TP4: Practical Evaluation

Alessio Pagliari and Shadi Ibrahim
March 29, 2022

Deadline for sending the report is **April 8th** at **23:59**

- Reports (in PDF) should be sent by email to **alessio.pagliari@irisa.fr**, the name of the pdf should be `cbd_report_`*surname*`.pdf`.
- A constructed answer of each question is expected: discuss why you conduct this experiment and explain the results and your observations.
- It will be great if you can discuss also the experimental setup and the metrics for each question.
- Two exercises are necessary for the evaluation: Exercise 1 is **mandatory**, then **chose one** between Exercise 2 and Exercise 3.

Exercise 1: Speculative Execution

Configure Hadoop with the default blocksize (128MB), set the dfs replication to default (3), disable speculative execution (set it to false) and deploy the platform to have at least 4 worker nodes. Provide a dataset of 20GB.

Question 1.1 Run either wordcount or sort application. Then, enable Speculative Execution and rerun the same application. See the execution times, the number of speculative copies, and average (maximum) task runtimes for both map and reduce tasks, what can you observe?

Question 1.2 With Speculative Execution disabled, enter one node and run a small program to overload the CPU. Re-run the application. Now enable Speculative Execution and rerun the same benchmark. See the execution times, the number of speculative copies, and average (maximum) task runtimes for both map and reduce tasks, what do you observe?

To stress the CPU you can use the `stress` command (you may need to install it via `apt install stress`):

```
# stress -help
    'stress' imposes certain types of compute stress on your system

    Usage:  stress [OPTION [ARG]] ...
     -n, -dry-run show what would have been done
     -t, -timeout N timeout after N seconds
     -c, -cpu N spawn N workers spinning on sqrt()

    Example:  stress -cpu 8 -timeout 10s
```

Question 1.3 (Optional) Now, always with Speculative Execution Enabled, enter one node and run a program to stress the disk (i.e. I/O). Run the same application. What do you observe?

You can do it using the dd command:

```
# dd -help
    Copy a file, converting and formatting according to the operands.

    Usage:  dd [OPERAND]...
     bs=BYTES read and write up to BYTES bytes at a time (default:  512)
     if=FILE read from FILE instead of stdin
     of=FILE write to FILE instead of stdout
     of=FILE oflag=FLAGS write as per the comma separated symbol list

    Each FLAG symbol may be:
     direct use direct I/O for data
     dsync use synchronized I/O for data

    Example:  dd if=/dev/zero of=/tmp/test1.img bs=1G count=1 oflag=dsync
```

Question 1.4 (Optional) Try to stress the network at the same manner as the previous questions.

Chose **at least one** among the two following exercises

Exercise 2: System Failure

Configure Hadoop with default blocksize (128MB), set the dfs replication to **1** and deploy the platform in 3 to 5 nodes. Provide a dataset of 20GB.

Question 2.1 Run one application (either sort or wordcount) that takes enough time to run. Run the same application in two scenarios: in the first one turn off one datanode during the map phase and in the second one turn off the same datanode during the reduce phase. What do you observe?

Question 2.2 In case you observe execution problems, how do you solve it?

Exercise 3: Slowstart

Configure Hadoop with default blocksize, set the dfs replication to default (3) and deploy the platform in 3 to 5 nodes. Use a data set of 20GB.

Question 3.1 Configure slowstart and run Wordcount and Sort benchmark for different values of slowstart (i.e., 0.05, 0.5 and 1). See the execution times, what can you observe?