Top 10 Lessons Learned from Deploying Hadoop in a Private Cloud

Rod Cope, CTO & Founder
OpenLogic, Inc.

Agenda

- Introduction
- The Problem
- The Solution
- Top 10 Lessons
- Final Thoughts
- Q & A

Introduction

- **Rod Cope**
  - CTO & Founder of OpenLogic
  - 25 years of software development experience
  - IBM Global Services, Anthem, General Electric

- **OpenLogic**
  - Open Source Support, Governance, and Scanning Solutions
  - Certified library w/SLA support on 500+ Open Source packages
    - http://olex.openlogic.com
  - Over 200 Enterprise customers

The Problem

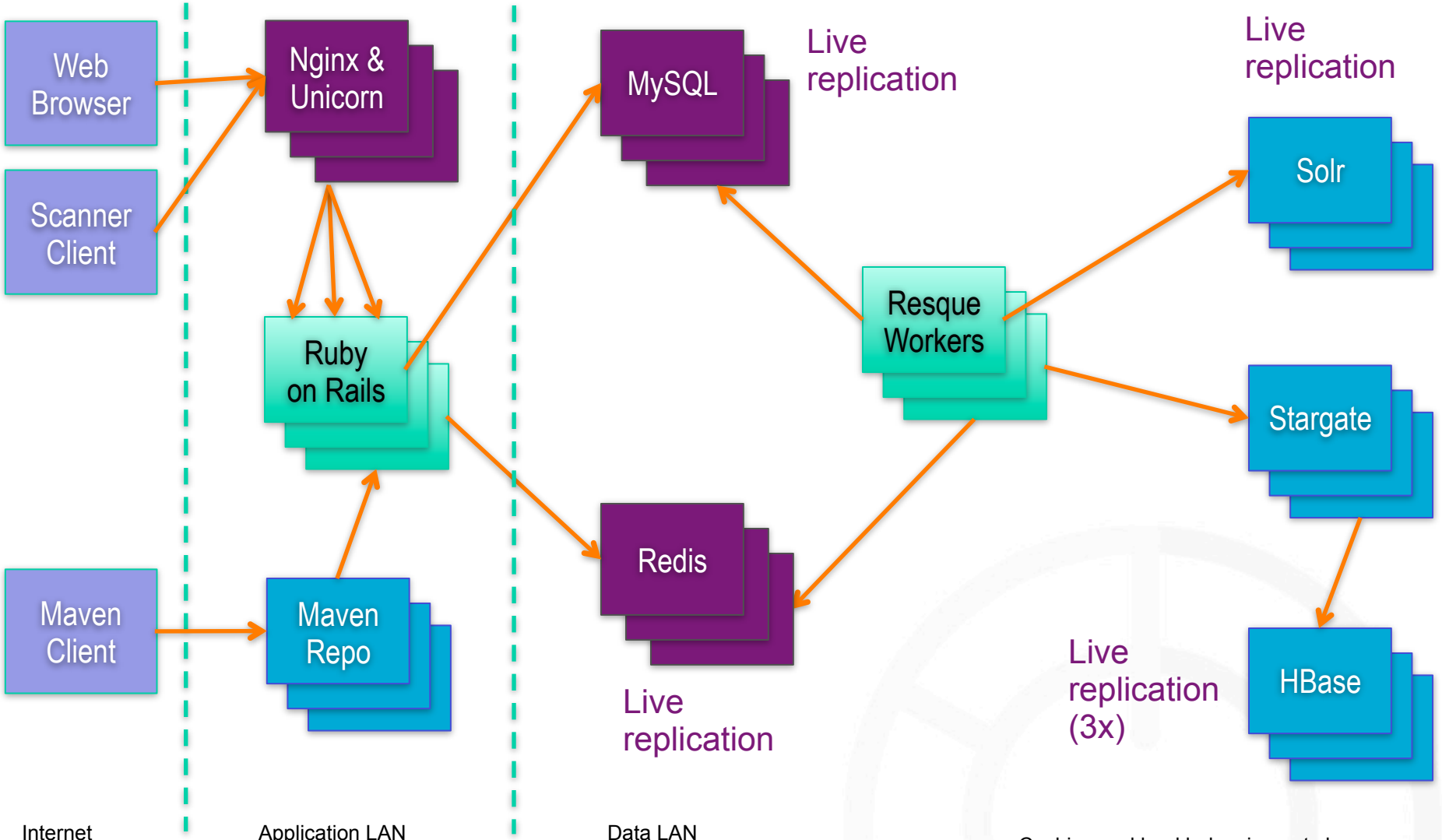
- **"Big Data"**
  - All the world's Open Source Software
  - Metadata, code, indexes
  - Individual tables contain many terabytes
  - Relational databases aren't scale-free

