THE IMPLICATIONS OF DIVERSE AND SCALABLE DATA SETS AND APPLICATIONS IN BENCHMARKING BIG DATA SYSTEMS

*Zhen Jia, Lei Wang,Wanling Gao，***Jianfeng Zhan***, Lixin Zhang*

Institute of Computing Technology, Chinese Academy of Sciences

http://prof.ict.ac.cn/jfzhan

中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

Some question

Consider a printer (***Its behavior is simple***):

- Printing 2000 pages consumes 50 seconds
- what about 4000 pages?

2000 pages

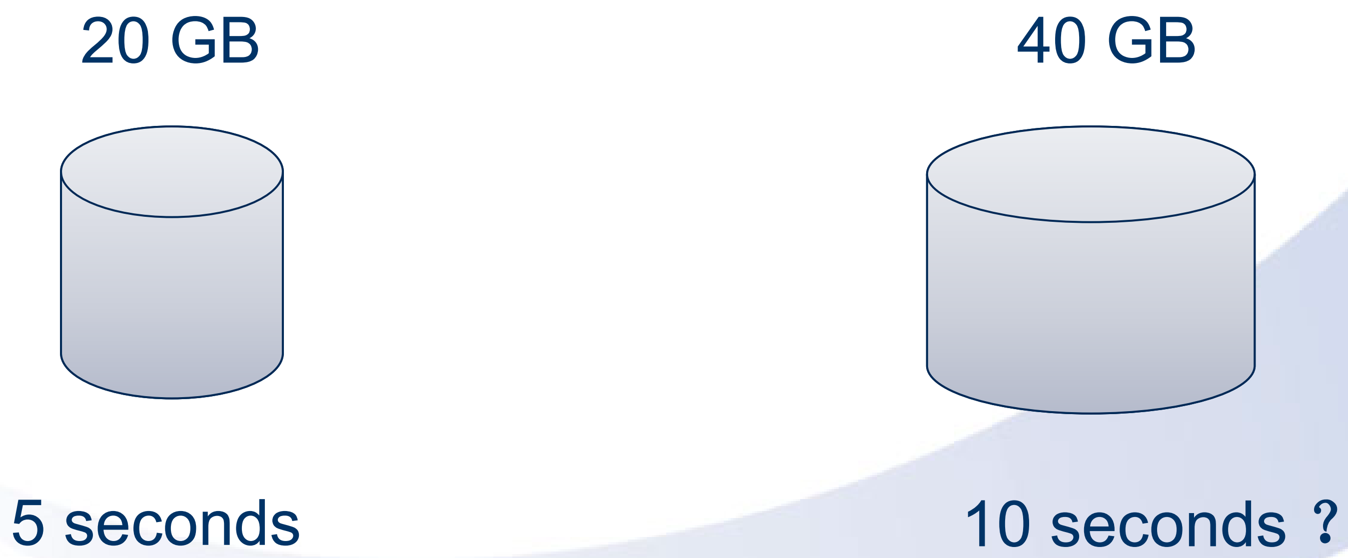
4000 pages

50 seconds

100 seconds

Some question

- If an algorithm (O(n)) processes 20GB data with 5 seconds
- What about 40GB ?
  - 10 seconds ? Why?



Benchmark for big data: State of practice

- MinuteSort, JouleSort, TeraByte Sort
  - Only one application
    - One-fits-all solution ?
    - Can it represent thousands of applications in big data field?
  - Fixed data scale
    - reflects the data processing capability using fixed data scale.
    - Other scales?

?
=

Sort

中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

Our observations

- Different applications have different sensitivity to big data
  - The sensitivity of applications to data scales must be considered when evaluating big data systems
- A single user-observed metric is not enough
  - Varied from different algorithms, data sets, and scales.

Four Workloads

- Sort
  - A representative I/O-intensive application
  - Use the MapReduce framework to sort the sequence files within a directory
- Word Count
  - A representative CPU-intensive application
  - It reads text files and counts how often words occurred

Four workloads (cont.)

