

MapReduce: Distributed Computing (the Google Way)

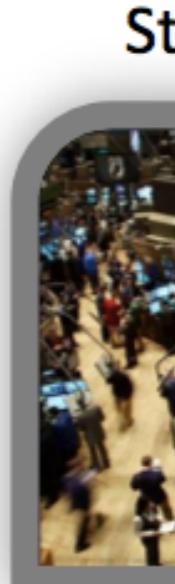
Assoc Prof. Krerk Piromsopa, Ph. D.
Department of Computer Engineering,
Chulalongkorn University

krerk.P@Chula.ac.th

Why do we care?

ການນິຍາມ big data ກ່ອນດູກຂະໜາຍດ້ວຍແລກ 5V

New York



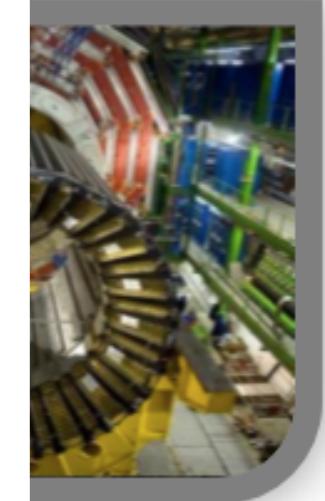
1 TE



Internet Archive



Hadron Collider
CERN



15 PB
year

sent across all
the world daily
everyday.



That is enough to fill...



1.7 million



9.2 million



63.9 trillion
diskettes

ចំណាំ ?

Cost/Performance

Server ក៏បែងការឱ្យគ្រប់

IBM NextScale Server M5

intel XEON (Up to 18 cores)

RAM 512Gb ៣១ core តម្លៃ 833\$



\$14,432.00 USD

ឧបតម្យលេខវិស័យ 32 / 64 មេដាច់

នៅរវាង memory bandwidth ទាំង ២ នៅក្នុង memory កំណត់ទៅរាយកើត

20-30 units

\$700.00 USD

ផ្ទា nuc
80 cores
RAM 320 Gb

Hardware vs. Software

IBM NextScale Server M5



មានប្រើបង្រៀន

- Hardware: Reliable
- Software: easy

សម្រាប់
code ក្រោម

Intel NUC farm



អ្វីមួយទៅ
តាមតម្លៃ network

- Hardware: Vulnerable
- Software : ????

នូវ នៅ

ເຕັມເວລາທີ່ໄດ້ໃຊ້ເຈັນ ໂກງ ແລະ ຢິພ ສຶກສອນຈະໜີ (Specific App)

Time to Product/Analysis

Before

ນາມ dev.
Development : weeks
Run: days/month

Today

Development : few days
Run: minutes

ນາມ mkt ກ່ຽວ 1 ນາທີ
ຕົວ suggest ໄນໂດຍ

ເປົ້າຂົວ
programming ດ້ວຍ mapReduce

Why MapReduce? Why not Grid/Cluster?

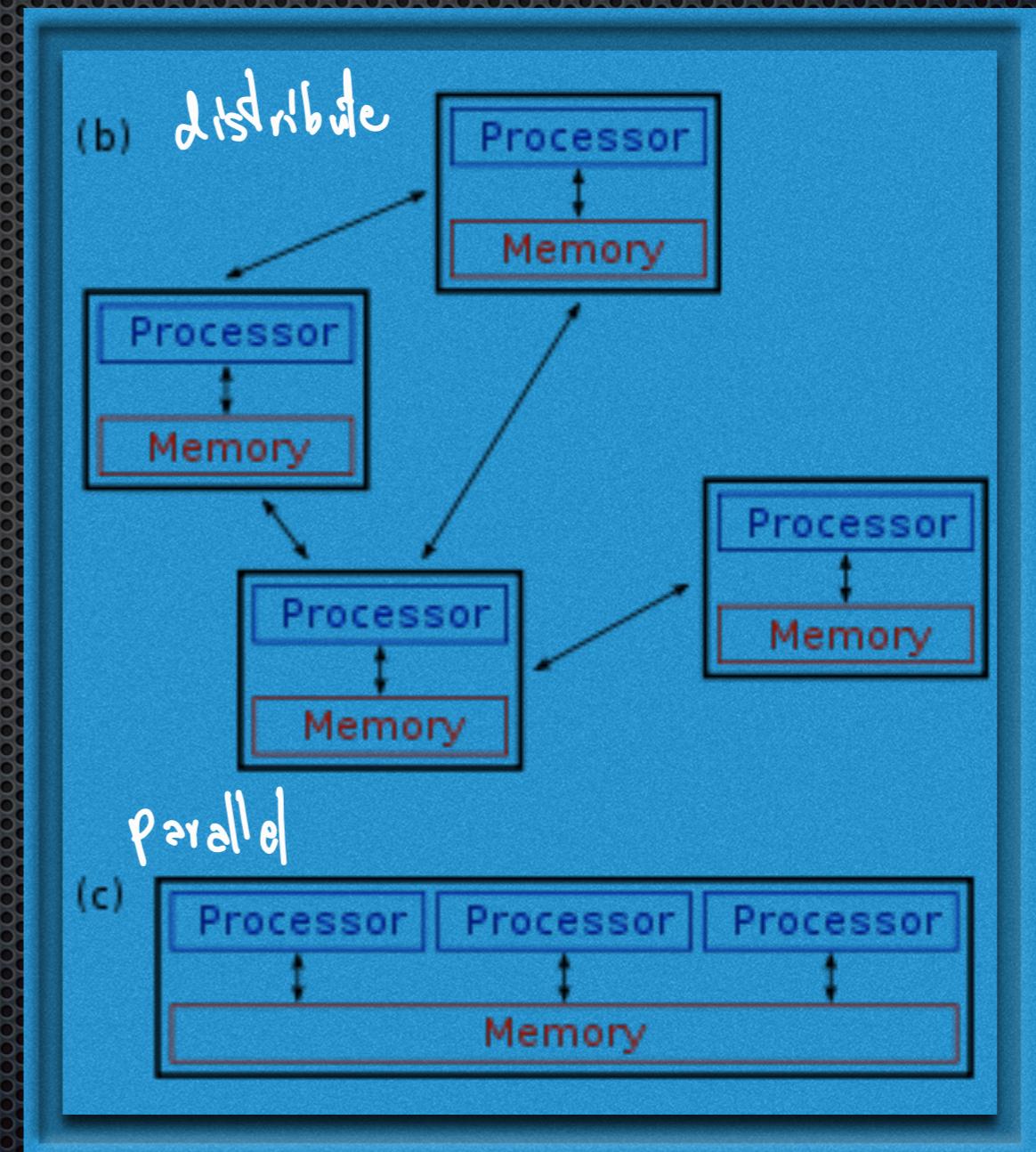
Google ໃນລົດ mapReduce ແຕ່ມີກຳນົດ practical

