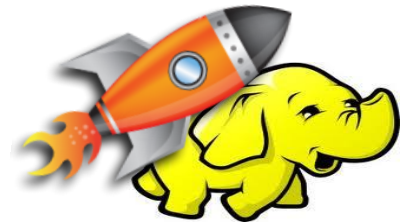


COMPUTERWORLD  
**OSBC** | **OPEN**  
BUSINESS CONFERENCE

**Cloud. Data. Mobile. Open Source.**

APRIL 29-30, 2013

# Turbo-Charging Open Source Hadoop for Faster, more Meaningful Insights



Gord Sissons  
Senior Manager, Technical Marketing  
IBM Platform Computing  
[gssissons@ca.ibm.com](mailto:gssissons@ca.ibm.com)

## Agenda

- Some Context – IBM Platform Computing
- Low-latency scheduling meets open-source
- Breakthrough performance
- Multi-tenancy (*for real!*)
- Cluster-sprawl - The *elephant* in the room
- Side step the looming challenges

## IBM Platform Computing

- Acquired by IBM in 2012
- 20 year history in high-performance computing
- 2000+ global customers
- 23 of 30 largest enterprises
- High-performance, mission-critical, extreme scale
- Comprehensive capability

*De facto standard for commercial high-performance computing*

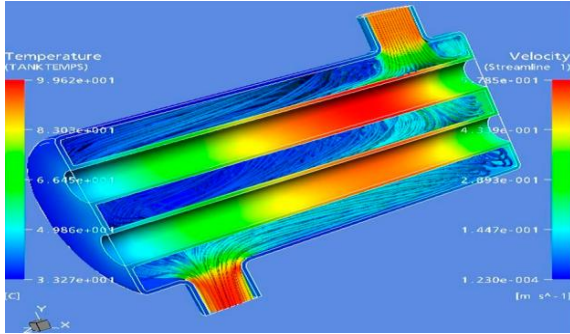
*Powers financial analytics grids for 60% of top investment banks*

*Over 5 million CPUs under management*

*Breakthrough performance in Big Data analytics*



## Technical Computing - HPC



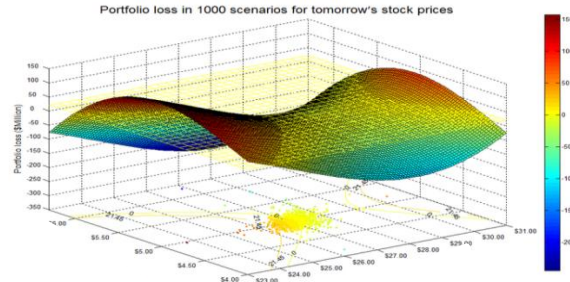
Platform LSF  
Family

Scalable, comprehensive workload management for demanding heterogeneous environments

Platform HPC

Simplified, integrated HPC management software bundled with systems

## Analytics Infrastructure Software



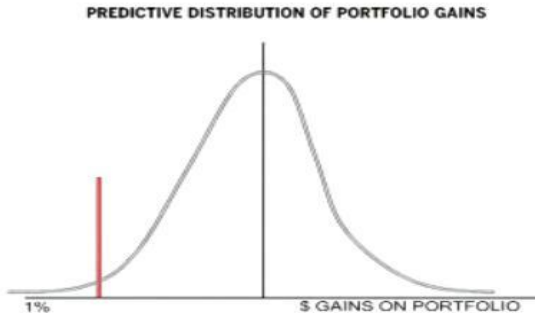
Platform  
Symphony Family

High-throughput, low-latency compute and data intensive analytics

- An SOA infrastructure for analytics
- Extreme performance and scale
- Complex Computations (i.e., risk)
- Big Data Analytics via MapReduce

## Our worldview – shaped by time critical analytics

- Financial firms compete on their ability to maximize use of capital
- Monte-Carlo simulation is a staple technique for simulating market outcomes
- Underlying instruments are increasingly complex
- A crush of new regulation



$$\begin{aligned}
 \sigma_{\lambda}^2 &= \sum_{i=1}^N L_i^2 \cdot E[(D_i^2 - 2D_i \cdot p_i + p_i^2)] + \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N L_i \cdot L_j \cdot E[(D_i D_j - p_i D_j - p_j D_i + p_i p_j)] \\
 &= \sum_{i=1}^N L_i^2 \cdot p_i \cdot (1 - p_i) + \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N L_i \cdot L_j \cdot [E(D_i D_j) - p_i p_j] \text{ because } D_i \in \{0; 1\} \text{ and hence } D_i^2 = D_i \\
 &= \sum_{i=1}^N L_i^2 \cdot p_i \cdot (1 - p_i) + \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N L_i \cdot L_j \cdot [Pr(D_i = 1 \cap D_j = 1) - p_i p_j] \text{ because of (2)} \\
 &= \sum_{i=1}^N L_i^2 \cdot p_i \cdot (1 - p_i) + \sum_{i=1}^N \sum_{\substack{j=1 \\ j \neq i}}^N L_i \cdot L_j \cdot (\Phi^{(2)}(\tau_i, \tau_j; w_i w_j) - p_i p_j) \text{ as in (6)}
 \end{aligned}$$