- Grep
  - Frequently used in data mining application
  - Extract matching strings from text files and count how many times they occurred
- Naïve Bayes
  - A simple probabilistic classifier which applies the Bayes' theorem with strong(naïve) independence assumptions

Computing complexity of four workloads

- The computing complexity
  - **Sort** : O(n*lgn)
  - **Word Count** and **Grep**: O(n)
  - **Naïve Bayes**: O(m*n)   [m: the length of dictionary]

- A user-perceived metric MB/second
  - The ratio of a specified data scale to **its running time**

Data processing capability in theory

- The data processing capability
  - Sort : n/(n*lg(n))=1/lg(n)
    - **Decrease in theory**
  - Word Count、Grep、Naïve Bayes
    - Unchanged in theory

Configuration

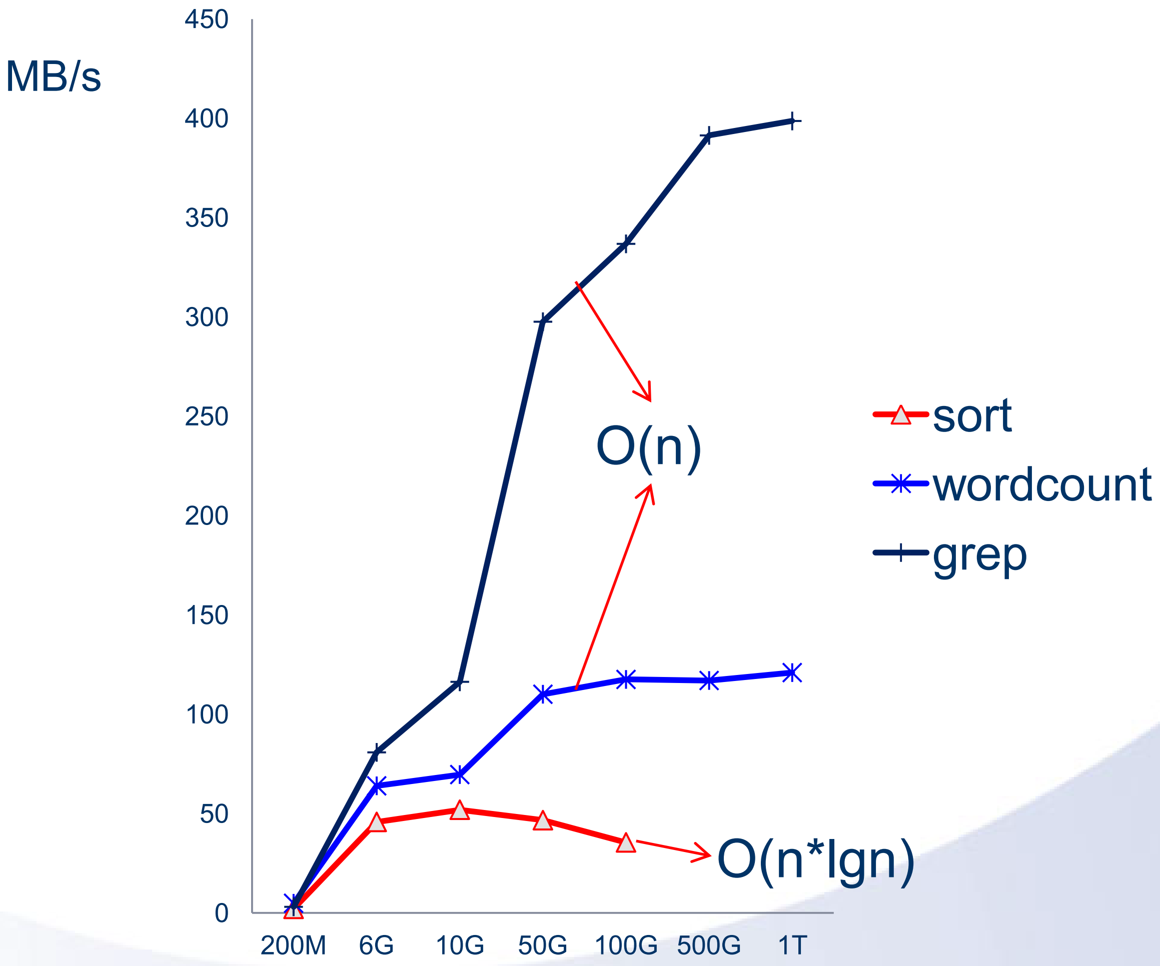
- 5-Node cluster
  - 2*Xeon E5645(6 cores with hyper-thread)
  - 16 GB MEM
  - 8 TB Disk

