# Tolling Assignment Advanced

## Objectives and Outcomes

* Understand how to add custom virtual machines to the Vagrant platform
* Understand the advantages of using Vagrant and Packer
* Configure specific software needed to use Big Data tooling platform, Spark, Hadoop, and Mariadb in virtual machines

## Packer and Vagrant Build Steps

* From the commandline, cd into the build directory of the packer-vagrant-build-scripts repo you cloned.
  + packer-vagrant-build-scripts\packer\build
* Find the build artifacts, there should be one \*.box file, similar to ubuntu-vanilla-18043-server-virtualbox-1580565828.box
  + Your number will be different as it is a timestamp
* Add this box to Vagrant
  + Initialize and connect to this virtual machine.

## Inside Vagrant Box Steps

* Change host name, add initials and system for local
* From the command line (non-admin) execute the command: vagrant plugin install vagrant-vbguest
  + This takes care a warning message from Vagrant about not being able to mount VirtualBox shared drives
* Install mariadb server
* Install java 8 openjdk
  + make sure it is default jdk
* Install R package
  + <https://www.digitalocean.com/community/tutorials/how-to-install-r-on-ubuntu-18-04>
* Update default Python version to 3.6
  + <https://askubuntu.com/questions/1065572/how-to-safely-switch-to-python3-as-default-after-upgrade-to-ubuntu-18-04>
* Install Spark 2.4
  + <http://mirrors.advancedhosters.com/apache/spark/spark-2.4.4/spark-2.4.4-bin-hadoop2.7.tgz>
* Install Hadoop 2.9.2
  + <http://apache.mirrors.hoobly.com/hadoop/common/hadoop-2.9.2/hadoop-2.9.2.tar.gz>
* Configure .bashrc for PATH variables
  + Add: export JAVA\_HOME=/usr
  + Add: export HADOOP\_HOME=/home/vagrant/hadoop-2.9.2
  + Add: export SPARK\_HOME=/home/vagrant/spark
  + Add: export HADOOP\_CLASSPATH=/usr/lib/jvm/java-8-openjdk-amd64/lib/tools.jar
  + Demonstrate installation success with --version commands
* Configure Vagrant paths, memory, and cpus

## Deliverable

* Proceed to finish the last part of chapter 2 lecture
  + Walking through the command line Spark exercises