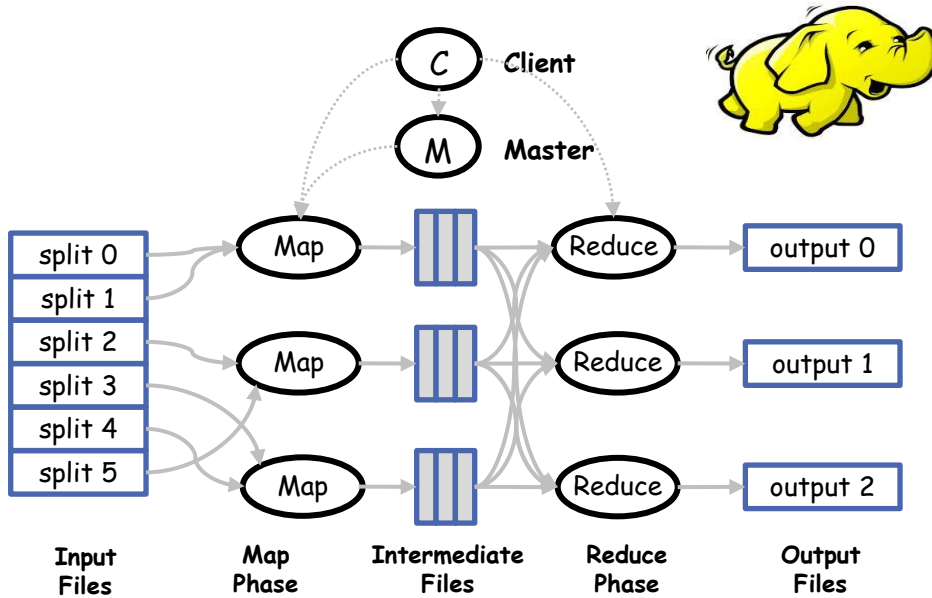
Compute this over 5,000 market scenarios comprised of 200 risk factors over 10 years for all instruments and all portfolios – NOW!

## IBM Platform Symphony

- A heterogeneous grid management platform
- A high-performance SOA middleware environment
- Supports diverse compute & data intensive applications
  - ISV applications – Many applications in this space are open source
  - In-house developed applications (C/C++, C#/.NET, Java, Excel, R etc)
  - Support for Linux / Power Linux, Windows + other OS
- React instantly to time critical-requirements
- A multi-tenant shared services platform
- Implements a fully compatible MapReduce run-time for open-source Hadoop



# Hadoop MapReduce

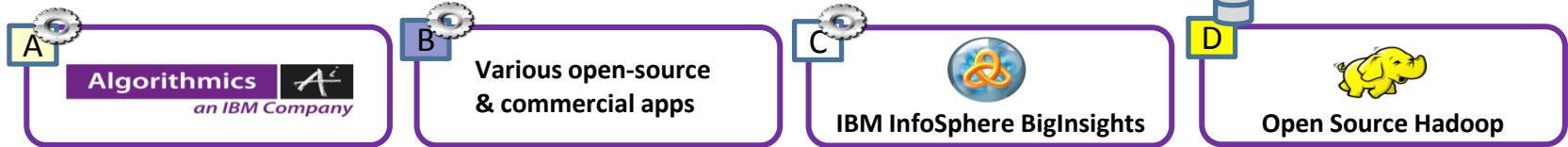


## De-facto "Big Data" standard

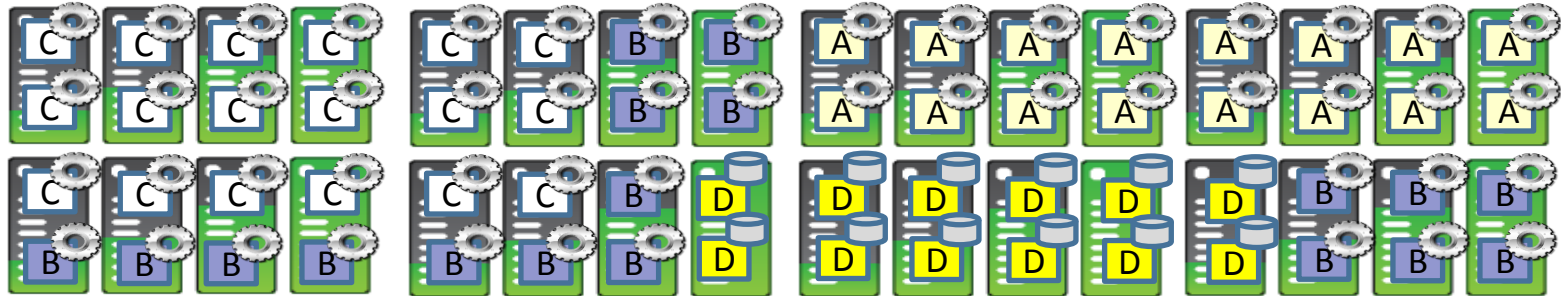
- Pioneered at Google / Yahoo!
- Framework for writing applications to rapidly process vast datasets
- More cost effective than traditional data warehouse / BI infrastructure
- Dramatic performance gains
- Java based
- From our perspective: *Just another distributed computing problem*



# IBM Platform Symphony

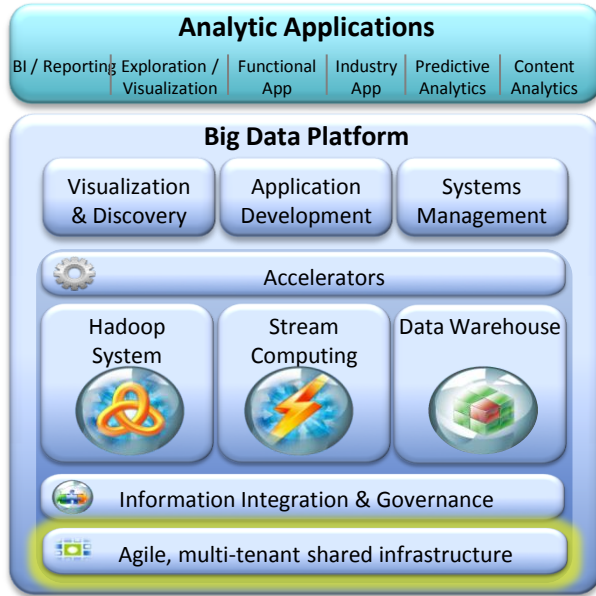


## Workload Manager

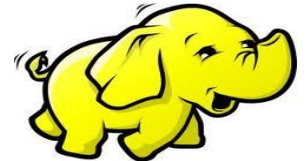


## Resource Orchestration

## IBM InfoSphere BigInsights & Platform Symphony



- Comprehensive platform
- Data at rest, data in motion
- Extensive library of data connectors
- Rich development tools
- Application accelerators
- Web-based management console