CPU Type	Intel CPU Core
Intel ®Xeon E5645	6 cores@2.40G

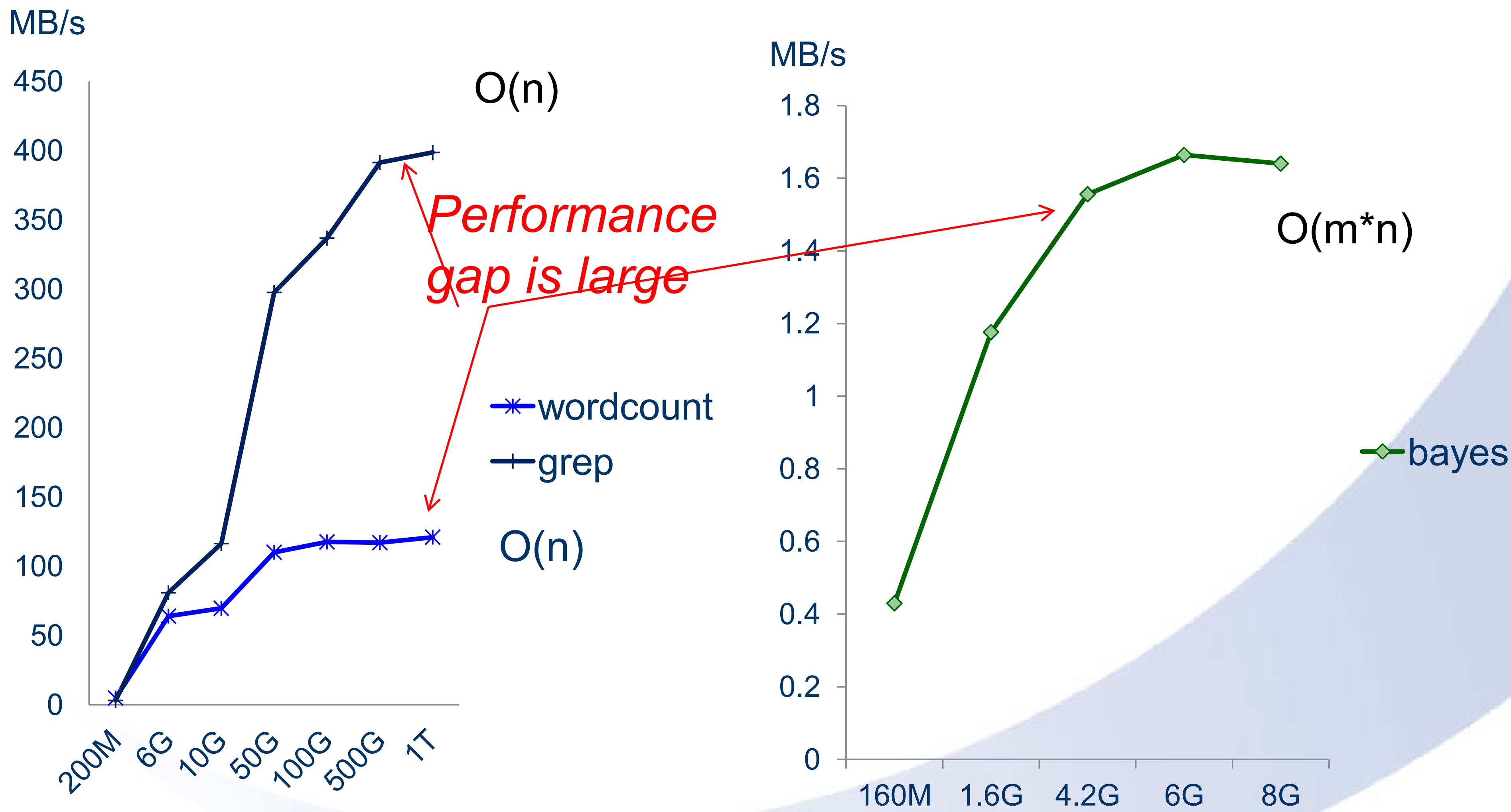
L1 DCache	L1 ICache	L2 Cache	L3 Cache
6 × 32 KB	6 × 32 KB	6 × 256 KB	12MB

