Yahoo! welcomes you

to the

Hadoop Bay Area User Group

September 23rd, 2009

Agenda

- Upgrading to the New MapReduce API
  - Owen O'Malley, Yahoo!
- Mining the web with Hadoop, Cascading & Bixo
  - Ken Krugler
- QnA and Open Discussion

Upgrading to the New MapReduce API

September 23rd, 2009

Top Level Changes

- Change all of the “mapred” packages to “mapreduce”.
- Methods can throw InterruptedException as well as IOException.
- Use Configuration instead of JobConf
- Library classes moved to mapreduce.lib.{input,map,output,partition,reduce}.*
- **Don’t Panic!**

Mapper

- Change map function signature from:
  - map(K1 key, V1 value,
    OutputCollector<K2,V2> output,
    Reporter reporter)
  - map(K1 key, V1 value, Context context)
  - context replaces output and reporter.
- Change close() to
  - cleanup(Context context)

Mapper (cont)

- Change output.collect(K,V) to
  - context.write(K,V)
- Also have setup(Context context) that can replace your configure method.

MapRunnable

- Use mapreduce.Mapper
- Change from:
  - void run(RecordReader<K1,V1> input,
                OutputCollector<K2,V2> output,
                Reporter reporter)
  - void run(Context context)

Reducer and Combiner

- Replace:
  - void reduce(K2, Iterator<V2> values, OutputCollector<K3,V3> output)
  - void reduce(K2, Iterable<V2> values, Context context)
- Also replace close() with:
  - void cleanup(Context context)
- Also have setup and run!

Reducer and Combiner (cont)

- Replace
  - while (values.hasNext()) {
    V2 value = values.next(); … }
  - for(V2 value: values) { … }
- Users of the grouping comparator can use context.getCurrentKey to get the real current key.

Submitting Jobs

- Replace the JobConf and JobClient with Job.
  - The Job represents the entire job instead of just the configuration.
  - Set properties of the job.
  - Get the status of the job.
  - Wait for job to complete.

Submitting Jobs (cont)

- Job constructor:
  - job = new JobConf(conf, MyMapper.class)
    job.setJobName(“job name”)
  - job = new Job(conf, “job name”)
    job.setJarByClass(MyMapper.class)
- Job has getConfiguration
- FileInputFormat in mapreduce.lib.input
- FileOutputFormat in mapreduce.lib.output

Submitting Jobs (cont)

- Replace:
  - JobClient.runJob(job)
  - System.exit(job.waitForCompletion(true)?0:1)

InputFormats

- Replace:
  - InputSplit[] getSplits(JobConf job, int numSplits)
  - List<InputSplit> getSplits(JobContext context)
- Replace:
  - RecordReader<K,V>
    getRecordReader(InputSplit split, JobConf job,
    Reporter reporter)
  - RecordReader<K,V>
    createRecordReader(InputSplit split,
    TaskAttemptContext context)

InputFormat (cont)

- There is no replacement for numSplits (mapred.map.tasks).
- FileInputFormat just uses:
  - block size
  - mapreduce.input.fileinputformat.minsize
  - mapreduce.input.fileinputformat.maxsize
- Replace MultiFileInputFormat with CombineFileInputFormat

RecordReader

- Replace:
  - boolean next(K key, V value)
  - K createKey()
  - V createValue()
- With:
  - boolean nextKeyValue()
  - K getCurrentKey()
  - V getCurrentValue()

RecordReader (cont)

- The interface supports generic serialization formats instead of just Writable.
- Note that the getCurrentKey and getCurrentValue may or may not return the same object.
- The getPos method has gone away since not all RecordReaders have byte positions.

OutputFormat

- Replace
  - RecordWriter<K,V> getRecordWriter(FileSystem ignored, JobConf conf, String name, Progressable progress)
  - RecordWriter<K,V> getRecordWriter(TaskAttemptContext ctx)

OutputContext (cont)

- Replace:
  - void checkOutputSpecs(FileSystem ignore,
    JobConf conf)
  - void checkOutputSpecs(JobContext ctx)
- The OutputCommitter is returned by the OutputFormat instead of configured separately!

Cluster Information (in 21)

- Replace JobClient with Cluster.
- Replace ClusterStatus with ClusterMetrics
- Replace RunningJob with JobStatus

Mining the web with Hadoop, Cascading & Bixo

September 23rd, 2009

Thank you.

See you at the next
Bay Area User Group
Oct 21st, 2009