

RED HAT SUMMIT 2017 LAB: LEARN HANDS-ON HOW TO MODERNIZE LEGACY APPLICATIONS

Introduction

Modern development using microservices architectures, containerized infrastructures and DevOps principles are much more effective than the legacy way to develop and maintain applications.

The exercises in this lab simulate a scenario in which a business with a mission-critical monolithic retail consumer application wants to modernize its business and its software. However, the business cannot afford to spend a lot of time and energy re-architecting the application in one big effort. Instead, they wish to re-platform their existing app to a modern deployment platform (in this case OpenShift), implement DevOps processes to increase speed and efficiency of updates, and then get to work incrementally replacing functionality from the old app with functionality developed using modern techniques like microservices architecture and modern application platforms like JBoss EAP and WildFly Swarm.

What You Will Learn

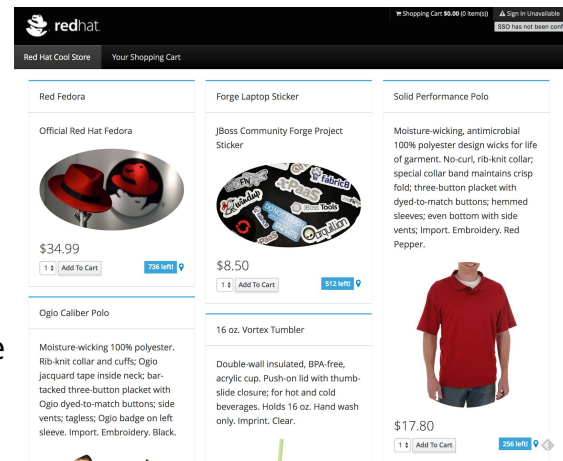
In this hands on lab you will learn how to modernize an existing legacy monolithic application by applying microservice architectural principles, using modern lightweight runtimes like WildFly Swarm and deploying to container-based infrastructure using OpenShift Container Platform. This combination of architectures and deployment platforms opens up new possibilities for applying DevOps processes to improve the speed of development and deployment.

Labs Details and Additional Resources

This lab is split into three main sections, taking you through the process of deploying a monolith to OpenShift, wrapping it with a CI/CD pipeline, developing microservices to start replacing functionality in the monolith, and integrating it all together to form the beginnings of a complete modernization of an existing app.

Lab 1: Building and Deploying a Fast-Moving Monolith

Large organizations have a tremendous amount of resources invested in existing monolith applications, and are looking for a sane way to capture the benefits of containers and orchestration. OpenShift provides the platform for existing apps, and provides a path forward for future modern architectures.

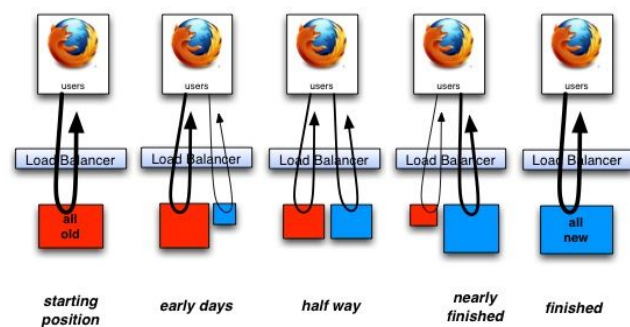


In this lab you will deploy a monolith and incorporate DevOps processes like deployment pipelines to increase the speed at which the monolith can evolve.

The monolith represents an online retail store where items can be viewed (including their inventory availability) and purchased through a “shopping cart” experience.

Lab 2: Strangling the Monolith

Strangling in the context of software means incrementally replacing some piece of functionality in an existing app with something better

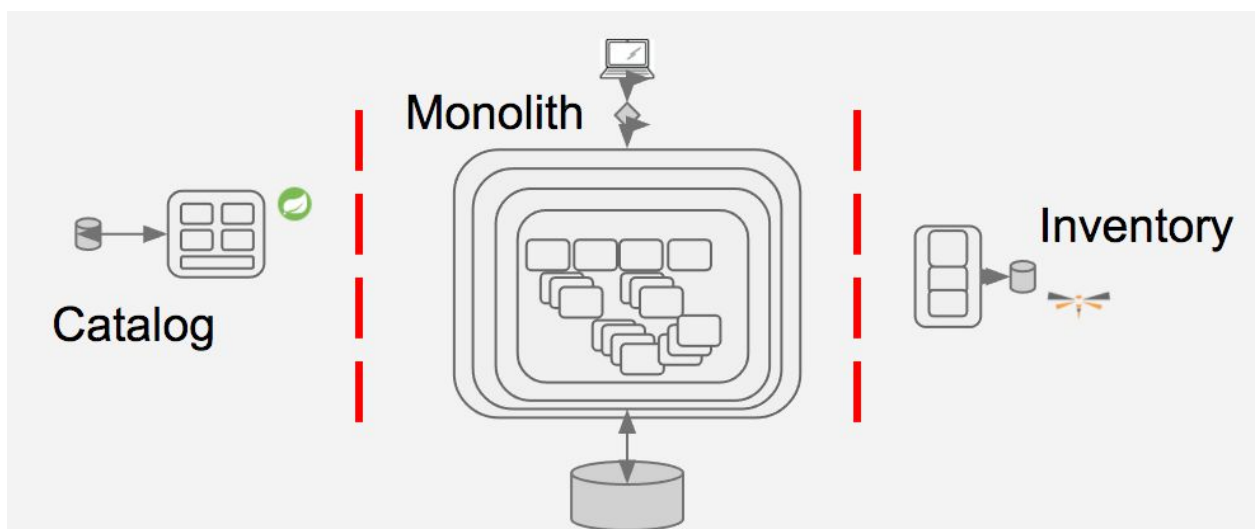


(cheaper, faster, easier to maintain). As functionality is replaced, the “dead” parts of existing app (in our case our monolith) can be removed or retired.