- **Growing every day**
- **Need real-time random access to all data**
- **Long-running and complex analysis jobs**

The Solution

- **Hadoop, HBase, and Solr**
  - Hadoop – distributed file system, map/reduce
  - HBase – "NoSQL" data store – column-oriented
  - Solr – search server based on Lucene
  - All are scalable, flexible, fast, well-supported, used in production environments
- **And a supporting cast of thousands…**
  - Stargate, MySQL, Rails, Redis, Resque, Nginx, Unicorn, HAProxy, Memcached, Ruby, JRuby, CentOS, …

Solution Architecture



Hadoop Implementation

- **Private Cloud**
  - 100+ CPU cores
  - 100+ Terabytes of disk
  - Machines don't have identity
  - Add capacity by plugging in new machines
- **Why not Amazon EC2?**
  - Great for computational bursts
  - Expensive for long-term storage of Big Data
  - Not yet consistent enough for mission-critical usage of HBase

Top 10 Lessons Learned

- Configuration is key
- "Commodity Hardware" is not an old desktop
- Hadoop & HBase crave bandwidth
- Big Data takes a long time…
- Big Data is hard
- Scripting languages can help
- Public clouds are expensive
- Not possible without Open Source
- Expect things to fail – a lot
- It's all still cutting edge

Configuration is Key

- **Many moving parts**
- **Pay attention to the details**
  - Operating system – max open files, sockets, and other limits
  - Hadoop – max Map/Reduce jobs, memory, disk
  - HBase – region size, memory
  - Solr – merge factor, norms, memory
- **Minor versions are very important**
  - Use a good known combination of Hadoop and HBase
  - Specific patches are critical
  - The fine print matters

Configuration Tips & Gotchas

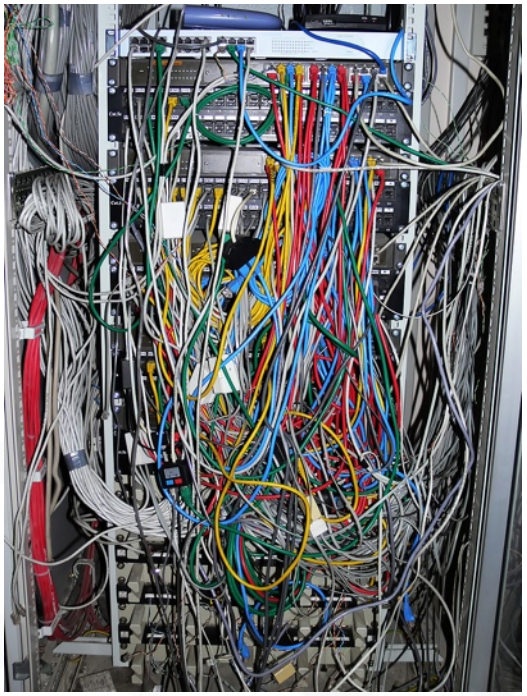
- **Follow all HBase configuration advice here:**
  - http://wiki.apache.org/hadoop/Hbase/Troubleshooting
  - Yes, that's a whole lot of configuration
  - Skip steps at your own peril!
- **If you really need HA Hadoop**
  - http://www.cloudera.com/blog/2009/07/hadoop-ha-configuration/
- **If you hit datanode timeouts while writing to sockets:**
  - dfs.datanode.socket.write.timeout = 0
  - Even though it should be ignored…

Configuration Tips & Gotchas (cont.)