# Big problems demand big infrastructure

- **Exploit threads**
  - Power 7+ - 2 threads per core vs. 2 threads per core
- **High Throughput**
  - Extreme memory and I/O bandwidth
- **Better Java implementation**
  - Optimized JVM on Power 7+
- **Superior I/O**
  - Massive I/O bandwidth
- **Parallel file system**
  - Your choice of HDFS or GPFS
- **Ideal match for Apache Hadoop MapReduce framework**
  - Massively parallel processing across Linux clusters

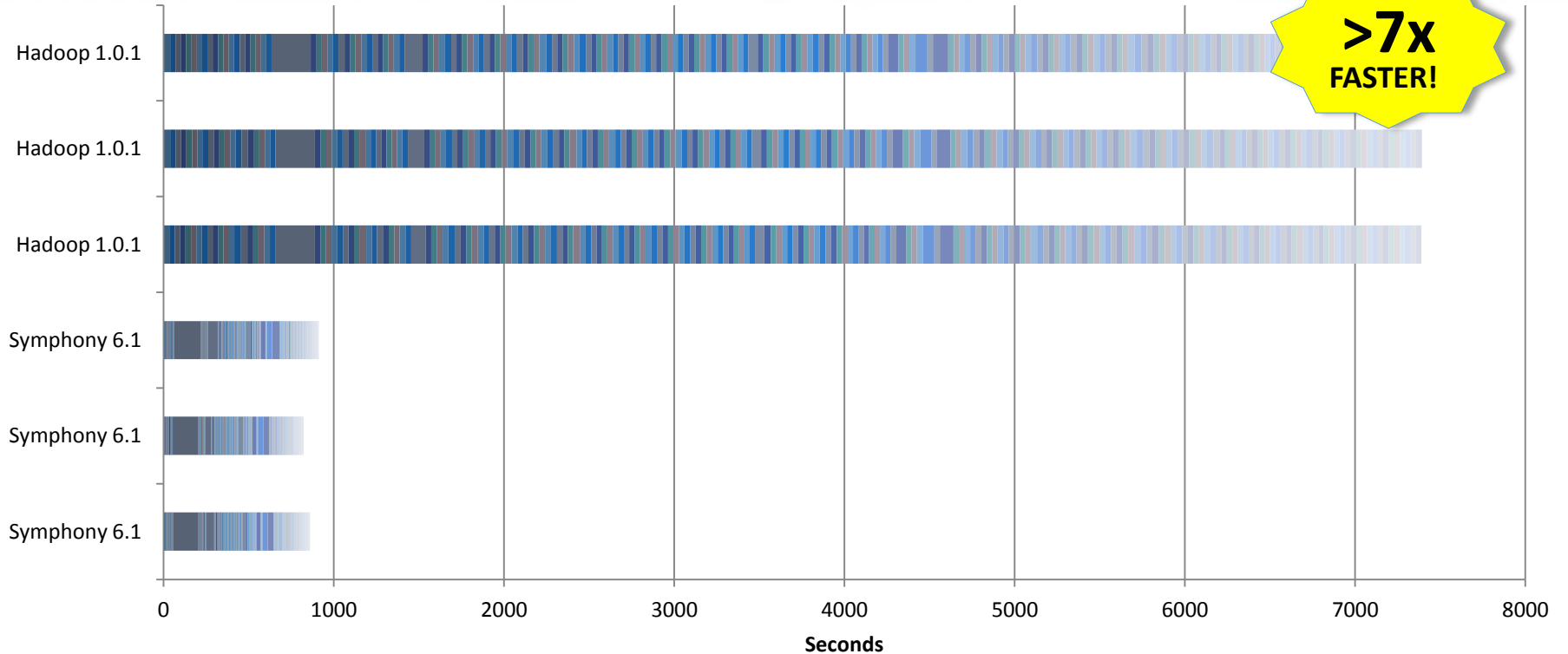


# PERFORMANCE

## Berkley SWIM

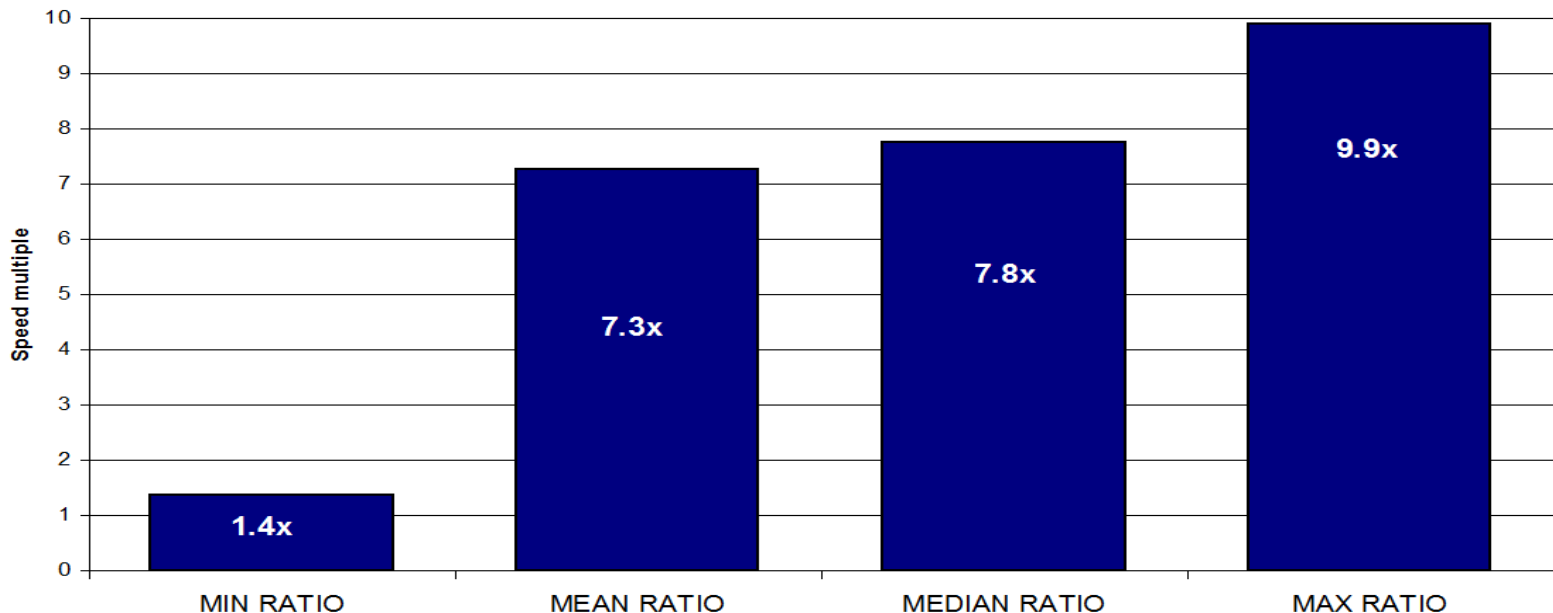
- “Real-world” MapReduce benchmark – synthesize and replay captured *real-world* workloads
- Developed by Yanpei Chen and others at @ UCB - <https://github.com/SWIMProjectUCB/SWIM/wiki>
- Viewed as an advance over existing synthetic MapReduce benchmarks including GridMix2, PigMix, Hive BM etc.
- Represents workloads comprised of short, large and huge jobs stressing disk, network IO, CPU and memory
- Promoted by Cloudera – advantages of SWIM promoted at Hadoop World 2011 - <http://www.slideshare.net/cloudera/hadoop-world-2011-hadoop-and-performance-todd-lipcon-yanpei-chen-cloudera>