ແລ້ວສິນໃຈຫຼຸດ ຖໍ່ມີ Google ເນື້ອງໄມ້ໄກ້ພໍາໄລ

Distributed vs. Parallel Computing

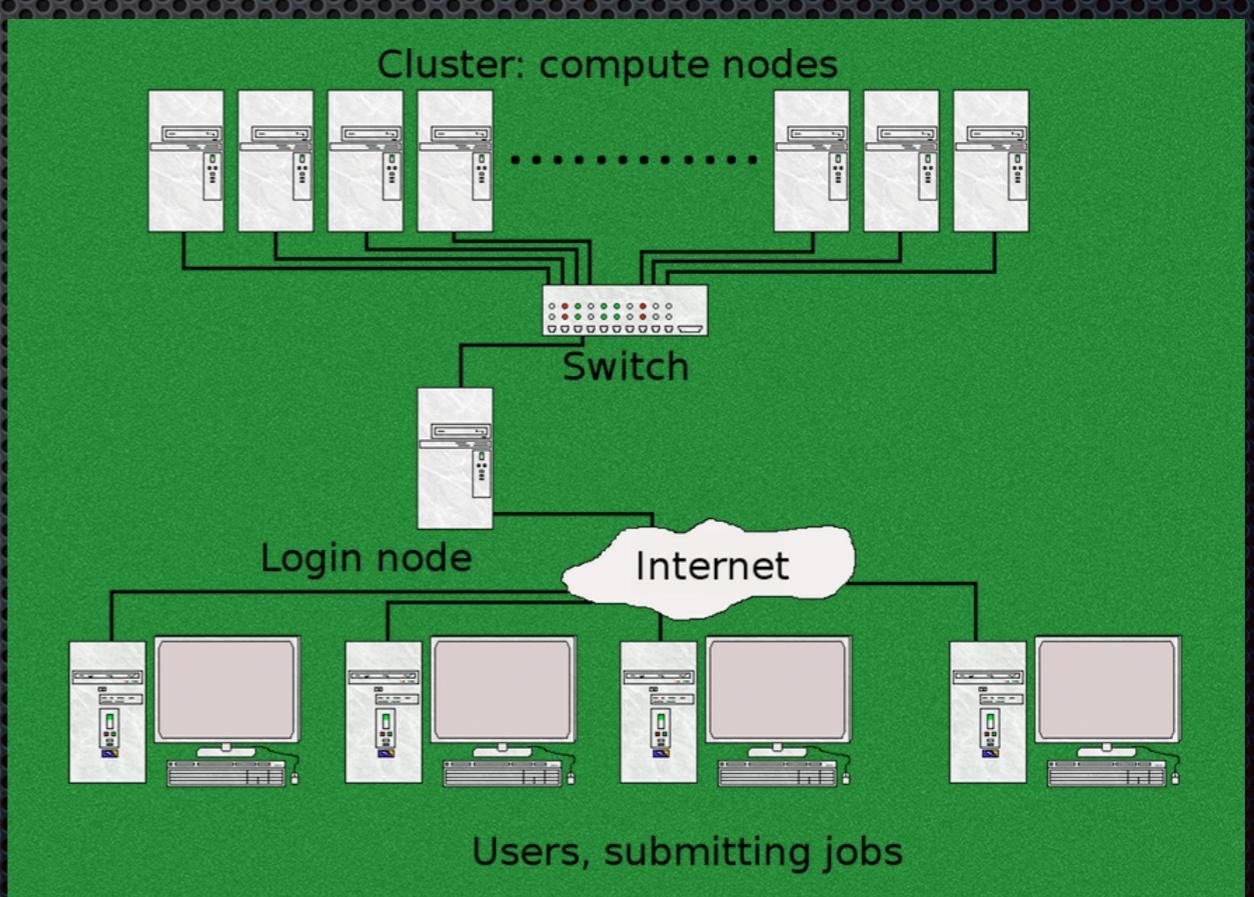
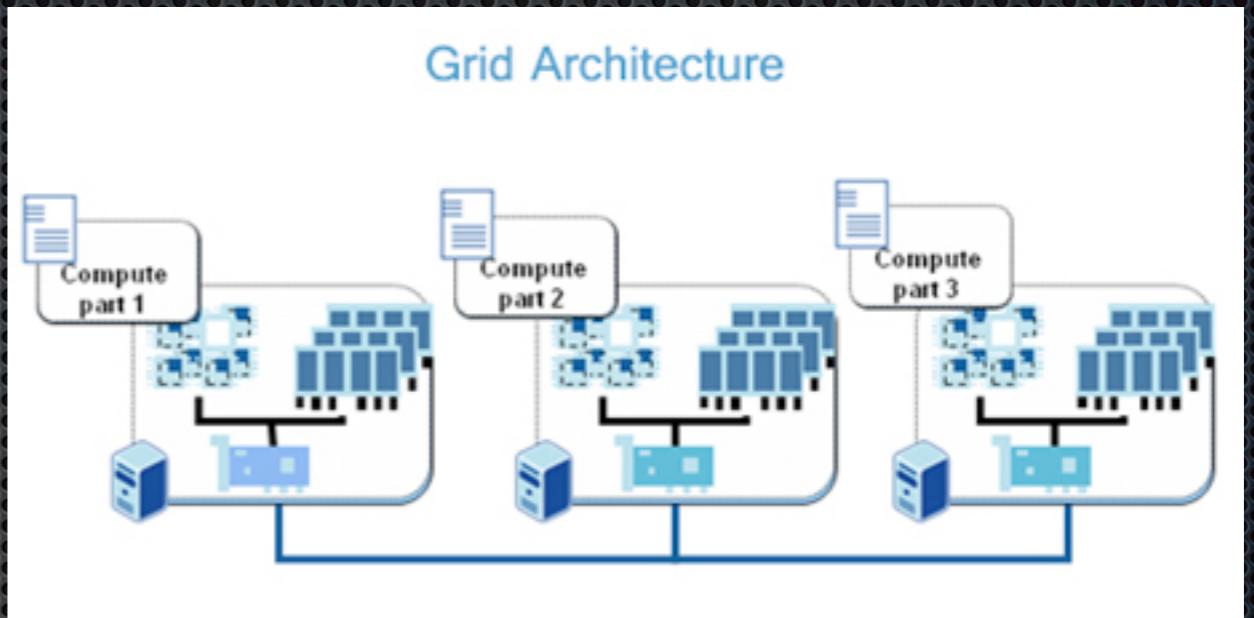
- Parallel Computing:
everything in memory
- Distributed Computing:
beyond one computer

