Assignment 1 Hints

CSCI4180

Qin Chuan

Assignment 1

- Due on **Oct. 24**
- Configure VMs & Azure platform
- Write Java program
  - Word length count
  - N-gram count
  - N-gram relative frequency
- Test on the KJV & shakespeare data
- Do some optimizations

Pass Arguments

```
public class MapRedProg {
// Define Mapper Class
    public static class MyMap extends Mapper<KEY_IN, VAL_IN, KEY_OUT, VAL_OUT> {
        ......
        protected void map(KEY_IN key, VAL_IN val, Context context) {
            Configuration conf = context.getConfiguration();
            gram = Integer.parseInt(conf.get(“ngram”));
        }
    }
// Main Function, Job Configuration and Starting Point
    public static void main(String [] args) {
        conf.set(“ngram”,args[2]);
        ......
    }
}
```

Part 3

- N-gram Initial
    - Eg. N = 3, for “who is it” we have (w i i 1)
    - N-gram means N consecutive words
    - Initial means first character of the word
    - **Alphabet** means A-Z and a-z
- N-gram across Rows
    - Eg. “how can I finish this assignment on time without the help of my groupmates?”
    - N = 3, “on time without” should count (o t w 1) and “time without the” should count (t w t 1)

Part 4

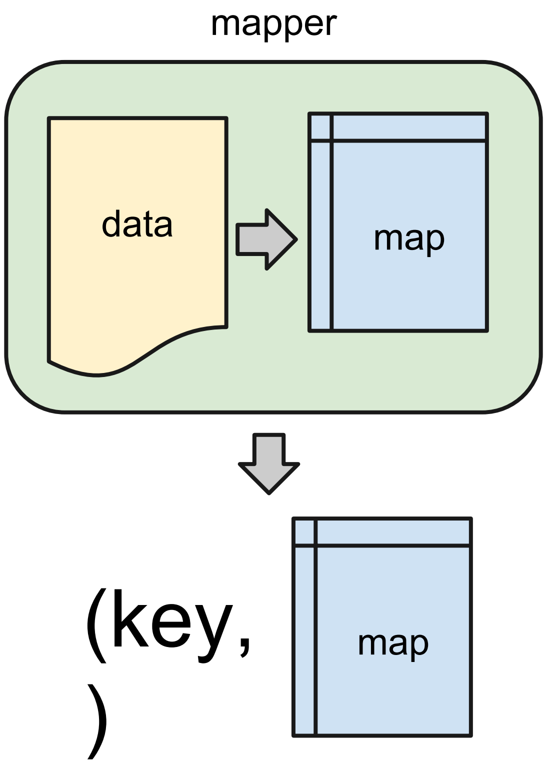
- N-gram Initial Relative Frequency
  - Eg. N = 3 “who is it? We want to know”
  - How frequent is initial w followed initial i i?
  - (w i i 1)(w w t 1)(w t k 1)
  - RF(w i i) = ⅓ = 0.333
- Only Alphabet counts
  - Eg. (w > i 1)(w “ a 1) won’t count
  - You need to think about data structure to store intermediate data to compute RF