## Benchmark: SWIM: Facebook 2010 Workload



## MapReduce/Symphony vs MapReduce/Hadoop

(Ratio of Symphony speed to Hadoop speed over 302 diverse MapReduce jobs)



Source: STAC®  
[www.STACresearch.com](http://www.STACresearch.com)  
Copyright © 2012 STAC

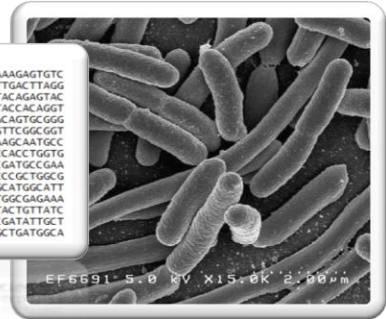
A subset of 302 jobs from Berkeley SWIM, generated from the Facebook 2010 trace.  
Ratio for each job = job duration using Hadoop / job duration using Symphony.  
Each job duration is the mean over three test runs.



## Benchmark: Contrail

- Open-source software for De Novo Genome Assembly – key contributors are Jeremy Lewi, Avijit Gupta, Ruschil Gupta, Michael Schatz and others
- Sequencing large genomes is too large a problem for conventional algorithms
- It turns out that the *deBruijn graph* fundamental to genome sequencing is readily represented as key-value pairs – ideal for processing with MapReduce
- Contrail runs a pipeline where each pipeline stage is implemented as a MapReduce job to exploit parallelism

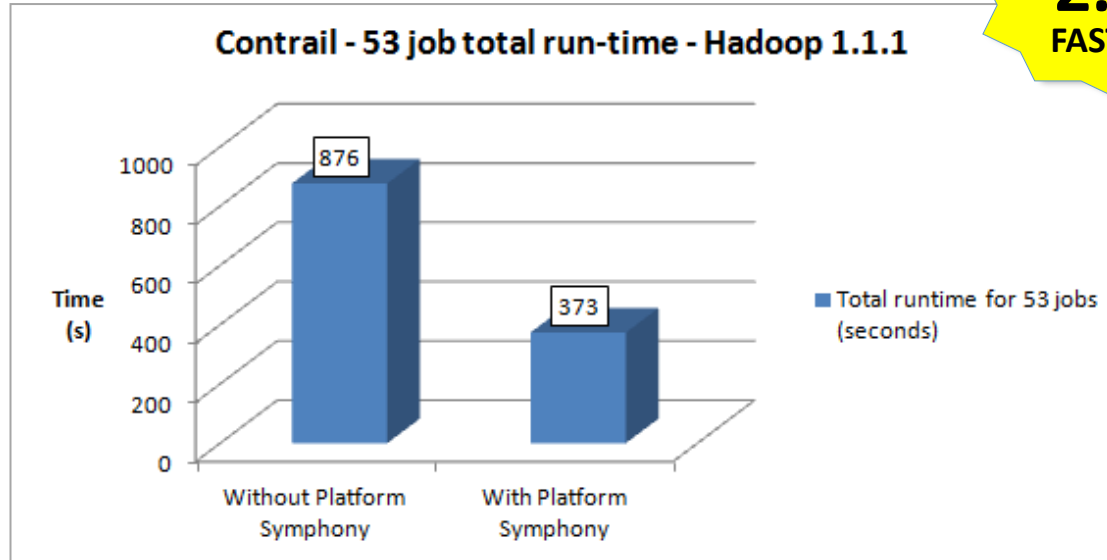
```
pORDQJRCMOIHDA | 101=9998 | cov=14.03
AGCTTTCCATTCTGACTGCTAAGCGGGCAATAGTCTCTGTGTGGATTAATAAAAAGAGTGTG
TGATAGCAGCTTCTGAACGGTGTACCGCCGTGAGTAAATTAATAATTTATTGACTTAGG
TCACATAAATACTTTAACCAATATAGGCATAGCGCACAGACAGATAAAAAATTACAGAGTAC
ACACATCCATGAAACGCAATTAGCACACCATTACACACCACATCCATTACACAGGT
AACGGTCCGGGGTGACGCCATACAGGAACACAGAAAAGCCCGCACCTGACAGTCCGGG
CTTTTTTTTTCGACAAAGGTAACGAGGTAAACCAATGCGAGGTGTGAAGTTCGGCGGT
ACATCAGTGGCAAAATGCGAGAACGTTTTTCGCGTGTGGCCGATATTCGGAAAGCAATGCC
AGGCAAGGGCCAGGTGGCCACCCTCTCTCTGCCCCGCCAAAATCACAAACACCTGGTG
CGATGATTAATAAAAACATTAGCGGGCCAGGATGCTTTACCCAAATACAGGATGCCGAA
CGATTTTTTGGCCGAACTTTTGACGGGACTCGCCGCCGCCAGCCGGGGTCCCGCTGGCG
CAATTGAAAACCTTTCGTCGATCAGGAATTTGCCAAATAAAMCATGTCCTGCAATGGCAAT
AGTTTTTGGGGCAGTCCCGGATAGCATCAACGCCGCTGATTTGCCGTGGCGAGAAA
ATGTCGATCCCATTAATGGCCGGCGTATTAGAAGCGCCGGTCAACACGTTACTGTTATC
GATCCGGTCAAAAACCTGCTGGCAGTGGGGCATTACCCTCAATCACAGTCCGATATTTGCT
GATGCCACCGCCGATTTGGCGCAAGCCGCATTCGGCTGATCACATGGTCTGATGGCA
```