នៅក្នុង network



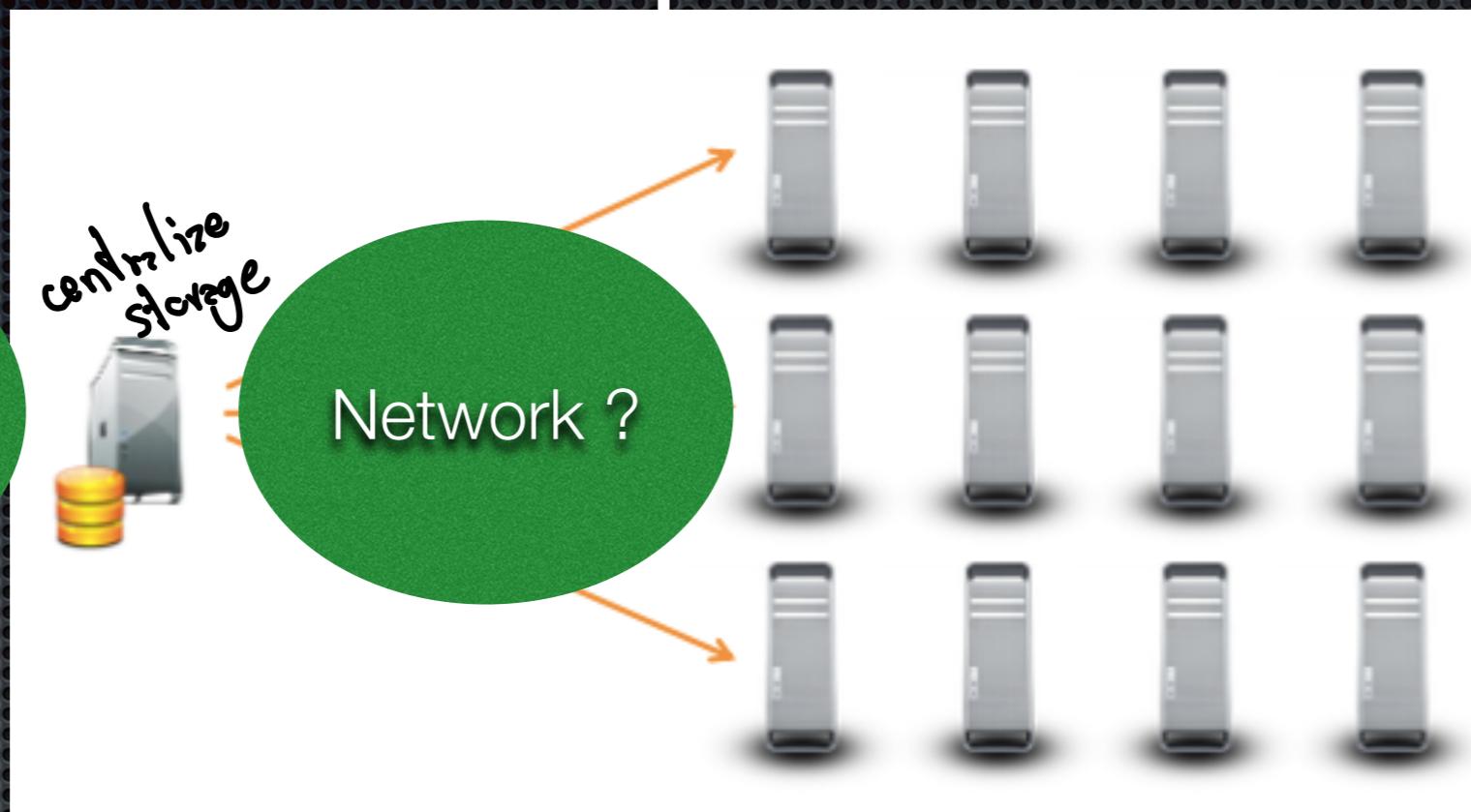
Traditional Distributed Systems

- Grid និង ប្រព័ន្ធដែលមានការចាប់ផ្តើមជាមុន
- Clusters និង សម្រាក់ប្រព័ន្ធ



Where is the problem?

1 Disk :
75-300 MB/s
ເວັບໄປ/ນ



- Traditional Grid/Cluster good for distributed workload
- One storage (SAN/NAS), multiple machines
- small data, long process

ເປັນໄຟລ໌ເຄີຍງົງ process ມານ່າງ ແລ້ວ ເວັບໄປກຳຈົດ
ນີ້ ສູ່ຮູ່ການ ແລ້ວຕ້ອງຟ້ານ 40 algorithm ດ້ວຍນິບຕື່ນ ຈົນດຸວ່ມ

Example

- 200 Gb
- more than 30 minutes to just read sequentially
- more than 5 hours to just transfer over fast ethernet

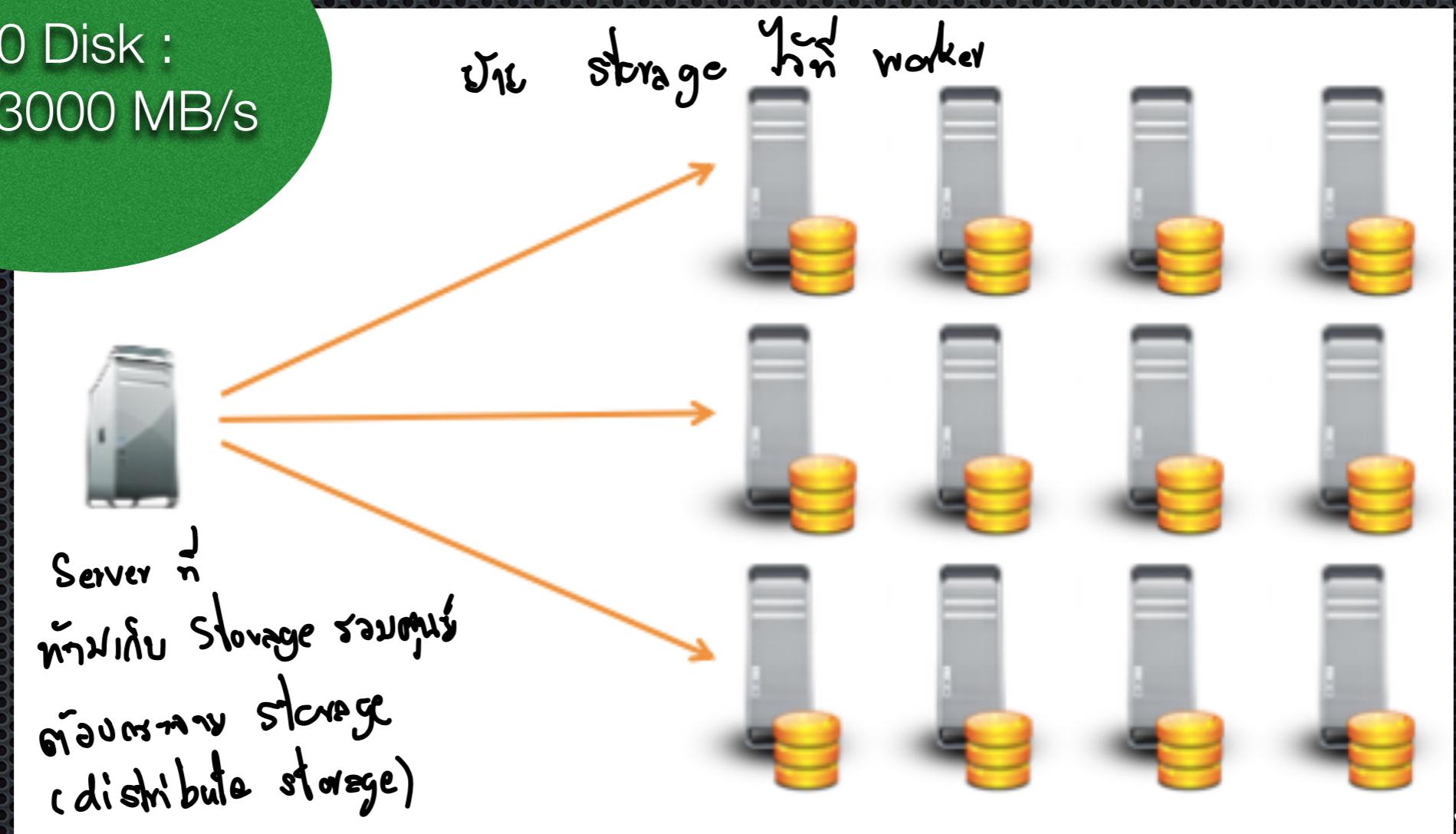
ჩს ქანუაყ ია:

Grid / Cluster რეჟიმ

1 Gbps ტერმინალი
10 Gbps ბანდვიდთ

MapReduce?

