General Environment Setup

Seminar 1 of *NoSQL Databases* (PA195)

Luděk Bártek, Vlastislav Dohnal
Faculty of Informatics, Masaryk University, Brno

Agenda

- Stratus – cloud computing platform
- Virtual machines
- Simple example in Hadoop Framework
  - MapReduce and Spark (as a subset of Tutorial 1)

Seminar Organization

- 2-3 tasks solved in groups of 3-4 students
  - groups may vary from seminar to seminar
- Each task will practice a NoSQL technology
  - on a “real-life” example
  - You will typically use a cluster to solve the task.
    - Cluster will be formed by the machines of students in the group.
  - Teacher will check completing the task within the seminar.
- You must succeed in all assigned tasks!

Faculty account vs IS account

- Faculty of Informatics has their own user accounts
  - We will reference it as faculty credentials or FI credentials.
- See the general information of user logins
  - Briefly, the login is generated automatically according to the faculty relationship
    - `xlastname` – internal students (FI branch),
    - `xučo` – external students (from another faculty, ERASMUS, …).
  - If you do not know your FI's password, please use this IS app
    - https://is.muni.cz/auth/system/heslo_fi

Stratus – cloud platform @FI

- Uses OpenNebula cloud and edge computing platform
- Setup you access to Stratus.FI
  - For details see info on FI Technical Info page
  - Log into to the stratus.fi.muni.cz
    - using FI credentials.

Stratus.fi.muni.cz

- Firstly, setup SSH keys:
  - Generate an ssh key pair (if you do not have any yet):
    `musa$ ssh-keygen` OR `aisa$ ssh-keygen`
    - by default, stored in `$HOME/.ssh/id_rsa` and `$HOME/.ssh/id_rsa.pub`
  - In stratus' menu, navigate to *Settings > Auth tab*
  - Edit *Public SSH Key*
    - Copy&paste the contents of `$HOME/.ssh/id_rsa.pub`
  - The private key is then used to log into VM as root
- Do not use root password setup please.

Creating the Hadoop Server

- On the left, select Templates and VMs
  - Locate the template “PA195-hadoop-single”
    - Select it and click “Instantiate”
  - Go to the menu *Instances > VMs* to find your new VM.
    - Wait for the ready state

musa (local PC)

- Log into the virtual server as root:
  - `$ ssh root@<VMs_IP>`
    - It uses the preconfigured SSH key set in the your user profile at stratus.

HDFS DFS (1)

stratus (VM)

- HDFS system monitoring & basic commands

```
$ hdfs dfs -help
```

- Documentation of HDFS DFS file system commands

- Get some data (complete Shakespeare's plays)

```
# su - hadoop
$ wget https://is.muni.cz/go/zp93wh -O shake.txt
$ hdfs dfs -put shake.txt
```

HDFS DFS (2)

stratus (VM)

- Other hdfs commands: file list, file removal, directory creation
  - (you may not perform them)

```
# su - hadoop
$ hdfs dfs -ls
$ hdfs dfs -rm shake.txt
$ hdfs dfs -mkdir input
```

musa (local PC)

Check HDFS files in the web browser

http://<VM_ip>:9870/explorer.html#/user

MapReduce using Spark

- Spark is a multi-language engine for executing data engineering, data science, and machine learning on single-node machines or clusters.
  - Installed in your VM: doc

Task: Calculate word frequency in a document, e.g., `shake.txt`

Spark: Simple Example

stratus (VM)

```
# su - hadoop
$ spark-shell --master yarn
scala> :help
scala> val file =
  sc.textFile("hdfs:///user/hadoop/shake.txt")
scala> val counts = file
  .flatMap(line => line.split(" "))
  .map(word => (word,1))
  .reduceByKey(_ + _)
scala> counts.saveAsTextFile("spark-output")
scala> :quit
$ hdfs dfs -get spark-output/
```

Lessons Learned & Cleanup

What lessons did we take from the following?

- Basic work with the HDFS distributed file system
- Hadoop MapReduce using Spark
  - simple word count

Delete large files from both HDFS and the your home dir in VM, and shutdown you Stratus VM, if not needed anymore.

stratus (VM)

```
# su - hadoop
$ hdfs dfs -rm -R wiki-input/
$ hdfs dfs -rm -R output
```