



MULE ESB-Availability

About high-reliable applications...

A high-reliability application must feature the following:

- Zero tolerance for message loss
- Reliable underlying infrastructure (ESB)
- Highly reliable individual connections

The feature known as transactionality tracks application event sequences to ensure that:

- each message-processing step gets completed successfully
- No messages get lost or processed incorrectly

If a step fails, for some reason, the transactional mechanism rolls back all previous processing events, then restarts the message processing sequence again.

Design considerations

Non-functional requirements largely determine your design

Using a cluster for HA?

Clustering or multiple, stand-alone runtimes?

What transactionality option?

- Atomic transactions
- Multiple-resource transactions
- XA transactions

High Availability in CloudHub environments

HA can be achieved by using multiple workers (horizontal scaling)

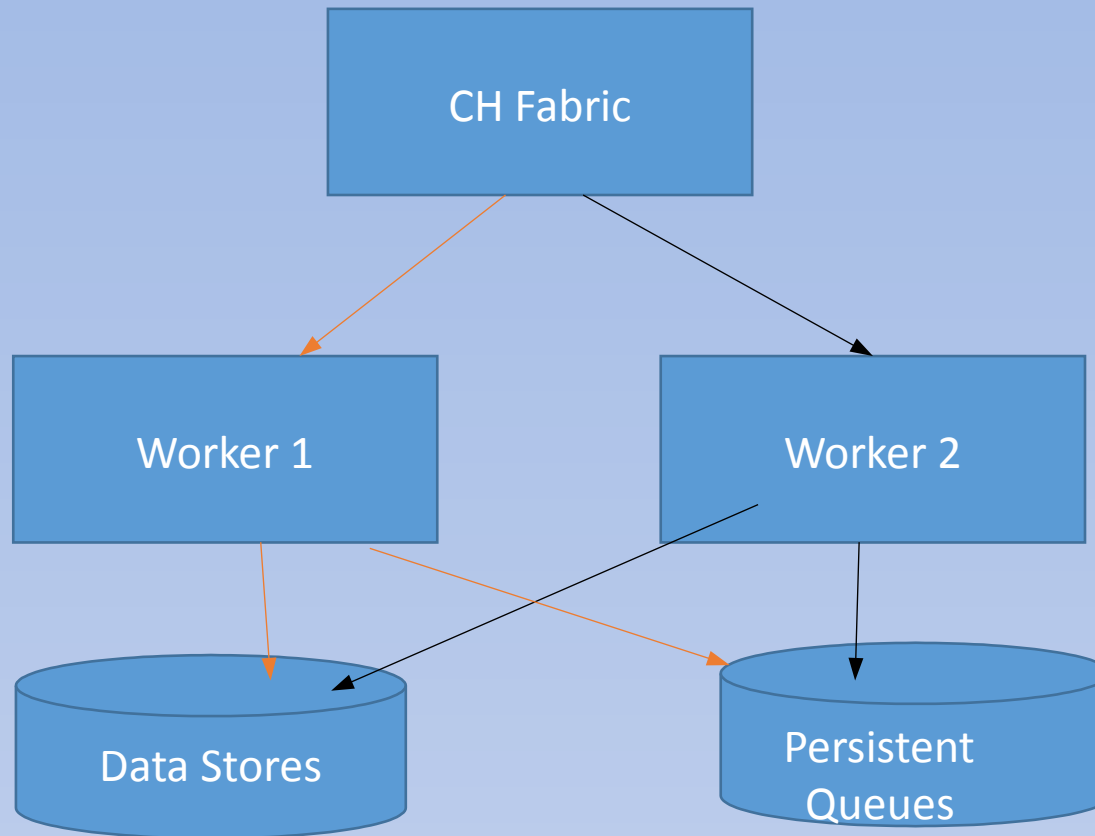
- Easily configurable
- Workers do not share memory
- Workers can use extra infrastructure in CloudHub for sharing data

