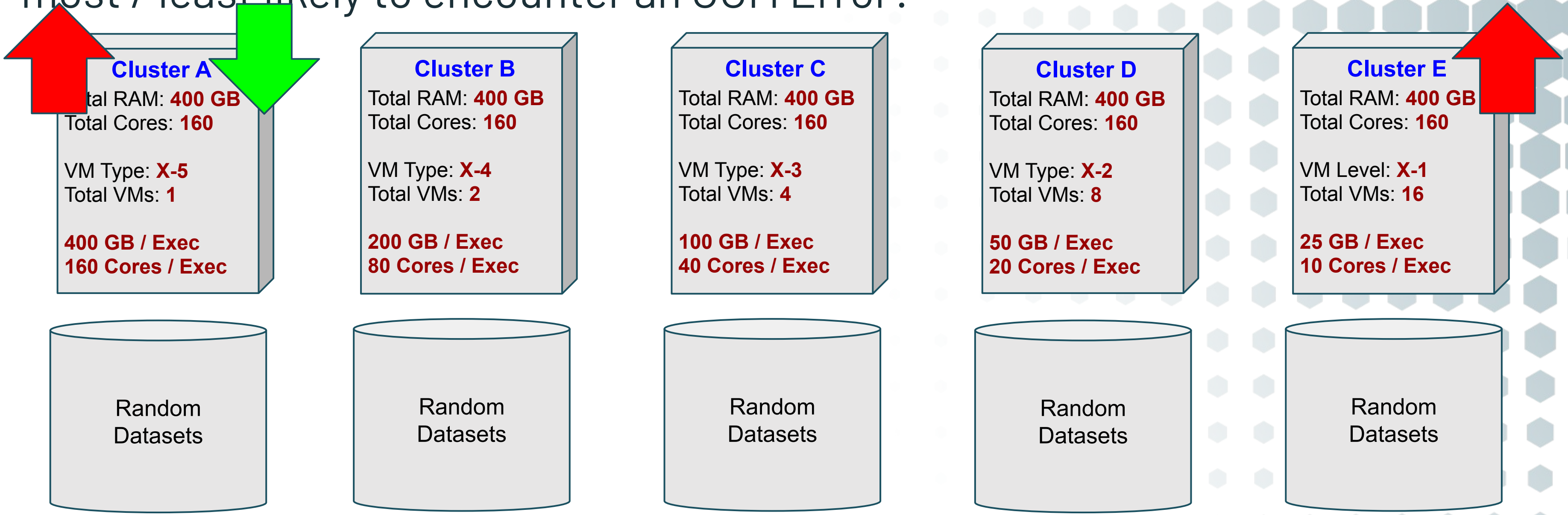
VM Level: **X-1**
Total VMs: **16**

25 GB / Exec
10 Cores / Exec

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios
General OOM Error

Which of the following cluster configurations is most / least likely to encounter an OOM Error?



Cluster A
tal RAM: 400 GB
Total Cores: 160

VM Type: X-5
Total VMs: 1

400 GB / Exec
160 Cores / Exec

Random Datasets

Cluster B
Total RAM: 400 GB
Total Cores: 160

VM Type: X-4
Total VMs: 2

200 GB / Exec
80 Cores / Exec

Random Datasets

Cluster C
Total RAM: 400 GB
Total Cores: 160

VM Type: X-3
Total VMs: 4

100 GB / Exec
40 Cores / Exec

Random Datasets

Cluster D
Total RAM: 400 GB
Total Cores: 160

VM Type: X-2
Total VMs: 8

50 GB / Exec
20 Cores / Exec

Random Datasets

Cluster E
Total RAM: 400 GB
Total Cores: 160

VM Level: X-1
Total VMs: 16

25 GB / Exec
10 Cores / Exec

Random Datasets

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios

Caching Induced OOM Error

Which of the following usage cases is most / least likely to induce an OOM Error induced by caching?

An ETL Job that is consuming CSV data, updating data types, removing duplicates and then writing it out to parquet

Not Caching

A report that joins three tables and writes the result to a Delta table used by BI tools

Not Caching

A team of 5 analyst engaged in heavy, ad hoc analysis against a single shared cluster

Excessive Caching

A data scientist that is training the first iteration of a model against a 1,000 GB dataset

Heavy Caching

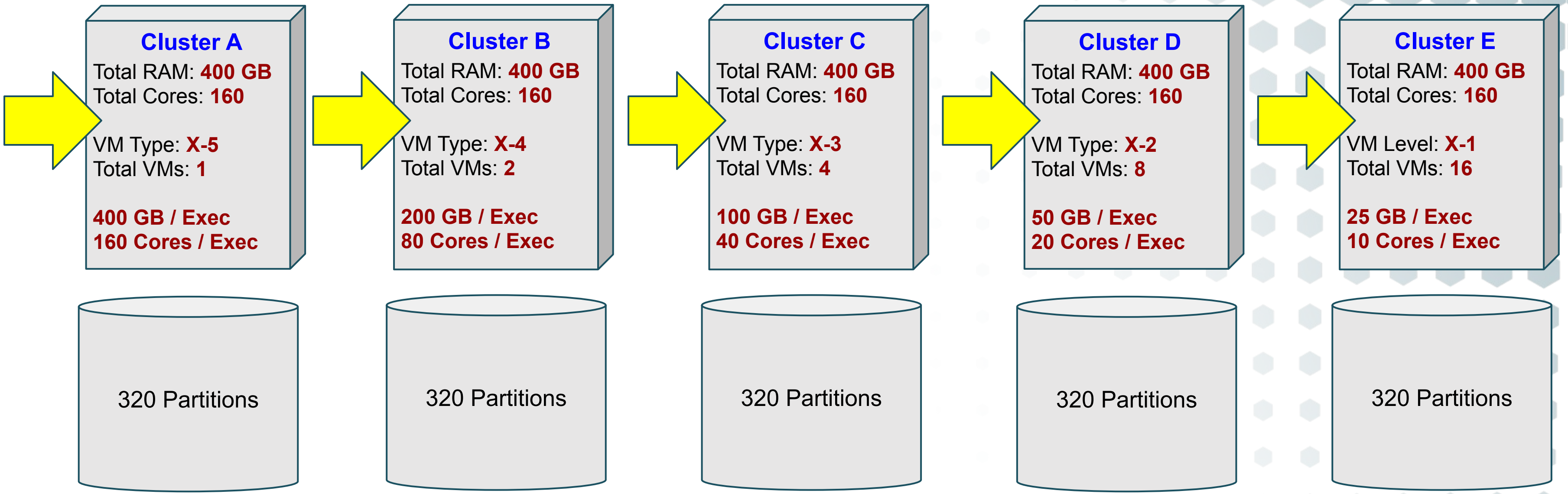
A single analyst attempting to validate sales-tax calculations for the previous year against a well formed 100 GB dataset

Light Caching

Cluster Configurations Scenarios

More Cores == More Money Version #1

Assuming the data in 320 partitions is equally distributed, which cluster configuration will cost the most / least amount of money for this job?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**
VM Type: **X-5**
Total VMs: **1**
400 GB / Exec
160 Cores / Exec

320 Partitions

Cluster B
Total RAM: **400 GB**
Total Cores: **160**
VM Type: **X-4**
Total VMs: **2**
200 GB / Exec
80 Cores / Exec

320 Partitions

Cluster C
Total RAM: **400 GB**
Total Cores: **160**
VM Type: **X-3**
Total VMs: **4**
100 GB / Exec
40 Cores / Exec

320 Partitions

Cluster D
Total RAM: **400 GB**
Total Cores: **160**
VM Type: **X-2**
Total VMs: **8**
50 GB / Exec
20 Cores / Exec

