Lecturer: Prof. Dr. Joeri De Koster (joeri.de.koster@vub.be)
Assistant: Jens Van der Plas (jens.van.der.plas@vub.be)

Cloud Computing and Big Data Processing
BDP: Using Isabelle for your project

Introduction

This document describes how you can use the Isabelle cluster to test and benchmark your solution of the project. To do so, you first need to book a slot during which you can use the server. In this document, we describe how to book slots, how to use the server during your slot, as well as some rules that you must follow when using the server and practical tips.

Carefully read this document in full as it contains important information related to your project.
Consult this document if you run into problems first before sending me an email.

1 – Reserving benchmarking slots

To test your application on Isabelle, you will need to book a slot during which you will have exclusive access to the server. Every student gets two slots of two hours to test their project on Isabelle. To this end, we foresee two sets of two-hour slots. You are allowed to book one slot for each of the slot sets. You are, however, not forced to book two slots: if you have enough with your first slot, you don't need to book a second slot. Do note that **you are not allowed to book two slots in the same set!**

Make sure to not book your second slot too close to your first slot in case you need to make changes. Instructions to reserve the slots will be announced on Canvas. Note that there may be a deadline to book the slots and that slots may fill up. Therefore, do not wait too long to book your (first) slot.

Recall that to be able to access the server, you need to have an active Wilma account (i.e., the account to access the computer rooms at E.1.), which you can activate here in case you have not done so. Do this as soon as possible and send us an email after activating your account. **Note that after creating your account, our system administrator needs to manually grant you access to the server. In case you fail to activate your account in time, you may not be able to benchmark on the server during your foreseen slot!**

2 – Running your application on Isabelle

This section gives a brief overview of how to run an application on Isabelle. Remember that we also used Isabelle during the lab sessions. This time, however, you'll need to use your own personal account and start and stop Isabelle yourself. Carefully read the instructions in this section. We refer to the assignment of the lab session on Isabelle where possible. Do carefully read the corresponding instructions again for more information.

Wherever you read `<username>` in this document you should substitute it by username (i.e., your netID).

2.1 Basic commands

You will be able to access Isabelle during your time slots through SSH. If you are using a unix-based system just open a shell. If you are using a version of Windows older than Windows 10, you may take a look at PuTTY.

The basic commands you need to connect to Isabelle using SSH are the following:

- To open a connection with SSH tunneling:
  `ssh -o ServerAliveInterval=59 -L 9999:localhost:9999 <username>@isabelle.vub.ac.be`
- To close a connection (during the SSH session):
  `exit`

After typing the opening connection command, your password will be requested. Normally it should be your netID password. If it does not correspond, contact us.

Always remember to **exit** from **all** your SSH sessions when you finished your operations!

Carefully read the information in exercise 1 of the lab session on Isabelle.

2.2 Setting up Spark

In order to execute your programs you need to start Spark. To this end, we have configured a script that starts (and one that stops) everything you will need.

In order to be able to run a Spark application and to have access to the Spark UI, follow the following steps:

a) Connect to Isabelle via SSH as described above and insert your password.

b) **(Only if this is the first time you run Spark on your account.)** Extract Spark in your home directory by executing `/data/spark/extract.sh`

c) To start Spark, execute the following command in your home directory:
`./spark/start-on-isabelle.sh`

If Spark cannot be executed DURING YOUR SLOT (e.g., you receive an error saying that is still running), check Section 2.5.

2.3 Executing your project

To execute your project, follow the next steps:

a) Remove the `conf.setMaster(local[x])` line from your project.

b) Build your project in an executable jar file.

c) Using `scp`, PuTTY, or a graphical tool like FileZilla, copy the jar file to your home. Example `scp` command (executed in the directory where the jar file is stored):
`scp Project.jar <username>@isabelle.vub.ac.be:/home/<username>/Project.jar`

d) When the upload is over, use your opened SSH session to submit the jar file to Spark in the following way:
`./spark/bin/spark-submit --master spark://134.184.43.10:7077 --deploy-mode cluster Project.jar`
In the case Spark cannot find the main class, add before the jar path the parameter `--class NameOfTheClass`, where `NameOfTheClass` is the name of the class that contains the `main` method to be run.

e) Open a browser on localhost:9999 to check that your application has been submitted (and its progress).

f) You access the **Spark history server** at http://localhost:9999/proxy:localhost:18080 to check how your application executed and to analyse execution times.

Carefully read the information in exercises 2 and 3 of the lab session on Isabelle. Also have a look at the solutions to common issues at the end of the assignment.