Application data is stored in CloudHub's object stores

Queues can be persisted, no need to change a deployed application

There is no clustering support in CloudHub

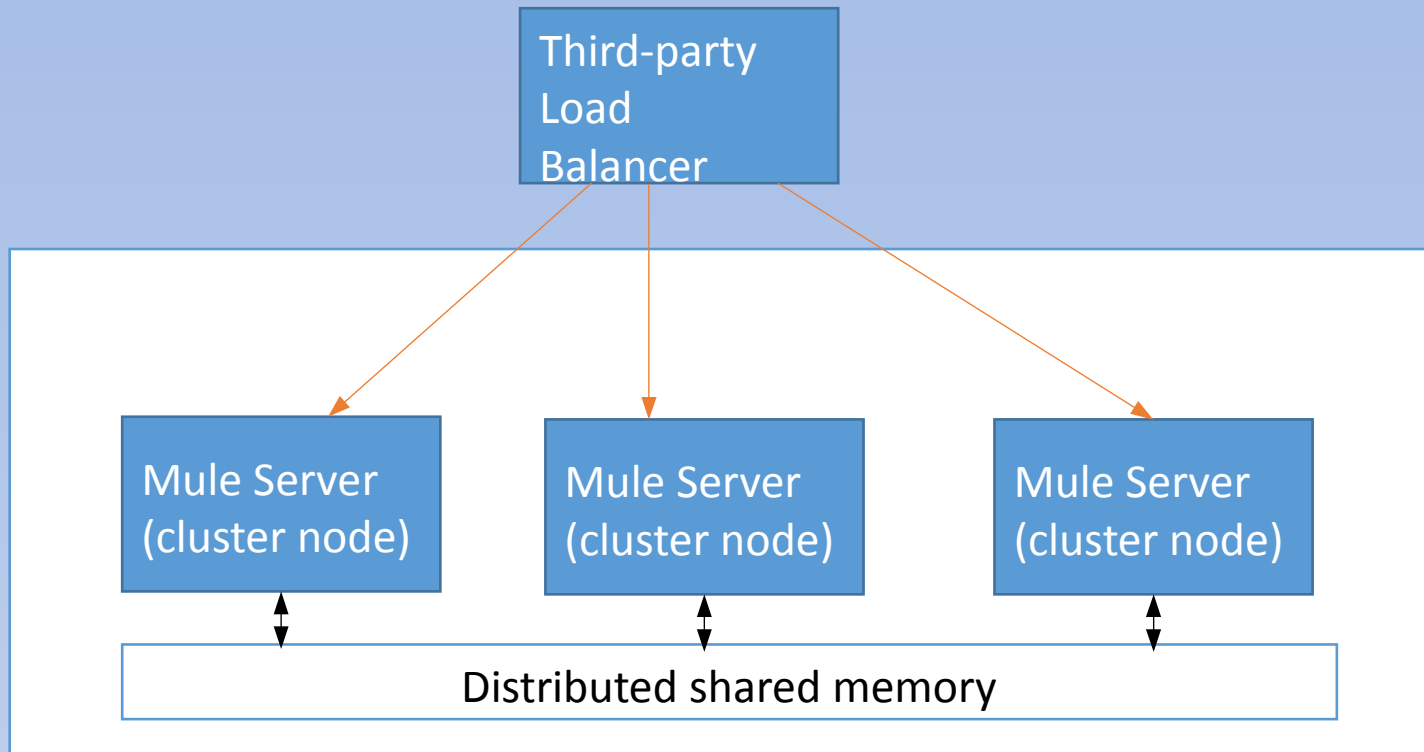
High Availability in CloudHub environments



High Availability in on-premise environments

High Availability can be achieved using

- Mule clustering
- Load balanced, stand-alone Mule instances



Improving High Availability

Improve reliability by persisting all data stores

- Requires third-party database solution

Improve reliability by replicating data to another data center

- Requires additional configuration for database replication/synchronization
- Requires fast network connection

Cross data center clustering is possible, however...

- Requires ultra-fast set up between data centers