Hadoop Cluster Anyone?

$ ./bdutil deploy
Mapreduce
A distributed computing paradigm that consists of:
● A map step, performed on subsets of the input data
● A reduce step, that combines the output together
A Mapreduce can run on structured or unstructured data

Hadoop
An Open Source implementation of Mapreduce
BigQuery and Datastore Connectors

- Datastore Connector
- Hadoop
- BigQuery Connector
- Cloud Storage Connector
- Google Cloud Storage
bdutil - Thin wrapper around Cloud SDK command-line tools

- Deploy Hadoop cluster on-demand
- Installs and configures GCS connector
- Extensible - add scripts to run during deployment
Let’s talk briefly about Big Data...

Courtesy of Nimbus Ninety – www.nimbusninety.com
Some examples

- Personalised insights into your customer base
  - Loyalty Cards, In App Purchases, Point of Sale offers
- User retention activities
  - For Games, Mobile Apps, anything that has pattern based usage
- Predicting health problems
- Website optimization through analytics
- Enabling future breakthroughs in biology and medicine
- Autonomous Traffic Lights and Flying Cars
Where is your data?

Personal Genome Project

Genomic, Environmental, and Human Trait data
Sometimes you just need to do something simple... but you need to do a **lot** of it!
How Big is a Human Genome?

$ gsutil ls gs://pgp-harvard-data-public/hu*//*//*/ASM/master*
  + 173 more...

$ wc masterVarBeta-GS000015172-ASM.tsv
  17615429  306261586  2349556328 masterVarBeta-GS000015172-ASM.tsv

One File Per Genome

Participant Identifier Buried in Object Name
What we have isn’t what we want

```
$ head -n 30 masterVarBeta-GS000015172-ASM.tsv | cut -f -10
#ASSEMBLY_ID  GS000015172-ASM
#CNV_DIPLOID_WINDOW_WIDTH  2000
#CNV_NONDIPLOID_WINDOW_WIDTH  100000
[...]
>locus ploidy chromosome  begin  end   zygosity  varType   reference allele1Seq
allele2Seq
1  2   chr1   0  10000  no-call  no-ref  =  ?  ?
2  2   chr1   10000  10476  no-call  complex  =  ?  ?
3  2   chr1   10476  10481  hom  ref=  =  =
4  2   chr1   10481  10518  no-call  complex  =  ?  ?
[...]
```
Wheelbarrow or Giant Truck?

$ ./cgi-mapper.py ./hu011C57/masterVarBeta-GS000015172-ASM.tsv

demo-m$ ./hadoop-install/bin/hadoop jar \
./hadoop-install/contrib/streaming/hadoop-streaming-1.2.1.jar \
-input gs://pgp-harvard-data-public/hu0*/*/*/*/ASM/master* \
-mapper cgi-mapper.py \
-file cgi-mapper.py \
--numReduceTasks 0 \
-output gs://big-data-roadshow/output
bd-nn Hadoop Map/Reduce Administration

**State:** RUNNING  
**Started:** Thu Jul 24 20:49:35 UTC 2014  
**Version:** 1.2.1, r1503152  
**Compiled:** Mon Jul 22 15:23:09 PDT 2013 by mattf  
**Identifier:** 201407242049  
**SafeMode:** OFF

---

**Cluster Summary (Heap Size is 687 MB/21.02 GB)**

<table>
<thead>
<tr>
<th>Running Map Tasks</th>
<th>Running Reduce Tasks</th>
<th>Total Submissions</th>
<th>Nodes</th>
<th>Occupied Map Slots</th>
<th>Occupied Reduce Slots</th>
<th>Res Map</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Hadoop job_201407242049_0001 on bd-nn

User: hadoop
Job Name: streamjob8138476836720102555.jar
Submit Host: bd-nn.c.big-data-roadshow.internal
Submit Host Address: 10.240.139.137
Job-ACLs: All users are allowed
Job Setup: Successful
Status: Succeeded
Started at: Thu Jul 24 20:57:44 UTC 2014
Finished at: Thu Jul 24 21:03:26 UTC 2014
Finished in: 5mins, 41sec
Job Cleanup: Successful

<table>
<thead>
<tr>
<th>Kind</th>
<th>% Complete</th>
<th>Num Tasks</th>
<th>Pending</th>
<th>Running</th>
<th>Complete</th>
<th>Killed</th>
<th>Failed/Killed Task Attempts</th>
</tr>
</thead>
<tbody>
<tr>
<td>map</td>
<td>100.00%</td>
<td>704</td>
<td>0</td>
<td>0</td>
<td>704</td>
<td>0</td>
<td>4 / 68</td>
</tr>
<tr>
<td>reduce</td>
<td>100.00%</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0 / 0</td>
</tr>
<tr>
<td>Counter</td>
<td>Map</td>
<td>Reduce</td>
<td>Total</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------------------------------------</td>
<td>-------</td>
<td>--------</td>
<td>------------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>File Input Format Counters</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bytes Read</td>
<td>0</td>
<td>0</td>
<td>42,473,494,841</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Job Counters</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLOTS_MILLIS_MAPS</td>
<td>0</td>
<td>0</td>
<td>89,547,604</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total time spent by all reduces waiting after reserving slots (ms)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rack-local map tasks</td>
<td>0</td>
<td>0</td>
<td>776</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total time spent by all maps waiting after reserving slots (ms)</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Launched map tasks</td>
<td>0</td>
<td>0</td>
<td>776</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SLOTS_MILLIS_REDUCES</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>File Output Format Counters</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bytes Written</td>
<td>0</td>
<td>0</td>
<td>434,716,013,268</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>FileSystemCounters</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GS_BYTES_READ</td>
<td>42,473,621,561</td>
<td>0</td>
<td>42,473,621,561</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FILE_BYTES_WRITTEN</td>
<td>30,785,106</td>
<td>0</td>
<td>30,785,106</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GS_BYTES_WRITTEN</td>
<td>434,716,013,268</td>
<td>0</td>
<td>434,716,013,268</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Map-Reduce Framework</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Map input records</td>
<td>0</td>
<td>0</td>
<td>3,060,990,274</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Physical memory (bytes) snapshot</td>
<td>0</td>
<td>0</td>
<td>734,499,840,000</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Spilled Records</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total committed heap usage (bytes)</td>
<td>0</td>
<td>0</td>
<td>952,810,078,208</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU time spent (ms)</td>
<td>0</td>
<td>0</td>
<td>81,531,230</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Map input bytes</td>
<td>0</td>
<td>0</td>
<td>409,871,015,097</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Virtual memory (bytes) snapshot</td>
<td>0</td>
<td>0</td>
<td>1,958,114,177,024</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SPLIT_RAW_BYTES</td>
<td>126,720</td>
<td>0</td>
<td>126,720</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Map output records</td>
<td>0</td>
<td>0</td>
<td>3,041,791,805</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Goodbye Cluster!

$ ./bdutil delete
But what did it cost?

434 GB, 3 billion rows
bdutil deploy: ~3.5 minutes
hadoop job: ~7 minutes
bdutil delete: ~1.5 minutes
Total time with manual entry: ~14 minutes
n1-standard-16 instances: $1.16 / hour
But what did it cost?

Compute: $5.68

What about storage? 434 GB @ $0.026 GB / month
$0.38 / day
$11.28 / month

What about networking?
No Charge!
But there’s more...
Google BigQuery
BigQuery: Big Data Analytics in the Cloud

Unrivaled Performance and Scale

- Scan multiple TB’s in seconds
- Interactive query performance
- No limits on amount of data

Ease of Use and Adoption

- No administration / provisioning
- Convenience of SQL
- Open interfaces (REST, WebUI, ODBC)
- First 1 TB of data processed per month is free

Advanced “Big Data” Storage

- Familiar database structure
- Easy data management and ACL’s
- Fast, atomic imports
wikimedia page views

```
SELECT
title,
SUM(views) AS views
FROM
[bigquery-samples:wikimedia_pageviews.201207],
[bigquery-samples:wikimedia_pageviews.201206],
[bigquery-samples:wikimedia_pageviews.201205],
[bigquery-samples:wikimedia_pageviews.201204],
[bigquery-samples:wikimedia_pageviews.201203],
[bigquery-samples:wikimedia_pageviews.201202],
[bigquery-samples:wikimedia_pageviews.201201]
WHERE
 NOT title CONTAINS ':'
AND wikipedia_project = 'wp'
AND language = 'en'
AND REGEXP_MATCH(title,
    r'G.*o.*e$')
GROUP BY
title
ORDER BY
views DESC
```
Create and Import

Choose job template  Choose destination  Select data  Specify schema  Advanced options

Source Format
- CSV
- JSON (Newline Delimited)
- AppEngine Datastore Backup

Load data from
- File upload
  - Google Cloud Storage
    - gs://big-data-roadshow/output/part-*