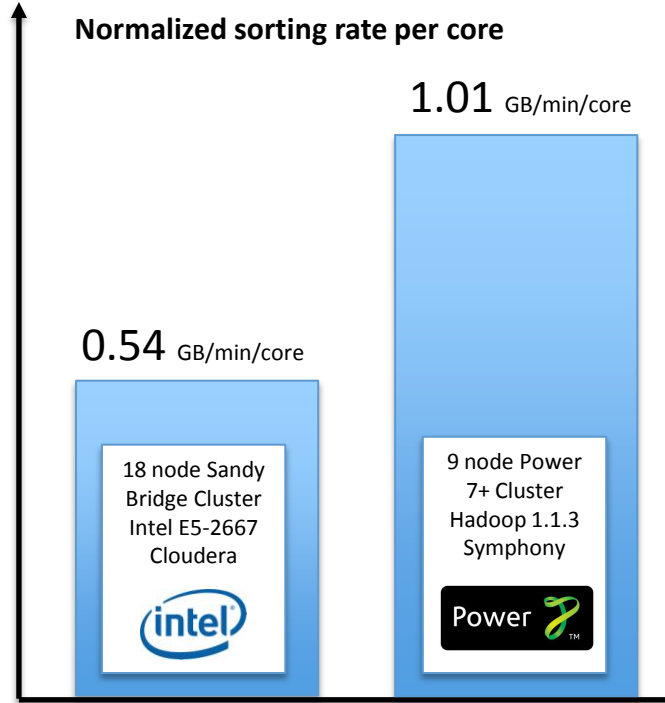
## Benchmark: Contrail

**2.3x**  
**FASTER!**



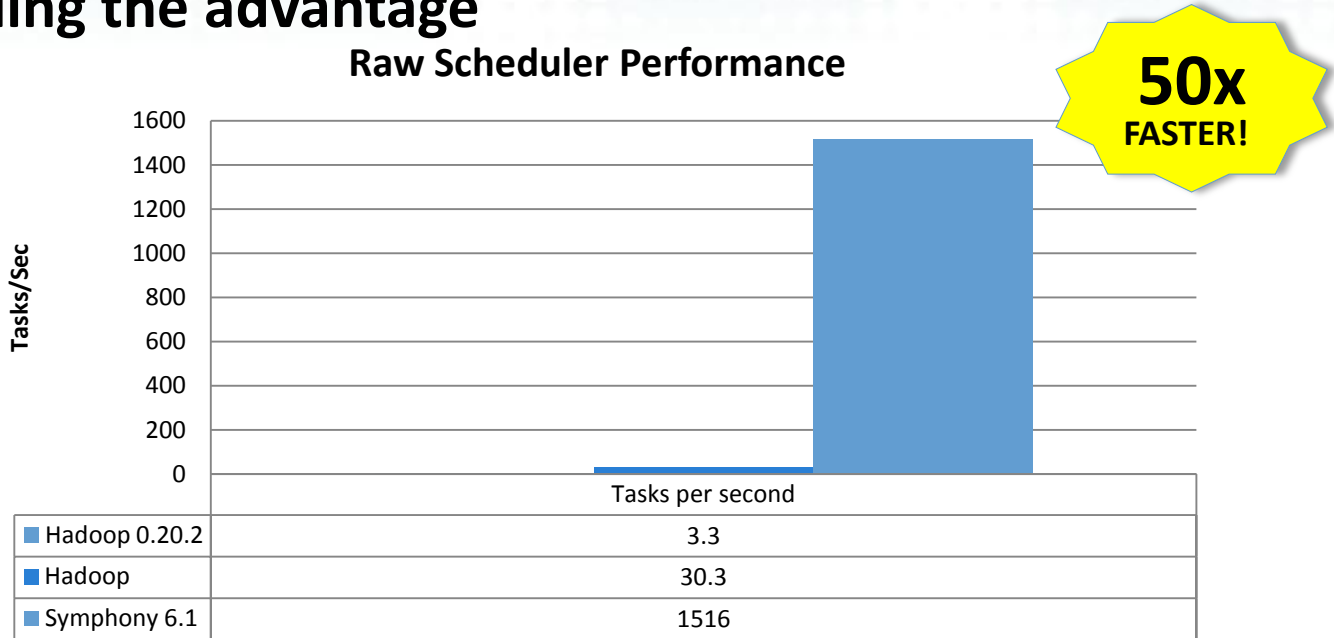
# Record Terasort results on Power 7+

Hardware	
Cluster	1 PowerLinux P7+ Master Node 9 PowerLinux P7+ Slave Nodes
CPU	16 processor cores per server (128 total)
Memory	128 GB per server (1280 total)
Internal Storage	6 600GB internal SAS drives per server (36 TB total)
Storage Expansion	24 600GB SAS drives in IBM EXP24S SFF Gen2-bay Drawer, per server(144 TB total)
Network	2 10Gbe connections per server
Switch	BNT BLACE RackSwitch G8264
Software	
OS	Red Hat Enterprise Linux 6.2
Java	IBM Java 64bit Version 7 SR1
HDFS	Hadoop v1.1.3 (1 node as NameNode and 9 nodes as DataNode)
Platform Symphony MapReduce	Advanced Edition 6.1.0.1 1 node as Management Host and 9 nodes as Compute Hosts



