TECH
VAULT

DISTRIBUTED DATABASES

AMR ELHELW

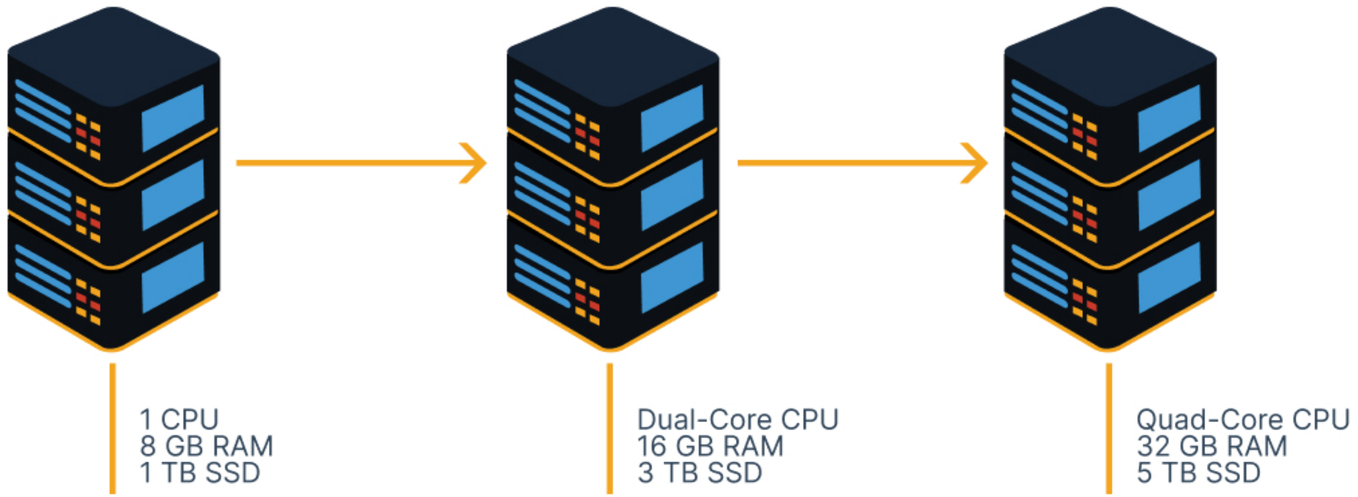
Centralized Databases Challenges

- Too much data
- Slow query processing
- Server failures

Scaling storage and processing

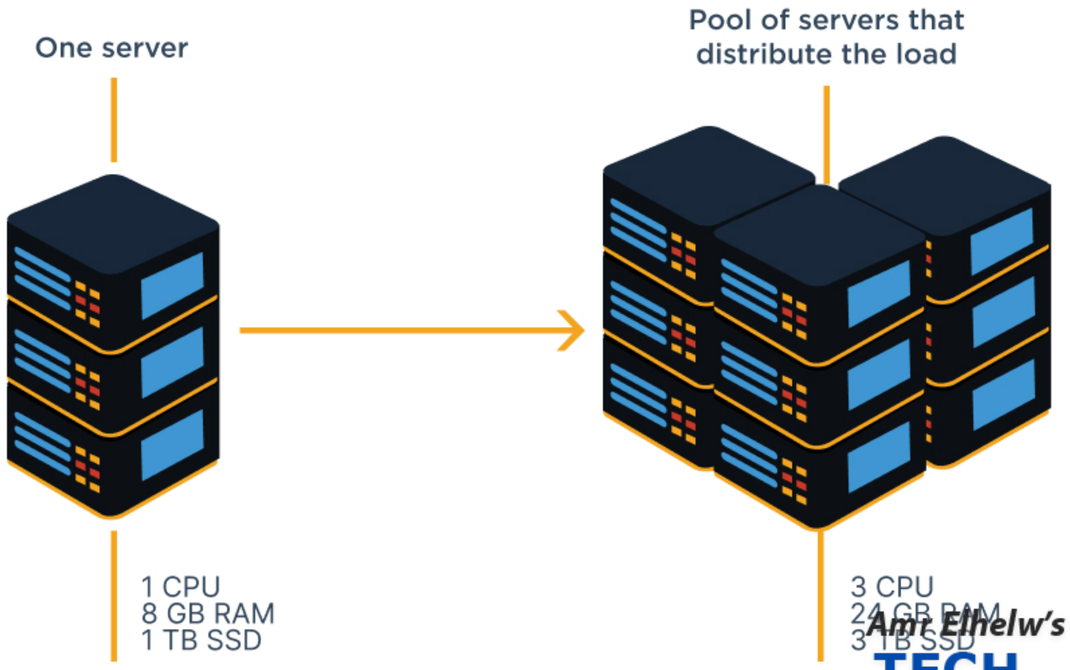
Vertical Scaling

(Improve or replace the existing server)



Horizontal Scaling

(Add more same-size nodes)

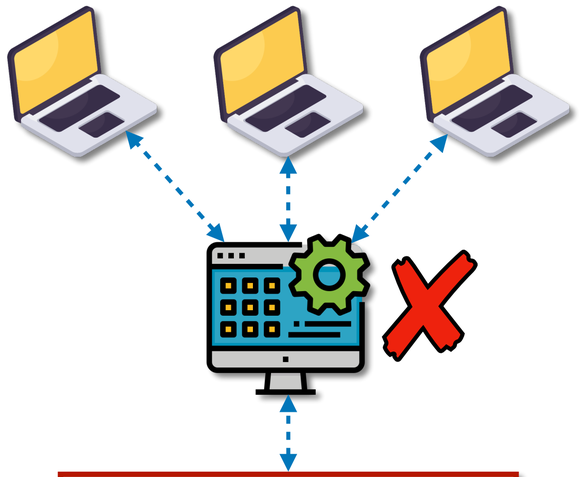


Server Failure

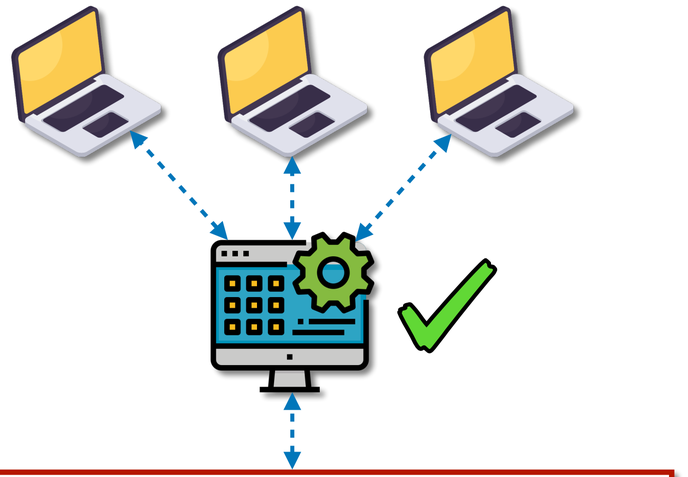
Users

Application

Database



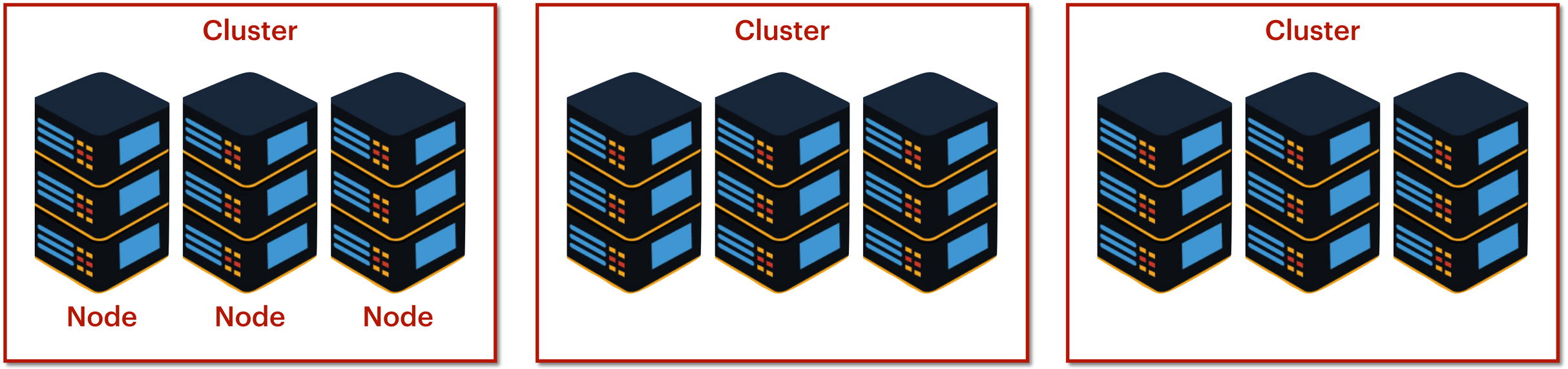
Centralized Database



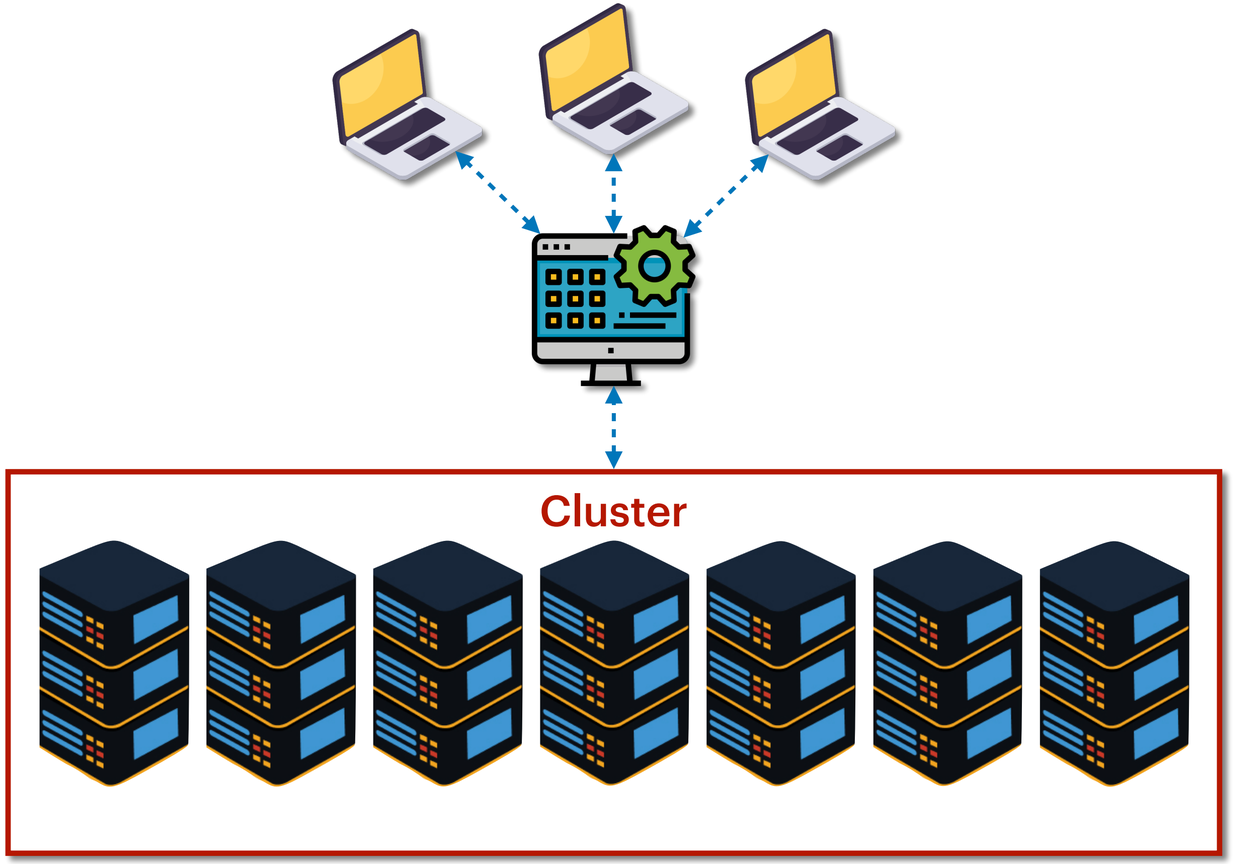
Distributed Database

Amr Elhelw's TECH VAULT

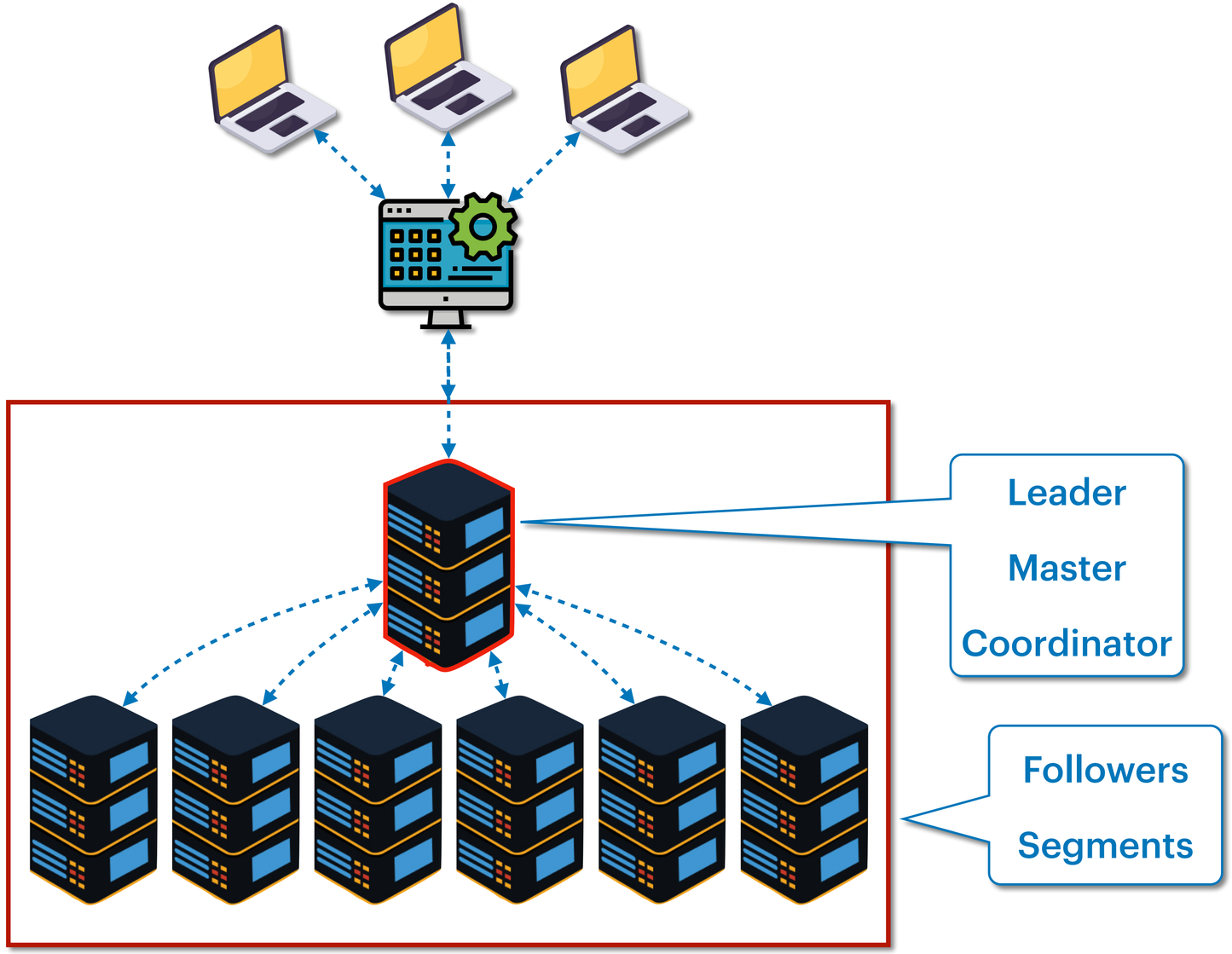
Distributed Database



Distributed Database



Distributed Database

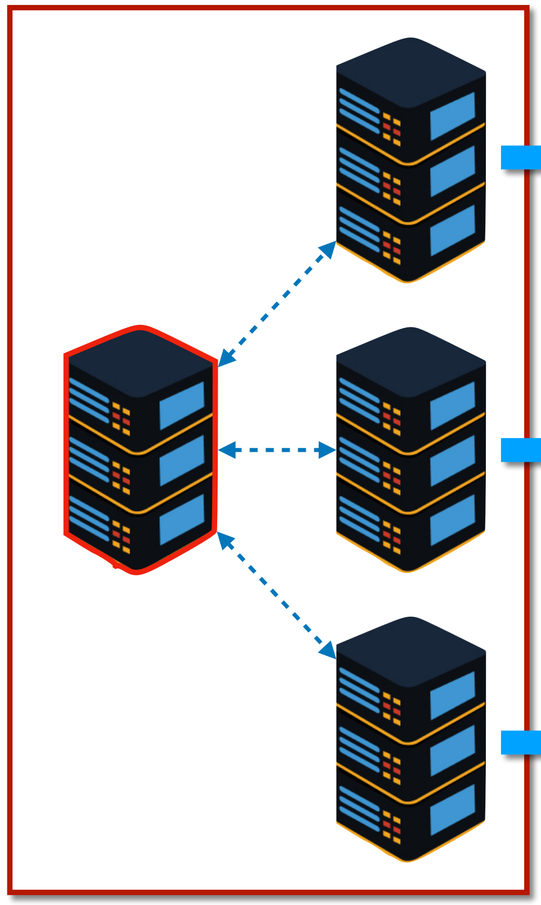


Sharding

Centralized Database

order_id	order_date	customer_id	amount
1573	2022-05-28	16	648
1574	2023-01-26	10	921
1575	2023-09-25	3	810
1576	2022-01-25	18	1063
1577	2022-06-28	8	753
1578	2021-01-28	13	548
1579	2021-02-07	2	572
1580	2023-03-23	11	953
1581	2023-09-11	16	734
1582	2022-09-12	19	1070
1583	2023-04-07	12	385
1584	2023-07-31	16	930
1585	2023-09-22	9	716
1586	2023-07-17	15	766
1587	2022-12-20	8	1000
1588	2022-11-28	2	595
1589	2022-06-30	20	949
1590	2023-04-05	11	297
1591	2022-07-01	2	640
1592	2023-05-18	2	596
1593	2023-03-10	8	280
1594	2022-06-16	8	971
1595	2022-07-07	18	796
1596	2021-06-05	17	796