320 Partitions

Cluster E
Total RAM: **400 GB**
Total Cores: **160**
VM Level: **X-1**
Total VMs: **16**
25 GB / Exec
10 Cores / Exec

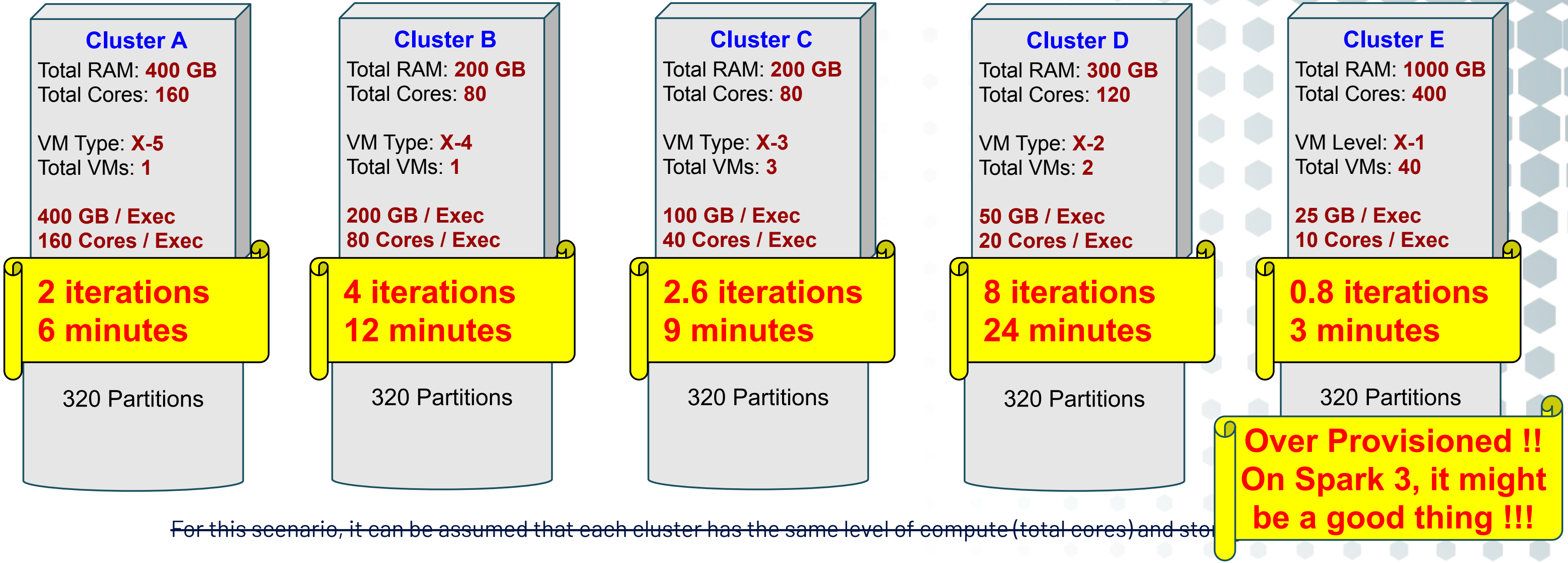
320 Partitions

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios

More Cores == More Money Version #2

Assuming each partition takes 3 minutes to process... Calculate the **compute-time**, **number of iterations** and **run-time** for each scenario:



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

2 iterations
6 minutes

320 Partitions

Cluster B
Total RAM: **200 GB**
Total Cores: **80**

VM Type: **X-4**
Total VMs: **1**

200 GB / Exec
80 Cores / Exec

4 iterations
12 minutes

320 Partitions

Cluster C
Total RAM: **200 GB**
Total Cores: **80**

VM Type: **X-3**
Total VMs: **3**

100 GB / Exec
40 Cores / Exec

2.6 iterations
9 minutes

320 Partitions

Cluster D
Total RAM: **300 GB**
Total Cores: **120**

VM Type: **X-2**
Total VMs: **2**

50 GB / Exec
20 Cores / Exec

8 iterations
24 minutes

320 Partitions

Cluster E
Total RAM: **1000 GB**
Total Cores: **400**

VM Level: **X-1**
Total VMs: **40**

25 GB / Exec
10 Cores / Exec

0.8 iterations
3 minutes

320 Partitions

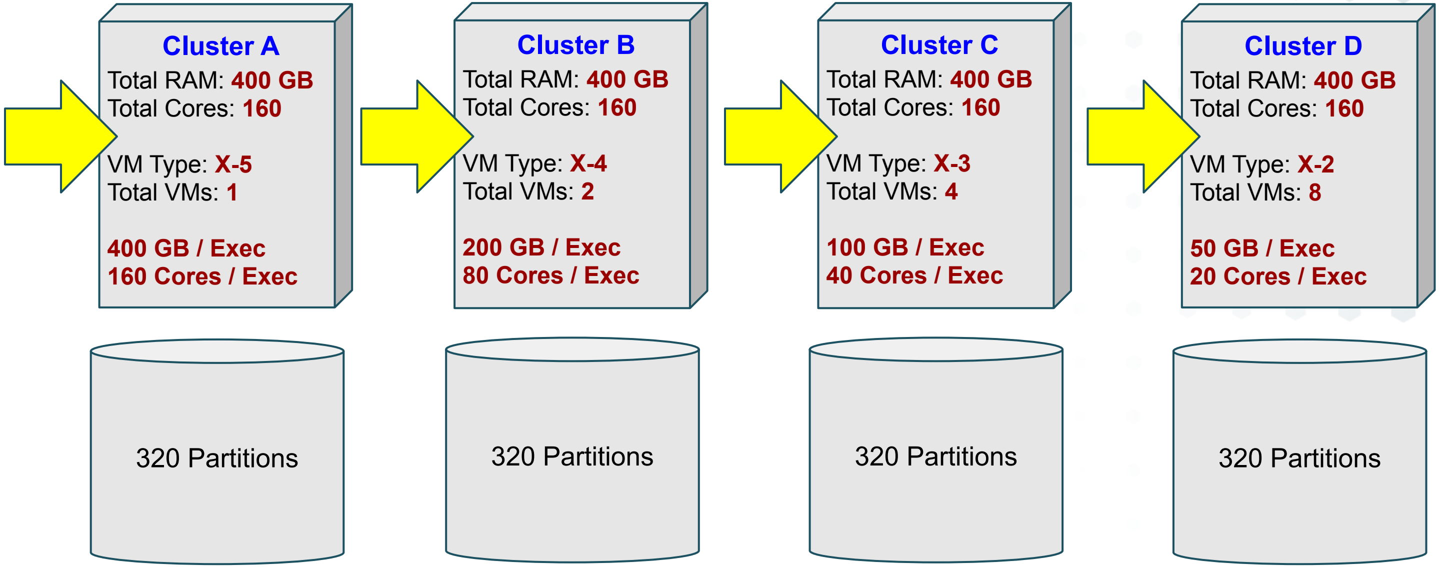
Over Provisioned !!
On Spark 3, it might be a good thing !!!

~~For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and sto~~

Cluster Configurations Scenarios

Batch ETL: Raw -> Bronze

Which of the following cluster configurations is best / least suited for a simple ETL job that does not employ wide transformations (no joins)?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

320 Partitions

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

320 Partitions

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

320 Partitions

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

320 Partitions

Category
- ❑ Memory Optimized
- 🟡 Compute Optimized
- ❑ Storage Optimized
- ❑ GPU Optimized
- ❑ General Purpose