Problem

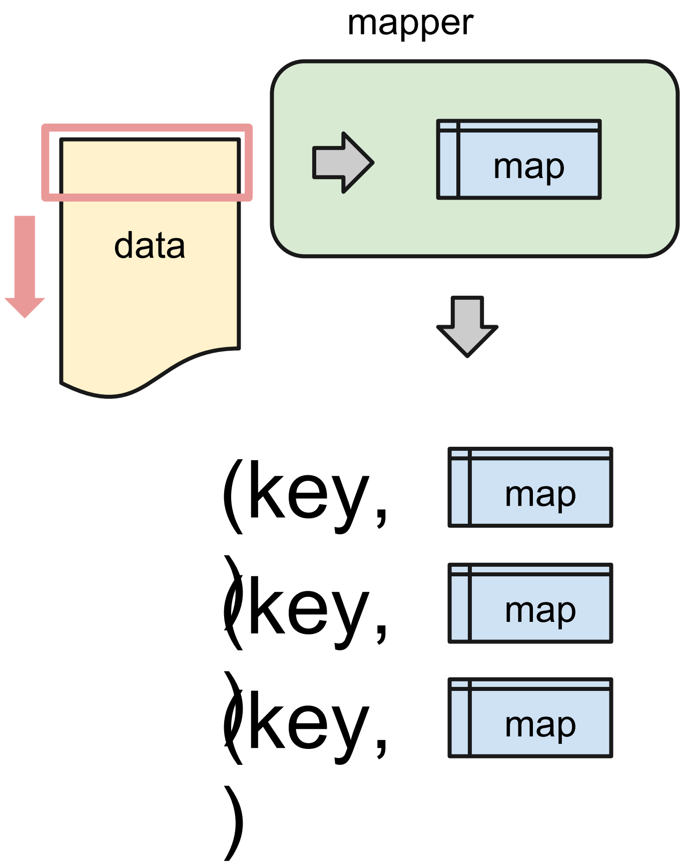
- Hadoop cannot handle too many emit pairs
  - We need to reduce the number of emit times
- Use in-mapper combining technique
  - Map, vector to centralize information