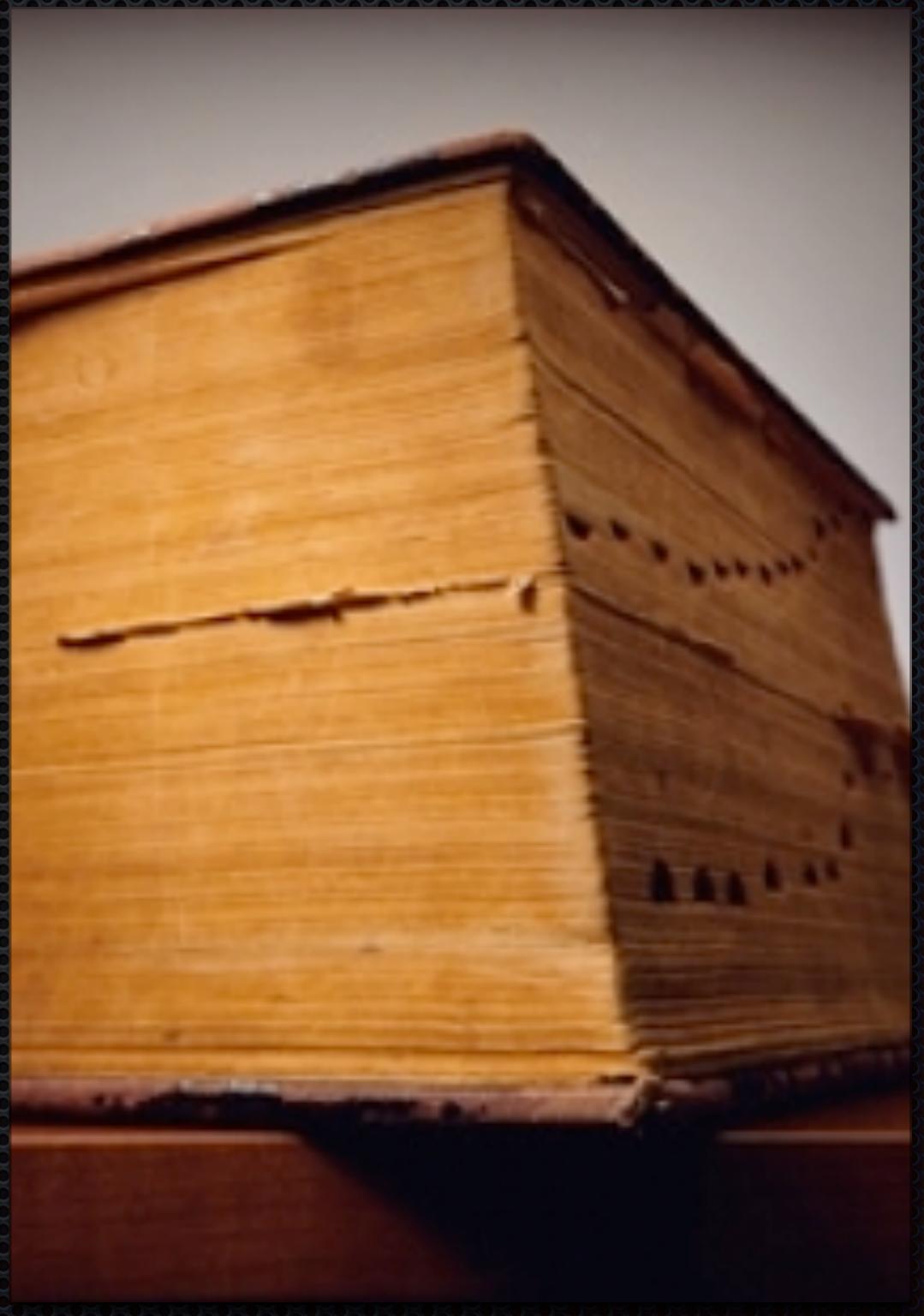
Google នៅលើការបង្កើត bandwidth ដូចជា bandwidth
100 Disk :
7500-3000 MB/s



Let's distribute the data/storage.

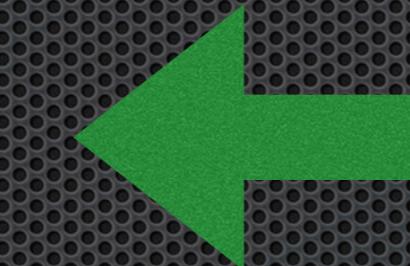
Example (Word frequency count)

Find frequency of each word in this big book?

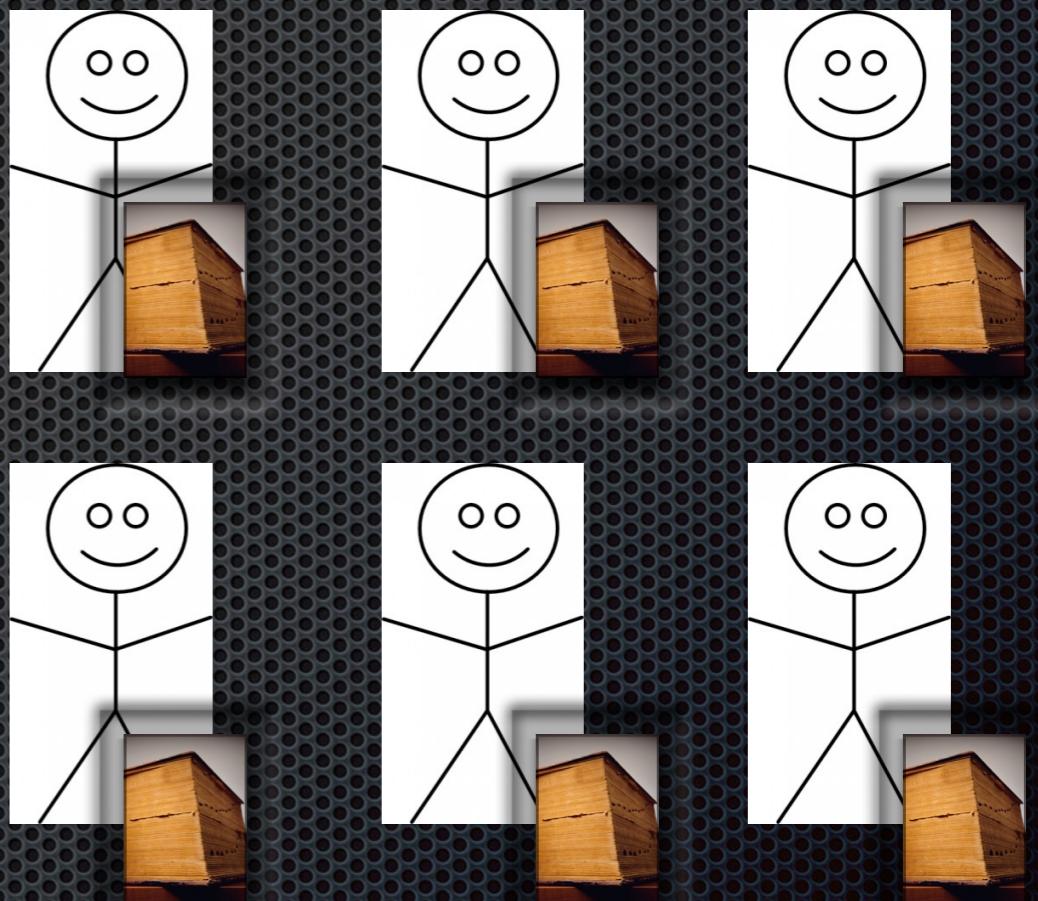


Word freq.: Grid/Cluster

Here is the data.
Please count.



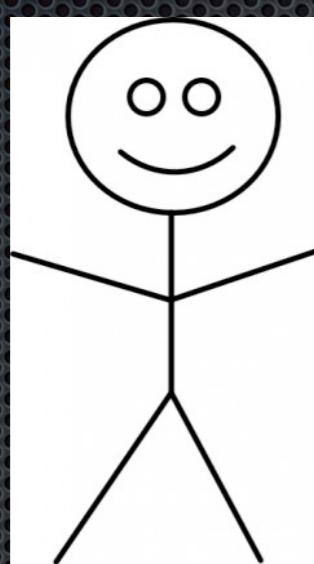
ແນວງນາກົງການມາຈຸດໜັງສື່ອ



Word freq.: MapReduce

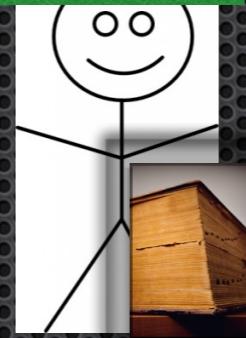
Store a part of data.

With your data,
please count.



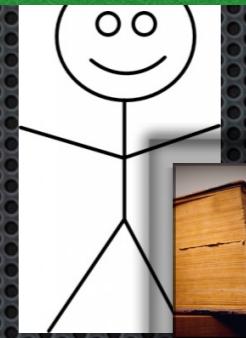
จิกาปุ่งหนังสือ บท 1

Map



บท 2

Map



บท 3

Map



คลิ๊กตาม
เรตติ้งมาดู
นะ!!

