

Only Aggressive Elephants are fast Elephants

Jens Dittrich, Jorge-Arnulfo Quiané-Ruiz, Stefan Richter,
Stefan Schuh, Alekh Jindal, Jörg Schad

Saarland University

<http://www.iStockphoto.com/istock-roboter-18423895-ref-courtesy-roboter-66237df>

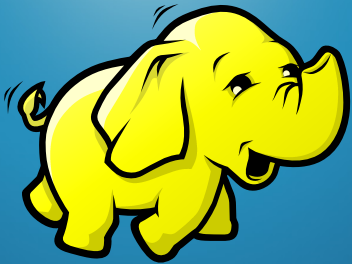
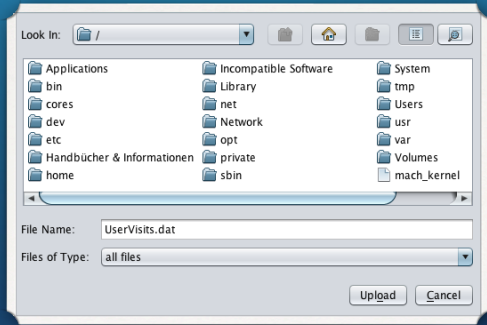
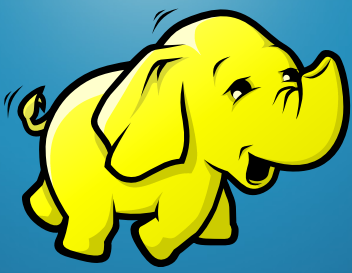
One Day in Bob's Life

```
bob$ head UserVisits.dat
170.122.19.7|0eqt.html|1994-4-27|336.43|NetAnts/1.2|MLI|MLI-HE|Hertzprung-Russell|7
161.110.31.8|0gzhwrmzfd.html|2003-7-16|176.15|MojeekBot|GUY|GUY-ZQ|astrochemistry|7
166.103.36.25|0fwzbsuzn.html|1991-2-14|182.915|KAIST_AITrc|CAF|CAF-AZ|molecules|1
175.101.2.40|0nfxnbnuu.html|1988-3-24|130.32|Mozilla/5.0|MKD|MKD-RQ|A&A|7
156.100.40.7|0iczechptml|1983-12-5|39.906|Mozilla/5.0|MOZ|MOZ-RF|cosmology:|5
174.118.37.24|0setzifibmk.html|1976-10-11|71.28|Robozilla/1.|MLT|MLT-HT|GALAXIES|5
173.125.36.22|0hhzmry.html|1974-8-27|15.58|Incutio_HttpClient|MTQ|MTQ-RE|microwave|2
161.110.5.14.15|0nfxnbnuu.html|2002-4-18|7.930|Mozilla/4.03|MYT|MYT-PK|techniques|1
161.110.5.14.15|0nfxnbnuu.html|2004-5-1|223.571|Pagestacker|ASM|ASM-WN|function|4
161.110.5.14.15|0nfxnbnuu.html|1975-1-20|273.854|Privoxy/3.0|BHR|BHR-MD|individual|6
158.118.25.1|0sawtqjwaavk.html|1977-5-23|160.077|Toutatis.x.x-|JOR|JOR-TW|nebulae|5
169.113.12.45|0vxwzb.html|2001-1-20|351.129|Wavefire/0.8-dev|TLS|TLS-PW|spots|2
155.106.15.8|0vmlwypbhhluer.html|1975-1-19|239.101|OutfoxBot/0.x|MDG|MDG-MG|radial|5
150.101.22.8|0aoclkhkqzcas.html|1973-4-30|220.202|MSIE 4.0|EGY|EGY-KB|occultations|8
175.109.8.39|0jsup.html|1983-9-8|352.6177|RSSOwl/1.2.3|ATA|ATA-BV|interferometers|7
163.111.9.17|0vefhggqwzbgh.html|1976-6-18|268.3|Mozilla/4.05|SOM|SOM-IV|surveys|8
160.122.23.10|0dnbzhr.html|1990-5-24|237.3636|NextGenSearchBot|VUT|VUT-ZL|infrared|6
160.106.6.40|0rwyjocdra.html|1981-10-29|94.965|libwww-perl/5.6|VIR|VIR-FI|Scuti|5
155.110.37.21|0az567.html|1979-1-27|375.101|Jakarta-HttpClient|LBN|LBN-XS|MEDIUM|6
```

SOURCE IP

URL

DURATION



Uploading UserVisits.dat (2.6 TB) to HDFS...

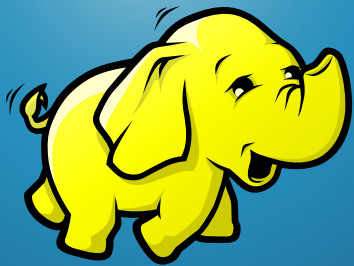


```
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

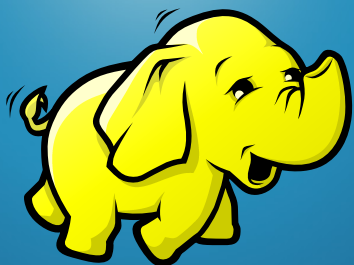
/**
 * User: Bob
 * Date: 29.08.12
 * Time: 10:45
 */
public class BobsMapper extends Mapper<IntWritable, Text, Text, Text> {
    @Override
    public void map(IntWritable key, Text value, Context context)
        throws IOException, InterruptedException {
        // implement me!
    }
}
```

30 minutes later...



Uploading UserVisits.dat (2.6 TB) to HDFS...

Stop



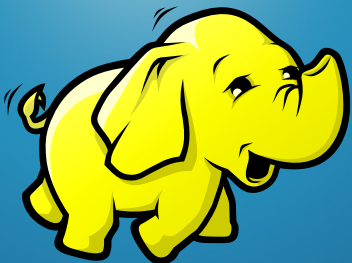
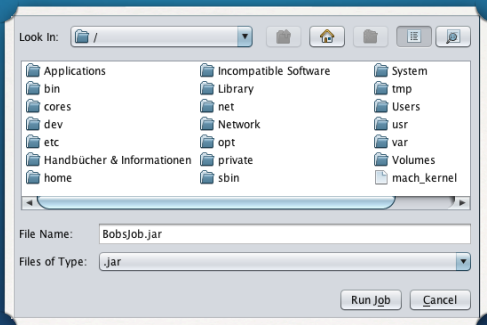
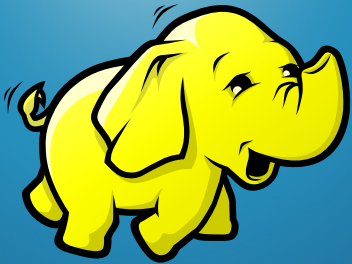
Uploading 2.6 TB of UserVisits.dat

completed

```
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

/**
 * User: Bob
 * Date: 29.08.12
 * Time: 10:45
 */
public class BobsMapper extends Mapper<IntWritable, Text, Text, Text> {
    @Override
    public void map(IntWritable key, Text value, Context context)
        throws IOException, InterruptedException {
        String[] lineFields = value.toString().split("\\|");
        if ("168.114.15.1".equals(lineFields[0])) {
            context.write(key, value);
        }
    }
}
```