- Hadoop Cluster
  - 1 master, 4 slaves
    - 24 map slots and 24 reduce slots per slave
  - Threshold size: the data amount simultaneously processed
    - 24*4* 64 MB (a data split)= *6 GB*

Performance metrics for applications with different complexity

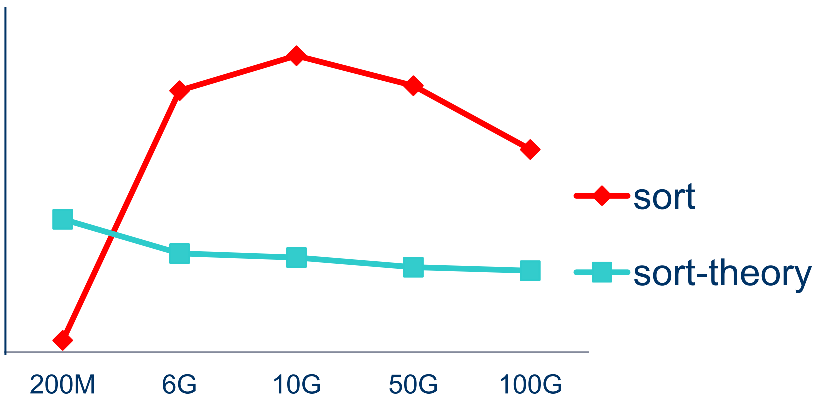


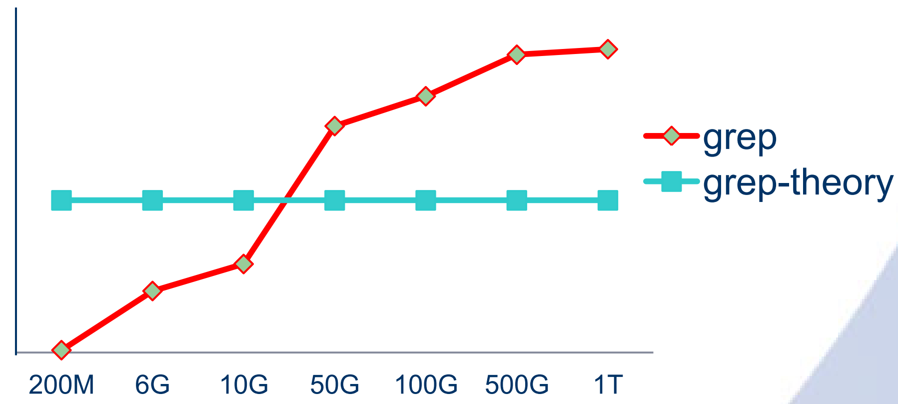
Metrics for applications with O(n) complexity

