



NumerInfo

Big Data Analysis With RHadoop

David Chiu

@COS

2014/05/25

About Me



- Co-Founder of NumerInfo
- Ex-Trend Micro Engineer
- ywchiu.com

R + Hadoop



NumerInfo



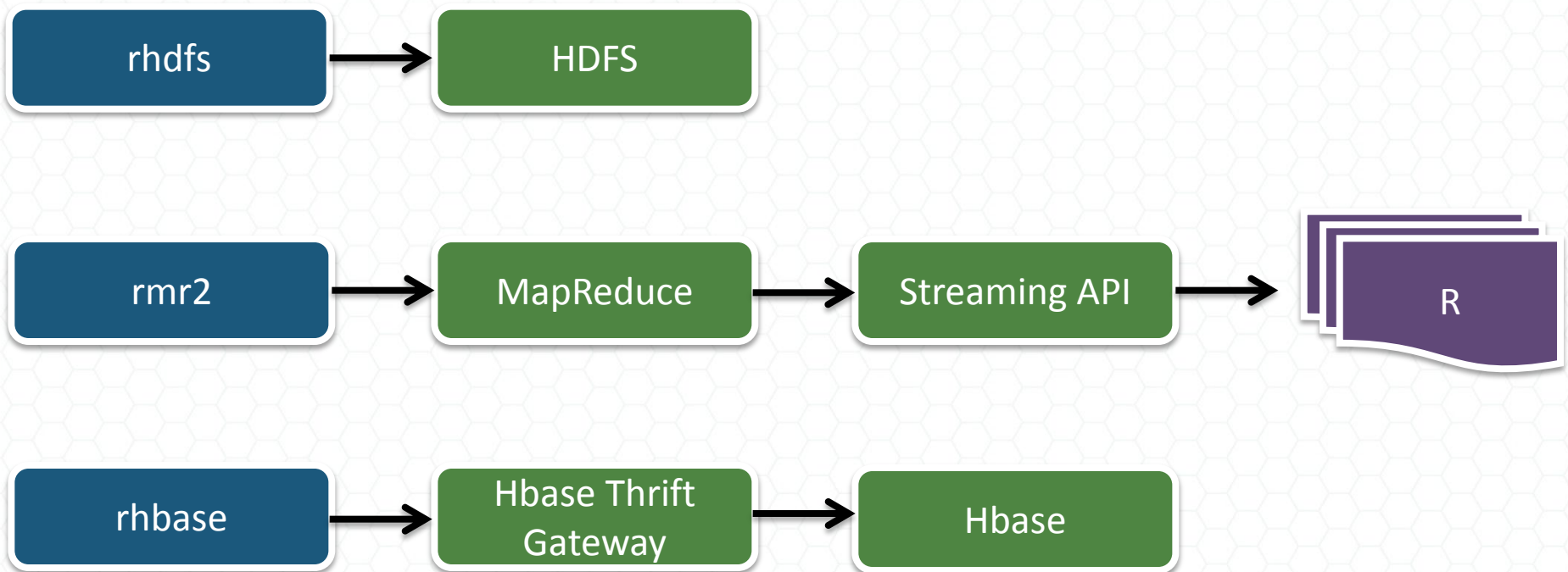


- Scaling R
 - Hadoop enables R to do parallel computing
- Do not have to learn new language
 - Learning to use Java takes time

一切都是為了
生產力



Rhadoop Architecture



- Enable developer to write Mapper/Reducer in any scripting language(R, python, perl)
- Mapper, reducer, and optional combiner processes are written to read from standard input and to write to standard output
- Streaming Job would **have additional overhead** of starting a scripting VM

■ Writing MapReduce Using R

■ mapreduce function

- Mapreduce(input output, map, reduce...)

■ Changelog

- rmr 3.0.0 (2014/02/10): 10X faster than rmr 2.3.0

- rmr 2.3.0 (2013/10/07): support **plyrmr**

- Access HDFS From R
- Exchange data from R dataframe and HDFS

- Exchange data from R to Hbase
- Using Thrift API

- Perform common data manipulation operations, as found in **plyr** and **reshape2**
- It provides a familiar plyr-like interface while hiding many of the mapreduce details
- **plyr: Tools for splitting, applying and combining data**

RHadoop Installation



NumerInfo

- R and related packages should be installed on each tasknode of the cluster
- A Hadoop cluster, CDH3 and higher or Apache 1.0.2 and higher but limited to mr1, not mr2. Compatibility with mr2 from Apache 2.2.0 or HDP2

Getting Ready (Cloudera VM)



NumerInfo

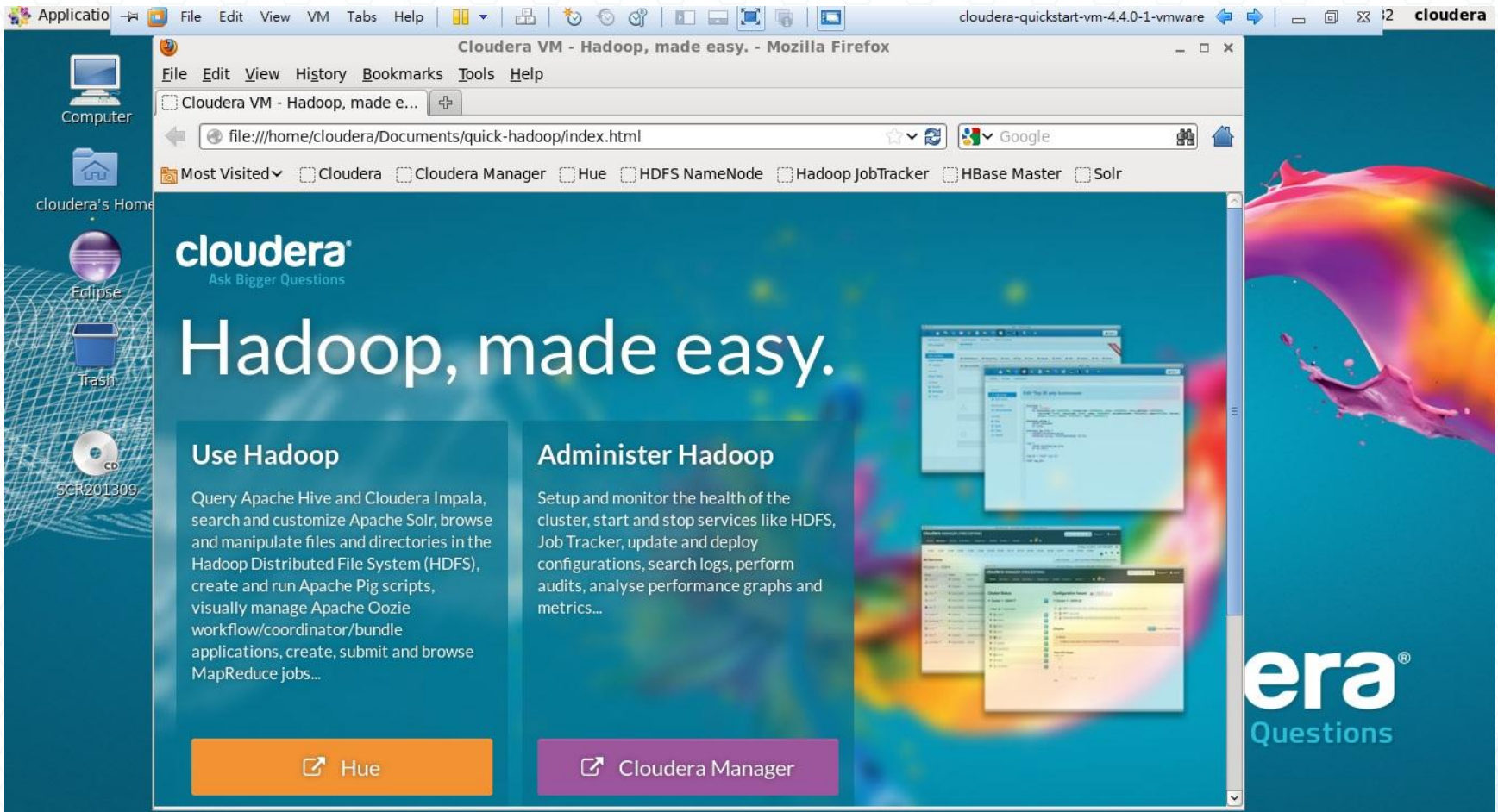
■ Download

http://www.cloudera.com/content/cloudera-content/cloudera-docs/DemoVMs/Cloudera-QuickStart-VM/cloudera_quickstart_vm.html

■ This VM runs

- ❑ CentOS 6.2
- ❑ CDH4.4
- ❑ R 3.0.1
- ❑ Java 1.6.0_32

CDH 4.4



The screenshot displays a virtual machine desktop for Cloudera VM - Hadoop, made easy. The desktop background is a vibrant blue and green abstract design. On the left side, there is a sidebar with icons for 'Computer', 'cloudera's Home', 'Eclipse', 'Trash', and a CD icon labeled 'SCR201309'. The main window is a Mozilla Firefox browser titled 'Cloudera VM - Hadoop, made easy. - Mozilla Firefox'. The address bar shows the file path 'file:///home/cloudera/Documents/quick-hadoop/index.html'. The browser's 'Most Visited' section lists several Cloudera-related services: Cloudera, Cloudera Manager, Hue, HDFS NameNode, Hadoop JobTracker, HBase Master, and Solr. The main content area of the browser window features the Cloudera logo with the tagline 'Ask Bigger Questions' and the heading 'Hadoop, made easy.'. Below this, there are two columns of text. The left column, titled 'Use Hadoop', describes various tasks like querying Apache Hive, using Cloudera Impala, and managing Apache Oozie workflows. The right column, titled 'Administer Hadoop', describes tasks like monitoring cluster health, starting/stopping services, and performing audits. At the bottom of each column are buttons for 'Hue' and 'Cloudera Manager' respectively. On the right side of the desktop, there is a vertical banner with the text 'era® Questions' and a colorful liquid splash graphic.

cloudera
Ask Bigger Questions

Hadoop, made easy.

Use Hadoop

Query Apache Hive and Cloudera Impala, search and customize Apache Solr, browse and manipulate files and directories in the Hadoop Distributed File System (HDFS), create and run Apache Pig scripts, visually manage Apache Oozie workflow/coordinator/bundle applications, create, submit and browse MapReduce jobs...

Administer Hadoop

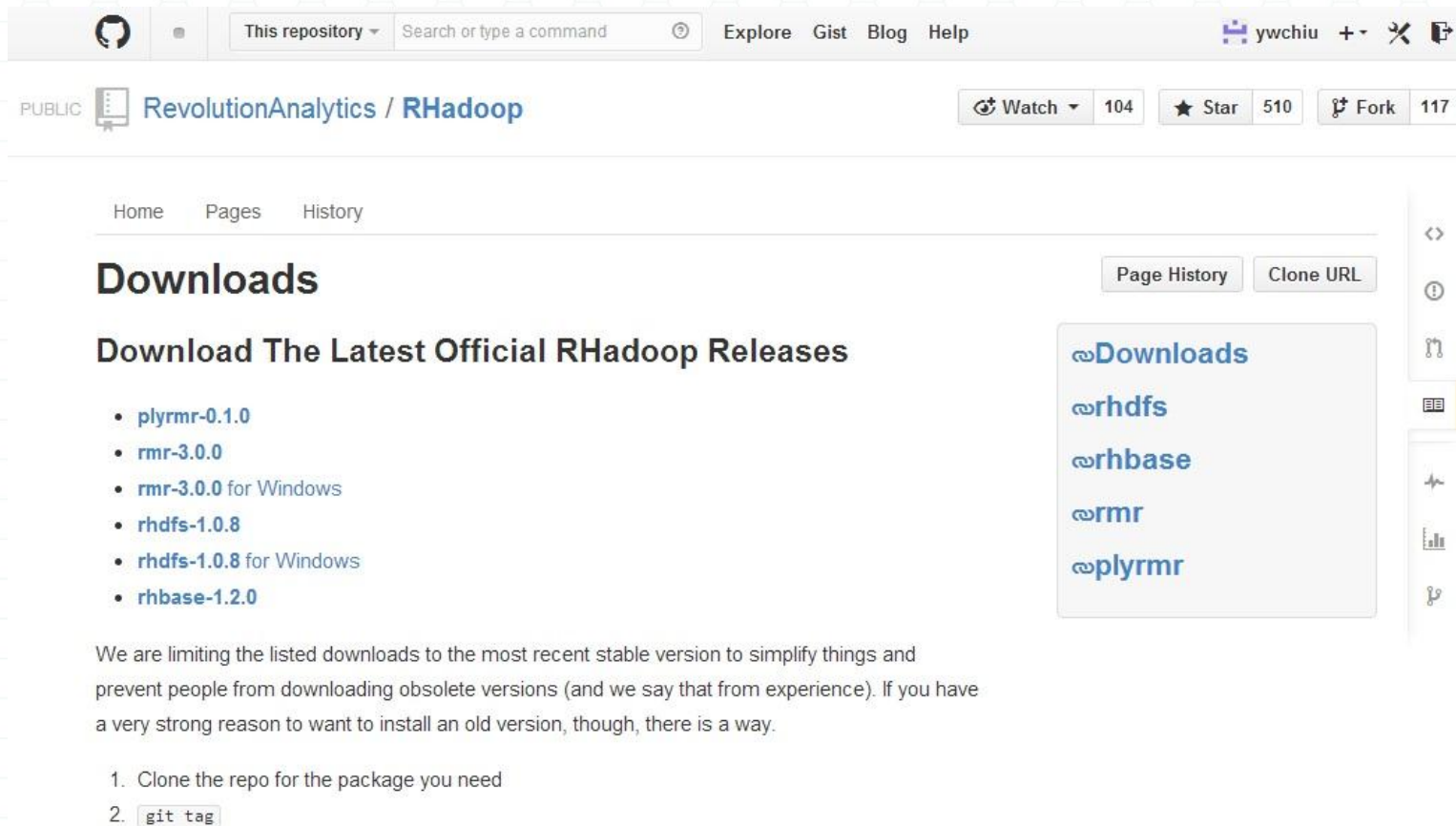
Setup and monitor the health of the cluster, start and stop services like HDFS, Job Tracker, update and deploy configurations, search logs, perform audits, analyse performance graphs and metrics...

[Hue](#) [Cloudera Manager](#)

era®
Questions

Get RHadoop

■ <https://github.com/RevolutionAnalytics/RHadoop/wiki/Downloads>



The screenshot shows the GitHub interface for the `RevolutionAnalytics/RHadoop` repository. The repository is public and has 104 watches, 510 stars, and 117 forks. The page title is "Downloads" and the subtitle is "Download The Latest Official RHadoop Releases". A list of releases is provided, including `plyrmr-0.1.0`, `rmr-3.0.0`, `rmr-3.0.0 for Windows`, `rhdfs-1.0.8`, `rhdfs-1.0.8 for Windows`, and `rhbase-1.2.0`. A sidebar on the right contains a "Downloads" section with links to `rhdfs`, `rhbase`, `rmr`, and `plyrmr`. The main content area includes a paragraph explaining the limitation to the most recent stable version and a list of instructions for cloning the repository and using `git tag`.

Home Pages History

Downloads

Download The Latest Official RHadoop Releases

- [plyrmr-0.1.0](#)
- [rmr-3.0.0](#)
- [rmr-3.0.0 for Windows](#)
- [rhdfs-1.0.8](#)
- [rhdfs-1.0.8 for Windows](#)
- [rhbase-1.2.0](#)

We are limiting the listed downloads to the most recent stable version to simplify things and prevent people from downloading obsolete versions (and we say that from experience). If you have a very strong reason to want to install an old version, though, there is a way.

1. Clone the repo for the package you need
2. `git tag`

Page History Clone URL

- Downloads
- rhdfs
- rhbase
- rmr
- plyrmr

Installing rmr2 dependencies

- Make sure the package is installed system wise

```
$ sudo R
```

```
> install.packages(c("codetools", "R", "Rcpp",  
"RJSONIO", "bitops", "digest", "functional", "stringr",  
"plyr", "reshape2", "rJava", "caTools"))
```


Install rmr2



NumerInfo

```
$ wget --no-check-certificate
```

```
https://raw.githubusercontent.com/RevolutionAnalytics/rmr2/3.0.0/build/rmr2\_3.0.0.tar.gz
```

```
$ sudo R CMD INSTALL rmr2_3.0.0.tar.gz
```

Installing...



NumerInfo

```
ePolicy>&) [with StoragePolicy = Rcpp::PreserveStorage]
/usr/lib64/R/library/Rcpp/include/Rcpp/RObject.h:49: note: Rcpp::RObject_Impl
<StoragePolicy>& Rcpp::RObject_Impl<StoragePolicy>::operator=(SEXP*) [with StoragePolicy =
Rcpp::PreserveStorage]
typed-bytes.cpp:267: error: ambiguous overload for 'operator=' in 'new_object = unserialize_2
55_terminated_list(const raw&, unsigned int&)(((unsigned int&)((unsigned int*)start)))'
/usr/lib64/R/library/Rcpp/include/Rcpp/RObject.h:35: note: candidates are: Rcpp::RObject_Impl
<StoragePolicy>& Rcpp::RObject_Impl<StoragePolicy>::operator=(const Rcpp::RObject_Impl<Storag
ePolicy>&) [with StoragePolicy = Rcpp::PreserveStorage]
/usr/lib64/R/library/Rcpp/include/Rcpp/RObject.h:49: note: Rcpp::RObject_Impl
<StoragePolicy>& Rcpp::RObject_Impl<StoragePolicy>::operator=(SEXP*) [with StoragePolicy =
Rcpp::PreserveStorage]
typed-bytes.cpp:270: error: ambiguous overload for 'operator=' in 'new_object = unserialize_m
ap(const raw&, unsigned int&)(((unsigned int&)((unsigned int*)start)))'
/usr/lib64/R/library/Rcpp/include/Rcpp/RObject.h:35: note: candidates are: Rcpp::RObject_Impl
<StoragePolicy>& Rcpp::RObject_Impl<StoragePolicy>::operator=(const Rcpp::RObject_Impl<Storag
ePolicy>&) [with StoragePolicy = Rcpp::PreserveStorage]
/usr/lib64/R/library/Rcpp/include/Rcpp/RObject.h:49: note: Rcpp::RObject_Impl
<StoragePolicy>& Rcpp::RObject_Impl<StoragePolicy>::operator=(SEXP*) [with StoragePolicy =
Rcpp::PreserveStorage]
make: *** [typed-bytes.o] Error 1
ERROR: compilation failed for package 'rmr2'
* removing '/usr/lib64/R/library/rmr2'
[cloudera@localhost ~]$
```




Downgrade Rcpp



NumerInfo

■ <http://cran.r-project.org/src/contrib/Archive/Rcpp/>

The screenshot shows a web browser displaying the Rcpp archive page. The address bar shows the URL `cran.r-project.org/src/contrib/Archive/Rcpp/`. The page lists various Rcpp tar.gz files with their release dates and sizes. The files are listed in descending order of release date, from 05-Apr-2011 to 03-Feb-2014. The files are named `Rcpp_0.9.3.tar.gz` through `Rcpp_0.11.0.tar.gz`. The sizes range from 1.9M to 2.3M. The page footer indicates the server is `Apache/2.2.22 (Debian) Server at cran.r-project.org Port 80`.

File Name	Date	Size
Rcpp_0.9.3.tar.gz	05-Apr-2011 21:03	2.0M
Rcpp_0.9.4.tar.gz	12-Apr-2011 18:22	1.9M
Rcpp_0.9.5.tar.gz	06-Jul-2011 20:56	2.0M
Rcpp_0.9.6.tar.gz	27-Jul-2011 15:41	2.0M
Rcpp_0.9.7.tar.gz	30-Sep-2011 07:51	2.0M
Rcpp_0.9.8.tar.gz	22-Dec-2011 09:26	2.0M
Rcpp_0.9.9.tar.gz	27-Dec-2011 11:05	2.0M
Rcpp_0.9.10.tar.gz	17-Feb-2012 08:38	2.0M
Rcpp_0.9.11.tar.gz	22-Jun-2012 17:08	2.2M
Rcpp_0.9.12.tar.gz	25-Jun-2012 08:15	2.0M
Rcpp_0.9.13.tar.gz	29-Jun-2012 08:32	2.0M
Rcpp_0.9.14.tar.gz	01-Oct-2012 08:36	2.0M
Rcpp_0.9.15.tar.gz	14-Oct-2012 11:12	2.0M
Rcpp_0.10.0.tar.gz	14-Nov-2012 08:28	2.2M
Rcpp_0.10.1.tar.gz	27-Nov-2012 07:43	2.3M
Rcpp_0.10.2.tar.gz	21-Dec-2012 16:39	2.3M
Rcpp_0.10.3.tar.gz	23-Mar-2013 17:05	2.3M
Rcpp_0.10.4.tar.gz	24-Jun-2013 16:25	2.3M
Rcpp_0.10.5.tar.gz	29-Sep-2013 11:02	1.9M
Rcpp_0.10.6.tar.gz	29-Oct-2013 15:11	1.9M
Rcpp_0.11.0.tar.gz	03-Feb-2014 07:12	1.9M

Apache/2.2.22 (Debian) Server at cran.r-project.org Port 80

Install Rcpp_0.11.0



NumerInfo


```
$ wget --no-check-certificate http://cran.r-project.org/src/contrib/Archive/Rcpp/Rcpp\_0.11.0.tar.gz
```

```
$sudo R CMD INSTALL Rcpp_0.11.0.tar.gz
```

Install rmr2 again

```
$ sudo R CMD INSTALL rmr2_3.0.0.tar.gz
```

```
cloudera@localhost:~  
File Edit View Search Terminal Help  
** help  
*** installing help indices  
  converting help for package 'rmr2'  
    finding HTML links ... done  
    bigdataobject      html  
    dfs.empty          html  
    equijoin           html  
    fromdfstodfs       html  
    keyval             html  
    make.io.format     html  
    mapreduce          html  
    rmr-package        html  
    rmr.options        html  
    rmr.sample         html  
    rmr.str            html  
    scatter            html  
    status             html  
    tomaptoreduce      html  
    vsum              html  
** building package indices  
** testing if installed package can be loaded  
* DONE (rmr2)  
Making 'packages.html' ... done  
[cloudera@localhost ~]$
```

嚇不到我的

Install RHDFS

```
$ wget -no-check-certificate  
https://raw.githubusercontent.com/RevolutionAnalytics/rhdfs/master/build/rhdfs\_1.0.8.tar.gz
```

```
$ sudo HADOOP_CMD=/usr/bin/hadoop R CMD  
INSTALL rhdfs_1.0.8.tar.gz
```

Enable hdfs



NumerInfo

```
> Sys.setenv(HADOOP_CMD="/usr/bin/hadoop")
> Sys.setenv(HADOOP_STREAMING="/usr/lib/hadoop-0.20-
mapreduce/contrib/streaming/hadoop-streaming-2.0.0-mr1-
cdh4.4.0.jar")
> library(rmr2)
> library(rhdfs)
> hdfs.init()
```

Be sure to run `hdfs.init()`

```
> hdfs.init()
```

```
14/03/16 00:55:13 ERROR security.UserGroupInformation: Unable to find JAAS classes:com.sun.se
curity.auth.UnixPrincipal not found in gnu.gcj.runtime.SystemClassLoader{urls=[file:/usr/lib6
4/R/library/rJava/java/boot/], parent=gnu.gcj.runtime.ExtensionClassLoader{urls=[], parent=nu
ll}}
```

```
14/03/16 00:55:14 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your p
latform... using builtin-java classes where applicable
```

```
Error in .jcall("RJavaTools", "Ljava/lang/Object;", "invokeMethod", cl, :
```

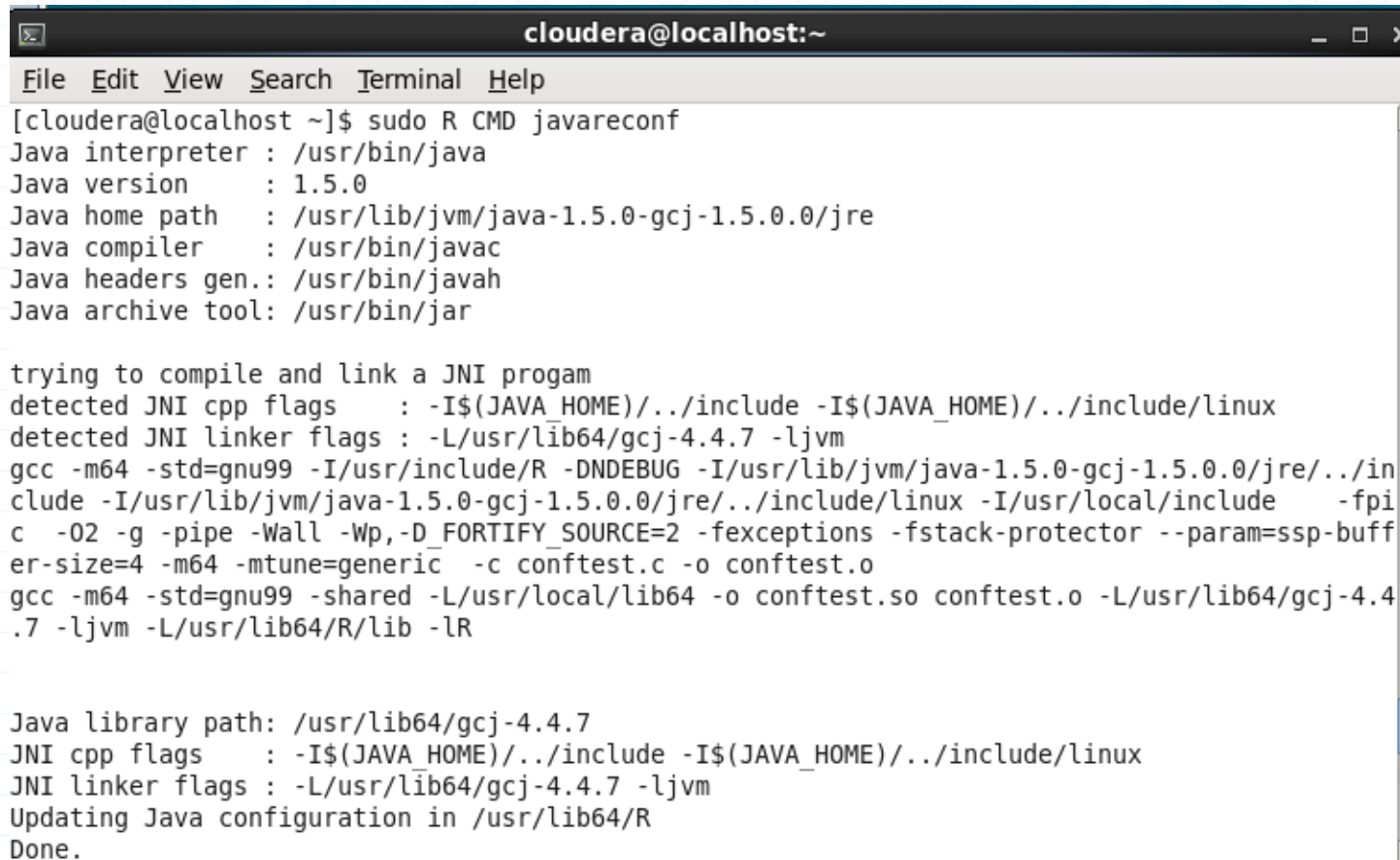
```
java.io.IOException: failure to login
```

```
>
```



Javareconf error

```
$ sudo R CMD javareconf
```



```
cloudera@localhost:~  
File Edit View Search Terminal Help  
[cloudera@localhost ~]$ sudo R CMD javareconf  
Java interpreter : /usr/bin/java  
Java version      : 1.5.0  
Java home path    : /usr/lib/jvm/java-1.5.0-gcj-1.5.0.0/jre  
Java compiler     : /usr/bin/javac  
Java headers gen.: /usr/bin/javah  
Java archive tool: /usr/bin/jar  
  
trying to compile and link a JNI program  
detected JNI cpp flags : -I$(JAVA_HOME)/../include -I$(JAVA_HOME)/../include/linux  
detected JNI linker flags : -L/usr/lib64/gcj-4.4.7 -ljvm  
gcc -m64 -std=gnu99 -I/usr/include/R -DNDEBUG -I/usr/lib/jvm/java-1.5.0-gcj-1.5.0.0/jre/./in  
clude -I/usr/lib/jvm/java-1.5.0-gcj-1.5.0.0/jre/./include/linux -I/usr/local/include -fpi  
c -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector --param=ssp-buff  
er-size=4 -m64 -mtune=generic -c conftest.c -o conftest.o  
gcc -m64 -std=gnu99 -shared -L/usr/local/lib64 -o conftest.so conftest.o -L/usr/lib64/gcj-4.4  
.7 -ljvm -L/usr/lib64/R/lib -lR  
  
Java library path: /usr/lib64/gcj-4.4.7  
JNI cpp flags : -I$(JAVA_HOME)/../include -I$(JAVA_HOME)/../include/linux  
JNI linker flags : -L/usr/lib64/gcj-4.4.7 -ljvm  
Updating Java configuration in /usr/lib64/R  
Done.
```

javareconf with correct JAVA_HOME



```
$ echo $JAVA_HOME
```

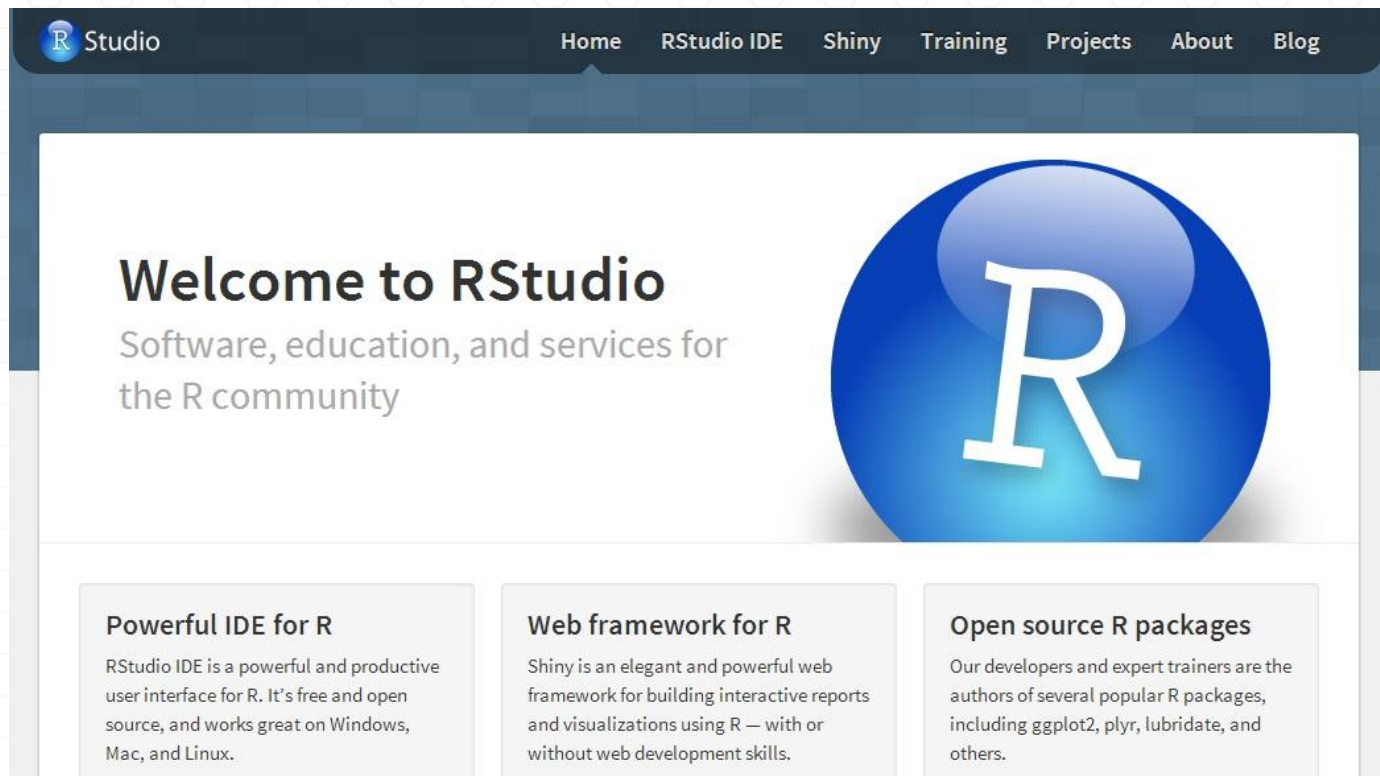
```
$ sudo JAVA_HOME=/usr/java/jdk1.6.0_32 R CMD javareconf
```

```
cloudera@localhost:~  
File Edit View Search Terminal Help  
[cloudera@localhost ~]$ sudo JAVA_HOME=/usr/java/jdk1.6.0_32 R CMD javareconf  
Java interpreter : /usr/java/jdk1.6.0_32/jre/bin/java  
Java version      : 1.6.0_32  
Java home path    : /usr/java/jdk1.6.0_32  
Java compiler     : /usr/java/jdk1.6.0_32/bin/javac  
Java headers gen. : /usr/java/jdk1.6.0_32/bin/javah  
Java archive tool : /usr/java/jdk1.6.0_32/bin/jar  
  
trying to compile and link a JNI program  
detected JNI cpp flags : -I$(JAVA_HOME)/include -I$(JAVA_HOME)/include/linux  
detected JNI linker flags : -L$(JAVA_HOME)/jre/lib/amd64/server -ljvm  
gcc -m64 -std=gnu99 -I/usr/include/R -DDEBUG -I/usr/java/jdk1.6.0_32/include -I/usr/java/jdk1.6.0_32/include/linux -I/usr/local/include -fpic -O2 -g -pipe -Wall -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector --param=ssp-buffer-size=4 -m64 -mtune=generic -c conftest.c -o conftest.o  
gcc -m64 -std=gnu99 -shared -L/usr/local/lib64 -o conftest.so conftest.o -L/usr/java/jdk1.6.0_32/jre/lib/amd64/server -ljvm -L/usr/lib64/R/lib -lR  
  
Java library path: $(JAVA_HOME)/jre/lib/amd64/server  
JNI cpp flags : -I$(JAVA_HOME)/include -I$(JAVA_HOME)/include/linux  
JNI linker flags : -L$(JAVA_HOME)/jre/lib/amd64/server -ljvm  
Updating Java configuration in /usr/lib64/R  
Done.  
  
[cloudera@localhost ~]$
```

Install Rstudio

```
$ wget http://download2.rstudio.org/rstudio-server-0.98.501-x86_64.rpm
```

```
$ sudo yum install --nogpgcheck rstudio-server-0.98.501-x86_64.rpm
```



Login into RStudio

Username: cloudera
Password: cloudera

Rstudio

Sign in to RStudio

Username:
cloudera

Password:

☐ Stay signed in

Sign In

MapReduce With RHadoop

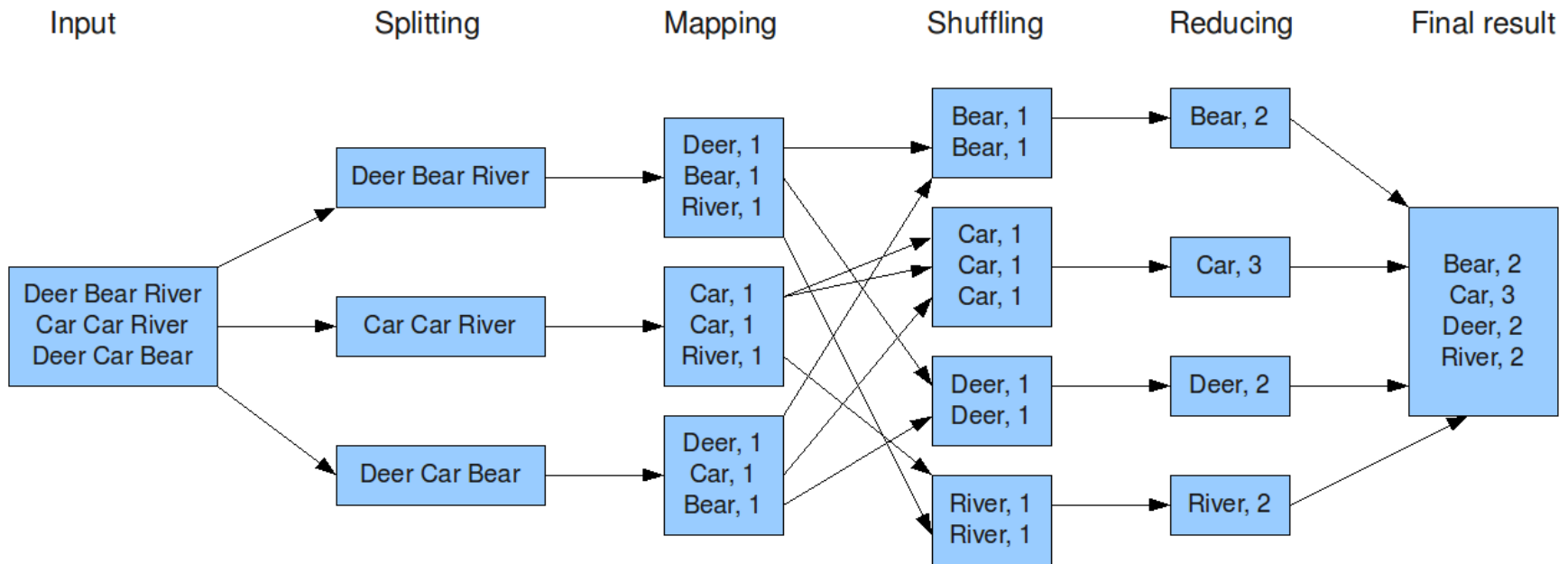


NumerInfo

- `mapreduce(input, output, map, reduce)`
- Like `sapply`, `lapply`, `tapply` within R

Hello World – For Hadoop

The overall MapReduce word count process



<http://www.rabidgremlin.com/data20/MapReduceWordCountOverview1.png>

Move File Into HDFS



NumerInfo

Put data into hdfs

```
Sys.setenv(HADOOP_CMD="/usr/bin/hadoop")
Sys.setenv(HADOOP_STREAMING="/usr/lib/hadoop-0.20-
mapreduce/contrib/streaming/hadoop-streaming-2.0.0-mr1-
cdh4.4.0.jar")
library(rmr2)
library(rhdfs)
hdfs.init()
hdfs.mkdir("/user/cloudera/wordcount/data")
hdfs.put("wc_input.txt", "/user/cloudera/wordcount/data")
```



```
$ hadoop fs -mkdir /user/cloudera/wordcount/data
$ hadoop fs -put wc_input.txt /user/cloudera/word/count/data
```

Wordcount Mapper



NumerInfo

#Mapper

```
map <- function(k,lines) {  
  words.list <- strsplit(lines, '\\s')  
  words <- unlist(words.list)  
  return( keyval(words, 1) )  
}
```



```
public static class Map extends MapReduceBase implements 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, OutputCollector<Text, IntWritable> output, Reporter  
reporter) throws IOException {  
    String line = value.toString();  
    StringTokenizer tokenizer = new StringTokenizer(line);  
    while (tokenizer.hasMoreTokens()) {  
      word.set(tokenizer.nextToken());  
      output.collect(word, one);  
    }  
  }  
}
```


Wordcount Reducer



NumerInfo

#Reducer

```
reduce <- function(word, counts) {  
  keyval(word, sum(counts))  
}
```



```
public static class Reduce extends MapReduceBase implements  
Reducer<Text, IntWritable, Text, IntWritable> {  
    public void reduce(Text key, Iterator<IntWritable> values,  
OutputCollector<Text, IntWritable> output, Reporter reporter) throws  
IOException {  
        int sum = 0;  
        while (values.hasNext()) {  
            sum += values.next().get();  
        }  
        output.collect(key, new IntWritable(sum));  
    }  
}
```

Call Wordcount

```
hdfs.root <- 'wordcount'
hdfs.data <- file.path(hdfs.root, 'data')
hdfs.out <- file.path(hdfs.root, 'out')

wordcount <- function (input, output=NULL) {
  mapreduce(input=input, output=output,
input.format="text", map=map, reduce=reduce)
}
out <- wordcount(hdfs.data, hdfs.out)
```

Read data from HDFS

```
results <- from.dfs(out)
```

```
results$key[order(results$val, decreasing  
= TRUE)][1:10]
```



```
$ hadoop fs -cat /user/cloudera/wordcount/out/part-00000 |  
sort -k 2 -nr | head -n 10
```


MapReduce Benchmark



NumerInfo

```
> a.time <- proc.time()
> small.ints2=1:100000
> result.normal = sapply(small.ints2, function(x) x^2)
> proc.time() - a.time
```

```
> b.time <- proc.time()
> small.ints= to.dfs(1:100000)
> result = mapreduce(input = small.ints, map = function(k,v)
  cbind(v,v^2))
> proc.time() - b.time
```

Elapsed 0.982 second

```
> a.time <- proc.time()
> small.ints2=1:100000
> result.normal = sapply(small.ints2, function(x) x^2)
> proc.time() - a.time
  user  system elapsed
0.323   0.292   0.982
```


mapreduce



NumerInfo

Elapsed 102.755 seconds

```
> b.time <- proc.time()
> small.ints= to.dfs(1:100000)
14/03/16 01:45:13 INFO zlib.ZlibFactory: Successfully loaded & initialized native-zlib 1
library
14/03/16 01:45:13 INFO compress.CodecPool: Got brand-new compressor [.deflate]
Warning message:
In to.dfs(1:1e+05) : Converting to.dfs argument to keyval with a NULL key
> result = mapreduce(input = small.ints, map = function(k,v) cbind(v,v^2))
packageJobJar: [/tmp/RtmpN2yqhK/rmr-local-env6219109c860a, /tmp/RtmpN2yqhK/rmr-global-en
v6219891ce2b, /tmp/RtmpN2yqhK/rmr-streaming-map6219fd52f9, /tmp/hadoop-cloudera/hadoop-u
njar4261759795176283750/] [] /tmp/streamjob2063587612495148779.jar tmpDir=null
14/03/16 01:45:22 WARN mapred.JobClient: Use GenericOptionsParser for parsing the argume
nts. Applications should implement Tool for the same.
14/03/16 01:45:23 INFO mapred.FileInputFormat: Total input paths to process : 1
14/03/16 01:45:24 INFO streaming.StreamJob: getLocalDirs(): [/tmp/hadoop-cloudera/mapred
/local]
14/03/16 01:45:24 INFO streaming.StreamJob: Running job: job_201403151827_0002
14/03/16 01:45:24 INFO streaming.StreamJob: To kill this job, run:
14/03/16 01:45:24 INFO streaming.StreamJob: UNDEF/bin/hadoop job -Dmapred.job.tracker=l
ocalhost.localdomain:8021 -kill job_201403151827_0002
14/03/16 01:45:24 INFO streaming.StreamJob: Tracking URL: http://0.0.0.0:50030/jobdetail
s.jsp?jobid=job_201403151827_0002
14/03/16 01:45:25 INFO streaming.StreamJob: map 0% reduce 0%
14/03/16 01:46:10 INFO streaming.StreamJob: map 50% reduce 0%
14/03/16 01:46:12 INFO streaming.StreamJob: map 100% reduce 0%
14/03/16 01:46:33 INFO streaming.StreamJob: map 100% reduce 100%
14/03/16 01:46:33 INFO streaming.StreamJob: Job complete: job_201403151827_0002
14/03/16 01:46:33 INFO streaming.StreamJob: Output: /tmp/RtmpN2yqhK/file621977a6faca
> proc.time() - b.time
      user  system elapsed
 28.903    1.812  102.755
```




咿什麼呀?!

- HDFS stores your files as data chunk distributed on multiple datanodes
- M/R runs multiple programs called mapper on each of the data chunks or blocks. The (key,value) output of these mappers are compiled together as result by reducers.
- It takes time for mapper and reducer being spawned on these distributed system.

Kmeans Clustering



NumerInfo

A large red prohibition sign (a circle with a diagonal slash) is centered over the code block, indicating that the code is incorrect or should not be used.

```
kcluster=kmeans(mydata, 4, iter.max=10)
```


Kmeans in MapReduce Style



NumerInfo

```
kmeans =  
  function(points, ncenters, iterations = 10, distfun = NULL) {  
    if(is.null(distfun))  
      distfun = function(a,b) norm(as.matrix(a-b), type = 'F')  
  
    newCenters =  
      kmeans.iter(  
        points,  
        distfun,  
        ncenters = ncenters)  
  
    # iteratively choosing new centers  
    for(i in 1:iterations) {  
      newCenters = kmeans.iter(points, distfun,  
        centers = newCenters)  
    }  
    newCenters  
  }  
}
```

Kmeans in MapReduce Style



NumerInfo

```
kmeans.iter =  
  function(points, distfun, ncenters = dim(centers)[1], centers = NULL)  
  {  
    from.dfs(mapreduce(input = points,  
      map =  
        if (is.null(centers)) { #give random point as sample  
          function(k,v) keyval(sample(1:ncenters,1),v)}  
        else {  
          function(k,v) { #find center of minimum distance  
            distances = apply(centers, 1, function(c) distfun(c,v))  
            keyval(centers[which.min(distances),], v)}},  
      reduce = function(k,vv) keyval(NULL,  
        apply(do.call(rbind, vv), 2, mean))),  
    to.data.frame = T)  
  }
```

One More Thing...

plyrmr



NumerInfo

- Perform common data manipulation operations, as found in **plyr** and **reshape2**
- It provides a familiar plyr-like interface while hiding many of the mapreduce details
- **plyr: Tools for splitting, applying and combining data**

Installation plymr dependencies



NumerInfo

```
$ yum install libxml2-devel  
$ sudo yum install curl-devel  
$ sudo R
```

```
> Install.packages(c(" Rcurl", "httr"), dependencies = TRUE  
> Install.packages("devtools", dependencies = TRUE)  
> library(devtools)  
> install_github("pryr", "hadley")  
> Install.packages(c(" R.methodsS3", "hydroPSO"), dependencies = TRUE)
```

Installation plyrmr



NumerInfo

```
$ wget https://raw.githubusercontent.com/RevolutionAnalytics/plyrmr/master/build/plyrmr\_0.1.0.tar.gz
```

```
$ sudo R CMD INSTALL plyrmr_0.1.0.tar.gz
```


Transform in plyrmr



NumerInfo

```
> data(mtcars)
> head(mtcars)
> transform(mtcars, carb.per.cyl = carb/cyl)
```



```
> library(plyrmr)
> output(input(mtcars), "/tmp/mtcars")
> as.data.frame(transform(input("/tmp/mtcars"),
carb.per.cyl = carb/cyl))
> output(transform(input("/tmp/mtcars"), carb.per.cyl =
carb/cyl), "/tmp/mtcars.out")
```

```
■ where(  
  select(  
    mtcars,  
    carb.per.cyl = carb/cyl,  
    .replace = FALSE),  
  carb.per.cyl >= 1)
```

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb	carb.per.cyl
Ferrari Dino	19.7	6	145	175	3.62	2.77	15.5	0	1	5	6	1
Maserati Bora	15.0	8	301	335	3.54	3.57	14.6	0	1	5	8	1

Group by

```
■ as.data.frame(  
  select(  
    group(  
      input("/tmp/mtcars"),  
      cyl),  
    mean.mpg = mean(mpg)))
```

	cyl	mean.mpg
1	6	19.74
1.1	4	26.66
1.2	8	15.10



- <https://github.com/RevolutionAnalytics/RHadoop/wiki>
- <http://www.slideshare.net/RevolutionAnalytics/rhadoop-r-meets-hadoop>
- http://www.slideshare.net/Hadoop_Summit/enabling-r-on-hadoop



■ Website:

■ ywchiu.com

■ Email:

■ tr.ywchiu@gmail.com

THANK YOU