Map Progress



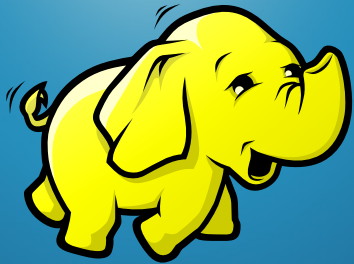
Reduce Progress

Stop


```
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Mapper;

import java.io.IOException;

/**
 * User: Bob
 * Date: 29.08.12
 * Time: 10:45
 */
public class BobsMapper extends Mapper<IntWritable, Text, Text, Text> {
    @Override
    public void map(IntWritable key, Text value, Context context)
        throws IOException, InterruptedException {
        String[] lineFields = value.toString().split("\\|");
        if ("168.114.15.1".equals(lineFields[0])) {
            context.write(key, value);
        }
    }
}
```

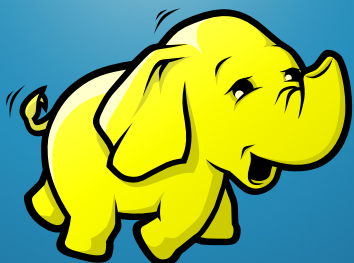


Map Progress



Reduce Progress

Stop

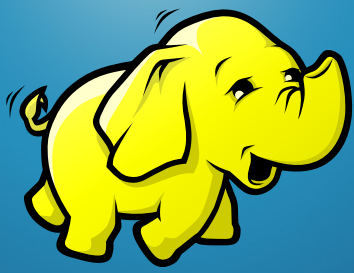


Map Progress



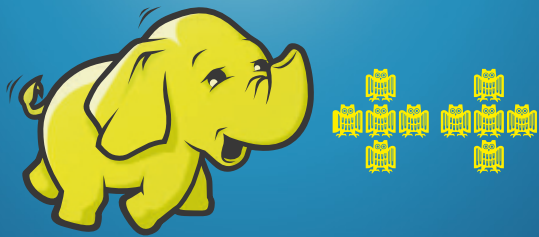
Reduce Progress

Stop



MapReduce Job 4711

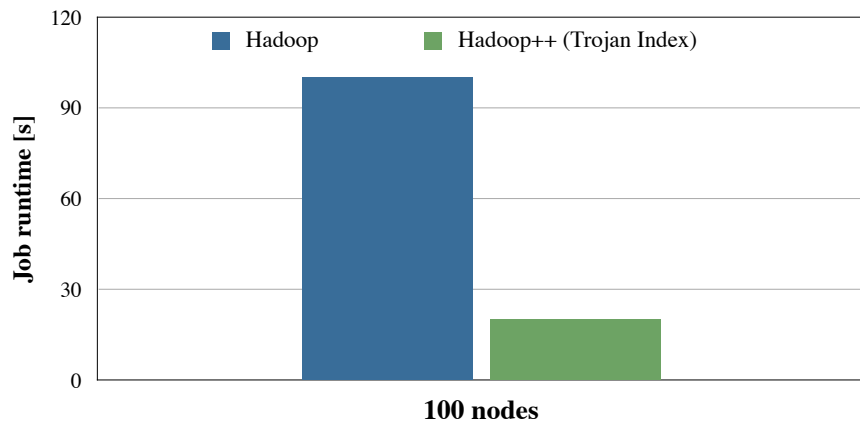
canceled

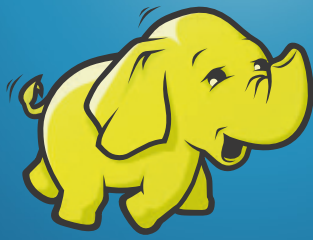


Creating Trojan Index on Attribute 0...

Stop

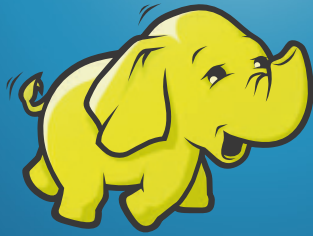
Trojan Index Query Time





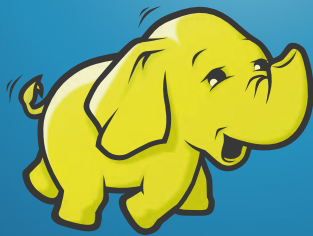
Creating Trojan Index on Attribute 0...

Stop



Creating Trojan Index on Attribute 0...