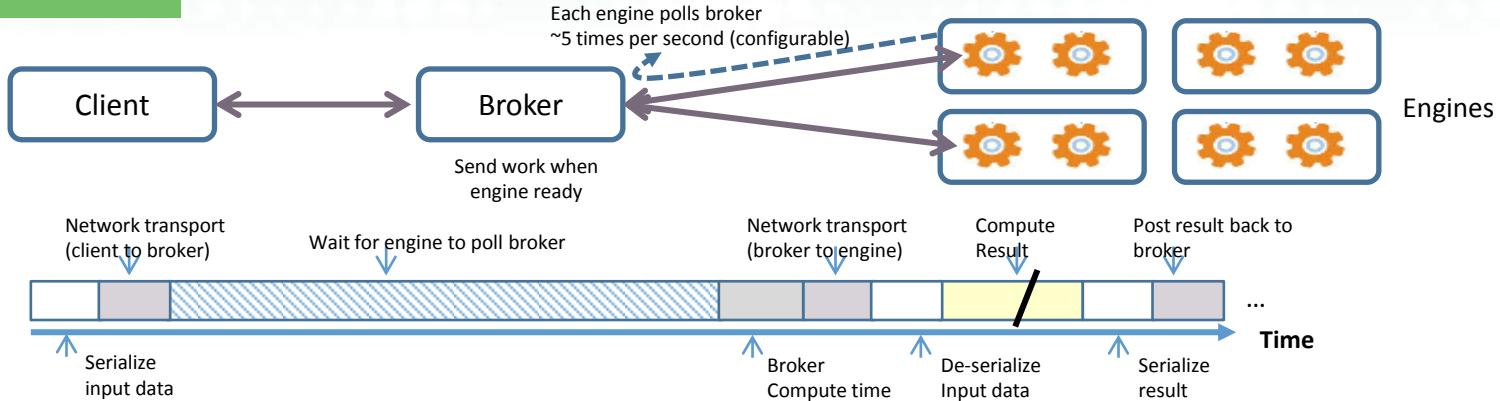
# Understanding the advantage

## Raw Scheduler Performance



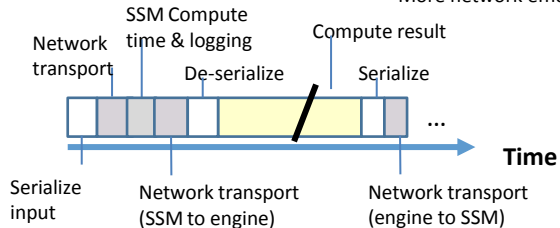
- Symphony 6.1 can schedule ~50x more tasks per second
- Hadoop results taken from Hadoop World 2011 performance presentation, Lipcon & Chen

## Other Grid Server



## Platform Symphony

No wait time due to polling, faster serialization/de-serialization, More network efficient protocol



### IBM Platform Symphony is (much) faster because:

- Efficient C language routines use CDR (common data representation) and IOCP rather than slow, heavy-weight XML data encoding
- Network transit time is reduced by avoiding text based HTTP protocol and encoding data in more compact CDR binary format
- Processing time for all Symphony services is reduced by using a native HPC C/C++ implementation for system services rather than Java
- Platform Symphony has a more efficient "push model" that avoids entirely the architectural problems with polling

## Many performance optimizations

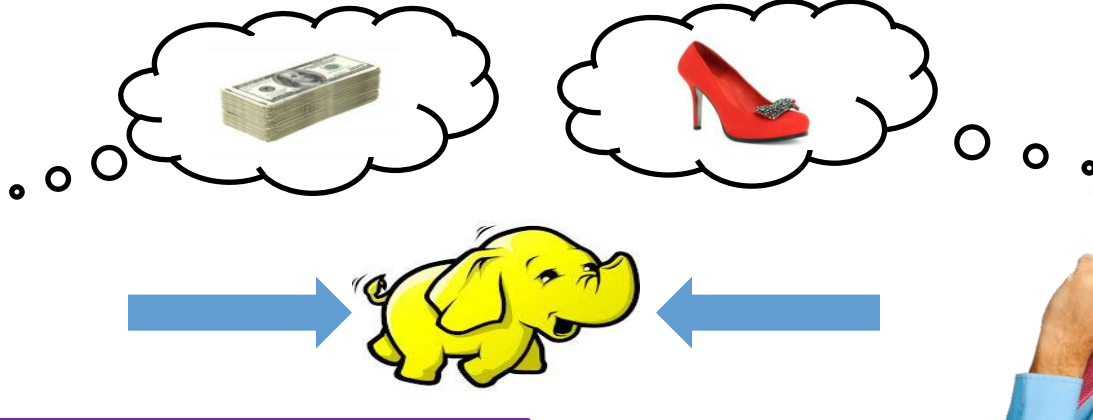
- C++ native code
- Optimized binary network protocols
- Fast object serialization
- JVM pre-start & re-use
- Generic slots enabling full cluster utilization
- Efficient push-based scheduling model
- Uses Symphony common data for JAR transport
- Shuffle-stage optimizations
- Intelligent pre-emption



