Agenda

- Hadoop “nuts and bolts”
- "Hello World" Hadoop example (distributed word count)
- Running Hadoop in “standalone” mode
- Running Hadoop on EC2
- Open-source Hadoop ecosystem
- Exercises and “office hours”
Hadoop “nuts and bolts”
Hadoop Zen

- Don’t get frustrated (take a deep breath)…
  - Remember this when you experience those W$*$#T@$F! moments
- This is bleeding edge technology:
  - Lots of bugs
  - Stability issues
  - Even lost data
  - To upgrade or not to upgrade (damned either way)?
  - Poor documentation (or none)
- But… Hadoop is the path to data nirvana?

Cloud

- Library used for teaching cloud computing courses at Maryland
- Demos, sample code, etc.
  - Computing conditional probabilities
  - Pairs vs. stripes
  - Complex data types
  - Boilerplate code for working various IR collections
- Dog food for research
- Open source, anonymous svn access
From Theory to Practice

1. Scp data to cluster
2. Move data into HDFS
3. Develop code locally
4. Submit MapReduce job
4a. Go back to Step 3
5. Move data out of HDFS
6. Scp data from cluster
Data Types in Hadoop

**Writable**
- Defines a de/serialization protocol.
- Every data type in Hadoop is a Writable.

**WritableComparable**
- Defines a sort order. All keys must be of this type (but not values).

**IntWritable**
- Concrete classes for different data types.

**LongWritable**
- Text

Complex Data Types in Hadoop

- How do you implement complex data types?
  - The easiest way:
    - Encoded it as Text, e.g., \((a, b) = \text{"a:b"}\)
    - Use regular expressions to parse and extract data
    - Works, but pretty hack-ish
  - The hard way:
    - Define a custom implementation of WritableComparable
    - Must implement: `readFields`, `write`, `compareTo`
    - Computationally efficient, but slow for rapid prototyping
  - Alternatives:
    - Cloud9 offers two other choices: Tuple and JSON
    - Plus, a number of frequently-used data types
What version should I use?
“Hello World” Hadoop example

Hadoop in “standalone” mode
Hadoop in EC2

From Theory to Practice

1. Scp data to cluster
2. Move data into HDFS
3. Develop code locally
4. Submit MapReduce job
   4a. Go back to Step 3
5. Move data out of HDFS
6. Scp data from cluster

You -> Hadoop Cluster
On Amazon: With EC2

1. Scp data to cluster
2. Move data into HDFS
3. Develop code locally
4. Submit MapReduce job
   4a. Go back to Step 3
5. Move data out of HDFS
6. Scp data from cluster
7. Clean up!

Uh oh. Where did the data go?

On Amazon: EC2 and S3

Copy from S3 to HDFS
Copy from HFDS to S3
Open-source Hadoop ecosystem

Hadoop/HDFS
Hadoop streaming

HDFS/FUSE
Pig

HBase
Hypertable

Hive
Mahout

Cassandra
CELL

Beware of toys!
Exercises

Questions?
Comments?

Thanks to the organizations who support our work: