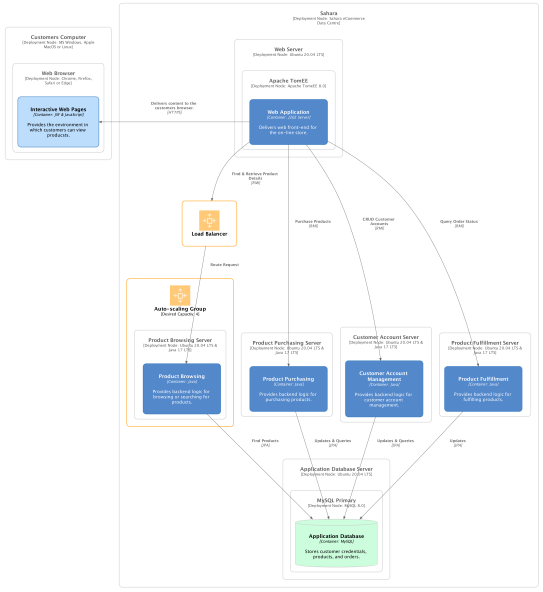
Distributed Computing II

CSSE6400

Brae Webb

April 4, 2022

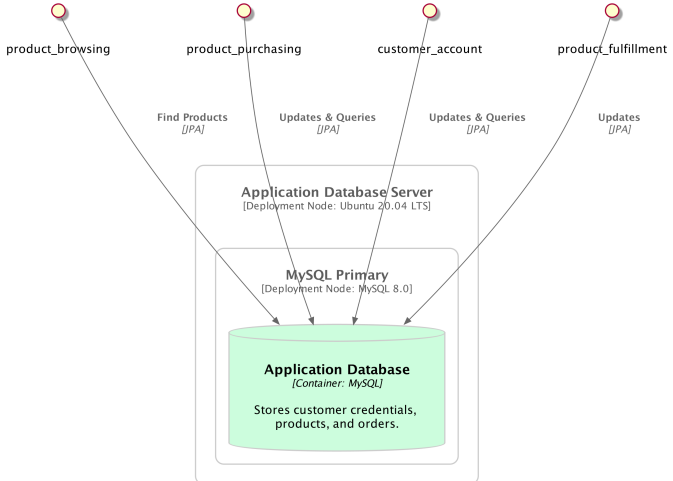
Previously in CSSE6400…



Things start going wrong

What is the likely cause?

Database


product_browsing
product_purchasing
customer_account
product_fulfillment
Find Products
[JPA]
Updates & Queries
[JPA]
Updates & Queries
[JPA]
Updates
[JPA]
Application Database Server
[Deployment Node: Ubuntu 20.04 LTS]
MySQL Primary
[Deployment Node: MySQL 8.0]
Application Database
[Container: MySQL]
Stores customer credentials, products, and orders.
Legend

Question

How do we fix database scaling issues?

Question

How do we fix database scaling issues?

Answer

- Replication
- Partitioning (or sharding)
- Independent databases

Question

How do we fix database scaling issues?

Answer

- *Replication*
- Partitioning (or sharding)
- Independent databases

Question

What is *replication*?

Definition 1. Replication

Data copied across multiple different machines.

Definition 2. Replica

Database node which stores a copy of the data.

Question

What are the advantages of *replication*?

Question

What are the advantages of *replication*?

Answer

- Scale out our database to cope with load.

Question

What are the advantages of *replication*?

Answer

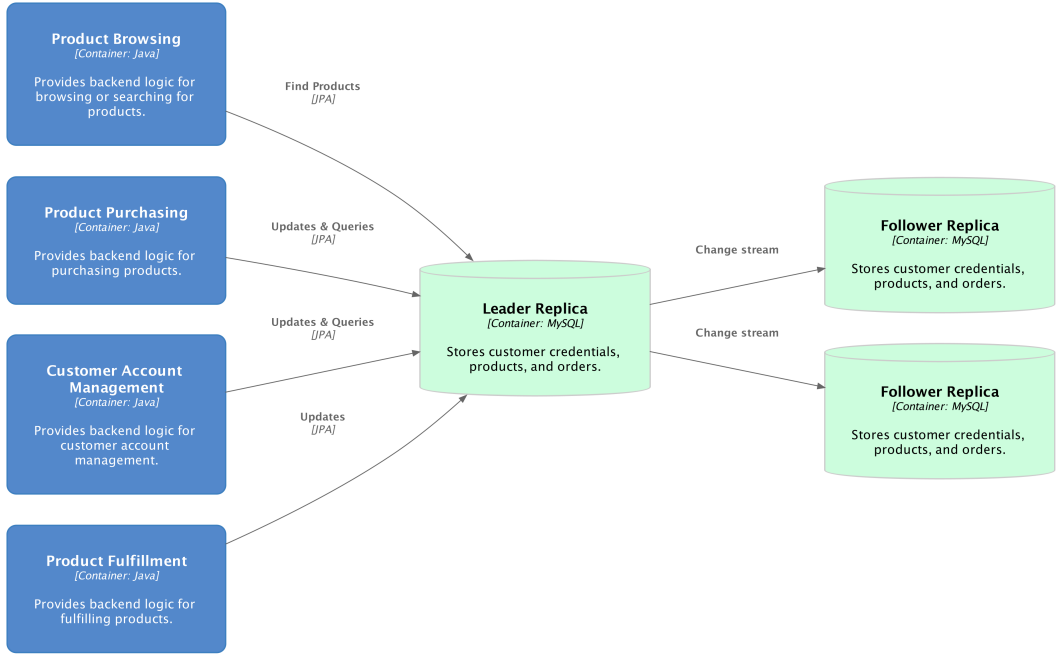
- Scale out our database to cope with load.
- Provide fault tolerance from a single database instance failure.