# MULTITENANCY

---

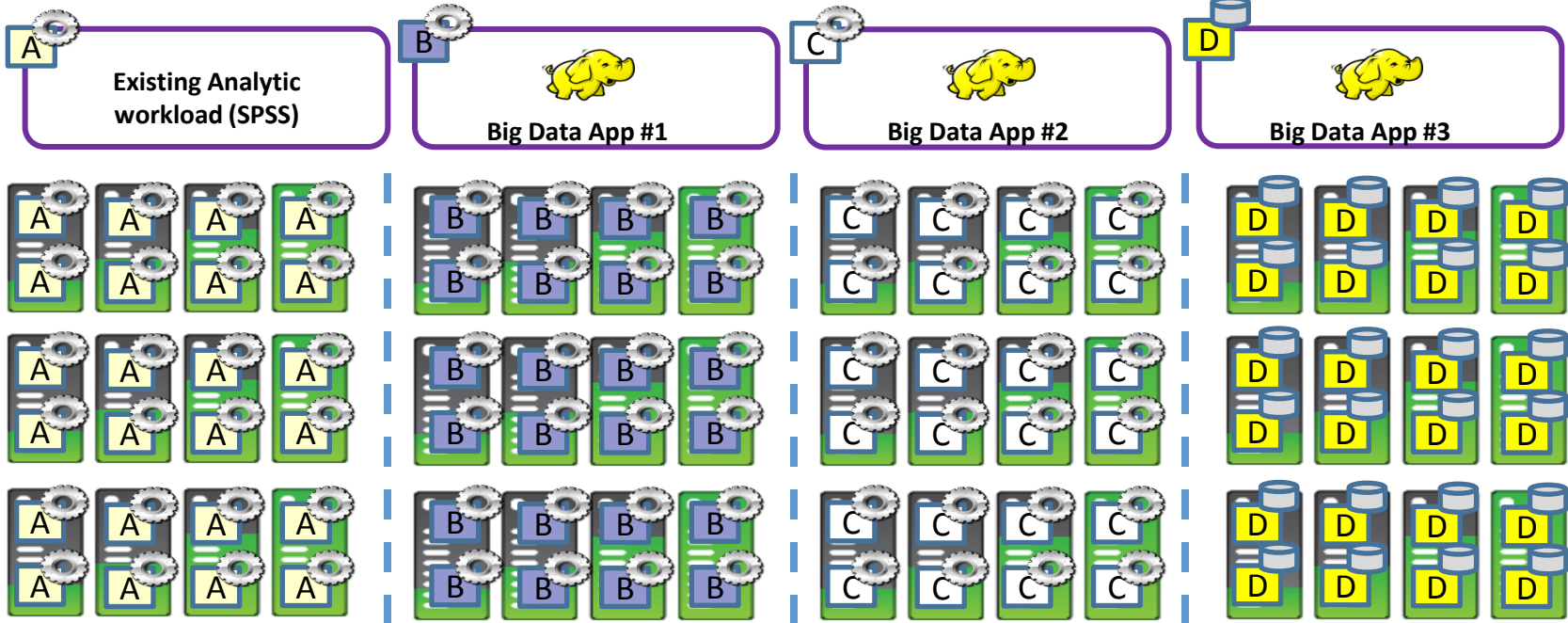
## Different workloads demand different SLAs



*“I need an updated counterparty credit risk analysis for the final earnings report by 2:00 pm”*

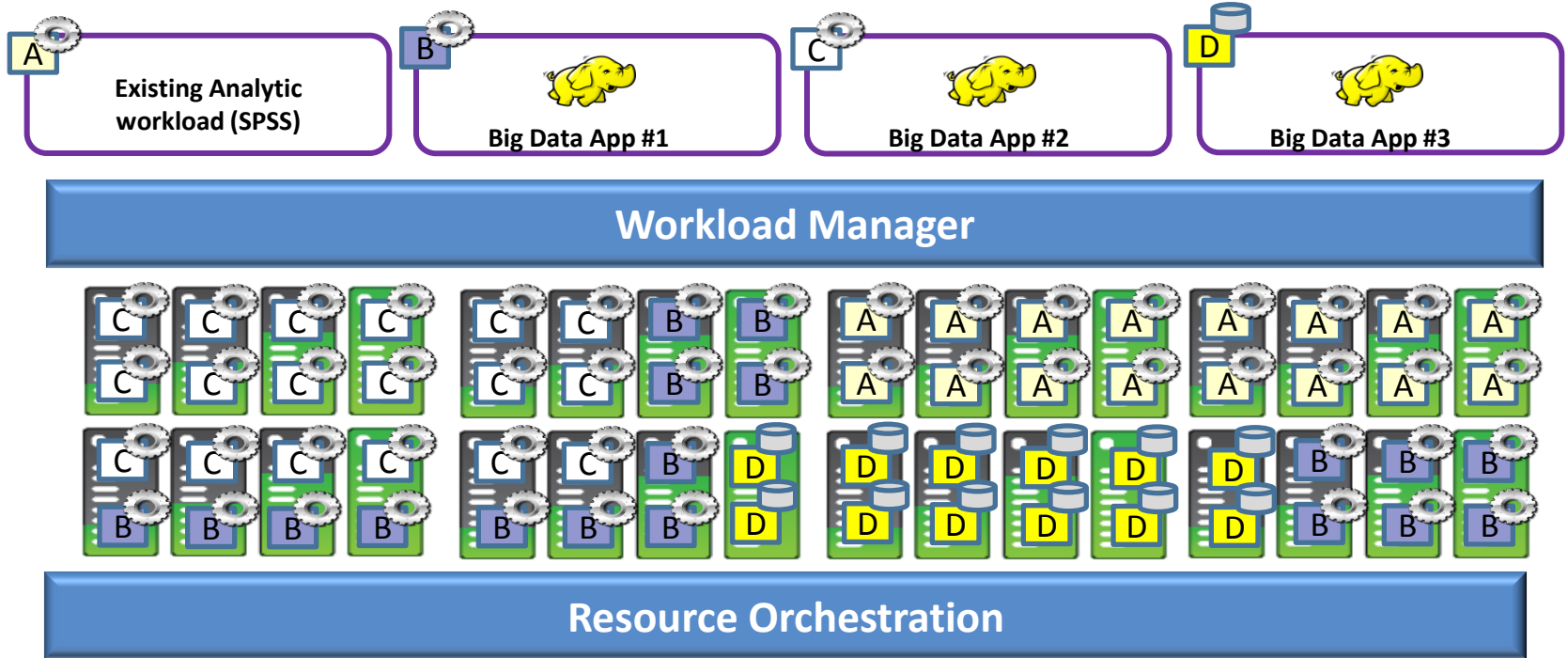
*“I wonder if teenagers in California still think red shoes are cool?”*