2.4 Closing everything

Before exiting your SSH sessions, you should stop the execution of Spark and the various proxies. **THIS IS THE MOST IMPORTANT STEP OF THE PROCESS.** If you do not stop the execution of Spark correctly, the people after you will not be able to instantiate Spark, and hence not able to test their project. **If you forget this step, your evaluation will be impacted.**

To correctly stop everything, execute the following command from within your home directory:

```
./spark/stop-everything.sh
```

Wait for about 30 seconds and exit all your SSH sessions.

2.5 Attention!

Please, remember to stop everything and exit your SSH session **before** the end of your slot.

If you are starting your slot, but you have problems starting Spark (with errors like "`Could not start master (or worker) at PID. Master already started.`"):

a) Execute a `stop-everything` (see Section 2.4). This step is needed even if Spark did not start correctly.

b) Wait for 5 minutes.

c) Try again.

If you keep having problems starting Spark after 15 minutes, e.g., because someone else is still using the cluster 15 minutes into your slot, contact us. Also send us an email if you are having problems ending your session (i.e., because of weird errors when stopping everything). You may not get an immediate response (especially if you chose an evening slot), but in case you miss your slot because of such problems don't worry too much; we will try to give you a slot at another time if possible.

3 – Rules for using Isabelle

When you use the server, you must obey the following rules:

- You are allowed to use the cluster **only during the slot(s) assigned to you** (i.e., the slots you selected in the Doodle) and only in the context of this project. Isabelle is a research cluster also used by researchers who execute experiments on them and by your colleagues to perform theirs. Be respectful to your fellow students, me, my colleagues, and researchers and thesis students that use the cluster daily for research purposes. **Accesses will be monitored and unauthorised accesses out of hours will be punished**. This includes slot times not assigned to you, as well as nights, weekends, and any other time outside your designated slot (unless approved by the assistant). As you cannot access the server outside of your slot, make sure to copy all necessary files from the server to your local machine. In case something goes wrong or you encountered major issues on the server, do not use the server outside of your slot but contact the assistant.
- Try to end your session 10 minutes before your slot ends, to avoid going over time.
- Always remember: execute the `start-on-isabelle` script to start, but (most importantly) execute the `stop-everything` script before the end of allocated slot. This will ensure that all resources are freed and cleaned up for the student following you.
- Even if you run into problems, make sure that you always execute stop-everything by the end time of your slot. This applies even if Spark crashes (which shouldn't happen but you never know!).
- At the end of your session, always use `exit` to close your SSH session.

Not following these rules may have an impact on your score.

4 – Tips for using Isabelle

- Keep into account that slots can fill up. Book your slots in due time!
- As the last days for testing your project when approaching the deadline are normally fully booked, it's very important that you at least test the basic functionalities (parsing!) of your project in the first slot!
- At the beginning of your slot (just after SSH-ing but before starting Spark), check http://localhost:9999. If this page is available and shows Spark information, it means the student before you is still not finished. In this case, you are allowed to contact them to make them execute the stop-everything script. Since delays can happen, please give your fellow student 10 minutes of buffer time when your slot starts.
- In case you are having problems executing your jar on Isabelle, pack your jar using sbt assembly as explained in exercise 2.2 of the lab session. You can find the file `sbtAssembly.sbt` together with the material for the lab session on Canvas. This can solve the classic 'ClassNotFound' problem, and should generally be the first thing you do if your jar does not want to run because of compilation-related errors.
- You will have two slots on the server. It is highly recommended to run at least something in your first slot, even if you don't have a complete implementation (e.g., just the parsing). This way, you know if everything runs all right for your second slot.
- In case you run into problems such as weird errors not really related to your code, first make sure to take a look at the list of frequent solutions at the end of the assignment of the lab session. In case you do not manage to solve your problem, contact us. Do not stress if you don't immediately get an answer or if you get an out-of-office email:

in case you don't manage to run anything in your slot because of problems with Spark we can reschedule your slot; just be patient, we will answer your email!

- Make sure you retrieve all files you need (e.g., your results) from the server before the end of your slot.
- On the cluster, apart from the full dataset, some smaller datasets are also available. Keep in mind that just the initial parsing of the full dataset will take more than 10 minutes. We suggest to first test on a reduced dataset if possible. If everything works, then try to test your solution on the big dataset as well. Make sure to mention the datasets you used in your report.
- Recall that you can use the Spark history server to see the statistics of terminated applications, as seen during the lab session on Isabelle.