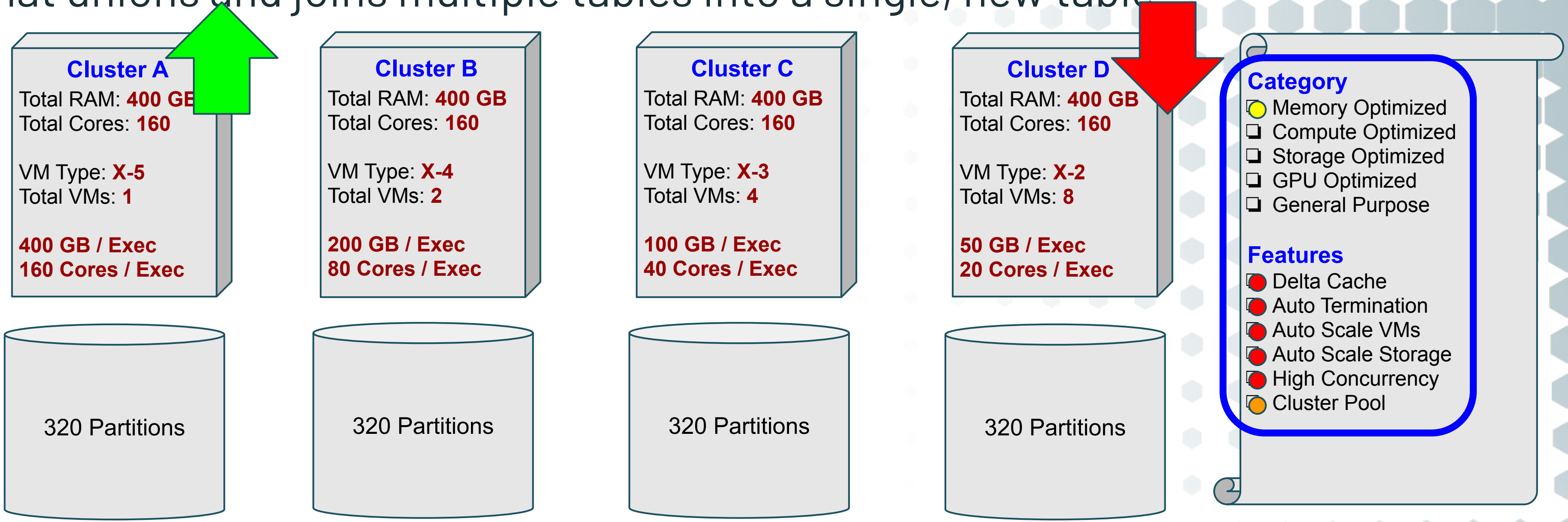
Features
- 🔴 Delta Cache
- 🔴 Auto Termination
- 🔴 Auto Scale VMs
- 🔴 Auto Scale Storage
- 🔴 High Concurrency
- 🟠 Cluster Pool

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

Cluster Configurations Scenarios

Batch ETL: Silver -> Gold

Which of the following cluster configurations is best / least suited for an ETL job that unions and joins multiple tables into a single, new table?



Cluster A
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-5**
Total VMs: **1**

400 GB / Exec
160 Cores / Exec

320 Partitions

Cluster B
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-4**
Total VMs: **2**

200 GB / Exec
80 Cores / Exec

320 Partitions

Cluster C
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-3**
Total VMs: **4**

100 GB / Exec
40 Cores / Exec

320 Partitions

Cluster D
Total RAM: **400 GB**
Total Cores: **160**

VM Type: **X-2**
Total VMs: **8**

50 GB / Exec
20 Cores / Exec

320 Partitions

Category
- ● Memory Optimized
- □ Compute Optimized
- □ Storage Optimized
- □ GPU Optimized
- □ General Purpose

Features
- ● Delta Cache
- ● Auto Termination
- ● Auto Scale VMs
- ● Auto Scale Storage
- ● High Concurrency
- ● Cluster Pool

For this scenario, it can be assumed that each cluster has the same level of compute (total cores) and storage (total RAM)

databricks