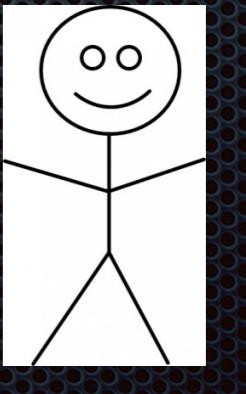
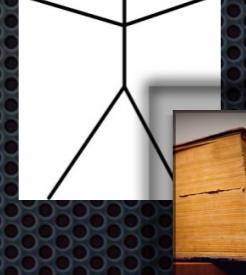


Map

Reduce



Map



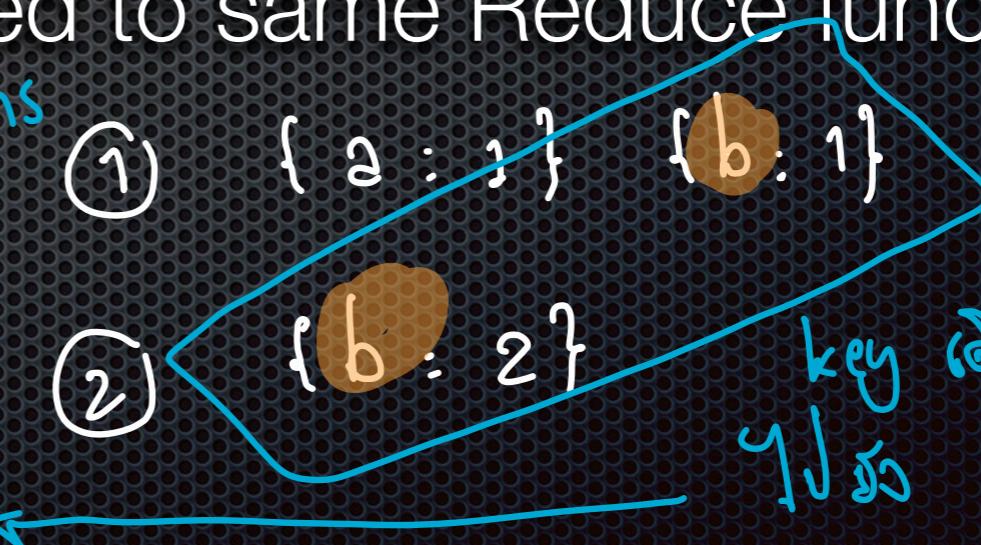
ไปมีชีวิตกัน เก็บเงินๆ กันที่ node ก็ไปมีชีวิตกัน

เลยเจ้มว่าการ rebalance cluster (node ใหม่ที่มา^{มา} เข้า ก็เปลี่ยน node ที่อยู่เดิม)

Fundamental of MapReduce

- Distributed storage *big data* គែងការណ៍នៃការផ្តល់ពេលវេលាដូចម៉ោង
- Do Map function with local data *run Fast* *map function តម្លៃបន្ថែមនៃការផ្តល់ពេលវេលាដូចម៉ោង*
- emit [key, value] pairs *key value នៅក្នុងបញ្ជី*
- Pairs with same key feed to same Reduce function *key value នៅក្នុងបញ្ជី*
- *reducer ត្រូវបានការពារឡើងដើម្បីបង្កើតការស្នើសុំការផ្តល់ពេលវេលាដូចម៉ោង*
- emit final value

R1 [a : [1]]
R2 [b : [1, 2]]



① { a : 1 } { b : 1 }
② { b : 2 } *key ត្រូវបានការពារឡើងដើម្បីបង្កើតការស្នើសុំការផ្តល់ពេលវេលាដូចម៉ោង*
key ត្រូវបានការពារឡើងដើម្បីបង្កើតការស្នើសុំការផ្តល់ពេលវេលាដូចម៉ោង

Disks

MapReduce ເປົ້າວົງເຄີນກໍສຸດ ລັງ ແຕ່ມະລກງົບຄວາມເຮົາ
ຕອນພາກເກີບຈັງສູງ ລະບຸວ່າເກີບຈວປາງຟັບ 2-3 node