- **Linux kernel is important – affects configuration switches, both required and optional**
  - Example: epoll limits required as of 2.6.27, then no longer required in newer kernels such as 2.6.33+
  - http://pero.blogs.aprilmayjune.org/2009/01/22/hadoop-and-linux-kernel-2627-epoll-limits/
- **Upgrade your machine BIOS, network card BIOS, and all hardware drivers**
  - Example: issues with certain default configurations of Dell boxes on CentOS/RHEL 5.x and Broadcom NIC's
    - Will drop packets & cause other problems under high load
  - Disable MSI in Linux & power saver (C-states) in machine BIOS

Configuration Debugging Tips

- **Many problems only show up under severe load**
  - Sustained, massive data loads running for 2-24 hours
- **Change only one parameter at a time**
  - Yes, this can be excruciating
- **Ask the mailing list or your support provider**
  - They've seen a lot, likely including your problem…but not always
  - Don't be afraid to dig in and read some code…



Comment found in Hadoop networking code

```
Ideally we should wait after transferTo returns 0. But
because of a bug in JRE on Linux
(http://bugs.sun.com/view_bug.do?bug_id=5103988),
which throws an exception instead of returning 0, we wait
for the channel to be writable before writing to it. If
you ever see IOException with message "Resource
temporarily unavailable" thrown here, please let us know.
Once we move to JAVA SE 7, wait should be moved to correct
place.
```

- Hadoop stresses every bit of networking code in Java and tends to expose all the cracks
- This bug was fixed in JDK 1.6.0_18 (after 6 years)

Commodity Hardware

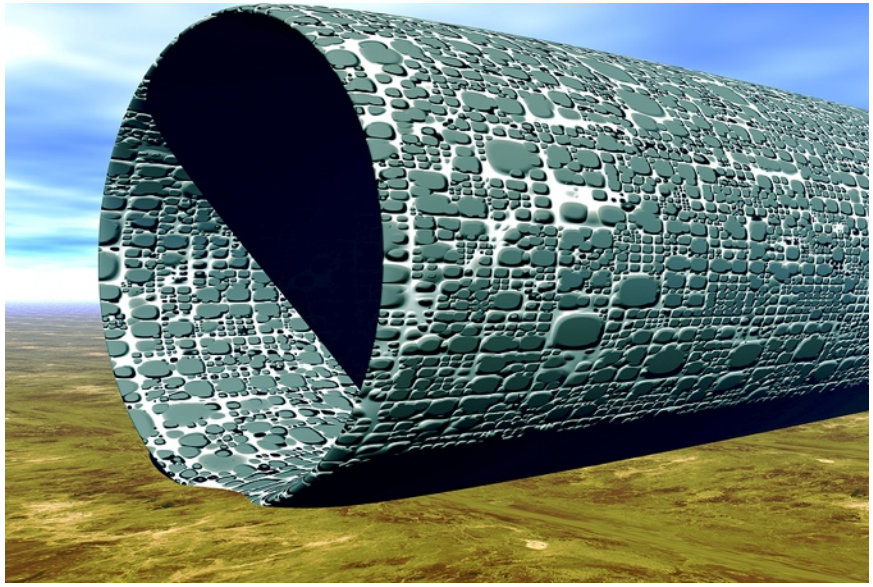
- **“Commodity hardware” != 3 year old desktop**
- **Dual quad-core, 32GB RAM, 4+ disks**
- **Don’t bother with RAID on Hadoop data disks**
  - Be wary of non-enterprise drives
- **Expect ugly hardware issues at some point**

OpenLogic's Hadoop Deployment

- **Dual quad-core and dual hex-core Dell boxes**
- **32-64GB RAM**
  - ECC (highly recommended by Google)
- **6 x 2TB enterprise hard drives**
- **RAID 1 on two of the drives**
  - OS, Hadoop, HBase, Solr, NFS mounts (be careful!), job code, etc.
  - Key "source" data backups
- **Hadoop datanode gets remaining drives**
- **Redundant enterprise switches**
- **Dual- and quad-gigabit NIC's**

Hadoop & HBase Crave Bandwidth and More

- **Hadoop**
  - Map/Reduce jobs shuffle lots of data
  - Continuously replicating blocks and rebalancing
  - Loves bandwidth – dual-gigabit network on dedicated switches
  - 10Gbps network can help



- **HBase**
  - Needs 5+ machines to stretch its legs
  - Depends on ZooKeeper – low-latency is important
  - Don't let it run low on memory