In this lab you will deploy two microservices:

- **Inventory Service** - supplies availability of a product through a RESTful interface using *WildFly Swarm*)
- **Catalog Service** - supplies the products that are available for sale (using *Spring Boot*)

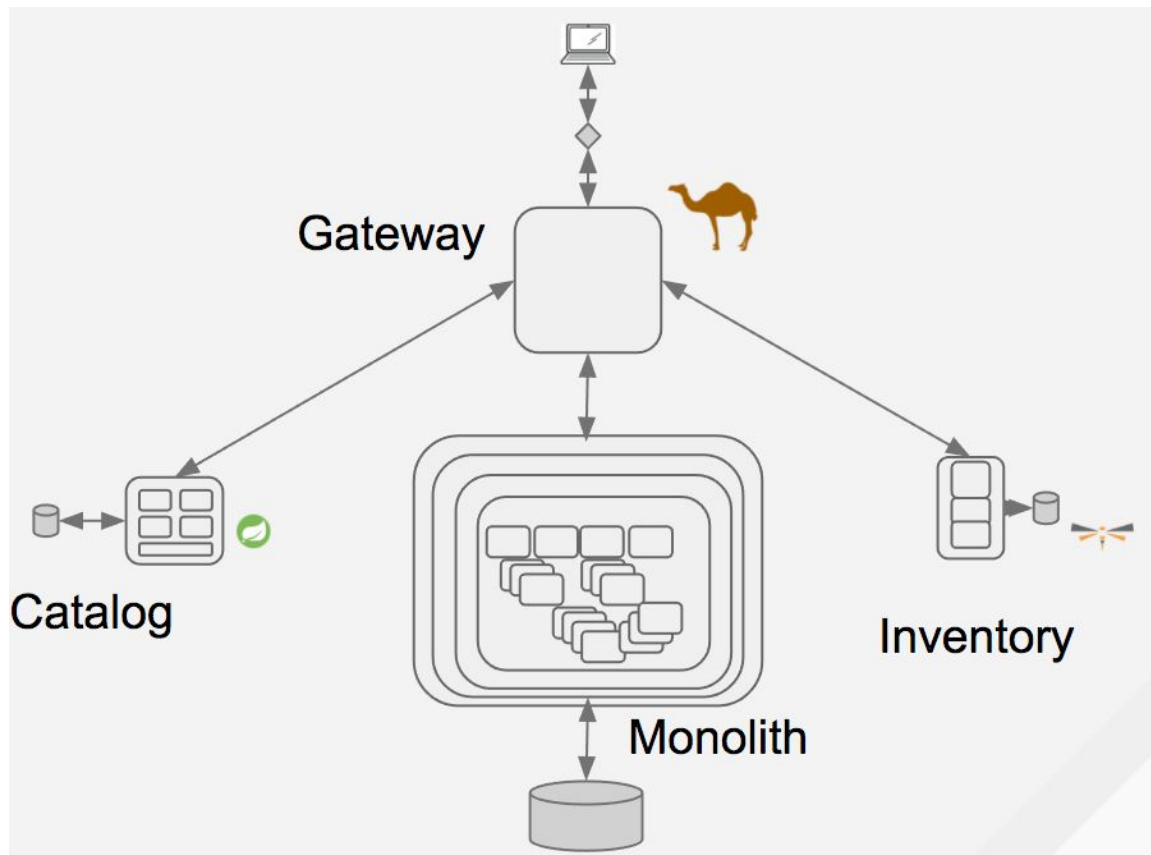
After deployment the structure will look like this:



Lab 3: Integrating the Microservices

In this lab, the 2 microservices are integrated with the monolith, so that when catalog or inventory data is requested by the monolith, the requests are redirected to our new microservices. This completes the beginning of the “strangulation”. A gateway component is also introduced to aggregate the microservices together such that the client (browser) only communicates with a single point, instead of having to aggregate from multiple different services.

At the end of the lab, the application will look like:



Additional Resources

- Lab Source Code & Guide
 - <https://github.com/modernize-legacy-apps>
 - This guide (also available at `${HOME}/projects/docs/lab-guide.pdf`):
<https://raw.githubusercontent.com/modernize-legacy-apps/docs/master/lab-guide.pdf>
- The Majestic Monolith - David Hansson
 - <https://m.signalvnoise.com/the-majestic-monolith-29166d022228>
- How We Sped Up Delivery From Every Three Months to Every Week
 - <http://red.ht/2jwR07s>
- Legacy Application Strangulation Case Studies - Paul Hammant
 - <http://paulhammant.com/2013/07/14/legacy-application-strangulation-case-studies>
- Red Hat Developers
 - <https://developers.redhat.com>