**Customizing a Buildpack**

**Goals**

• Fork the Java buildpack to make a minor change.

**Prerequisites**

To complete these steps you will need:

* A Cloud Foundry account (either on your company’s CF installation or on Pivotal Web Services)
* Either the CLoud Foundry CLI or an IDE installed (with Cloud Foundry support)

Most importantly: a GitHub account. Unlike other labs, a github account is mandatory for this lab.

**Steps**

In this section you will fork the Java buildpack, make a minor change to the desired Java version, then use your modified buildpack to push an application to Cloud Foundry, and observe the results

1. Push the spring-mvc-demo application to Cloud Foundry (if needed)

a. Use either the CLI or Eclipse/STS, whichever you find most convenient. If you use Eclipse/STS it would be best to save a manifest file for the later steps.

b. Observe the console output during the push process. Note the version of Java being used. Our goal in this lab will be to specify a different Java version.

1. Fork the Java Buildpack.
   1. Using a browser, go to <https://github.com/cloudfoundry/java-buildpack> (if you are not signed into GitHub, do so now).
   2. Click the “Fork” button to create a fork of this repository under your own GitHub account.   (You may clone a copy of this repository locally, though it is not necessary for the steps here)
2. Alter the Java version
   1. While still in the browser viewing your forked copy of the Java buildpack, drill-down to config/open\_jdk\_jre.yml. Click the edit button to modify this file in place.
   2. Find the line (near line 20) that indicates the Java version: version: 1.8.0\_+
   3. Alter this line to deliberately specify an older Java version: version:1.8.0\_11
   4. Save your work using the ‘Commit’ button.
3. Re-push the application using the altered Buildpack
   1. This step is performed differently depending on whether you have an existing manifest file or not. Assuming you do:
      1. Alter the manifest file. Add a line to indicate you new buildpack (change “yourGitId” and “java-buildpack” as needed to match). Be sure your indentation is correct:
      2. buildpack: https://github.com/[yourGitId]/[java-buildpack]
      3. Re-push the application using cf push. If you originally used Eclipse to push the app, there is no convenient way to tell Eclipse to re-push with the manifest, so either 1) specify the buildpack option using Eclipse or 2) use cf push from the CLI.
   2. If you did not have a manifest file, and you used the CLI, repush using the -boption to specify the buildpack (alter “yourGitId” and “java-buildpack” as needed):
   3. -b https://github.com/[yourGitId]/[java-buildpack]
   4. Observe the console output during the push process. Note the version of Java being used. It should be the version you specified above.

Congratulations, you have finished this exercise.