Big Data Takes a Long Time…

- **…to do anything**
  - Load, list, walk directory structures, count, process, test, back up
  - I'm not kidding
- **Hard to test, but don't be tempted to skip it**
  - You'll eventually hit every corner case you know and don't know
- **Backups are difficult**
  - Consider a backup Hadoop cluster
  - HBase team is working on live replication
  - Solr already has built-in replication

Advice on Loading Big Data into HBase

- **Don't use a single machine to load the cluster**
  - You might not live long enough to see it finish
- **At OpenLogic, we spread raw source data across many machines and hard drives via NFS**
  - Be very careful with NFS configuration – can hang machines
- **Load data into HBase via Hadoop map/reduce jobs**
  - Turn off WAL for much better performance
  - put.setWriteToWAL(false)
- **Avoid large values (> 5MB)**
  - Works, but may cause instability and/or performance issues
  - Rows and columns are cheap, so use more of them instead
- **Be careful not to over-commit to Solr**

Advice on Getting Big Data out of HBase

- **HBase → NoSQL**
  - Think hash table, not relational database
- **How do find my data if primary key won't cut it?**
- **Solr to the rescue**
  - Very fast, highly scalable search server with built-in sharding and replication – based on Lucene
  - Dynamic schema, powerful query language, faceted search, accessible via simple REST-like web API w/XML, JSON, Ruby, and other data formats

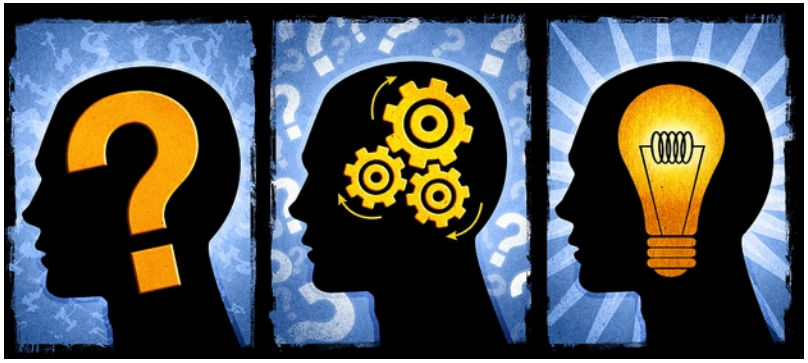
Solr

- **Sharding**
  - Query any server – it executes the same query against all other servers in the group
  - Returns aggregated result to original caller

- **Async replication (slaves poll their masters)**
  - Can use repeaters if replicating across data centers

- **OpenLogic**
  - Solr farm, sharded, cross-replicated, fronted with HAProxy
    - Load balanced writes across masters, reads across slaves and masters
  - Billions of lines of code in HBase, all indexed in Solr for real-time search in multiple ways
  - Over 20 Solr fields indexed per source file

Big Data is Hard



- **Expect to learn and experiment quite a bit**
  - Many moving parts, lots of decisions to make
  - You won't get them all right the first time
- **Expect to discover new and better ways of modeling your data and processes**
  - Don't be afraid to start over once or twice
- **Consider getting outside help**
  - Training, consulting, mentoring, support

Scripting Languages Can Help

- Scripting is faster and easier than writing Java
- Great for system administration tasks, testing
- Standard HBase shell is based on JRuby
- Very easy Map/Reduce jobs with J/Ruby and Wukong
- Used heavily at OpenLogic
  - Productivity of Ruby
  - Power of Java Virtual Machine
  - Ruby on Rails, Hadoop integration, GUI clients

Java (27 lines)

```
public class Filter {
  public static void main( String[] args ) {
    List list = new ArrayList();
    list.add( "Rod" );
    list.add( "Neeta" );
    list.add( "Eric" );
    list.add( "Missy" );

    Filter filter = new Filter();
    List shorts = filter.filterLongerThan( list, 4 );
    System.out.println( shorts.size() );

    Iterator iter = shorts.iterator();
    while ( iter.hasNext() ) {
      System.out.println( iter.next() );
    }
  }
  public List filterLongerThan( List list, int length ) {
    List result = new ArrayList();
    Iterator iter = list.iterator();
    while ( iter.hasNext() ) {
      String item = (String) iter.next();
      if ( item.length() <= length ) {
        result.add( item );
      }
    }
    return result;
  }
```

