Hadoop Code Tracing Techniques & Operating flow

Chun-Chen Tu

timtu@umich.edu

Outline

- Useful tools and technique
  - Eclipse
  - Java stack trace
- Hadoop operating flow
  - Job execution.
    - Job submit
    - JobTracker
    - TaskTracker
      - Map
      - Reduce

Eclipse

- Why choose Eclipse?
  - Free
  - GUI : easy to browse (compared to vim)
  - Several utilities : jump to declaration, syntax checking, search … etc.
  - People use Eclipse, thus many reference on the Internet.

Code tracing …

- Print out message or variable content.

```
hadoop@ubuntu:~/hadoop$ hadoop jar hadoop-examples-1.2.1.jar pi 1 100
Hi Mom, it's me
Number of Maps  = 1
Samples per Map = 100
14/02/15 17:05:23 INFO util.NativeCodeLoader: Loaded the native-hadoop library
Wrote input for Map #0
Starting Job
```

- Print stack trace

stdout logs

```
StackTrace:
nulljava.lang.Thread.getStackTrace(Thread.java:1588)
org.myorg.mywordcount$Map.map(mywordcount.java:24)
org.myorg.mywordcount$Map.map(mywordcount.java:11)
org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:50)
org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:430)
org.apache.hadoop.mapred.MapTask.run(MapTask.java:366)
org.apache.hadoop.mapred.Child$4.run(Child.java:255)
java.security.AccessController.doPrivileged(Native Method)
javax.security.auth.Subject.doAs(Subject.java:415)
org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1190)
org.apache.hadoop.mapred.Child.main(Child.java:249)
```

Java stack trace

```
public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter reporter)
{
        String line = value.toString();
        StringTokenizer tokenizer = new StringTokenizer(line);
        while (tokenizer.hasMoreTokens())
        {
                word.set(tokenizer.nextToken());
                output.collect(word, one);
        }
        StackTraceElement [] ste = Thread.currentThread().getStackTrace();
        String msg = null;
        for(StackTraceElement st: ste)
                msg=msg+st.toString()+"\n";
        System.out.println("StackTrace:"+"\n"+msg);
}
```

stdout logs

```
StackTrace:
nulljava.lang.Thread.getStackTrace(Thread.java:1588)
org.myorg.mywordcount$Map.map(mywordcount.java:24)
org.myorg.mywordcount$Map.map(mywordcount.java:11)
org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:50)
org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:430)
org.apache.hadoop.mapred.MapTask.run(MapTask.java:366)
org.apache.hadoop.mapred.Child$4.run(Child.java:255)
java.security.AccessController.doPrivileged(Native Method)
javax.security.auth.Subject.doAs(Subject.java:415)
org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1190)
org.apache.hadoop.mapred.Child.main(Child.java:249)
```

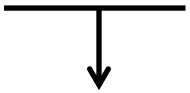
How to get this?

- Add code in target function.
- Compile
- Run hadoop
- Check console or log

Brief flow tracing for a job

What will actually operate after you type

hadoop jar hadoop-examples-1.2.1.jar wordcount input output



shell script under
$HADOOP_HOME/bin

```
hadoop@ubuntu:~/hadoop$ file bin/hadoop
bin/hadoop: a bash script, ASCII text executable
```

In this shell script, the remaining arguments will be parsed and finally invoke relative java program.

Brief flow tracing for a job

What will actually operate after you type

hadoop jar hadoop-examples-1.2.1.jar wordcount input output

In shell script, java CLASS is defined according to “jar” argument.

```
elif [ "$COMMAND" = "jar" ] ; then
  CLASS=org.apache.hadoop.util.RunJar
  HADOOP_OPTS="$HADOOP_OPTS $HADOOP_CLIENT_OPTS"
```

```
# run it
exec "$JAVA" -Dproc_$COMMAND $JAVA_HEAP_MAX $HADOOP_OPTS -classpath "$CLASSPATH" $CLASS "$@"
```

exec /usr/lib/jdk/bin/java -Dproc_jar ..*Several classpath*… \
org.apache.hadoop.util.RunJar hadoop-examples-1.2.1.jar wordcount input output

```
public static void main(String[] args) throws Throwable
{
        … Load and unjar JAR file…
        …Get the main Java class…
        …Get the argument…
        …Invoke main function in JAR file…
}
```


```
String[] newArgs = Arrays.asList(args)
  .subList(firstArg, args.length).toArray(new String[0]);
try {
  main.invoke(null, new Object[] { newArgs });
} catch (InvocationTargetException e) {
  throw e.getTargetException();
}
```