Distributed Database

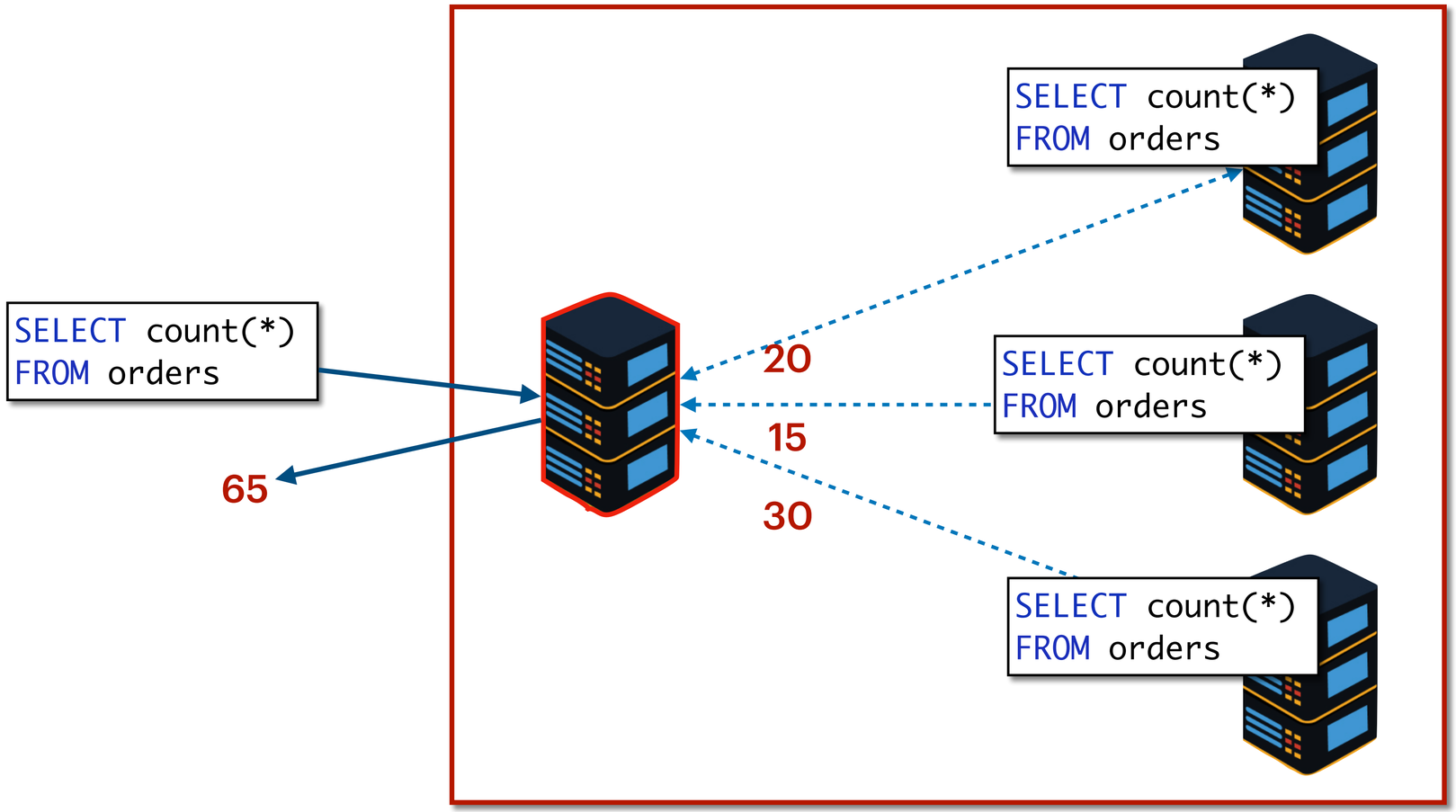


order_id	order_date	customer_id	amount
177	2021-01-11	9	581
181	2021-03-15	14	931
191	2021-03-11	13	445
207	2021-01-06	20	553
210	2021-01-13	17	471
260	2021-01-20	11	620
273	2021-01-20	12	444
289	2021-01-20	12	590

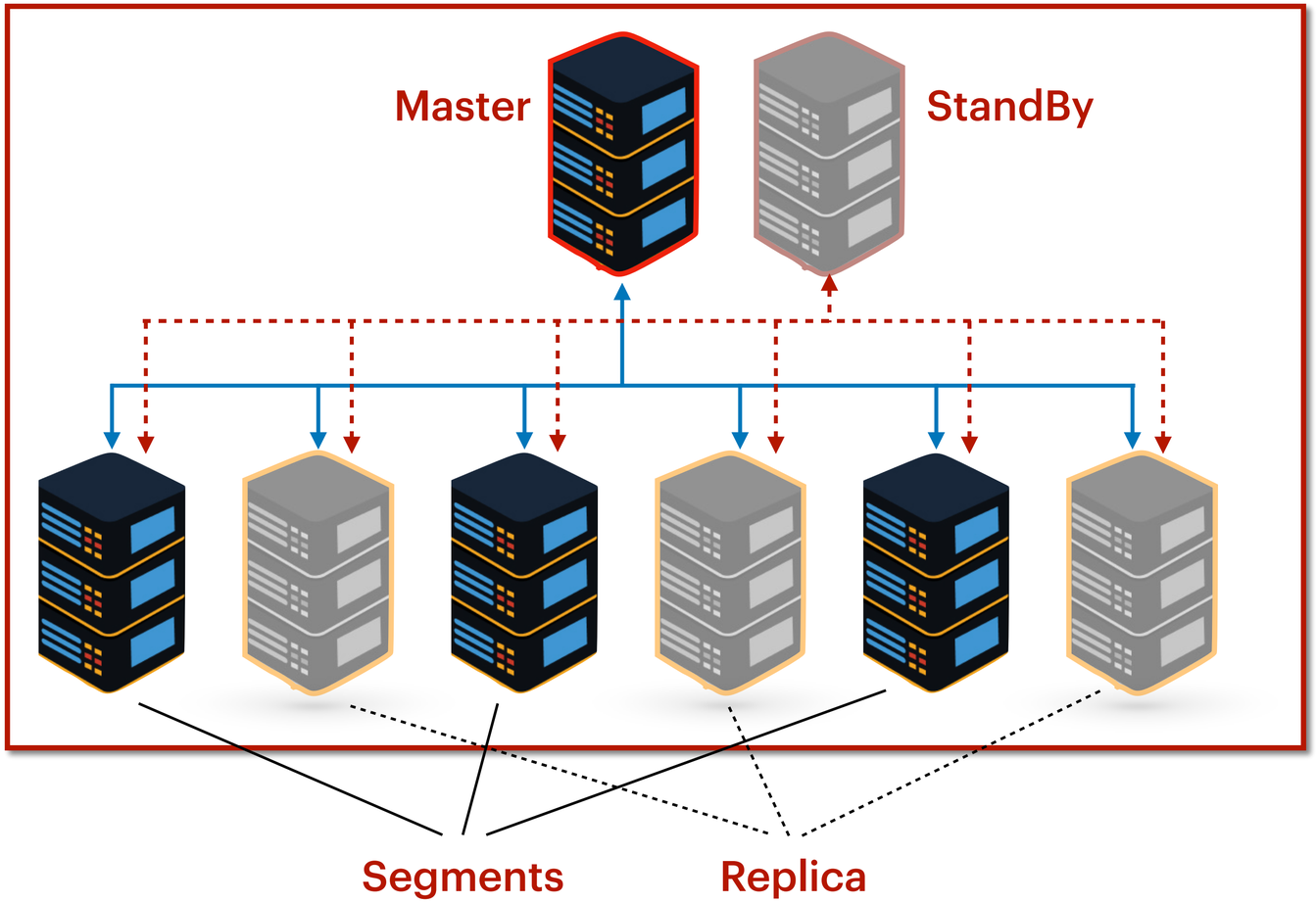
order_id	order_date	customer_id	amount
150	2021-06-24	19	1302
159	2021-05-10	7	192
162	2021-04-10	6	767
175	2021-06-21	8	764
183	2021-05-11	12	1079
187	2021-05-20	12	444

order_id	order_date	customer_id	amount
229	2021-08-04	14	1093
231	2021-08-15	7	167
251	2021-09-12	19	145
269	2021-08-26	19	458
317	2021-09-10	11	413
339	2021-08-23	19	303
340	2021-07-17	5	706
351	2021-08-17	14	144

