

Data Management in Large-Scale Distributed Systems

Instructions to prepare the lab on Apache Spark

Thomas Ropars

`thomas.ropars@univ-grenoble-alpes.fr`

`http://tropars.github.io/`

2018

Introduction

These slides provide the basic information to configure your personal laptop for the labs on Apache Spark.

Warning

- The instructions assume that you run on Linux
- It will most probably work also on MacOS
- Not sure about Windows machines

Description

We are going to use a docker container that is already configured with Spark properly installed.

Main steps

1. Installing and configuring Docker
2. Downloading the Spark container
3. Testing that everything works well

Installing docker

- We will use Docker CE (Community Edition)
- To install Docker CE on Ubuntu, please follow the instructions here:
 - ▶ <https://docs.docker.com/install/linux/docker-ce/ubuntu/#install-docker-ce>
- On the same page you can find instructions for other Linux distributions
- By default, Docker containers can be launched only by the *root* user. This can be inconvenient. To allow a normal user to run a docker container, follow these instructions:
 - ▶ <https://docs.docker.com/install/linux/linux-postinstall/>

More about Docker

Once you are done with installing Docker, you should test that it works by running the *hello world* of docker in a terminal:

```
docker run hello-world
```

- If everything works, congrats, you are almost done :-)
- If you want to know more about Docker, you may start from here:
 - ▶ <https://docs.docker.com/engine/docker-overview/>

Getting the Spark Docker container

One Docker is installed, you should pull the Docker image containing Spark that we will use. To do so, run in a terminal:

```
docker pull jupyter/pyspark-notebook
```

WARNING

- The image is big. It will take time.

Last step: testing that everything works

To test that you are able to run Spark on your machine, run the following command in a terminal:

```
docker run -it --rm -p 4040:4040 \  
    jupyter/pyspark-notebook \  
    /usr/local/spark-2.3.1-bin-hadoop2.7/bin/pyspark
```

The last slide

At this point, Spark is running on your machine and you have started the Python interactive console (pyspark).

- You can run your first Spark command by checking the default level of parallelism used by Spark on your machine, by typing in pyspark:

```
print(sc.defaultParallelism)
```

- You can probably even connect to the Spark Web UI here: <http://localhost:4040/>
- To exit pyspark and terminate Spark, simply type Ctrl-D

You are ready for the lab :-)