1 Mean Time Between Failure
ອາຍຸແລ້ວຂຳກ່າວທັງ

- MTBF is 1,200,00 hours
- With 10,000 disks,
one will crash every 5
days

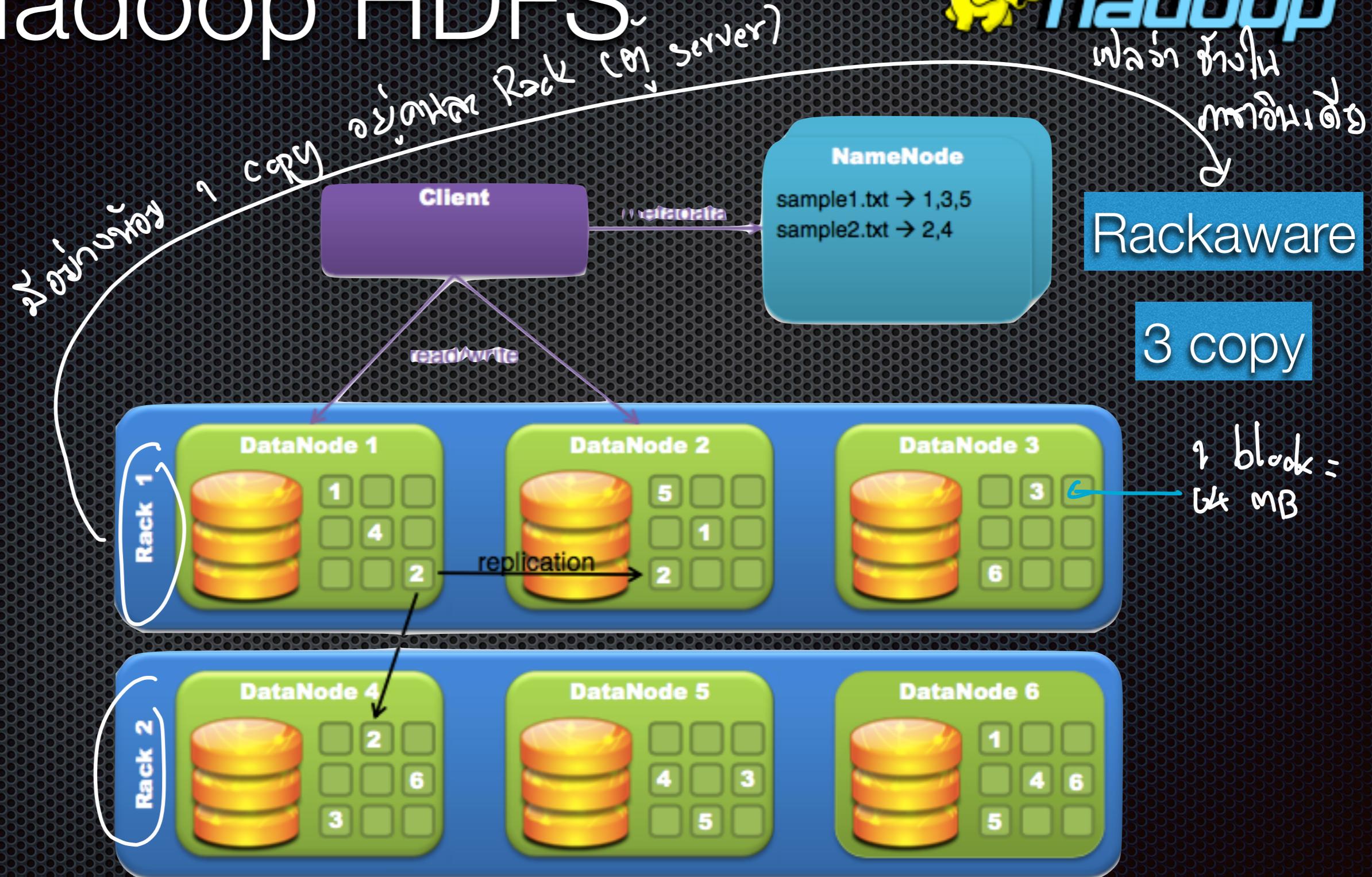
MySQL ລຸ່າງໝໍ ມີ ແລະ ສົບໃຈ

Source: google

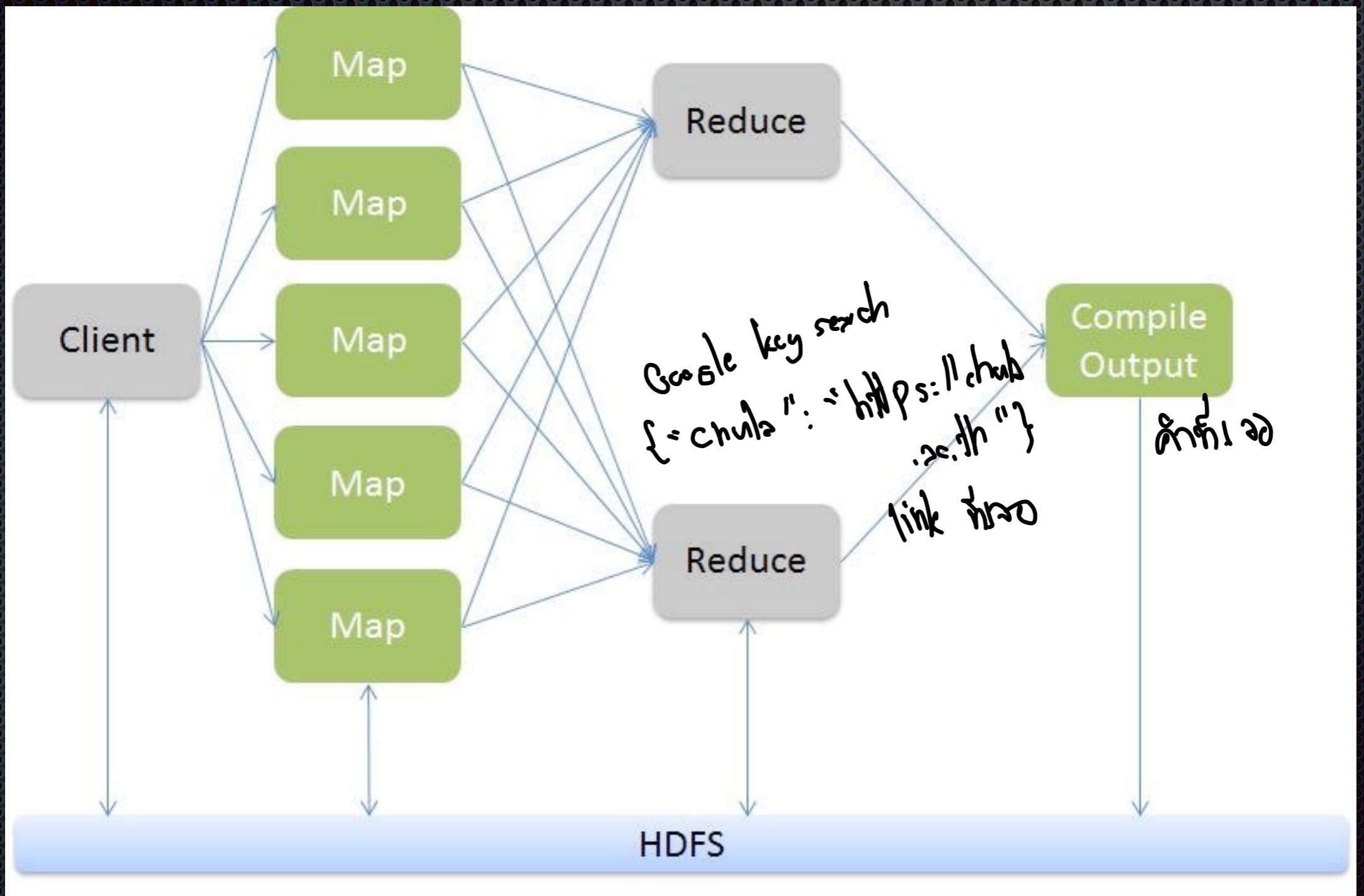


ယើត់ទៅ backup នៅ redundant 3 copies ឲ្យមក

Hadoop HDFS



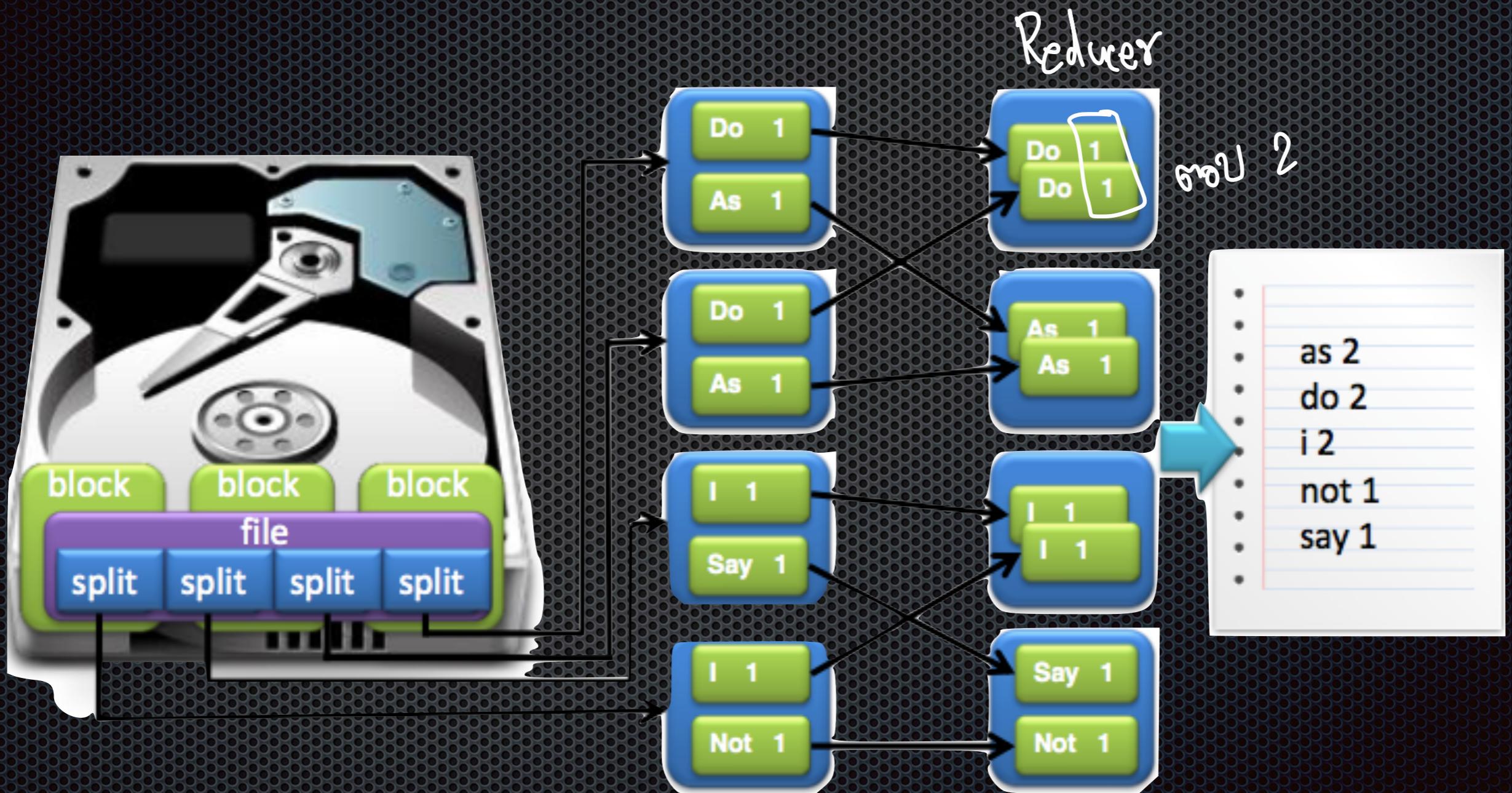
How MapReduce work?