  - Emit combined pairs
- Memory Limit
  - We cannot hold everything for large case
  - Set a bound for map size, emit when full

Solution

Original:



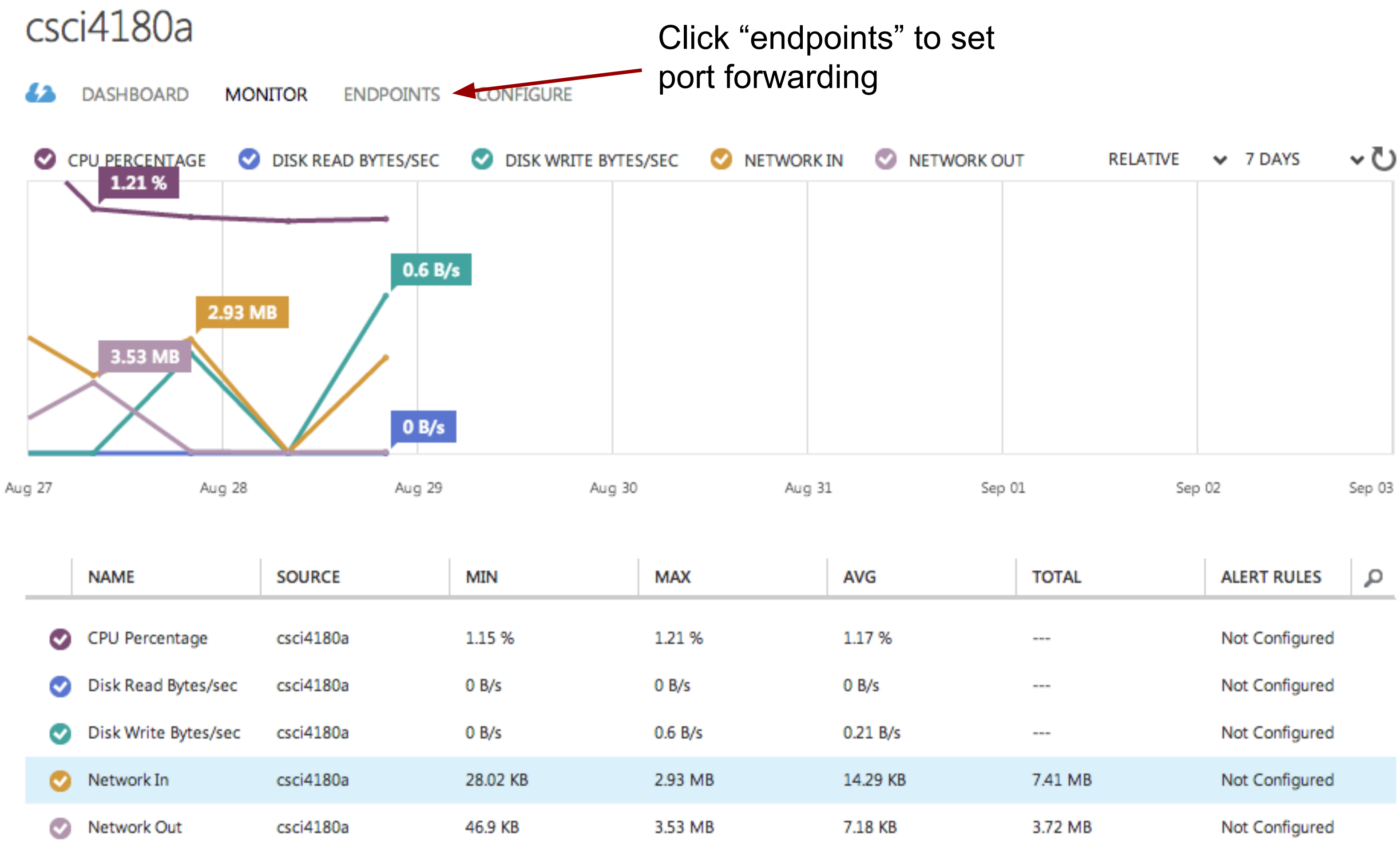
Modified:



Part 5

- Redeem the Azure Code
- Create 4 VMs
- Install Hadoop and set the cluster
- Configure the hadoop
- Start the hadoop service
- Compile the sample wordcount.java
- Run wordcount on the given data sets

Port Forwarding

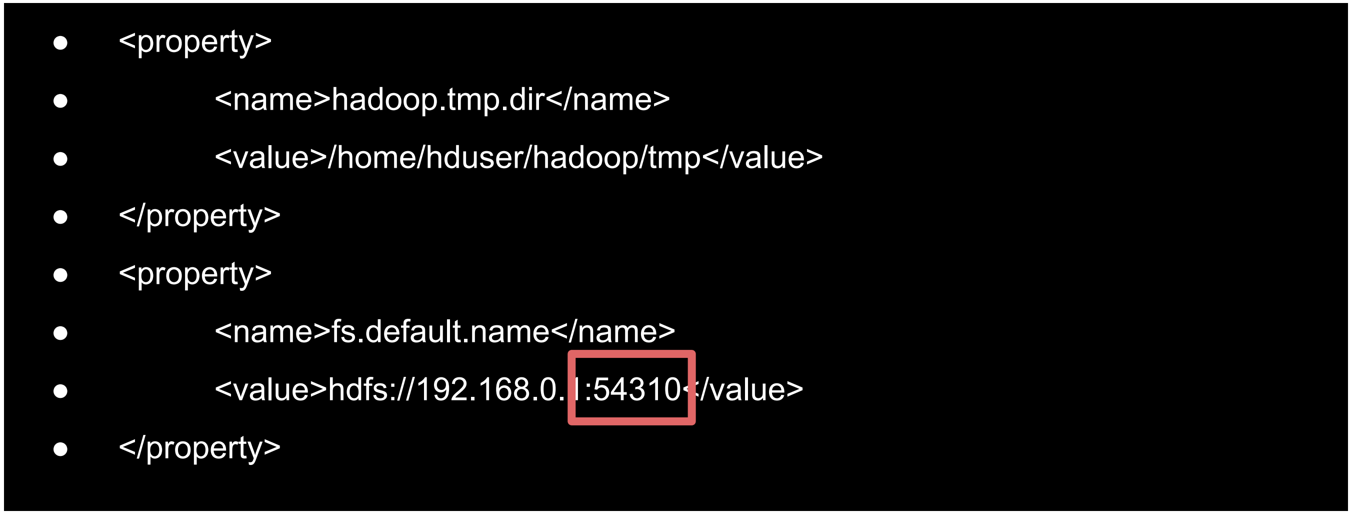


Click “endpoints” to set port forwarding

	NAME	SOURCE	MIN	MAX	AVG	TOTAL	ALERT RULES
✓	CPU Percentage	csci4180a	1.15 %	1.21 %	1.17 %	---	Not Configured
✓	Disk Read Bytes/sec	csci4180a	0 B/s	0 B/s	0 B/s	---	Not Configured
✓	Disk Write Bytes/sec	csci4180a	0 B/s	0.6 B/s	0.21 B/s	---	Not Configured
✓	Network In	csci4180a	28.02 KB	2.93 MB	14.29 KB	7.41 MB	Not Configured
✓	Network Out	csci4180a	46.9 KB	3.53 MB	7.18 KB	3.72 MB	Not Configured

Port Forwarding

- Set port for hadoop core
  - In **hadoop/conf/core-site.xml**



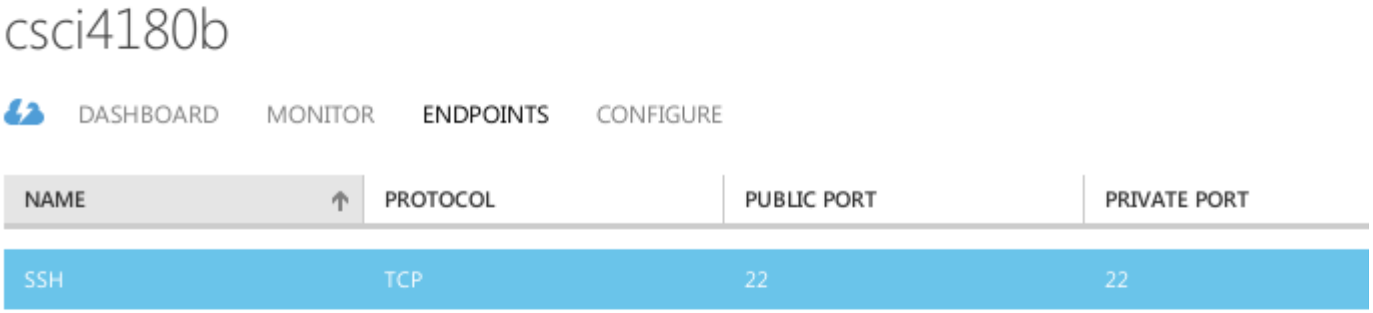
```
●   <property>
●           <name>hadoop.tmp.dir</name>
●           <value>/home/hduser/hadoop/tmp</value>
●   </property>
●   <property>
●           <name>fs.default.name</name>
●           <value>hdfs://192.168.0.1:54310</value>
●   </property>
```

Port Forwarding

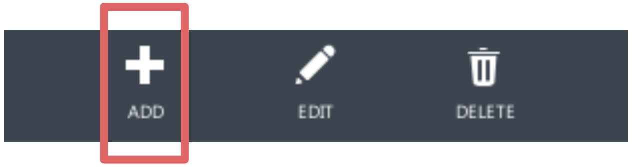
- Set port for hadoop mapred-site
  - In **hadoop/conf/mapred-site.xml**

```
●    <property>
●              <name>mapred.job.tracker</name>
●              <value>192.168.0.1:54311</value>
●    </property>
```