## Cluster Sprawl - Silos of underutilized, incompatible clusters





## Dynamic resource sharing among heterogeneous tenants



# Ensuring SLAs is critical

Consumers Resource Distribution Plan

Resource Group: Management Time Intervals and Settings

00:00 18:00

Ownership

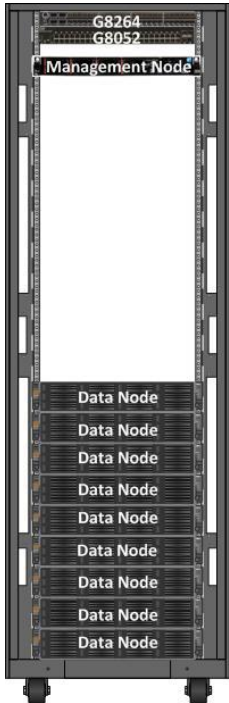
Consumer	Owned Slots	Consumer Rank	Lend	Limit	Borrow
demo2	24				
System	10	10			
Extended	4	70	<input type="checkbox"/>		
Standard	6	70	<input type="checkbox"/>		
<b>Total</b>	<b>10</b>	-	-	-	-
<b>Balance</b>	<b>0</b>	-	-	-	-
Application	10	5			
Payroll	1	30	<input type="checkbox"/>		
SymSOATest	3	30	<input type="checkbox"/>		
StoreFront	2	30	<input type="checkbox"/>		
Risk	1	30	<input type="checkbox"/>		
Tibco	3	30	<input type="checkbox"/>		
<b>Total</b>	<b>10</b>	-	-	-	-
<b>Balance</b>	<b>0</b>	-	-	-	-
<b>Total</b>	<b>20</b>	-	-	-	-
<b>Balance</b>	<b>4</b>	-	-	-	-

Agile sharing at run-time while preserving ownership and application SLAs

Cluster partition usage over the last week

Report period: from 11-19 00:00 to 11-26 00:00

## Multiple deployment options

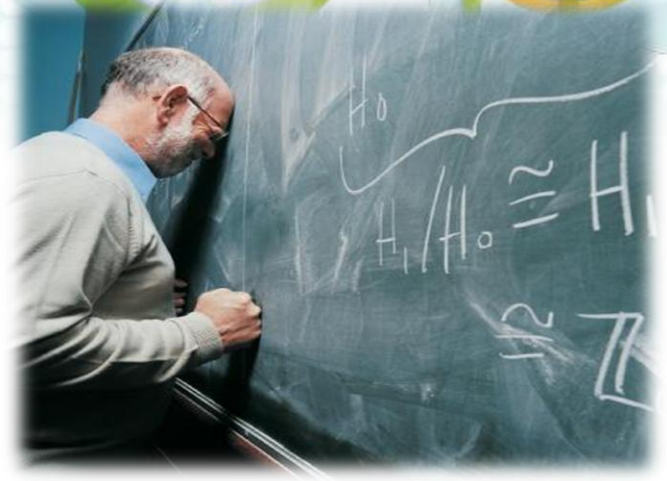


- Pure Data for Hadoop appliances
- Power and Intel based Big Data Reference Architectures
- Your choice of distribution
  - IBM BigInsights, Cloudera, MAPR, Apache, Hortonworks etc..*
- Your choice of file system
  - HDFS or GPFS*

## Summing up

A unique solution for open-source Big Data Analytics

- Exceptional performance
- Lower infrastructure cost
- Multiple Hadoop distributions
- Simplified application life cycle management
- Sophisticated multi-tenancy
- Optional GPFS file system



## Next steps

- Review the benchmarks
- Take the TCO challenge
- Contact us



**IBM Platform Symphony**  
Total Cost of Ownership Calculator

**Infrastructure**

Number of servers in the cluster

Average cost per server

Annual power costs per server

**Application Environment**

Number of Compute Intensive applications

Number of big data / Hadoop applications

**Additional Application Detail**

Percentage of jobs with short-running tasks

Efficiency gain for short-running tasks

Efficiency gain for long-running tasks

**Personnel**

System administrators

**Business Assumptions**

Cluster growth rate per year

**IBM Platform Symphony can provide significant savings for organizations deploying distributed applications and big data analytic workloads. Symphony is usually more cost efficient for the following reasons:**

- The low-latency scheduling and middleware processes tasks faster meaning that less hardware is required to meet performance goals
- Symphony's sophisticated resource sharing policies enable multiple departments, applications and users to share a common grid reducing the amount of infrastructure investment required
- Because Symphony can drive much high resource utilization than competing grid managers, the rate of cluster growth can be slowed, further reducing infrastructure, power and facilities costs.

The results of our high-level analysis are shown below. You can request that a more detailed report be sent to you by e-mail.

**Platform Symphony saves \$8,031,876 over three years.**

	EOY 1	EOY 2	EOY 3
Other Grid Manager	\$5,808,663	\$6,725,101	\$7,684,427
IBM Platform Sympt			
IBM Platform Sympt			

**IBM Sym:**

46  
42  
38  
34  
30  
EOY

<http://www.ibm.com/platformcomputing/products/symphony/>

<http://www.ibm.com/platformcomputing/products/symphony/highperfhadoop.html>

<http://www-03.ibm.com/systems/power/software/linux/powerlinux/>

<http://bigdatauniversity.com>

**Hadoop on the IBM SmartCloud Enterprise**



Learn to create your own Hadoop cluster on the IBM SmartCloud Enterprise with this **FREE** course.

**Enroll now!**

**Gord Sissons - gsissons@ca.ibm.com**

