Problem-

A monolithic java application stack, running on Apache web server, Apache tomcat application server with ActiveMQ & Oracle & MongoDB.

Have to be migrated to cloud platform (let’s say AWS)

Solution-

Step 1-

Consider the architecture of the existing application.

Create EC2 instance which will map the equivalent hardware configuration of physical server. Also consider combined storage and bandwidth requirements. This will be our tier one.

Then Install Apache web server and tomcat application server on the same EC2 instance.

This created EC2 instance will be scalable, so that it can handle high traffic.

Step 2-

We will use ELB to handle all the requests coming from DNS to web server (Which in this case will be Apache web server), from web server we will give request application server (tomcat application server) to get the work done. Application server will be tier 2.

Step 3-

Now we will think about migration of database layer that is tier 3. Now this application is using 3 database i.e ActiveMQ ,Oracle, MongoDB

To achieve zero downtime migration there are many tools available. The AWS database migration service(DMS) supports may database engines.

With DMS the source database remains functional while migration is happening.

We can in this case migrate from Oracle database to amazon aurora database.

The AWS schema conversion tool makes heterogeneous migrations easy.

Source -> Schema conversion tool -> target database

Source -> AWS DMS -> target database