Port Forwarding

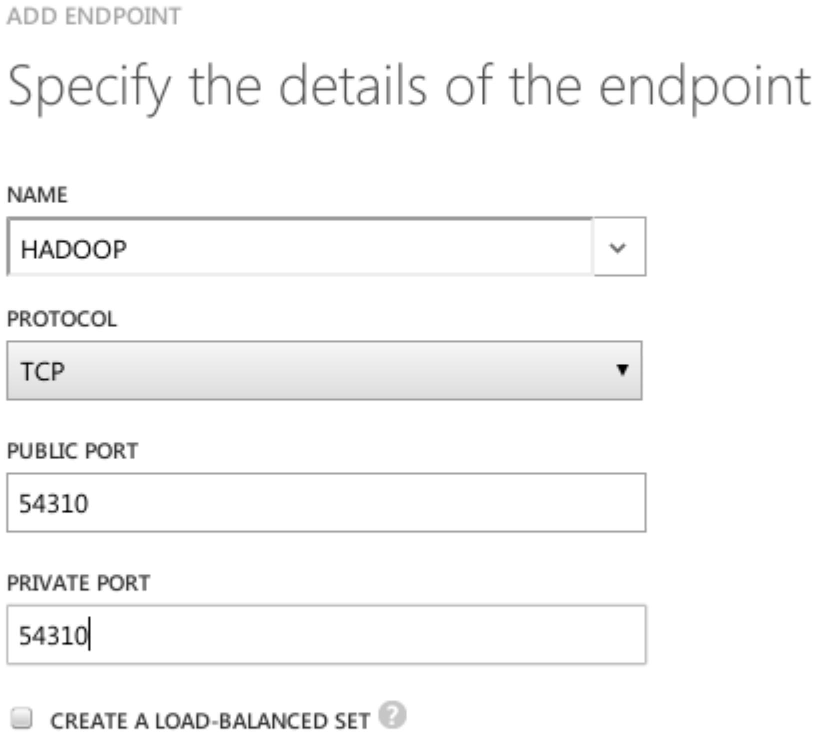


NAME	PROTOCOL	PUBLIC PORT	PRIVATE PORT
SSH	TCP	22	22



- Originally only port 22 is forwarded
- Click add to continue

Port Forwarding



ADD ENDPOINT

Specify the details of the endpoint

NAME

HADOOP

PROTOCOL

TCP

PUBLIC PORT

54310

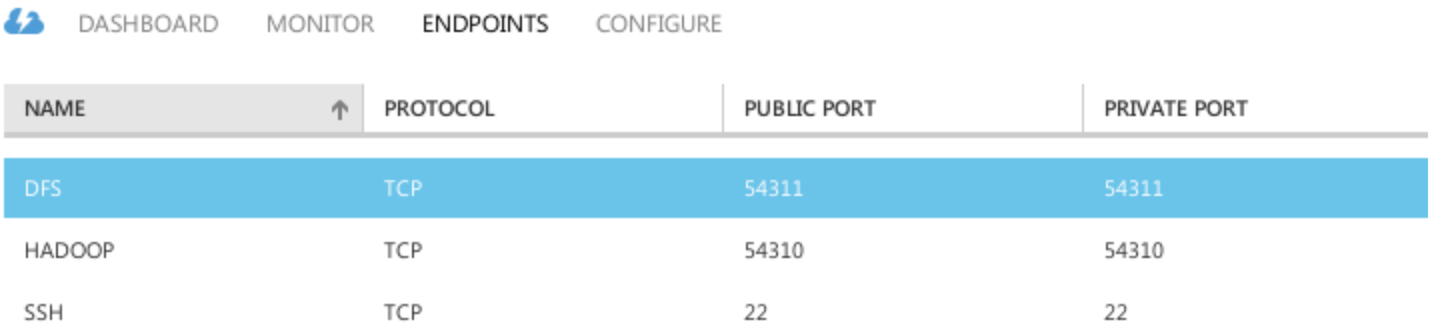
PRIVATE PORT

54310

□ CREATE A LOAD-BALANCED SET

- Add both ports for hadoop service and DFS

Port Forwarding



NAME	PROTOCOL	PUBLIC PORT	PRIVATE PORT
DFS	TCP	54311	54311
HADOOP	TCP	54310	54310
SSH	TCP	22	22

- After both ports are forwarded, we can use public IP to access the hadoop service

Thank you