Question

What are the advantages of *replication*?

Answer

- Scale out our database to cope with load.
- Provide fault tolerance from a single database instance failure.
- Locate databases closer to end-users.


Product Browsing
[Container: Java]
Provides backend logic for browsing or searching for products.
Product Purchasing
[Container: Java]
Provides backend logic for purchasing products.
Customer Account Management
[Container: Java]
Provides backend logic for customer account management.
Product Fulfillment
[Container: Java]
Provides backend logic for fulfilling products.
Find Products
[JPA]
Updates & Queries
[JPA]
Updates & Queries
[JPA]
Updates
[JPA]
Leader Replica
[Container: MySQL]
Stores customer credentials, products, and orders.
Change stream
Change stream
Follower Replica
[Container: MySQL]
Stores customer credentials, products, and orders.
Follower Replica
[Container: MySQL]
Stores customer credentials, products, and orders.

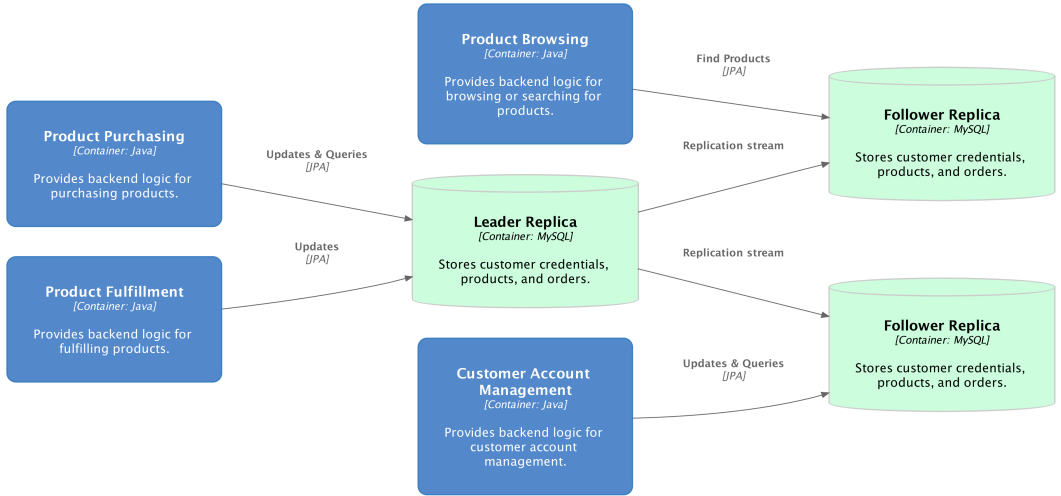
Leader-based Replication

On write Writes sent to leader, change is propagated via change stream.

Leader-based Replication

On write Writes sent to leader, change is propagated via change stream.

On read Any replica can be queried.


Product Purchasing
[Container: Java]
Provides backend logic for purchasing products.
Product Fulfillment
[Container: Java]
Provides backend logic for fulfilling products.
Product Browsing
[Container: Java]
Provides backend logic for browsing or searching for products.
Customer Account Management
[Container: Java]
Provides backend logic for customer account management.
Leader Replica
[Container: MySQL]
Stores customer credentials, products, and orders.
Follower Replica
[Container: MySQL]
Stores customer credentials, products, and orders.
Follower Replica
[Container: MySQL]
Stores customer credentials, products, and orders.
Updates & Queries
[JPA]
Updates
[JPA]
Find Products
[JPA]
Replication stream
Replication stream
Updates & Queries
[JPA]

Propogating changes

Synchronous vs. *Asynchronous*

When things go wrong

Handling outages

Follower failure

Leader failure

Question

How do we fix database scaling issues?

Question

How do we fix database scaling issues?

Answer

- *Replication*
- Partitioning (or sharding)
- Independent databases

Question

How do we fix database scaling issues?

Answer

- Replication
- *Partitioning (or sharding)*
- Independent databases

Question

How do we fix database scaling issues?

Question

How do we fix database scaling issues?

Answer

- Replication
- *Partitioning (or sharding)*
- Independent databases

Question

How do we fix database scaling issues?

Answer

- Replication
- Partitioning (or sharding)
- *Independent databases*