The invoked class inside hadoop-examples-1.2.1.jar

```
public static void main(String[] argv) throws Exception {
  System.exit(ToolRunner.run(null, new PiEstimator(), argv));
}
```

```
public int run(String[] args) throws Exception {
  if (args.length != 2) {
    System.err.println("Usage: "+getClass().getName()+" <nMaps> <nSamples>")
    ToolRunner.printGenericCommandUsage(System.err);
    return -1;
  }

  final int nMaps = Integer.parseInt(args[0]);
  final long nSamples = Long.parseLong(args[1]);

  System.out.println("Number of Maps  = " + nMaps);
  System.out.println("Samples per Map = " + nSamples);

  final JobConf jobConf = new JobConf(getConf(), getClass());
  System.out.println("Estimated value of Pi is "
      + estimate(nMaps, nSamples, jobConf));
  return 0;
}
```

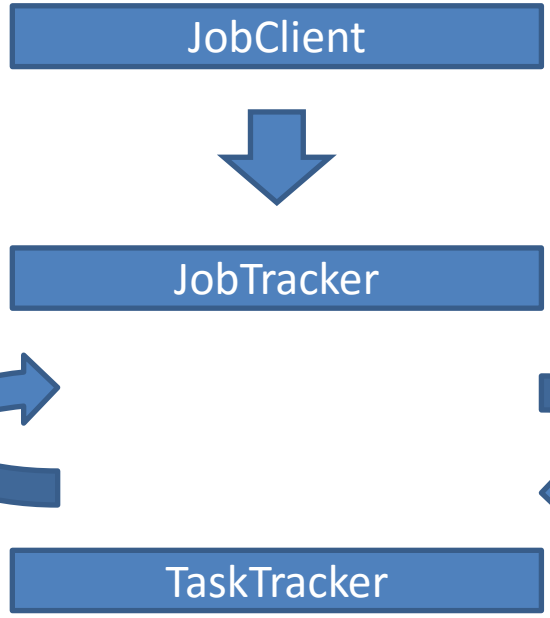
```
public static BigDecimal estimate(int numMaps, long numPoints, JobConf jobConf
    ) throws IOException {
  //setup job conf
  jobConf.setJobName(PiEstimator.class.getSimpleName());

  jobConf.setInputFormat(SequenceFileInputFormat.class);

  jobConf.setOutputKeyClass(BooleanWritable.class);
  jobConf.setOutputValueClass(LongWritable.class);
  jobConf.setOutputFormat(SequenceFileOutputFormat.class);

  jobConf.setMapperClass(PiMapper.class);
  jobConf.setNumMapTasks(numMaps);

  jobConf.setReducerClass(PiReducer.class);
  jobConf.setNumReduceTasks(1);
  final Path inDir = new Path(TMP_DIR, "in");
  final Path outDir = new Path(TMP_DIR, "out");
  FileInputFormat.setInputPaths(jobConf, inDir);
  FileOutputFormat.setOutputPath(jobConf, outDir);
  //start a map/reduce job
  System.out.println("Starting Job");
  final long startTime = System.currentTimeMillis();
  JobClient.runJob(jobConf);
  final double duration = (System.currentTimeMillis() - startTime)/1000.0;
  System.out.println("Job Finished in " + duration + " seconds");
```



TaskTracker Routine:

TaskTracker.java

1736: State offerService()

1826: HeartbeatResponse heartbeatResponse = transmitHeartBeat(now);