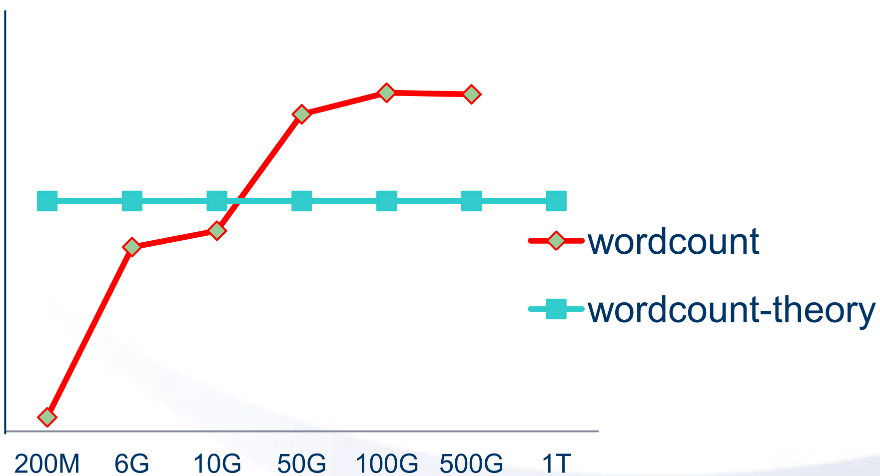


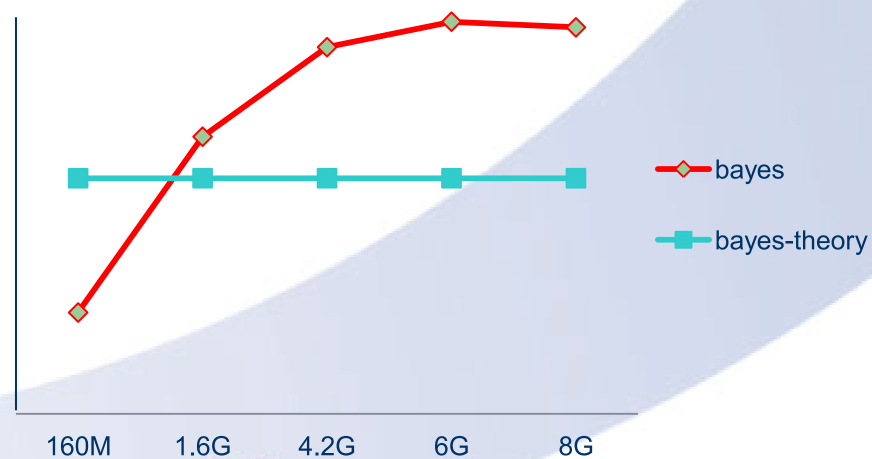
Reported value vs. performance trend in theory

