**Class 1 :**

**Q.1 Study Oozie components in details, For reference follow the link** <http://oozie.apache.org/docs/4.3.0/index.html>

Answer: Ooozie is an extremely complicated equivalent of a queue manager that requires a lot of dependencies and takes by average approx. 2 hours to compile with all tests. Therefore, for a lab bench it seems to be an overkill and its better to use commands dierectly (as per below).

**Q.2 Try out below commands and see the impact :**

a. moveToLocal

b. touchz

c. moveToLocal

d. copyFromLocal

e. appendToFile

f. getfattr

g. setrep

h. getmerge

|  |  |  |
| --- | --- | --- |
| **Command** | **Full syntax example** | **Comment** |
| moveToLocal |  |  |
| touchz |  |  |
| moveToLocal |  |  |
| copyFromLocal |  |  |
| appendToFile |  |  |
| getfattr |  |  |
| setrep |  |  |
| getmerge |  |  |

Answer: Check out <https://hadoop.apache.org/docs/r2.4.1/hadoop-project-dist/hadoop-common/FileSystemShell.html>

**Q.3 What is check pointing in HDFS ?**

Answer: In Hadoop world the 'Edit' is an equivalent of a CRUD or a transaction if you like. By default a size of such operation in HDFS (Hadoop File System) would rande from >=10b to X kb... now, because HDFS concurrency model is calculated in memore and peristence can be postponed due to calculation a specific “transaction” or a whole node can get out of sync. Since heach transaction is bearing a specific “TransactionID” resolved by getrasactionId()

In a shortest definition the “check pointing” is a form of (resource-expensive) arbitration whether a “commit” (the complete save/update operation was conducted) or... should the content (if exists) for read ops be served from a secondary source (node).

**Q.4 Secondary name node myths and truths.** [**http://blog.cloudera.com/blog/2009/02/multi-host-secondarynamenode-configuration/**](http://blog.cloudera.com/blog/2009/02/multi-host-secondarynamenode-configuration/)

Secondaty node is not a slave. It can facilitate a re-build of the “NameNode” by the virtue of providing for it a reference point of the most recent (communicated by the NameNode) snapshot (without the overhanging transaction history and context).

So, largely the difference is that it is NOT A SLAVE, but part of the flexible ecosystem empowering the Hadoop resiliance.

---------------------------

PRE-REQS

Config of the hadoop bench...

**Ad. 1 – Installation of Oozie for inspection**

Dependencies required to install and run oozie (from Apache Oozie Project website):

* Unix box (tested on Mac OS X and Linux)
* Java JDK 1.7+
* Maven 3.0.1+
* Pig 0.7+
* Hadoop 0.20.2+

**Step 1 – Validating we are on linux box.**

Run following command:

|  |
| --- |
| lsb\_release -a 2>/dev/null |

Expected:

Distributor ID: Ubuntu

Description: Ubuntu 14.04.5 LTS

Release: 14.04

Codename: trusty

**Step 2 – Installation of maven (and JDK for Java 8/1.8)**

Install maven and Java as its side-effect.

|  |  |
| --- | --- |
| sudo apt-get update; sudo apt-cache search maven; sudo apt-get install maven | |
|  | (1) Semicolon ; denotes a new command (new line for shell/bash) (2) Ubuntu and Debian do not support Oracle's Java anymore, but installation via maven would install Java8 for machines that do not have any JDK installed yet |

Now verify maven installation:

|  |
| --- |
| mvn --version |

Expected:

Apache Maven 3.0.5

Maven home: /usr/share/maven

Java version: 1.8.0\_144, vendor: Oracle Corporation

Java home: /usr/lib/jvm/java-8-oracle/jre

Default locale: en\_GB, platform encoding: UTF-8

OS name: "linux", version: "4.4.0-93-generic", arch: "amd64", family: "unix"

now for java ...

|  |
| --- |
| java -version |

Expected:

java version "1.8.0\_144"

Java(TM) SE Runtime Environment (build 1.8.0\_144-b01)

Java HotSpot(TM) 64-Bit Server VM (build 25.144-b01, mixed mode

Gradle etc.

curl -s "https://get.sdkman.io" | bash

source "$HOME/.sdkman/bin/sdkman-init.sh"

sdk version

sdk install gradle 4.3

**Step 3 – Installation of pig (*the highest version to date is 0.17 so the dependencies above from Oozie must be wrong*)**

Download first... and check size before unpacking...

|  |
| --- |
| wget http://www-us.apache.org/dist/pig/latest/pig-0.17.0.tar.gz  ls -lah pig-0.17\* |

Expected:

-rw-rw-r-- 1 {group} {user} 220M Jun 20 10:31 pig-0.17.0.tar.gz

unpack and move pig to target location

|  |
| --- |
| tar -xzf pig-0.17.0.tar.gz sudo mv pig-0.17.0 /usr/bin  sudo cd /usr/bin  sudo chmod 777 pig-0.17.0/  sudo ln -s pig-0.17.0/ apache-pig  cd # return to your home directory |

Create all environment variables needed to run pig for your user:

|  |
| --- |
| cat > .apache\_pig.conf  # Set PIG\_HOME  export PIG\_HOME=/usr/bin/apache-pig  export PATH=$PATH:/usr/bin/apache-pig/bin  export PIG\_CLASSPATH=$HADOOP\_CONF\_DIR # end with <Ctrl+D> |

Load this file for your user in .bashrc (and source your bash env)

# Pig config

source ~/.apache\_pig.conf

validate installation...

|  |
| --- |
| pig --version |

Expected:

Apache Pig version 0.17.0 (r1797386)

compiled Jun 02 2017, 15:41:58

HERE installation of Hadoop and Oozie.... then Q1

Install **oozie**

wget <http://archive.apache.org/dist/oozie/4.3.0/oozie-4.3.0.tar.gz>

Download first... and check size before unpacking...

|  |
| --- |
| wget http://archive.apache.org/dist/oozie/4.3.0/oozie-4.3.0.tar.gz  ls -lah oozie\* |

Expected:

-rw-rw-r-- 1 {group} {user} 220M Jun 20 10:31 pig-0.17.0.tar.gz

unpack and move pig to target location

|  |
| --- |
| tar -xzf oozie-4.3.0.tar.gz sudo mv oozie-4.3.0 /usr/bin  sudo cd /usr/bin  sudo chmod 777 oozie-4.3.0/  sudo ln -s oozie-4.3.0/ oozie  cd # return to your home directory |

Create all environment variables needed to run pig for your user:

|  |
| --- |
| cat > .apache\_oozie.conf  # Set OOZIE\_HOME  export OOZIE\_HOME=/usr/bin/oozie  export PATH=$PATH:/usr/bin/oozie/bin |