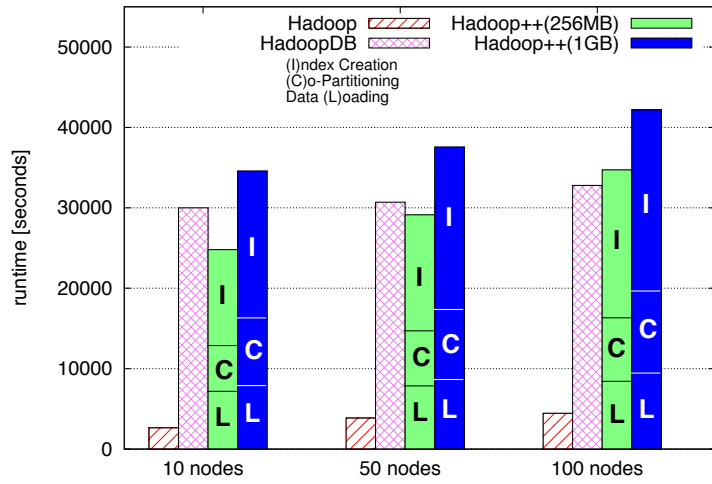
Stop



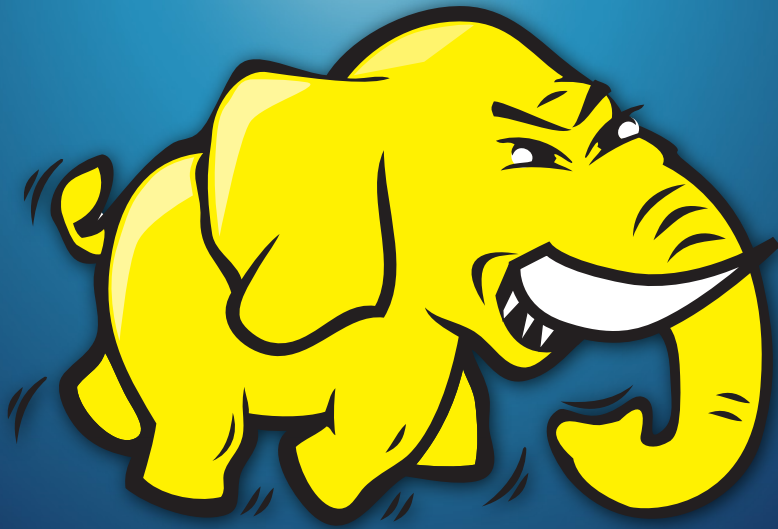
MapReduce Job 1337

anceled

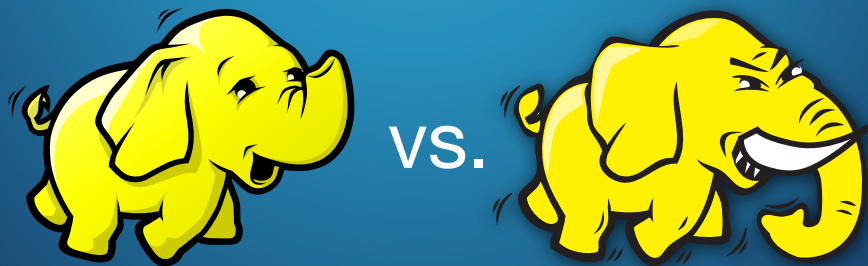
Trojan Index Creation Time

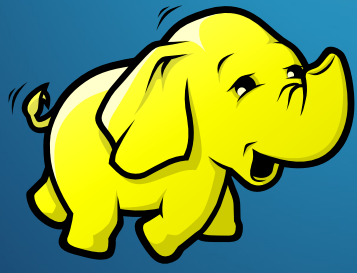


[J. Dittrich, J.-A. Quiane-Ruiz, A. Jindal, Y. Kargin, V. Setty, and J. Schad. Hadoop++: Making a Yellow Elephant Run Like a Cheetah (Without it Even Noticing). PVLDB 2010.]



Hadoop **A**ggressive **I**ndexing **L**ibrary



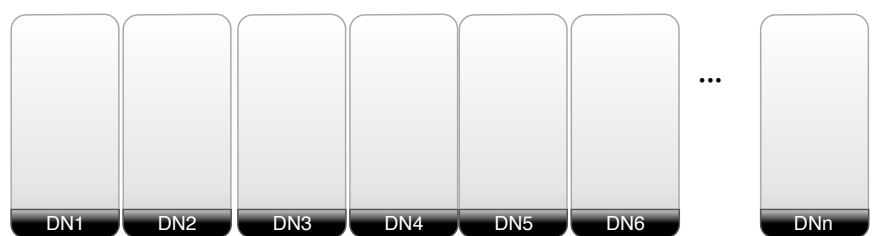


HDFS + MapReduce

HDFS

Bob   <http://www.istockphoto.com/stock-photo-5311184-king-ivor-the-king.php>

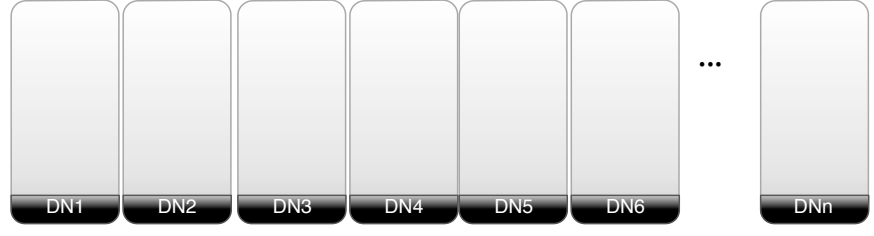
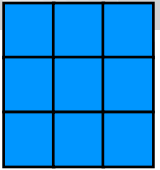
HDFS





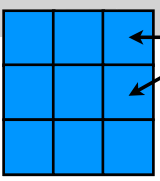
HDFS

horizontal partitions

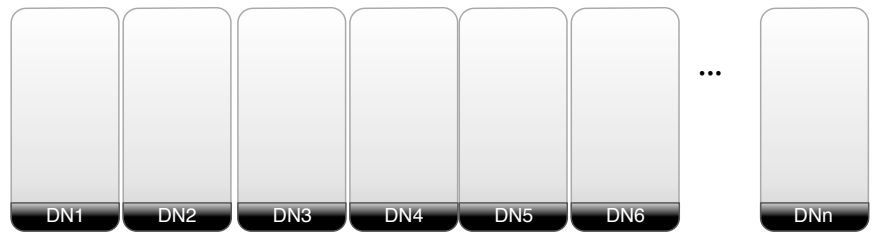


HDFS

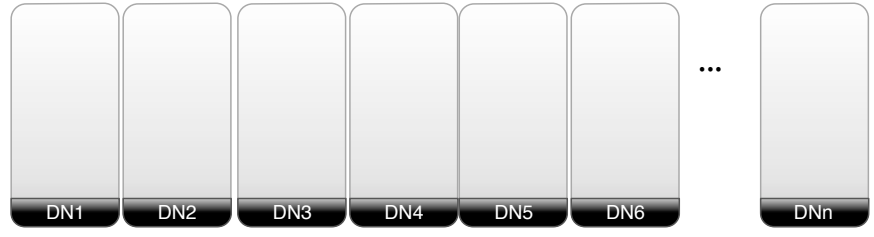
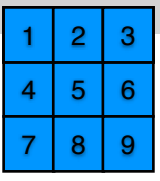
horizontal partitions



HDFS blocks
64MB (default)