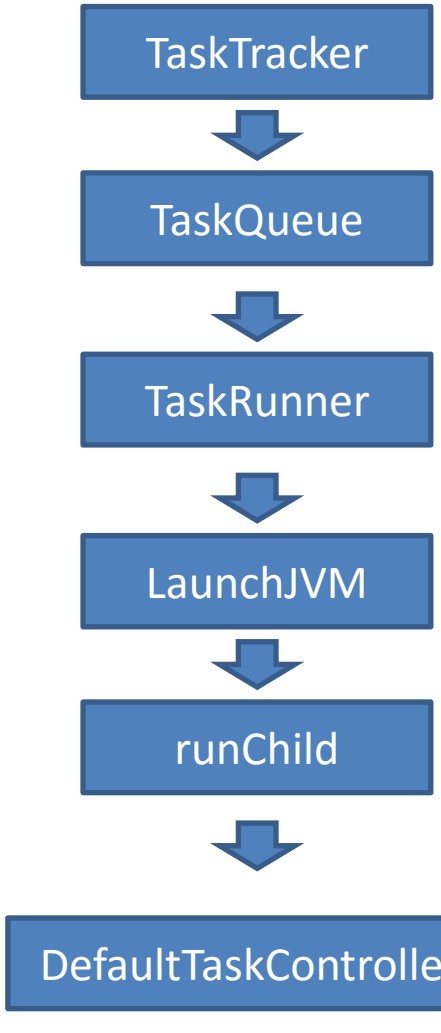
Assign Task according to heartbeat

```
1864: TaskTrackerAction[] actions = heartbeatResponse.getActions();
1879: if (actions != null){
        for(TaskTrackerAction action: actions) {
          if (action instanceof LaunchTaskAction) {
                  addToTaskQueue((LaunchTaskAction)action);
                  }
                  …Other operations…
          }
```

1: *JobTracker.java* 2933: public synchronized HeartbeatResponse heartbeat

2: *TaskTracker.java* 1950: HeartbeatResponse transmitHeartBeat(long now)

TaskTracker: Launch a task



Shell script

exec setsid '/usr/lib/jdk/jre/bin/java'
'-Djava.library.path=/home/hadoop/hadoop/libexec/../lib/native/Linux-amd64-64'
'-Xmx200m'
'-Djava.io.tmpdir=/tmp/hadoop-hadoop/mapred/local/taskTracker/hadoop/jobcache/job_201404211533_0003/attempt_201404211533_0003_m_000001_0/work/tmp'
'-classpath' ***..Some classpath…***
'-Dhadoop.log.dir=/home/hadoop/hadoop/libexec/../logs'

'org.apache.hadoop.mapred.Child'
'127.0.0.1'
'33873'
'attempt_201404211533_0003_m_000001_0'
'/home/hadoop/hadoop/libexec/../logs/userlogs/job_201404211533_0003/attempt_201404211533_0003_m_000001_0'
'223969752'
< /dev/null 1>>
/home/hadoop/hadoop/libexec/../logs/userlogs/job_201404211533_0003/attempt_201404211533_0003_m_000001_0/stdout 2>>
/home/hadoop/hadoop/libexec/../logs/userlogs/job_201404211533_0003/attempt_201404211533_0003_m_000001_0/stderr

- Write shell script command
- Execute shell script

Shell script

```
'org.apache.hadoop.mapred.Child'
 '127.0.0.1'
 '33873'
 'attempt_201404211533_0003_m_000001_0'
'/home/hadoop/hadoop/libexec/../logs/userlogs/job_201404211533_0003/attempt_2014042
11533_0003_m_000001_0'
 '223969752'
< /dev/null
 1>>
/home/hadoop/hadoop/libexec/../logs/userlogs/job_201404211533_0003/attempt_2014042
11533_0003_m_000001_0/stdout
2>>
/home/hadoop/hadoop/libexec/../logs/userlogs/job_201404211533_0003/attempt_2014042
11533_0003_m_000001_0/stderr
```

Child.java

```
public static void main(String[] args){
   String host = args[0];
   int port = Integer.parseInt(args[1]);
   final TaskAttemptID firstTaskid = TaskAttemptID.forName(args[2]);
   final String logLocation = args[3];
   int jvmIdInt = Integer.parseInt(args[4]);

   …Some operations…
   childUGI.doAs(new PrivilegedExceptionAction<Object>() {
      public Object run() throws Exception {
         taskFinal.run(job, umbilical);
      }
   …Other operations…
}
```

stdout logs

```
StackTrace:
nulljava.lang.Thread.getStackTrace(Thread.java:1588)
org.myorg.mywordcount$Map.map(mywordcount.java:24)
org.myorg.mywordcount$Map.map(mywordcount.java:11)
org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:50)
org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:430)
org.apache.hadoop.mapred.MapTask.run(MapTask.java:366)
org.apache.hadoop.mapred.Child$4.run(Child.java:255)
java.security.AccessController.doPrivileged(Native Method)
javax.security.auth.Subject.doAs(Subject.java:415)
org.apache.hadoop.security.UserGroupInformation.doAs(UserGroupInformation.java:1190)
org.apache.hadoop.mapred.Child.main(Child.java:249)
```