Sharding



Replication



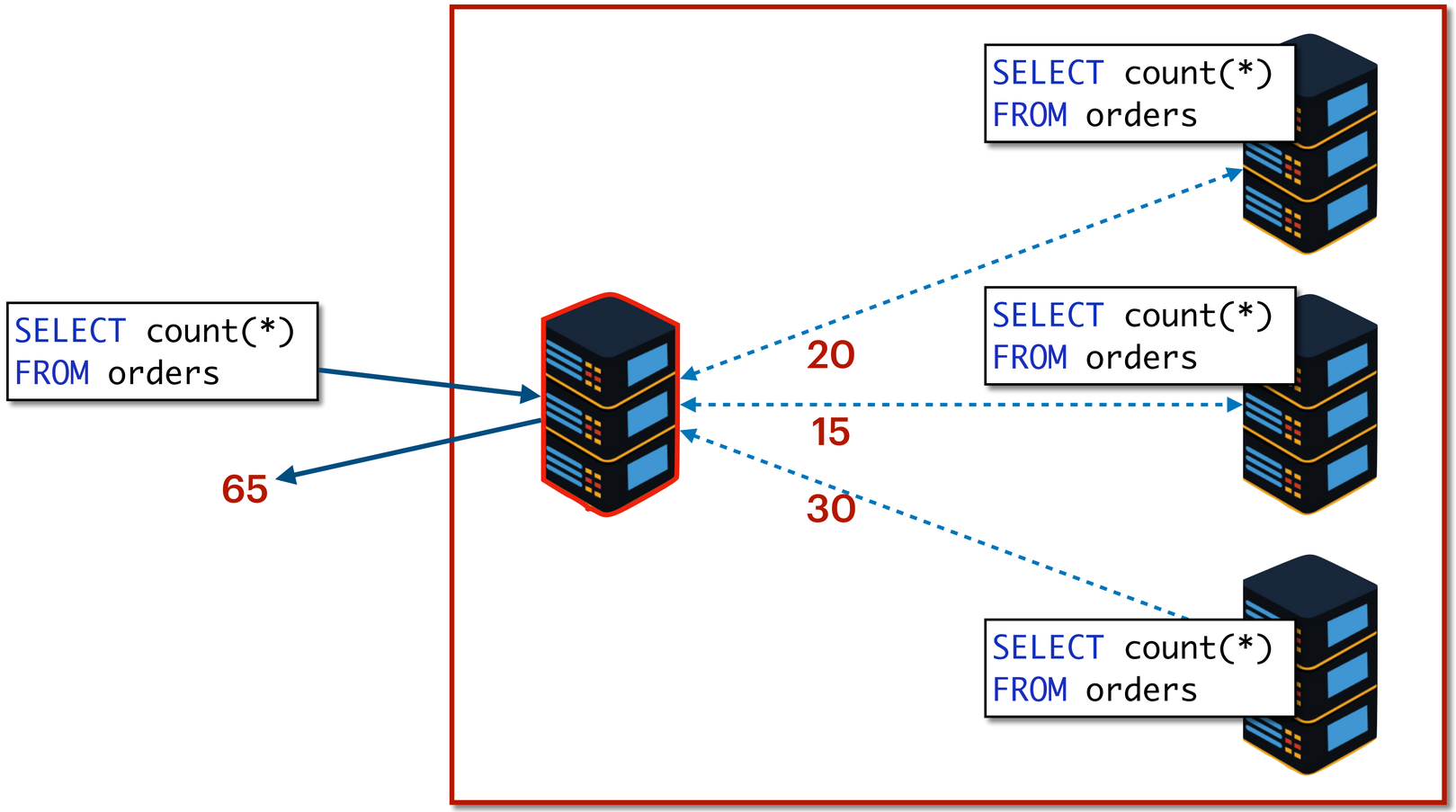
Replication

- Synchronous
- Asynchronous

Distributed Databases
Advantages

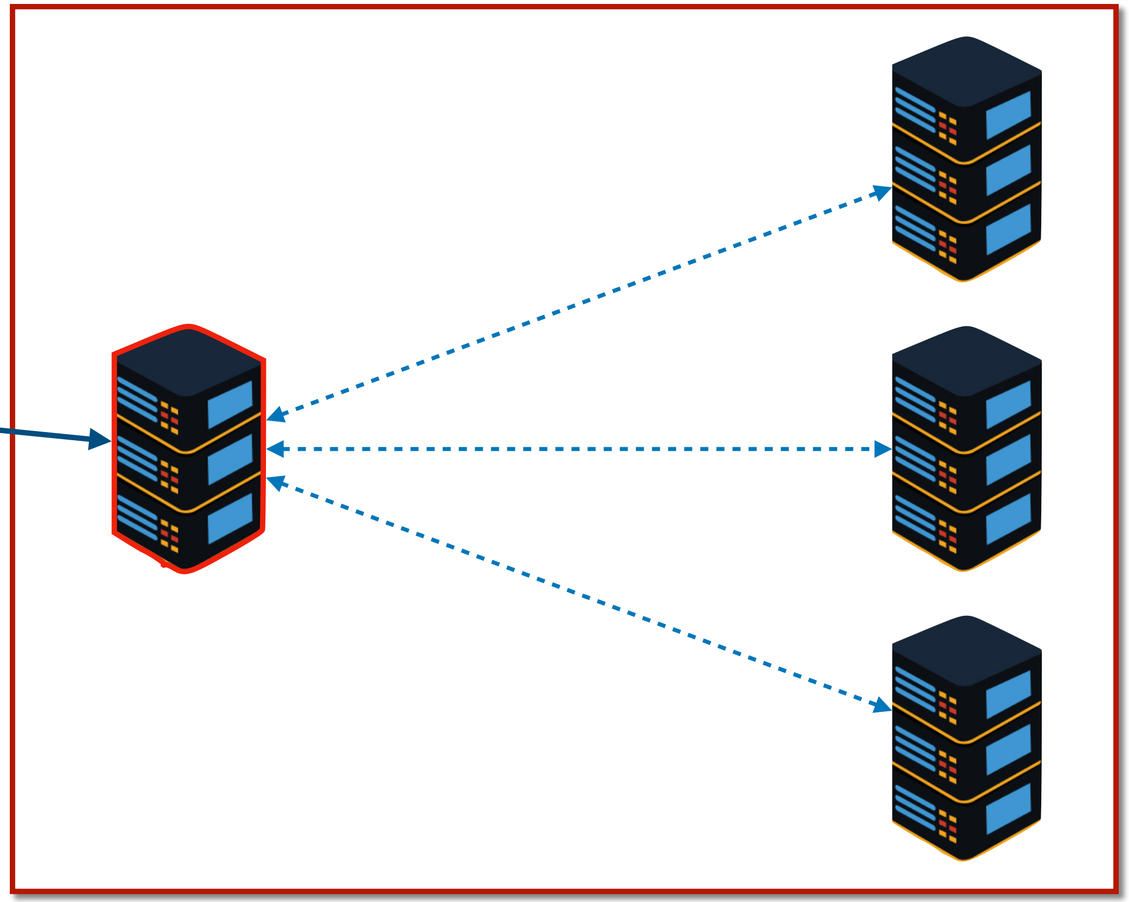
- Improved Performance
- Scalability
- High Availability
- Geographic Distribution

Query Processing



Query Processing

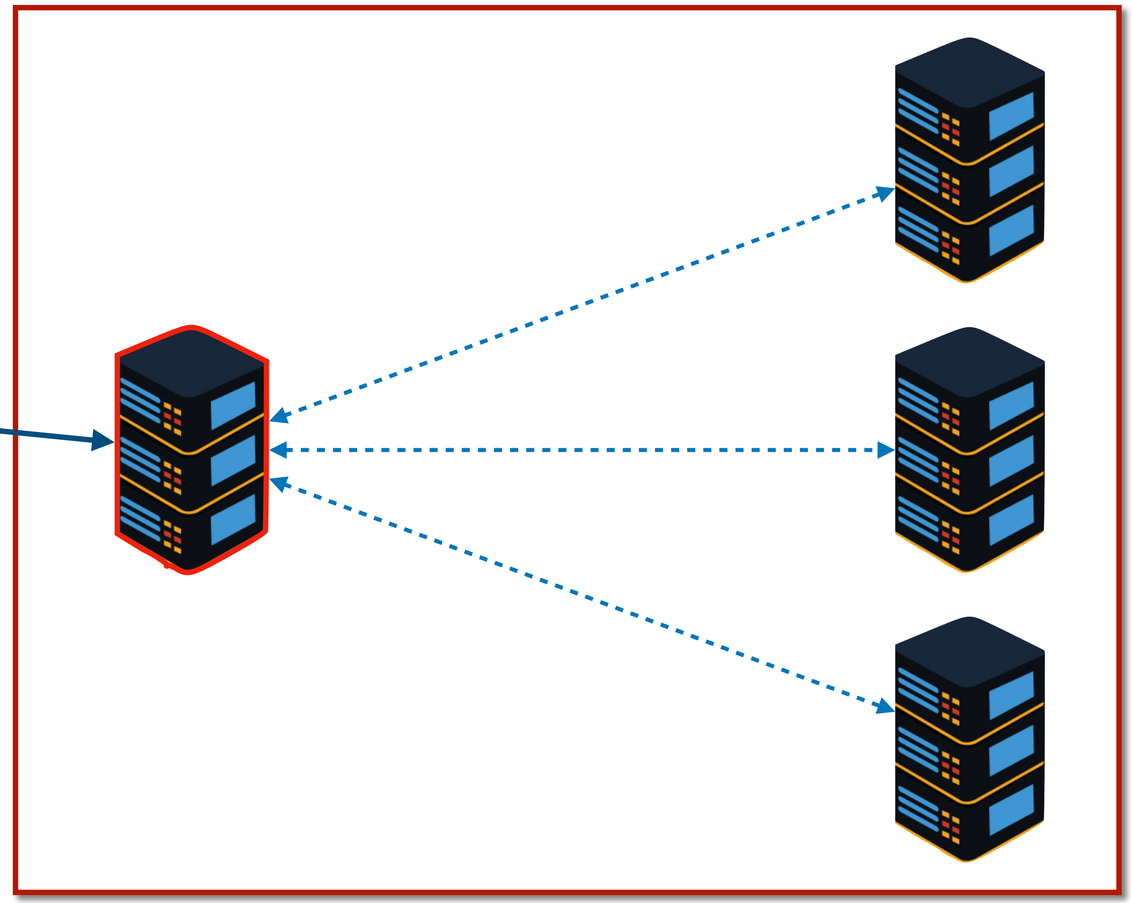
```
SELECT count(*)
FROM orders
```



```
SELECT count(DISTINCT customer_id)
FROM orders
```

Query Processing

```
SELECT count(DISTINCT customer_id)
FROM orders
```



```
SELECT count(DISTINCT customer_id)
FROM orders
```