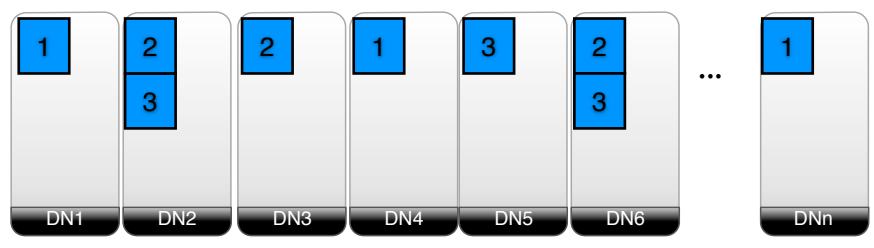
HDFS





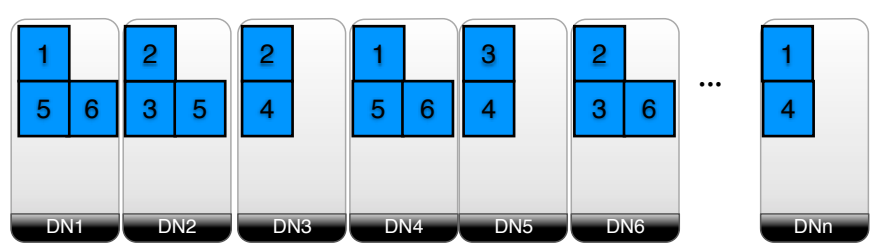
HDFS

4	5	6
7	8	9

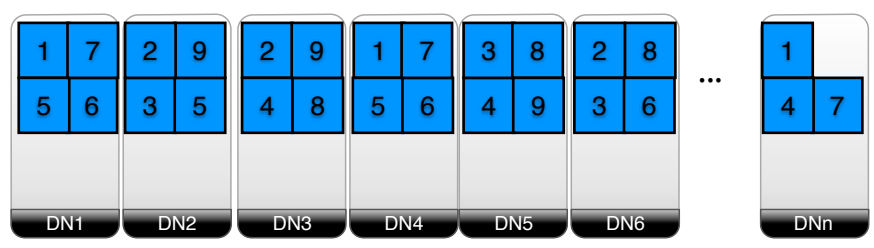


HDFS

7	8	9
---	---	---

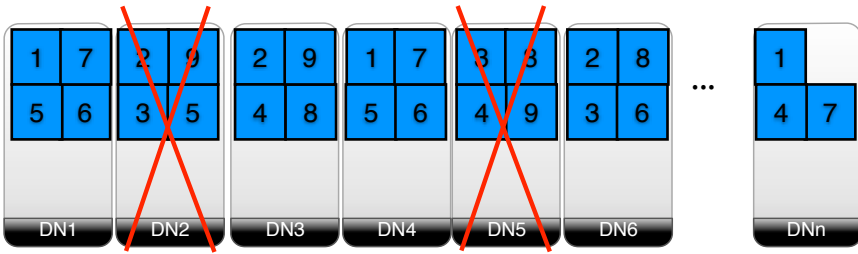


HDFS



Failover

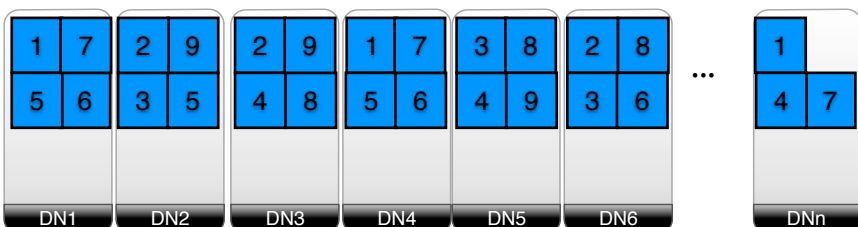
HDFS



MapReduce



HDFS



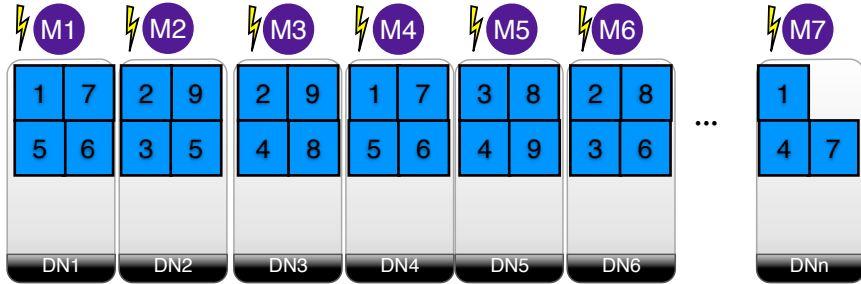
Map Phase



MapReduce

map(row) -> set of (ikey, value)

HDFS



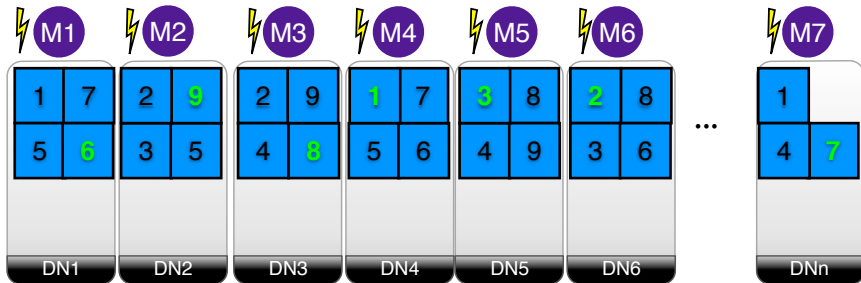
Map Phase



MapReduce

map(row) -> set of (ikey, value)

HDFS



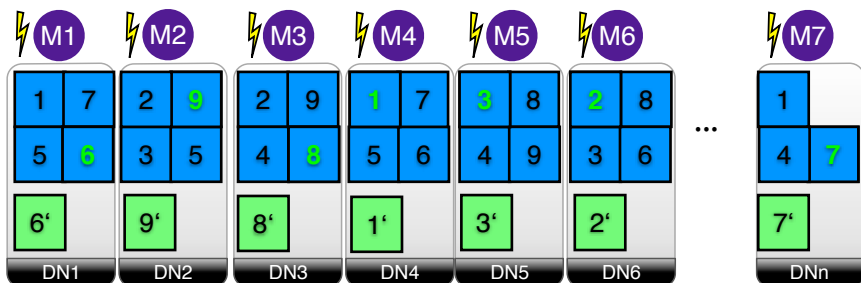
Map Phase



MapReduce

map(row) -> set of (ikey, value)

HDFS

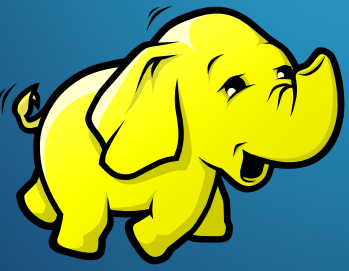
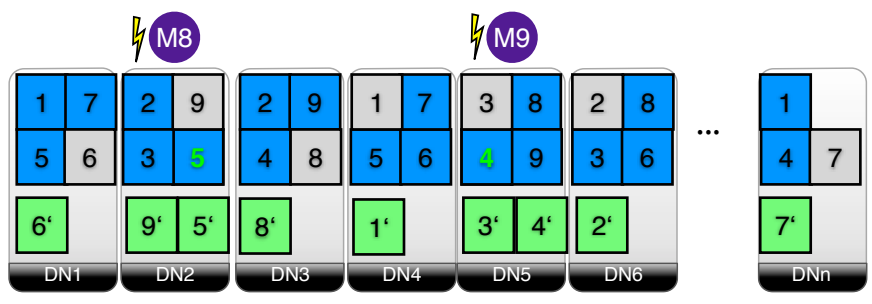


