

Bluemix DevOps Toolchain Lab

Objective

This series of labs shows how to set up a productive toolchain with a sample that consists of three microservices. After you finish this part of the series, you will be familiar with a toolchain that demonstrates practices from the IBM® Bluemix® Garage Method. Toolchains are available in the US South region only.

To create this toolchain, we will use a sample to create an online store that consists of three microservices: a Catalog API, an Orders API, and a UI that calls both of the APIs. The toolchain is pre-configured for continuous delivery, source control, blue-green deployment, functional testing, issue tracking, online editing, and alert notification. We will explore the various integrations.

Online Store sample

The online store consists of three microservices:

1. Catalog API: A back-end RESTful API that tracks all of the items in the store.
2. Orders API: A back-end RESTful API that tracks all store orders.
3. UI: A simple UI that displays all of the items in the store catalog, and that can create orders. This PHP UI calls both of the REST APIs.

The Catalog and Orders API are backed by a Cloudant store. As part of deploying this application a no cost Cloudant service instance will be created.

Each microservice is deployed to up to three environments: development, test, and production. You end up with multiple deployed microservices.

You will be creating a toolchain and adding the following integrations:

- Three GitHub repositories (repos), with GitHub Issues enabled. Each microservice has one GitHub repo.
- Three delivery pipelines, one for each GitHub repo. The pipelines are independent and can run in parallel.
- The Eclipse Orion Web IDE, which you can use to edit your code and deploy it with the pipeline from a web browser.
- Slack, which you can use to get real-time notifications about the status of builds and deployments. [comment: #](#) (- [IBM Alert Notification] is configured to alert the team when events happen) [comment: #](#) (- Bluemix Availability Monitoring is configured to monitor the application in production and alert the team when outages occur)

Labs

- [Lab 0: Getting setup](#)
- [Lab 1: Read Template Paper](#)
- [Lab 2: Create Order Toolchain Using Pre-Built Template](#)
- [Lab 3: Create Catalog Toolchain by Hand](#)
- [Lab 4: Create UI Toolchain from deployed application](#)
- [Lab 5: Integrate Slack](#)