Hadoop Architecture

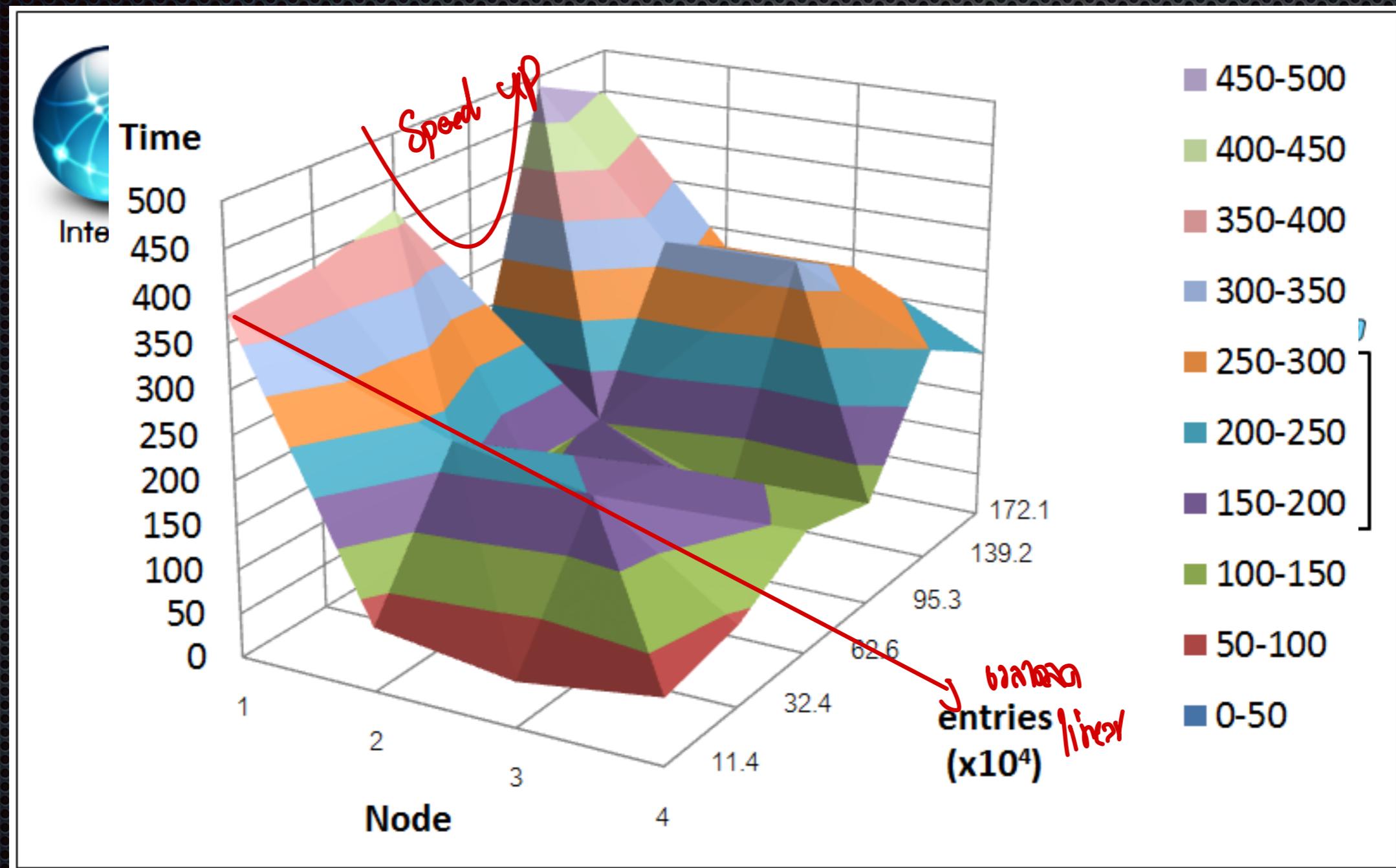


hadoop



MapReduce in action

Large-Scale log analysis



See demo

Sample Data

Personid	Gender	Salary	Name	Grade
A0001	M	1,000	A	A
A0002	F	2,000		B
A0003	M	3,000	C	A

Sample Data

Perso	Gend	Salary	Name	Grade
A000	M	1,000	A	A
A000	F	2,000		B
A000	M	3,000	C	A

[A : 1] → [A : 1]
[A : 1] → [B : 1] —> { B : [1] }

map

Grade	Count
A	2
B	1

```
function Map(doc) {  
    emit(Grade, 1);  
}      {A, B, C}
```

```
function Reduce(key, values)  
{  
    return sum(values);  
}
```

ការណែនាំក្នុងក្រុង

“MapReduce is just A Major Step Backwards!!!”

– *Dewitt and StoneBraker in January 17, 2008*

Let's debate!

Major Step Backwards!!!

- No schema, type (Garbage) ໄມ້ສົກລວມ ເກພຍາຮັງປຳລັງ ? key ມີ ຖະແຫຼງ value ມີ ຖະແຫຼງ
- No standard access language (e.g. SQL) ນາທຽບນຸ່ານຫຍຸ້ງ ?
- No index, but brute force mapReduce ໃຊ້ brute force
≈ 100,000 record ກໍຖາເລືອນກົງ
ຝ່າຍເກືອງນິກ where SQL
- No transaction DB ຫັດກັບຕົວ ແລ້ວ ຕັ້ງຈະຈະນູນ
- No integrity (e.g. foreign key)

Let's fight back!

MapReduce is not DBMS

- No schema, type (Garbage)
- No standard access language (e.g. SQL)
- No index, but brute force
- No transaction
- No integrity (e.g. foreign key)

MapReduce is a big forward in.

DB និង scale នៃ computing mapReduce ត្រូវ scale ឡើង

មានលក្ខណៈ mapReduce នាមខាង DB

- Scalability (Scale out)
- Reliable software model for unreliable hardware

What have we learned?

- MapReduce is a software solution for:
 - processing software on unreliable hardware
 - distributing I/O (data as well as workload)
- MapReduce is not DBMS.
- Think in Map function and Reduce function

SW node | ໜ້າໄກເປົ້າມ
HW | ກວດວິນດີ

RDBMS vs. M

Use th

RDBMS vs. MapReduce

	RDBMS	MapReduce
Data size	gigabytes	petabytes
Access	interactive and batch	batch
Updates	read and write many times	write once read many times
Structure	static schema	dynamic schema
Integrity	high	low
Scaling	nonlinear	linear

MapReduce is a screwdriver.

good for:

- unstructured data
- data intensive computation
- batch operations
- scale horizontal