Mapper.java

```
public void run(Context context) throws IOException, InterruptedException {
  setup(context);
  try {
    while (context.nextKeyValue()) {
      map(context.getCurrentKey(), context.getCurrentValue(), context);
    }
  } finally {
    cleanup(context);
  }
}
```

wordcount.java (User-defined code)

```
public static class Map extends Mapper<LongWritable, Text, Text, IntWritable>
{
    private final static IntWritable one = new IntWritable(1);
    private Text word = new Text();

    public void map(LongWritable key, Text value, Context context) throws IOException, InterruptedException
    {
        String line = value.toString();
        StringTokenizer tokenizer = new StringTokenizer(line);
        while (tokenizer.hasMoreTokens())
        {
            word.set(tokenizer.nextToken());
            context.write(word, one);
        }
    }
}
```

Reducer.java

```
public void run(JobConf job, final TaskUmbilicalProtocol umbilical)
  throws IOException, InterruptedException, ClassNotFoundException {
  this.umbilical = umbilical;
  job.setBoolean("mapred.skip.on", isSkipping());

  if (isMapOrReduce()) {
    copyPhase = getProgress().addPhase("copy");
    sortPhase  = getProgress().addPhase("sort");
    reducePhase = getProgress().addPhase("reduce");
  }
```

copy phase

```
if (!isLocal) {
  reduceCopier = new ReduceCopier(umbilical, job, reporter);
  if (!reduceCopier.fetchOutputs()) {
    if(reduceCopier.mergeThrowable instanceof FSError) {
      throw (FSError)reduceCopier.mergeThrowable;
    }
    throw new IOException("Task: " + getTaskID() +
        " - The reduce copier failed", reduceCopier.mergeThrowable);
  }
}
copyPhase.complete();                         // copy is already complete
```

sort phase

```
setPhase(TaskStatus.Phase.SORT);
statusUpdate(umbilical);

final FileSystem rfs = FileSystem.getLocal(job).getRaw();
RawKeyValueIterator rIter = isLocal
  ? Merger.merge(job, rfs, job.getMapOutputKeyClass(),
      job.getMapOutputValueClass(), codec, getMapFiles(rfs, true),
      !conf.getKeepFailedTaskFiles(), job.getInt("io.sort.factor", 100),
      new Path(getTaskID().toString()), job.getOutputKeyComparator(),
      reporter, spilledRecordsCounter, null)
  : reduceCopier.createKVIterator(job, rfs, reporter);

// free up the data structures
mapOutputFilesOnDisk.clear();

sortPhase.complete();                         // sort is complete
```

reduce phase

```
setPhase(TaskStatus.Phase.REDUCE);
statusUpdate(umbilical);
Class keyClass = job.getMapOutputKeyClass();
Class valueClass = job.getMapOutputValueClass();
RawComparator comparator = job.getOutputValueGroupingComparator();

if (useNewApi) {
  runNewReducer(job, umbilical, reporter, rIter, comparator,
                keyClass, valueClass);
} else {
  runOldReducer(job, umbilical, reporter, rIter, comparator,
                keyClass, valueClass);
}
done(umbilical, reporter);
```

Reducer.java

```
public void run(Context context) throws IOException, InterruptedException {
  setup(context);
  try {
    while (context.nextKey()) {
      reduce(context.getCurrentKey(), context.getValues(), context);
    }
  } finally {
    cleanup(context);
  }
}
```

wordcount.java (User-defined code)

```
public static class Reduce extends Reducer<Text, IntWritable, Text, IntWritable>
{
    public void reduce(Text key, Iterable<IntWritable> values, Context context) throws IOException
    {
        int sum = 0;
        for (IntWritable val : values)
        {
            sum+=val.get();
        }

        context.write(key, new IntWritable(sum));
    }
}
```