customer_id
9
14
13
20
17
11
11
14

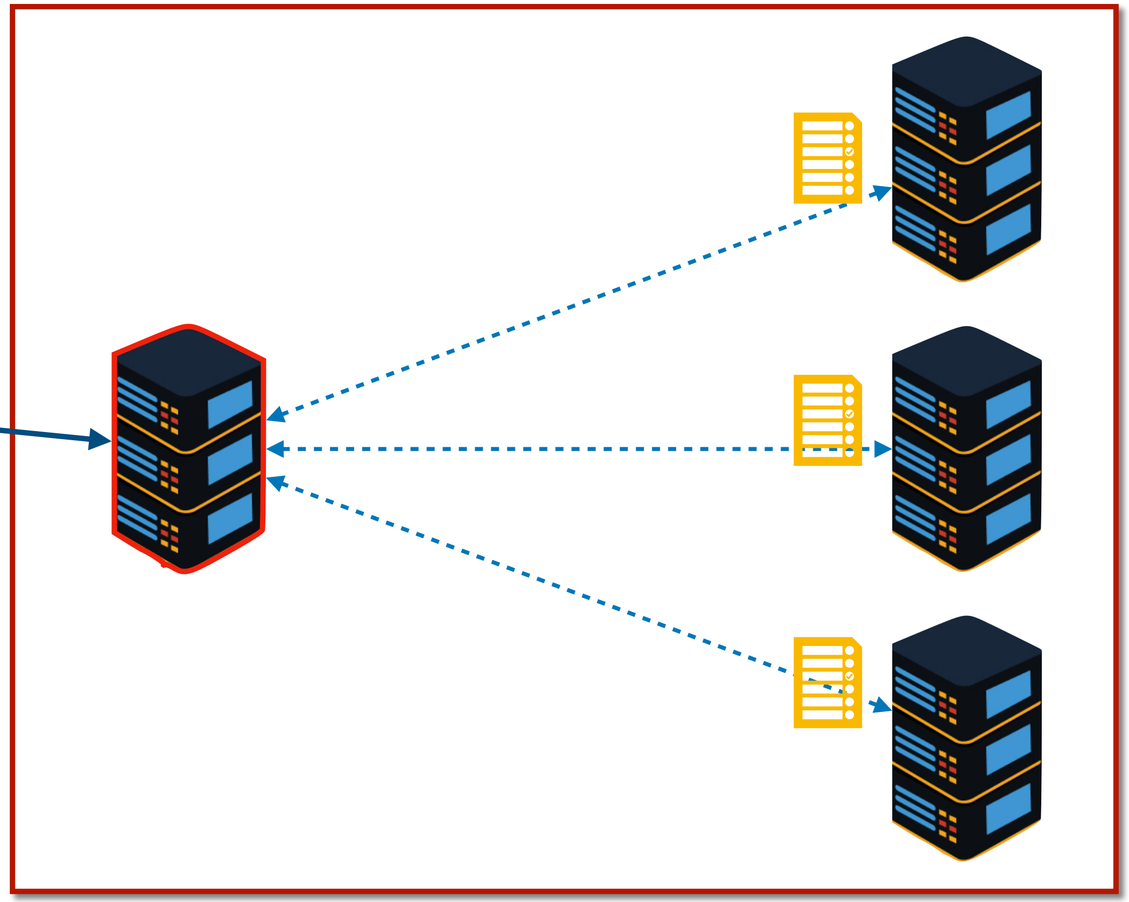
customer_id
19
7
6
8
12
13
6
5

customer_id
14
7
19
19
11
19
5

Amr Elhelw's TECH VAULT

Query Processing

```
SELECT customer_id
FROM orders
```



customer_id
9
14
13
20
17
11
11
14

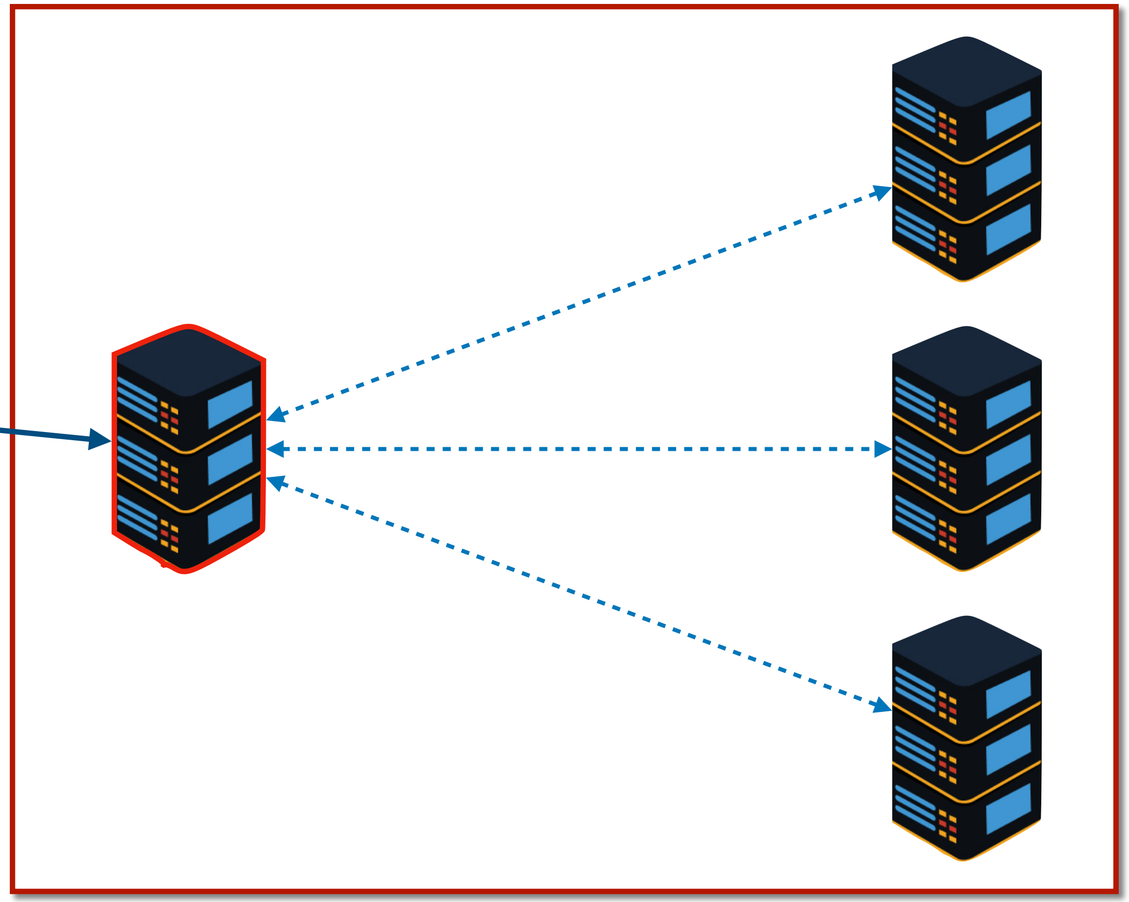
customer_id
19
7
6
8
12
13
6
5

customer_id
14
7
19
19
11
19
5

```
SELECT count(DISTINCT customer_id)
FROM orders
```

Query Processing

```
SELECT DISTINCT customer_id
FROM orders
```



```
SELECT count(DISTINCT customer_id)
FROM orders
```

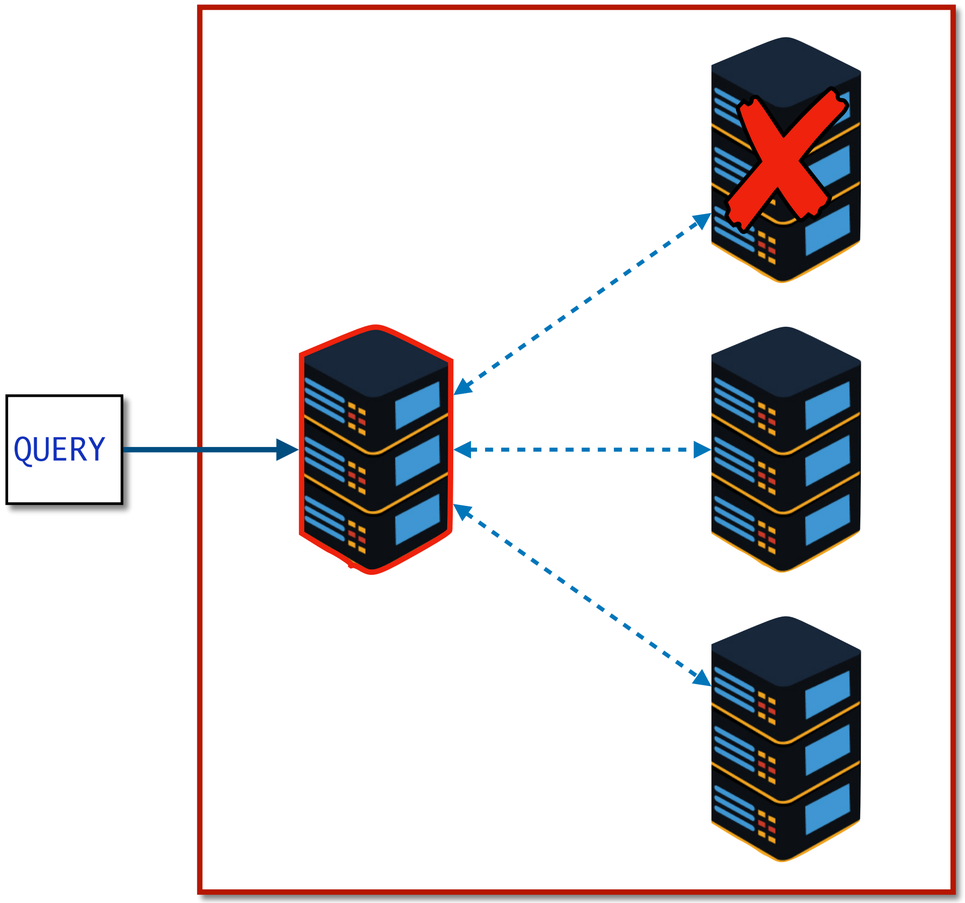
customer_id
9
14
13
20
17
11
11
14

customer_id
19
7
6
8
12
13
6
5

customer_id
14
7
19
19
11
19
5

Amr Elhelw's TECH VAULT

Consistency



- Wait until node is back online?
  - How long to wait?
- Is there a backup/replica?
- Return results from the other nodes?
- ...