Map Phase

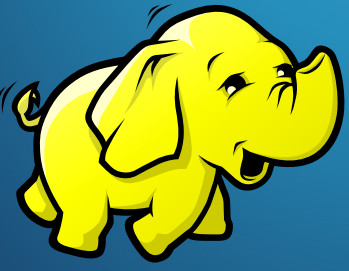


MapReduce `map(docID, document) -> set of (term, docID)`

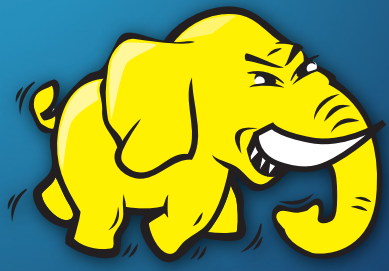
HDFS



HDFS
+ MapReduce

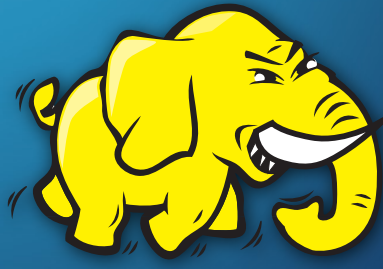


vs.



HDFS
+ MapReduce

~~HDFS~~ HAIL
+ MapReduce



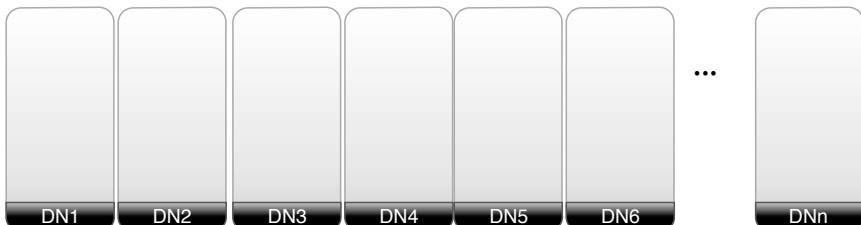
~~HDFS~~ HAIL
+ MapReduce

HAIL

Bob



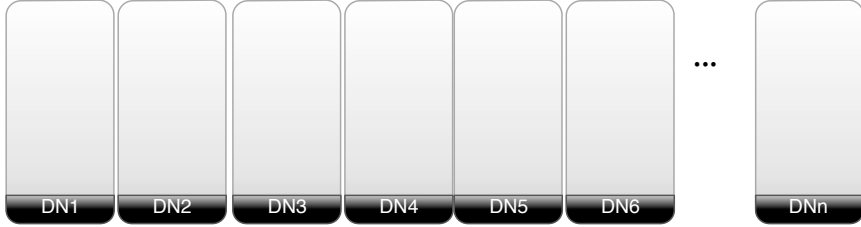
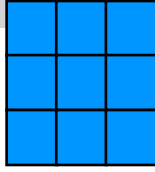
HAIL





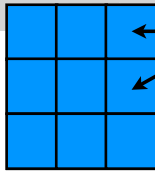
HAIL

horizontal partitions

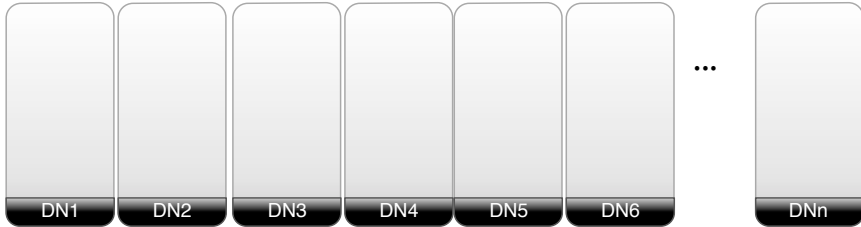


HAIL

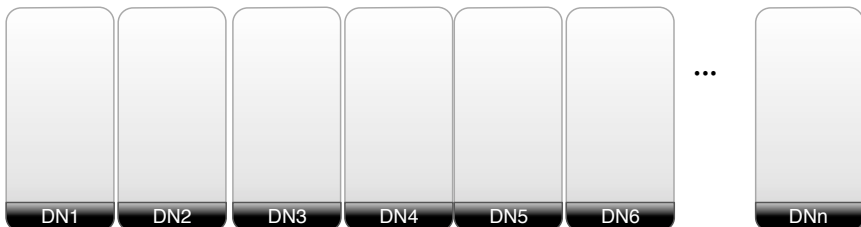
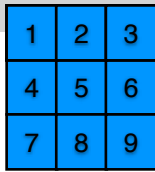
horizontal partitions



HDFS blocks
64MB (default)



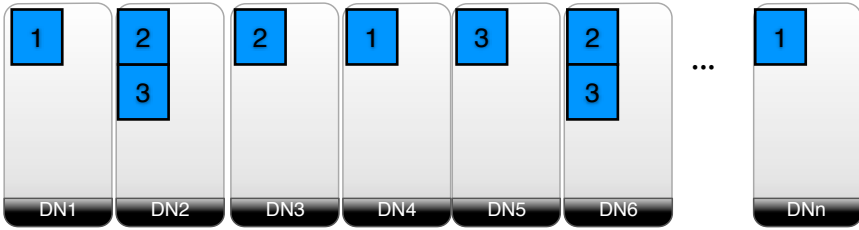
HAIL





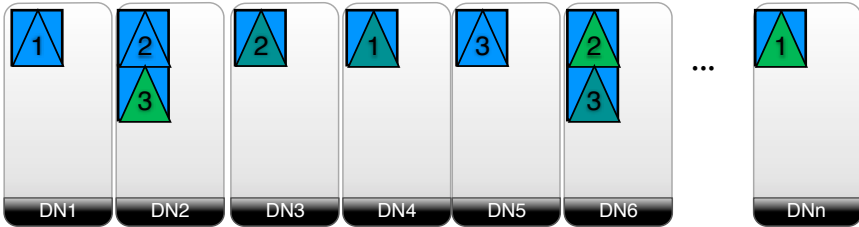
HAIL

4	5	6
7	8	9



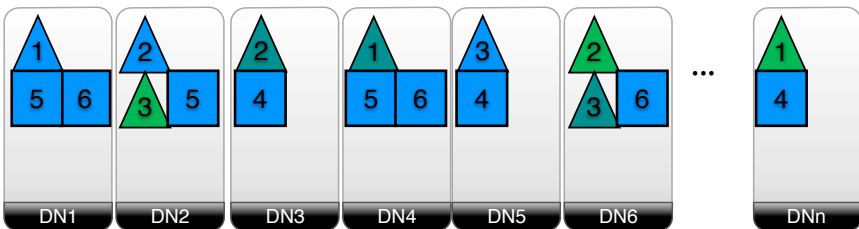
HAIL

4	5	6
7	8	9



HAIL

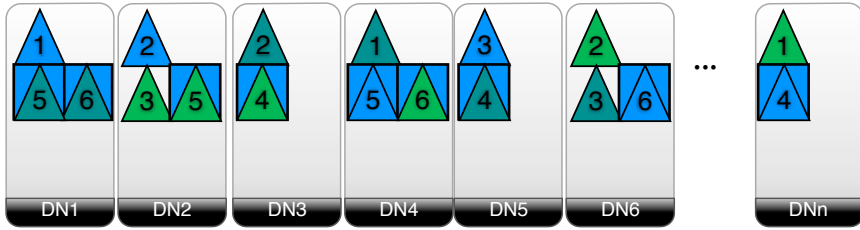
7	8	9
---	---	---





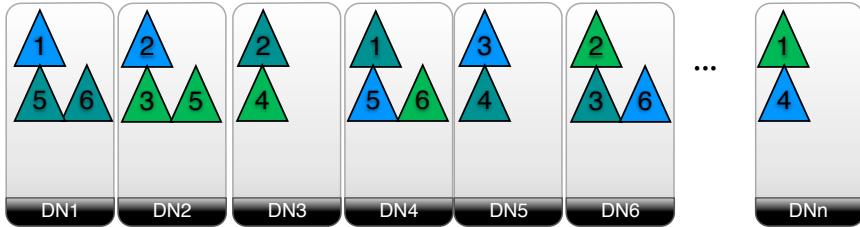
HAIL

7 8 9

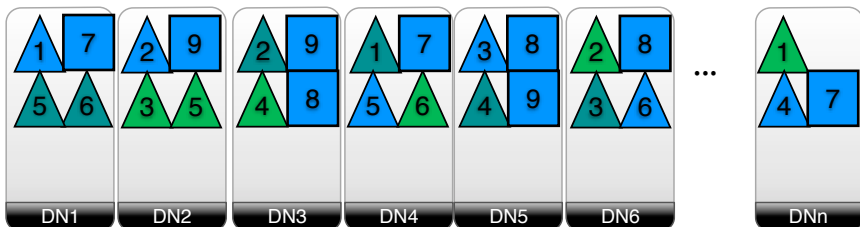


HAIL

7 8 9

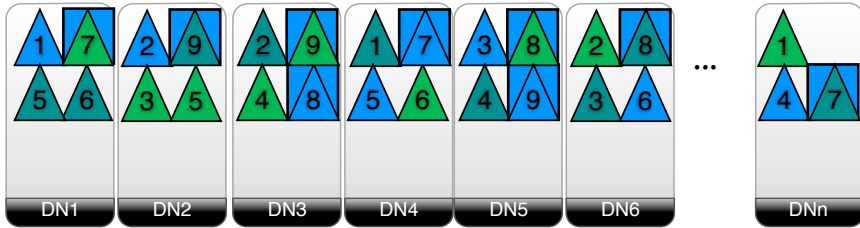


HAIL



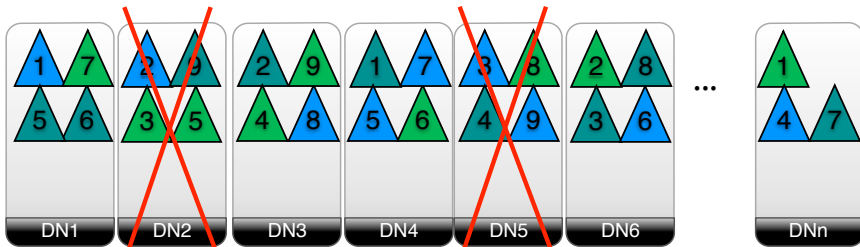


HAIL



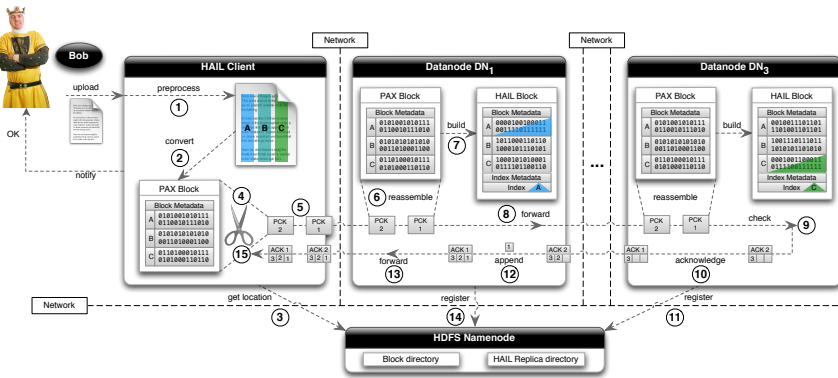
Failover

HAIL

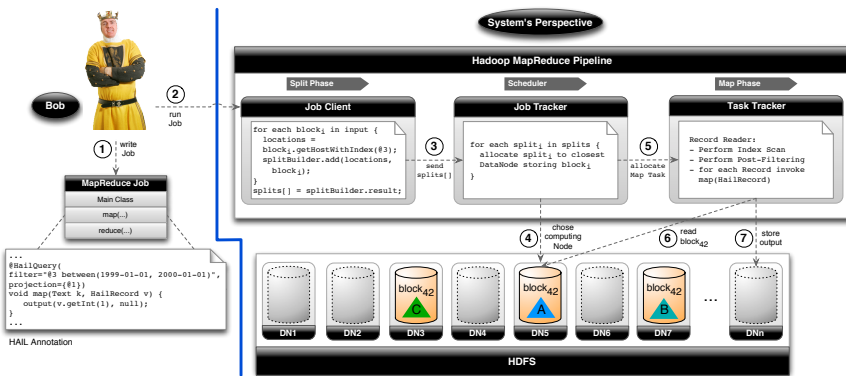


Details

HAIL Upload Pipeline



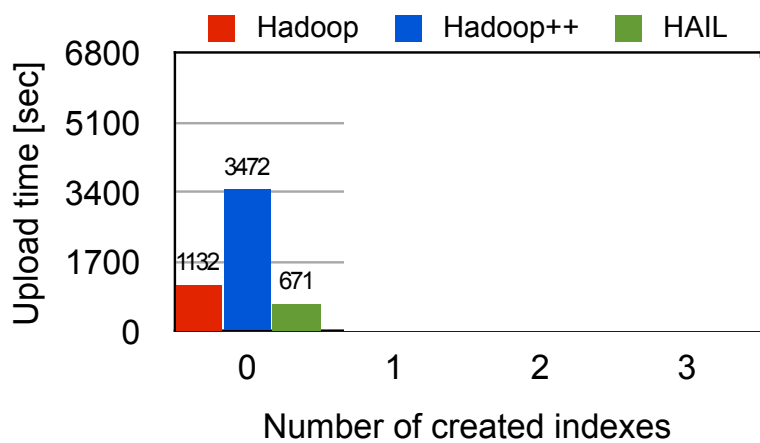
HAIL Query Pipeline



Experiments

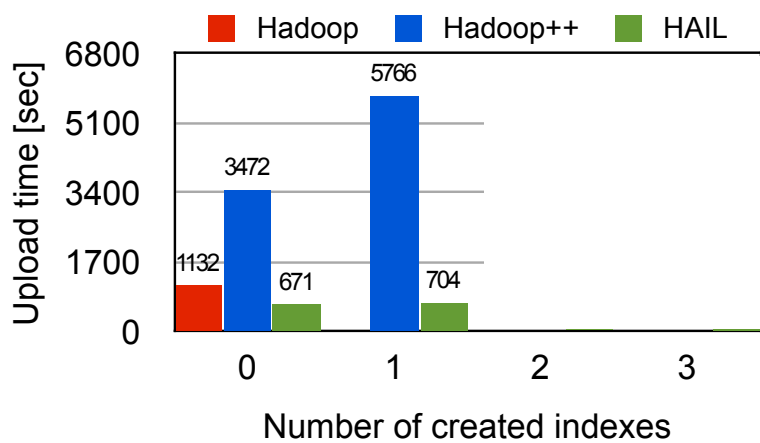
Upload Times

Upload Time



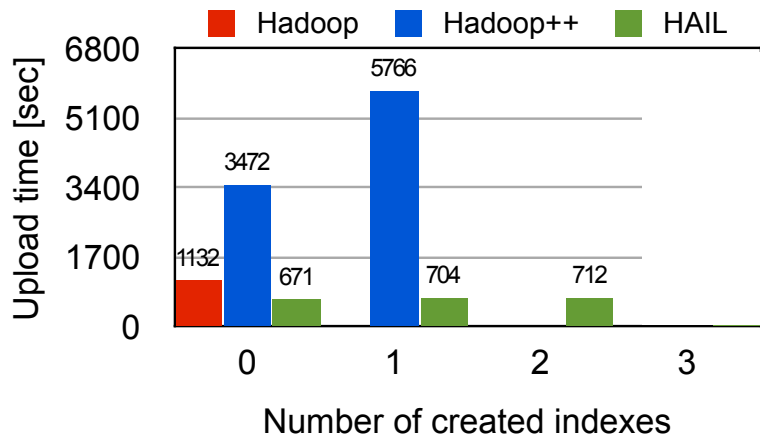
all with 3 replicas

Upload Time



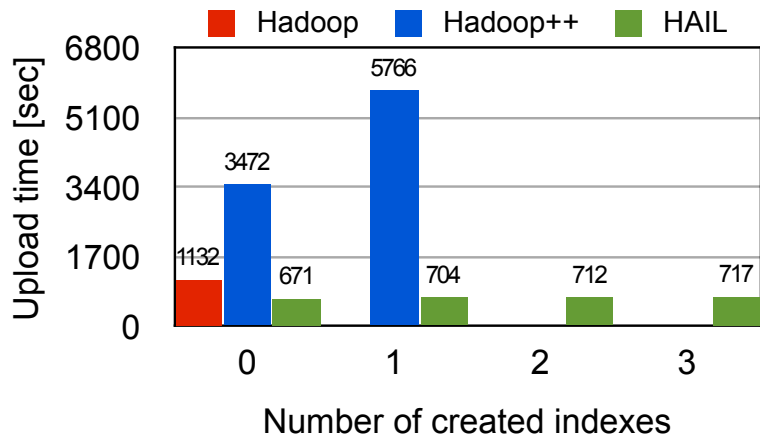
all with 3 replicas

Upload Time



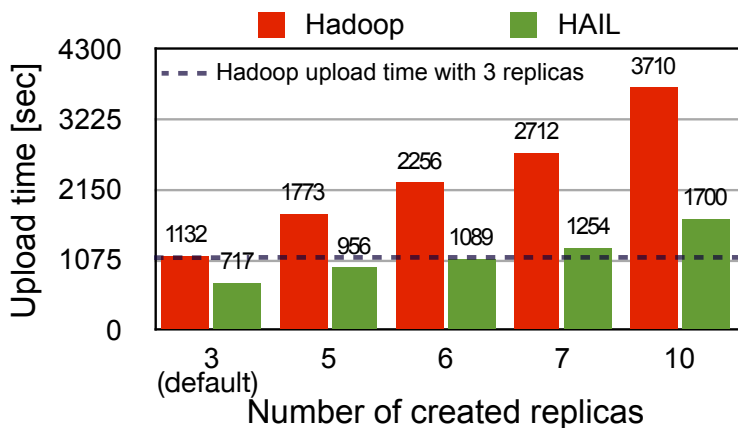
all with 3 replicas

Upload Time

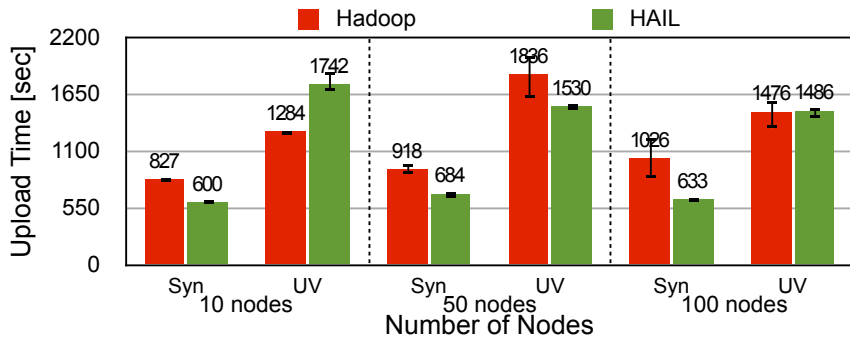


all with 3 replicas

Replica Scalability

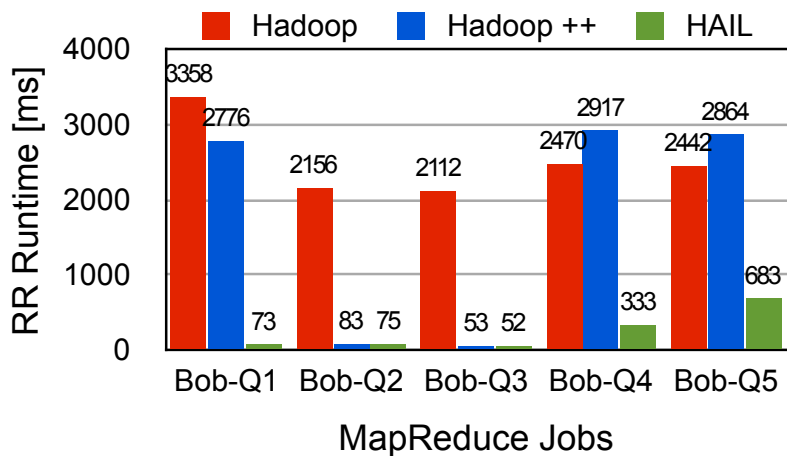


Scale-Out

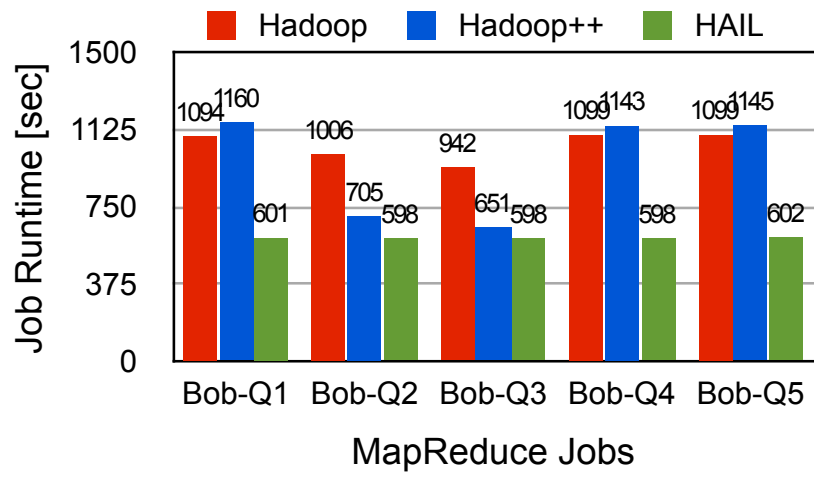


Query Times

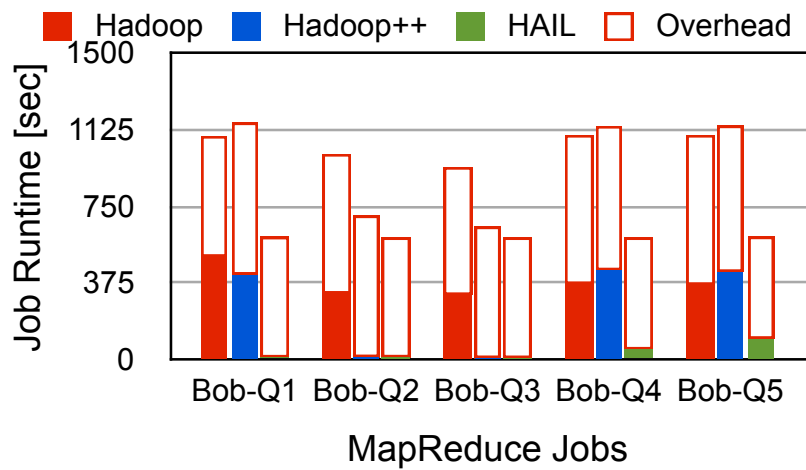
Individual Jobs: Weblog, RecordReader



Individual Jobs: Weblog, Job



Scheduling Overhead



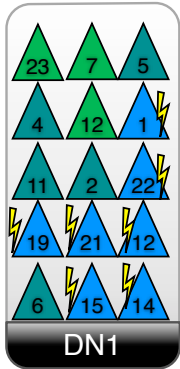
HAIL Scheduling

Hadoop Scheduling

MapReduce

map(row) -> set of (ikey, value)

HAIL



■ sort order

⇒ 7 map tasks (aka waves)

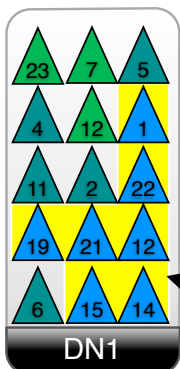
⇒ 7 times scheduling overhead

HAIL Scheduling

MapReduce

map(row) -> set of (ikey, value)

HAIL



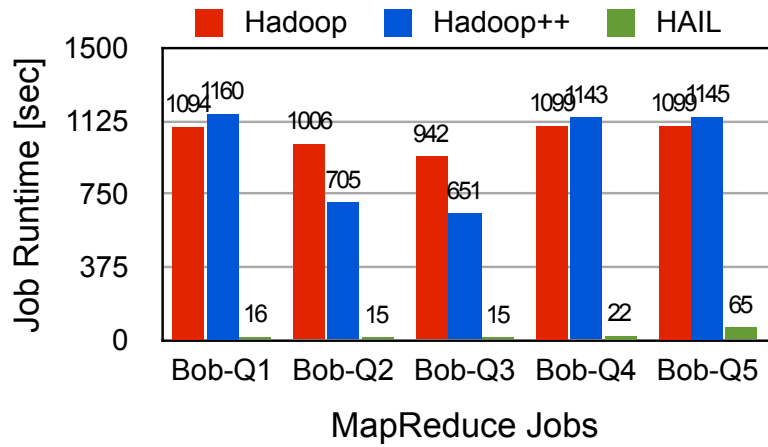
■ sort order

⇒ 1 map task (aka wave)

⇒ 1 times scheduling overhead

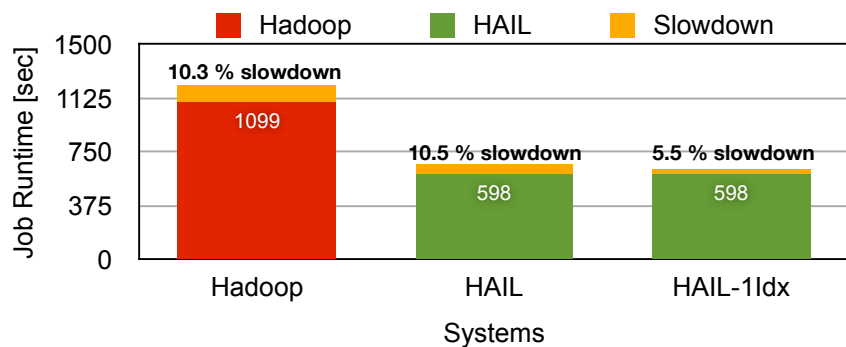
Query Times
with HAIL Scheduling

Individual Jobs: Weblog



Failover

Failover



10 nodes, one killed

Summary(Summary(...))