Buttons: Back  Next  Submit  Cancel
### Table Details: cgi_variants

#### Schema

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Nullable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>sample_id</td>
<td>STRING</td>
<td>NULLABLE</td>
<td>Sample ID</td>
</tr>
<tr>
<td>locus</td>
<td>INTEGER</td>
<td>NULLABLE</td>
<td>Integer ID where regi fully no-call</td>
</tr>
<tr>
<td>ploidy</td>
<td>INTEGER</td>
<td>NULLABLE</td>
<td>Number of</td>
</tr>
<tr>
<td>chromosome</td>
<td>STRING</td>
<td>NULLABLE</td>
<td>Chromosome</td>
</tr>
<tr>
<td>locusBegin</td>
<td>INTEGER</td>
<td>NULLABLE</td>
<td>Locus start</td>
</tr>
<tr>
<td>locusEnd</td>
<td>INTEGER</td>
<td>NULLABLE</td>
<td>Locus end</td>
</tr>
<tr>
<td>zygosity</td>
<td>STRING</td>
<td>NULLABLE</td>
<td>Call compl - no-call: / - hap: Hap - half: Dipl - full: Doo</td>
</tr>
</tbody>
</table>

---

**Big Data Roadshow**

- example
- google.com:biggene:1000gen...
- google.com:biggene:pgp
  - brca1_freq
  - cgi_variants
  - gvcf_variants
  - phenotypes
  - variants
SELECT COUNT(*) as RowCount FROM [google.com:biggene:pgp.cgi_variants]
For Immediate Release: Friday, May 9, 2014

Longevity gene may boost brain power

*NIH-funded researchers discover the gene may enhance cognitive abilities*

Scientists showed that people who have a variant of a longevity gene, called KLOTHO, have improved brain skills such as thinking, learning and memory regardless of their age, sex, or whether they have a genetic risk factor for Alzheimer’s disease. Increasing KLOTHO gene levels in mice made them smarter, possibly by increasing the strength of connections between nerve cells in the brain. The study was partly funded by the National Institutes of Health.

### Reference SNP (rsSNP) Cluster Report: rs9536314

<table>
<thead>
<tr>
<th>RefSNP</th>
<th>Allele</th>
<th>HGVS Names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organism: human (<em>Homo sapiens</em>)</td>
<td>SNV: single nucleotide variation</td>
<td>NC_000013.11:g.33054001T&gt;G</td>
</tr>
<tr>
<td>Molecule Type: Genomic</td>
<td>RefSNP Alleles: G/T (FWD)</td>
<td>NG_011485.1:g.42568T&gt;G</td>
</tr>
<tr>
<td>Created/Updated in build: 119/141</td>
<td>Allele Origin:</td>
<td>NM_004795.3:c.1064T&gt;G</td>
</tr>
<tr>
<td>Map to Genome Build: 106/Weight</td>
<td>Ancestral Allele: T</td>
<td>NP_004786.2:p.Phe352Val</td>
</tr>
<tr>
<td>Validation Status:</td>
<td>Clinical Channel: unknown</td>
<td>NT_024524.15:g.14645885T&gt;G</td>
</tr>
<tr>
<td>Citation: PubMed</td>
<td>Clinical Significance: NA</td>
<td>XM_006719895.1:c.133T&gt;G</td>
</tr>
<tr>
<td>MAF/MinorAlleleCount: G=0.130/284</td>
<td>MAF Source: 1000 Genomes</td>
<td>XP_006719958.1:p.Phe45Val</td>
</tr>
</tbody>
</table>

**SNP Details are organized in the following sections:**

- GeneView
- Map
- Submission
- Fasta
- Resource
- Diversity
- Validation

**Integrated Maps (Hint: click on 'Chr Pos' or 'Contig Pos' column value to see variation in NCBI sequence viewer):**

<table>
<thead>
<tr>
<th>Assembly</th>
<th>Genome Build</th>
<th>Chr</th>
<th>Chr Pos</th>
<th>Contig</th>
<th>Contig Pos</th>
<th>SNP to Chr</th>
<th>Contig allele</th>
<th>Contig to Chr</th>
<th>Neighbor SNP</th>
<th>Map Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRCh38</td>
<td>106</td>
<td>13</td>
<td>33054001</td>
<td>NT_024524.15</td>
<td>14645865</td>
<td>Fwd</td>
<td>T</td>
<td>Fwd</td>
<td>view</td>
<td>mapup</td>
</tr>
<tr>
<td>GRCh37.p13</td>
<td>105</td>
<td>13</td>
<td>33628138</td>
<td>NT_024524.14</td>
<td>14608136</td>
<td>Fwd</td>
<td>T</td>
<td>Fwd</td>
<td>view</td>
<td>blast</td>
</tr>
</tbody>
</table>