CAP Theorem

Consistency

- Every read receives the latest write or an error (no stale results)

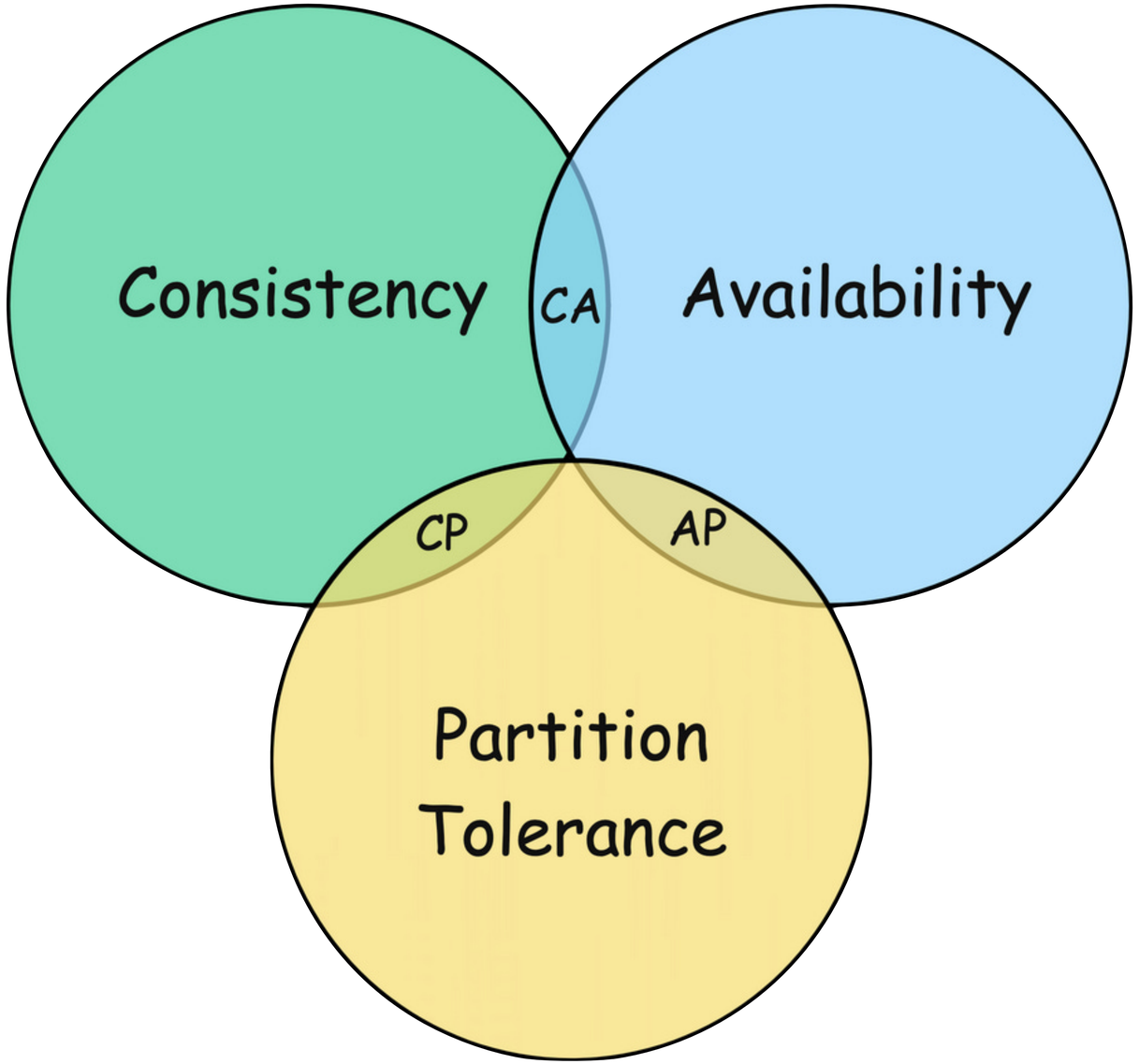
Availability

- Every request receives a response (but not necessarily the latest data)

Partition Tolerance

- The system continues to operate even if some nodes are unavailable

CAP Theorem



CA Centralized Databases

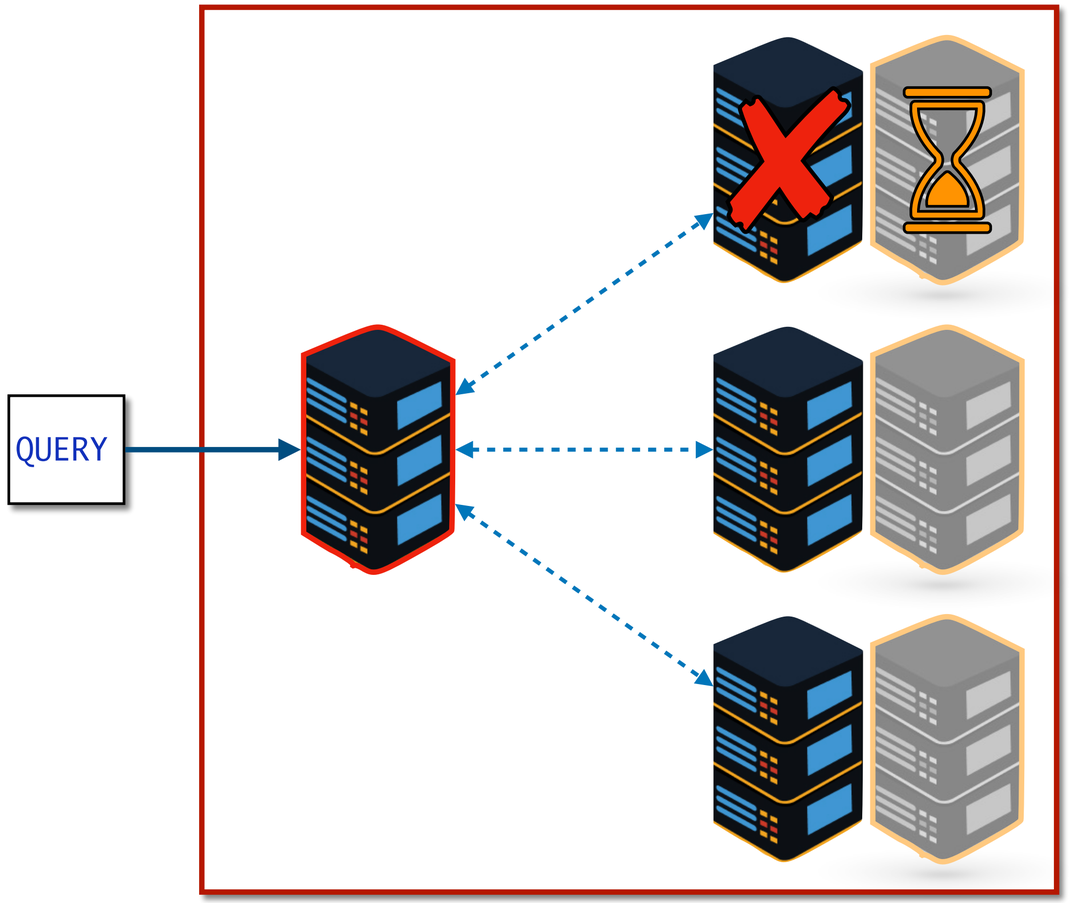
CP

AP

Distributed Databases

Amr Elhelw's TECH VAULT

CAP Theorem



- CP (prioritize consistency)
  - No stale results
  - Return error (unavailable)
- CA (prioritize availability)
  - Return stale result (not consistent)