Data processed per second

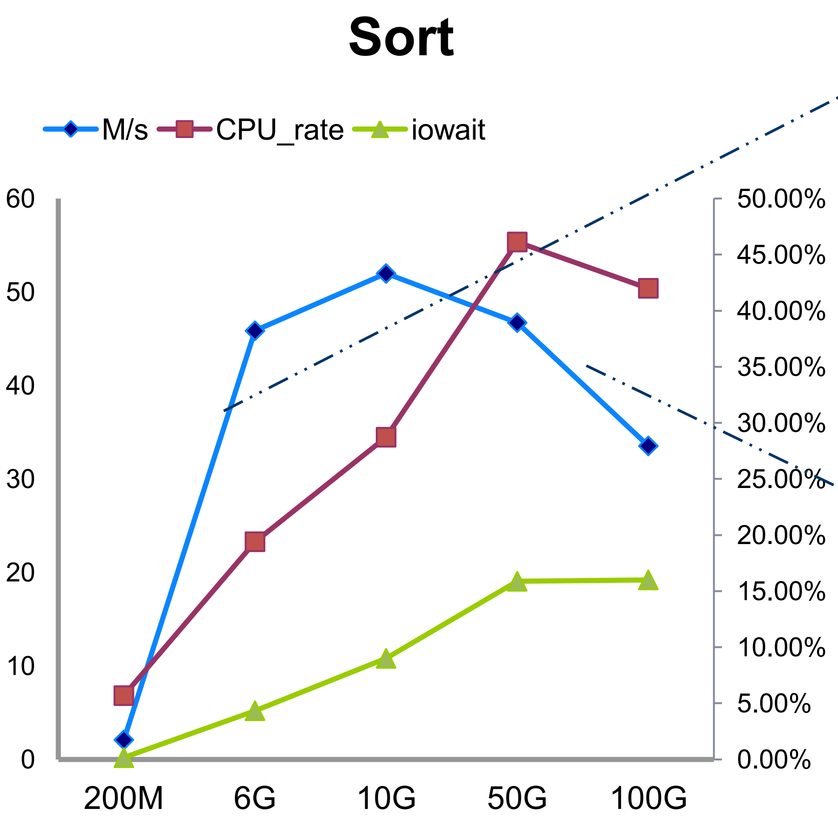






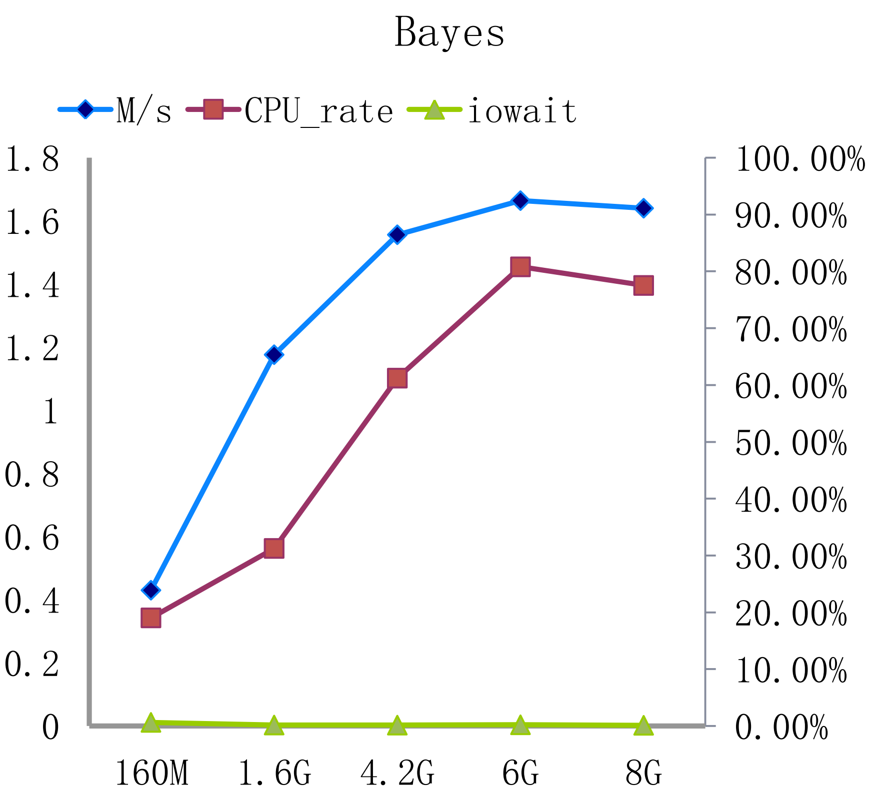


Preliminary analysis results



- Increasing phase
  - The data processing capability increases with data scale increases
  - CPU utilization goes up
    - using the hadoop slots more efficiently
- Decreasing phase
  - **Overloaded** situation
  - I/O wait increases intensively
  - Page fault increase

Preliminary analysis results



- The Bayes's programming logic is much complex than other three's
  - Multi-stage
  - O(m*n), m is large
- Iowait is low.

中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

Lessons from above observations

- MinuteSort, JouleSort, and TeraByte Sort have their limitations.
  - Only reflects the data processing capability in terms of a specific application with a fixed data scale.
- A **single** user-observed metric is not enough
  - **Varied from different algorithms, data sets, and scales.**
- Different applications have different sensitivity to big data
  - The sensitivity of benchmarks to data scales must be considered when evaluating big data systems

HPCA 2013 tutorial： HVC

High Volume Computing: The Motivations, Metrics, and Benchmarks Suite for Data Center Computer Systems

- A full day tutorial
- Topic:
  - What is High Volume Computing (HVC)
  - How to evaluate a data center computer system
  - HVCbench: HVC benchmarks

Feb 23,2013, Shenzhen, China.

中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

HVCbench

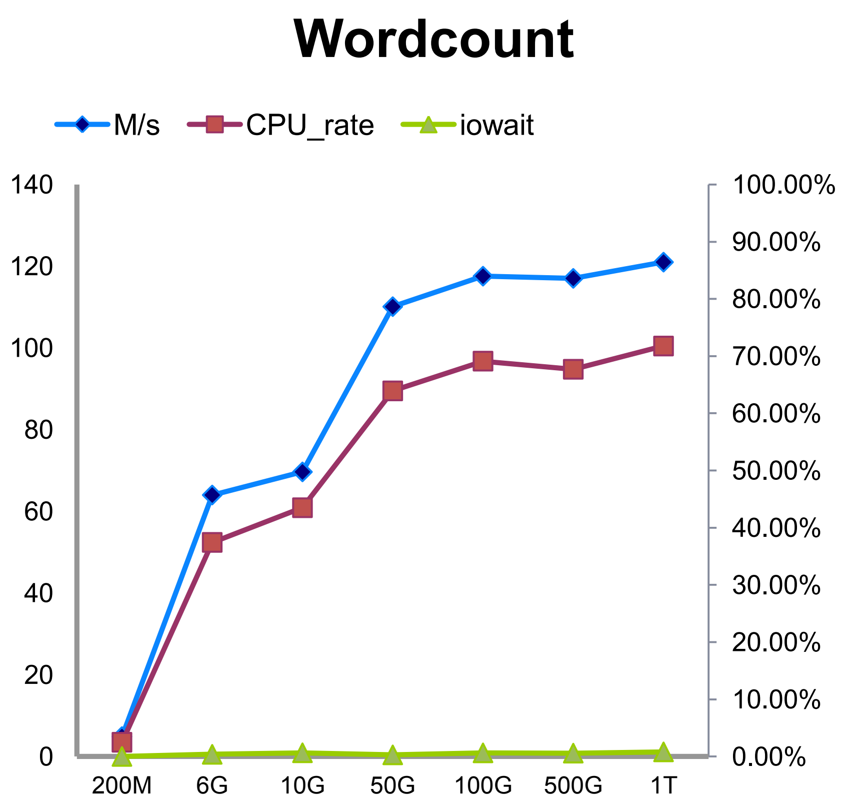
- A benchmark suite for data center workloads
  - 26 representative workloads
  - Six dwarfs
  - 4 programming models:
- Release soon on our web site http://prof.ict.ac.cn