Scripting languages (4 lines)

JRuby

```
list = ["Rod", "Neeta", "Eric", "Missy"]
shorts = list.find_all { |name| name.size <= 4 }
puts shorts.size
shorts.each { |name| puts name }
```

```
-> 2
-> Rod
   Eric
```

Groovy

```
list = ["Rod", "Neeta", "Eric", "Missy"]
shorts = list.findAll { name -> name.size() <= 4 }
println shorts.size
shorts.each { name -> println name }
```

```
-> 2
-> Rod
   Eric
```

Public Clouds and Big Data

- **Amazon EC2**
  - EBS Storage
    - 100TB * $0.10/GB/month = **$120k/year**
  - Double Extra Large instances
    - 13 EC2 compute units, 34.2GB RAM
    - 20 instances * $1.00/hr * 8,760 hrs/yr = $175k/year
    - 3 year reserved instances
      - 20 * 4k = $80k up front to reserve
      - (20 * $0.34/hr * 8,760 hrs/yr * 3 yrs) / 3 = **$86k/year** to operate
  - Totals for 20 virtual machines
    - 1st year cost: $120k + $80k + $86k = $286k
    - 2nd & 3rd year costs: $120k + $86k = $206k
    - Average: ($286k + $206k + $206k) / 3 = **$232k/year**

Private Clouds and Big Data

- **Buy your own**
  - 20 * Dell servers w/12 CPU cores, 32GB RAM, 5 TB disk = $160k
    - Over 33 EC2 compute units each
  - Total: **$53k/year** (amortized over 3 years)

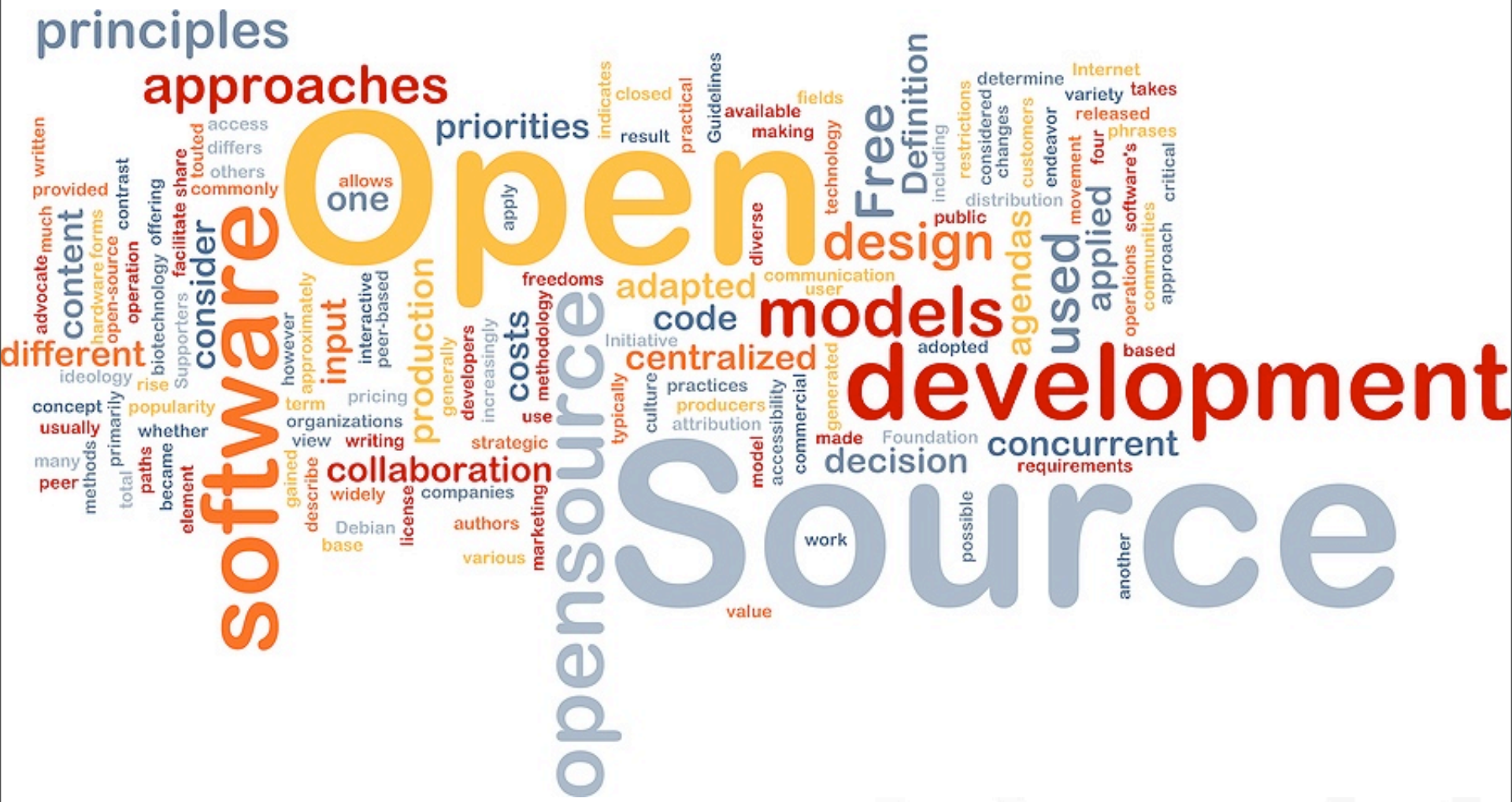
Public Clouds are Expensive for Big Data