### Insight: What do we expect to see

<table>
<thead>
<tr>
<th>Subject</th>
<th>chromosome</th>
<th>reference</th>
<th>allele1Seq</th>
<th>allele2Seq</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>huCCAFD0</td>
<td>chr13</td>
<td>=</td>
<td>=</td>
<td>=</td>
<td>No Variant</td>
</tr>
<tr>
<td>huE58004</td>
<td>chr13</td>
<td>T</td>
<td>G</td>
<td>T</td>
<td>Variant in one Allele</td>
</tr>
<tr>
<td>hu040C0A</td>
<td>chr13</td>
<td>T</td>
<td>G</td>
<td>G</td>
<td>Variant in both Alleles</td>
</tr>
<tr>
<td>huF1DC30</td>
<td>chr13</td>
<td>=</td>
<td>?</td>
<td>?</td>
<td>Not Sure</td>
</tr>
</tbody>
</table>

G = Guanine  
T = Thymine
# Sample level data for Klotho variant rs9536314

```sql
SELECT
    sample_id,
    chromosome,
    locusBegin,
    locusEnd,
    reference,
    allele1Seq,
    allele2Seq,
FROM
    [google.com:biggene:pgp.cgi_variants]
WHERE
    chromosome = "chr13"
    AND locusBegin <= 33628137
    AND locusEnd >= 33628138;
```
<table>
<thead>
<tr>
<th>Row</th>
<th>sample_id</th>
<th>chromosome</th>
<th>locusBegin</th>
<th>locusEnd</th>
<th>reference</th>
<th>allele1Seq</th>
<th>allele2Seq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>hu132B5C</td>
<td>chr13</td>
<td>33627829</td>
<td>33628967</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td>2</td>
<td>hu92FD55</td>
<td>chr13</td>
<td>33627829</td>
<td>33628608</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td>3</td>
<td>hu040C0A</td>
<td>chr13</td>
<td>33628137</td>
<td>33628138</td>
<td>T</td>
<td>G</td>
<td>G</td>
</tr>
<tr>
<td>4</td>
<td>huFAF983</td>
<td>chr13</td>
<td>33627829</td>
<td>33629510</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td>5</td>
<td>huDE435D</td>
<td>chr13</td>
<td>33627829</td>
<td>33628440</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td>6</td>
<td>hu5FB1B9</td>
<td>chr13</td>
<td>33627829</td>
<td>33628988</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td>7</td>
<td>huBEDA0B</td>
<td>chr13</td>
<td>33627829</td>
<td>33629979</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td>8</td>
<td>huBA30D4</td>
<td>chr13</td>
<td>33627829</td>
<td>33629512</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
</tbody>
</table>
# Sample counts for Klotho variant rs9536314

```
SELECT
    COUNT(sample_id) AS sample_counts,
    chromosome,
    reference,
    allele1Seq,
    allele2Seq,
FROM
    [google.com:biggene:pgp.cgi_variants]
WHERE
    chromosome = "chr13"
    AND locusBegin <= 33628137
    AND locusEnd >= 33628138
GROUP BY
    chromosome,
    reference,
    allele1Seq,
    allele2Seq
ORDER BY
    sample_counts DESC
```
### Query Results  4:22pm, 23 Jul 2014

<table>
<thead>
<tr>
<th>Row</th>
<th>sample_counts</th>
<th>chromosome</th>
<th>reference</th>
<th>allele1Seq</th>
<th>allele2Seq</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>135</td>
<td>chr13</td>
<td>=</td>
<td>=</td>
<td>=</td>
</tr>
<tr>
<td>2</td>
<td>32</td>
<td>chr13</td>
<td>T</td>
<td>G</td>
<td>T</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>chr13</td>
<td>T</td>
<td>G</td>
<td>G</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>chr13</td>
<td>=</td>
<td>?</td>
<td>?</td>
</tr>
</tbody>
</table>
From Data to Meaning

What questions do you want to ask?

Where’s your Data? Is it in the right place? Is it in the right format?

Data → Extract, Transform, Load → Meaning
Coming Soon

**Apache Spark**
- bdutil support
- Machine learning via MLlib
- SQL via Shark

**Cloud Pub/Sub**
- Managed Service
- many-to-many, async messaging between apps
- Limited Preview

**Cloud Dataflow**
- Managed Service
- Unified Batch and Streaming
- Automatic Pipeline Optimization
- Private Beta
Resources

Hadoop on Google Cloud Platform - http://goo.gl/33PoKG
(Covers the Hadoop Connectors and bdutil)

Personal Genome Project - http://goo.gl/j1oTTF

Cloud Developers console - https://console.developers.google.com

Google BigQuery - https://developers.google.com/bigquery
BigQuery Console - https://bigquery.cloud.google.com