Thank you!

http://prof.ict.ac.cn/jfzhan

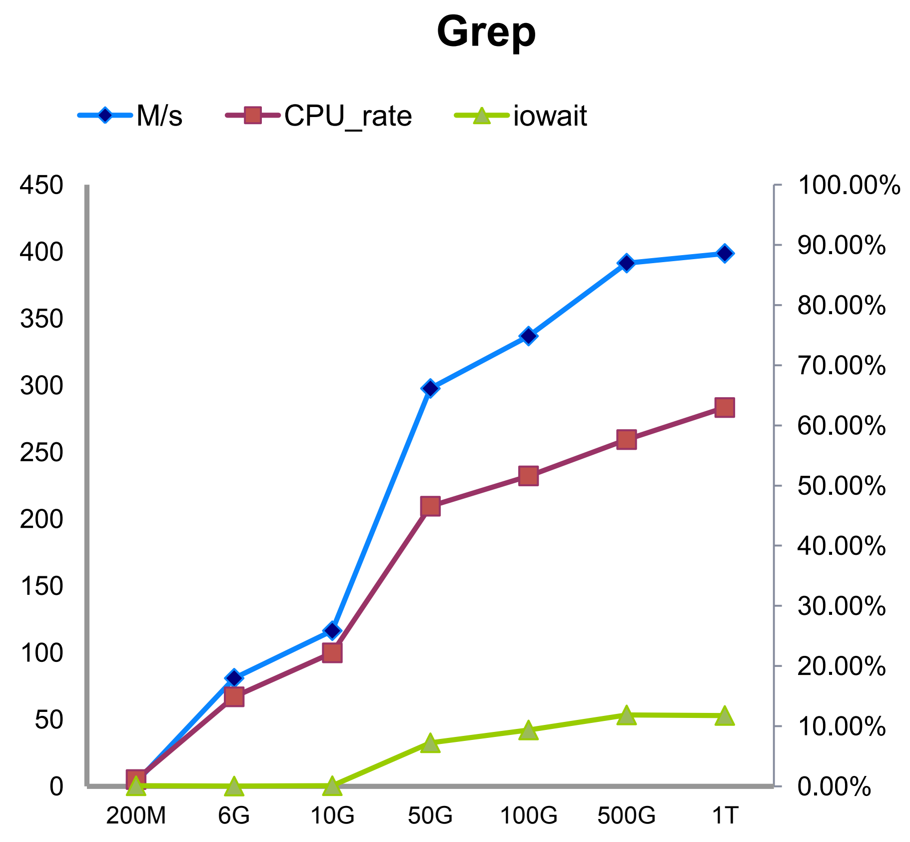
中国科学院计算技术研究所
INSTITUTE OF COMPUTING TECHNOLOGY, CHINESE ACADEMY OF SCIENCES

Different Workloads Processing Capability



- Increasing
  - The data processing capability increases with data scale increases
    - using the hadoop slots more efficiently
  - The CPU utilization goes up with the same trend
  - Less I/O wait ratio
- "threshold size"
  - Not in the range of our input data sets

Different Workloads Processing Capability



- Increasing
  - The data processing capability increases with data scale increasing
    - using the hadoop slots more efficiently
  - The CPU utilization goes up with the same trend
  - I/O wait ratio increases, but do not affect much
- "threshold size"
  - Not in the range of our input data sets