- **Amazon EC2**
  - 20 instances * 13 EC2 compute units = 260 EC2 compute units
  - Cost: $232k/year

- **Buy your own**
  - 20 machines * 33 EC2 compute units = 660 EC2 compute units
  - Cost: $53k/year
  - Does not include hosting & maintenance costs

- **Don't think system administration goes away**
  - You still "own" all the instances – monitoring, debugging, support

Not Possible Without Open Source



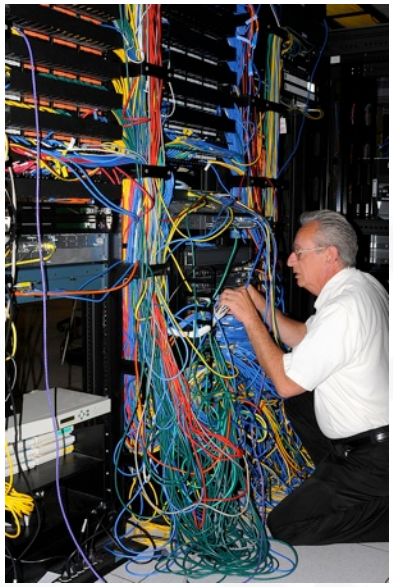
Not Possible Without Open Source

- Hadoop, HBase, Solr
- Apache, Tomcat, ZooKeeper, HAProxy
- Stargate, JRuby, Lucene, Jetty, HSQLDB, Geronimo
- Apache Commons, JUnit
- CentOS
- Dozens more

- Too expensive to build or buy everything

Expect Things to Fail – A Lot

- **Hardware**
  - Power supplies, hard drives
- **Operating System**
  - Kernel panics, zombie processes, dropped packets
- **Hadoop and Friends**
  - Hadoop datanodes, HBase regionservers, Stargate servers, Solr servers
- **Your Code and Data**
  - Stray Map/Reduce jobs, strange corner cases in your data leading to program failures



It's All Still Cutting Edge

- **Hadoop**
  - SPOF around Namenode, append functionality
- **HBase**
  - Backup, replication, and indexing solutions in flux
- **Solr**
  - Several competing solutions around cloud-like scalability and fault-tolerance, including ZooKeeper and Hadoop integration

Final Thoughts

- **You can host big data in your own private cloud**
  - Tools are available today that didn't exist a few years ago
  - Fast to prototype – production readiness takes time
  - Expect to invest in training and support

- **Public clouds**
  - Great for learning, experimenting, testing
  - Best for bursts vs. sustained loads
  - Beware latency, expense of long-term Big Data storage
- **You still need "Small Data"**
  - SQL and NoSQL coexist peacefully
  - OpenLogic uses MySQL & Redis in addition to HBase, Solr, Memcached

Q & A

Any questions for Rod?

rod.cope@openlogic.com

Slides: http://www.openlogic.com/news/presentations.php