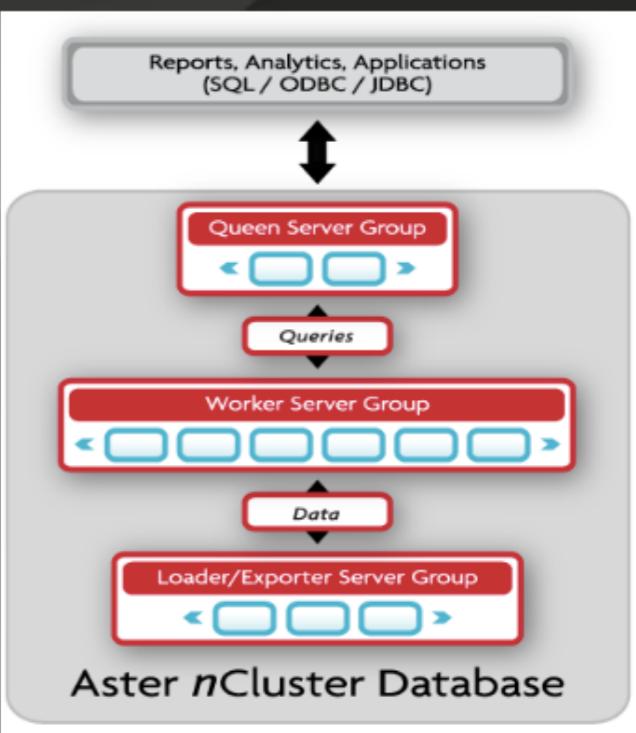
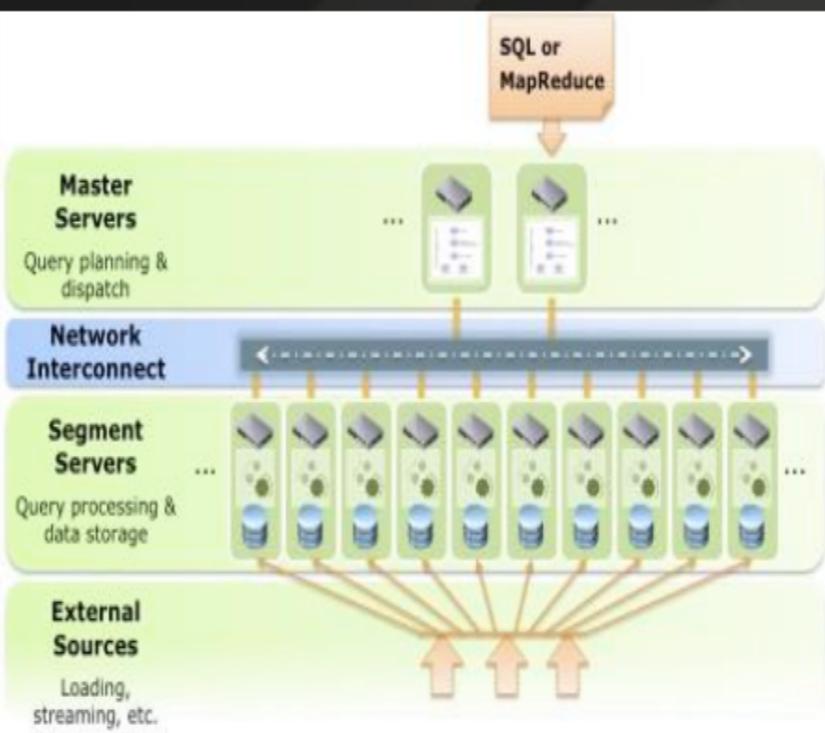
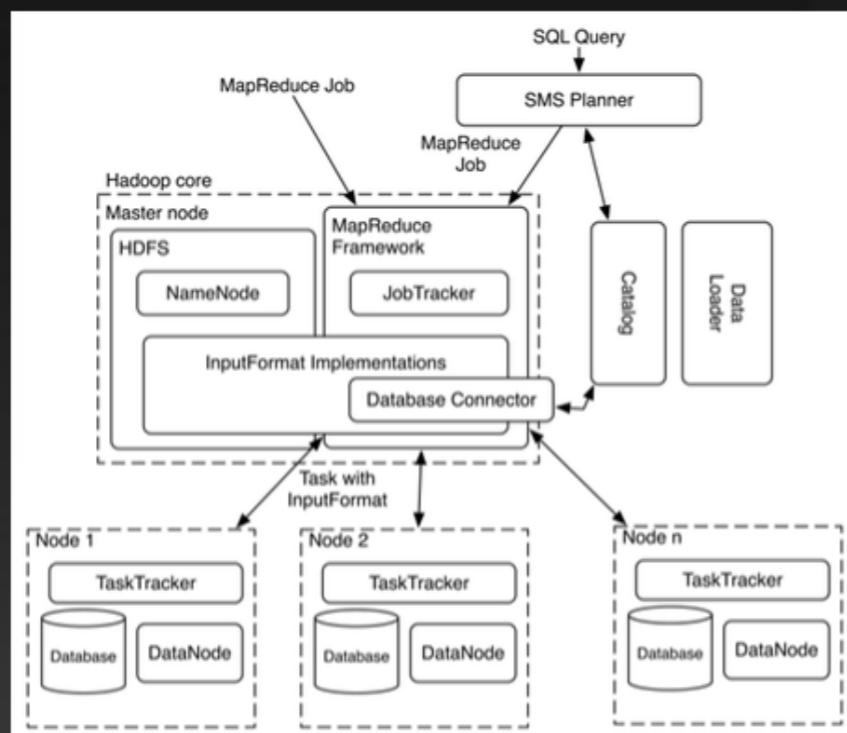
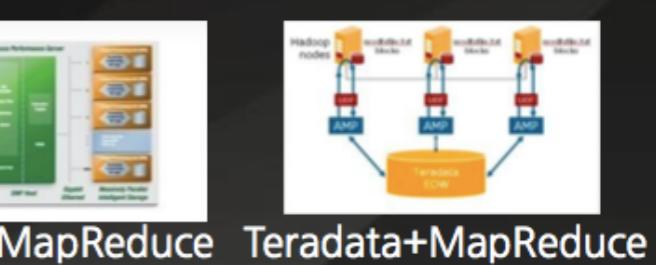
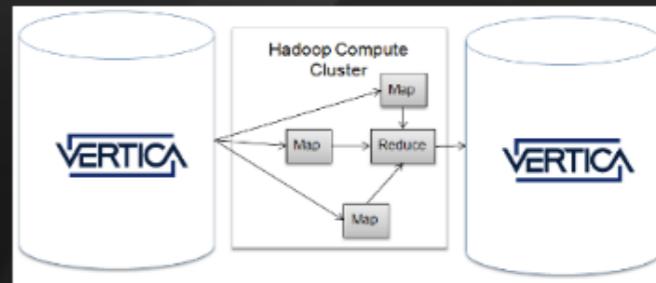
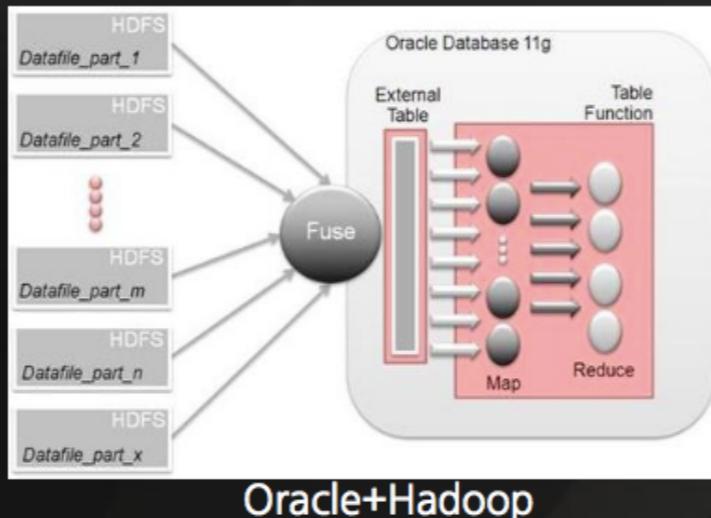
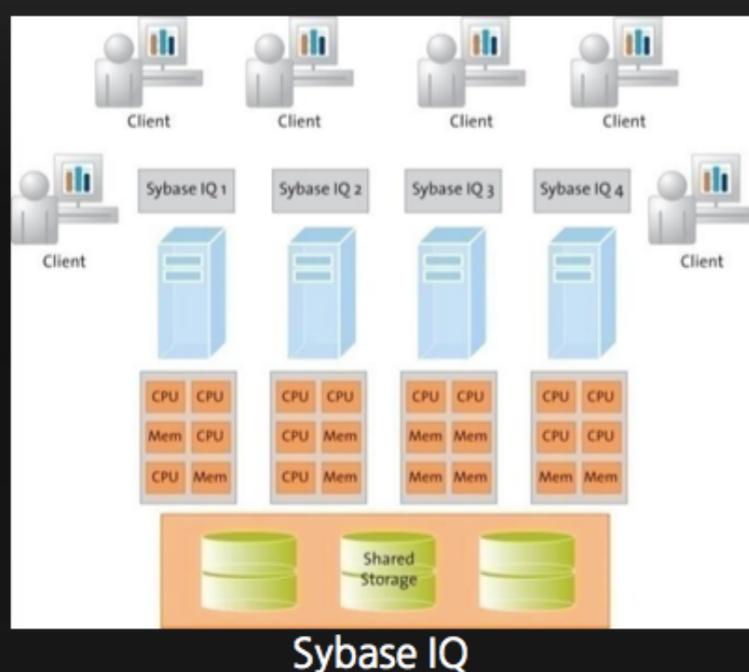


good for:

- structured data
- transactions
- interactive requests
- scale vertically

Databases are hammers.

Why don't use both?



Hadoop is good for

மின்மான்றும் embalancing parallel உயர்தீவுகள்
நகர் dependencies ஏற்று

- Large-scale data analysis
- Search engines
- etc...

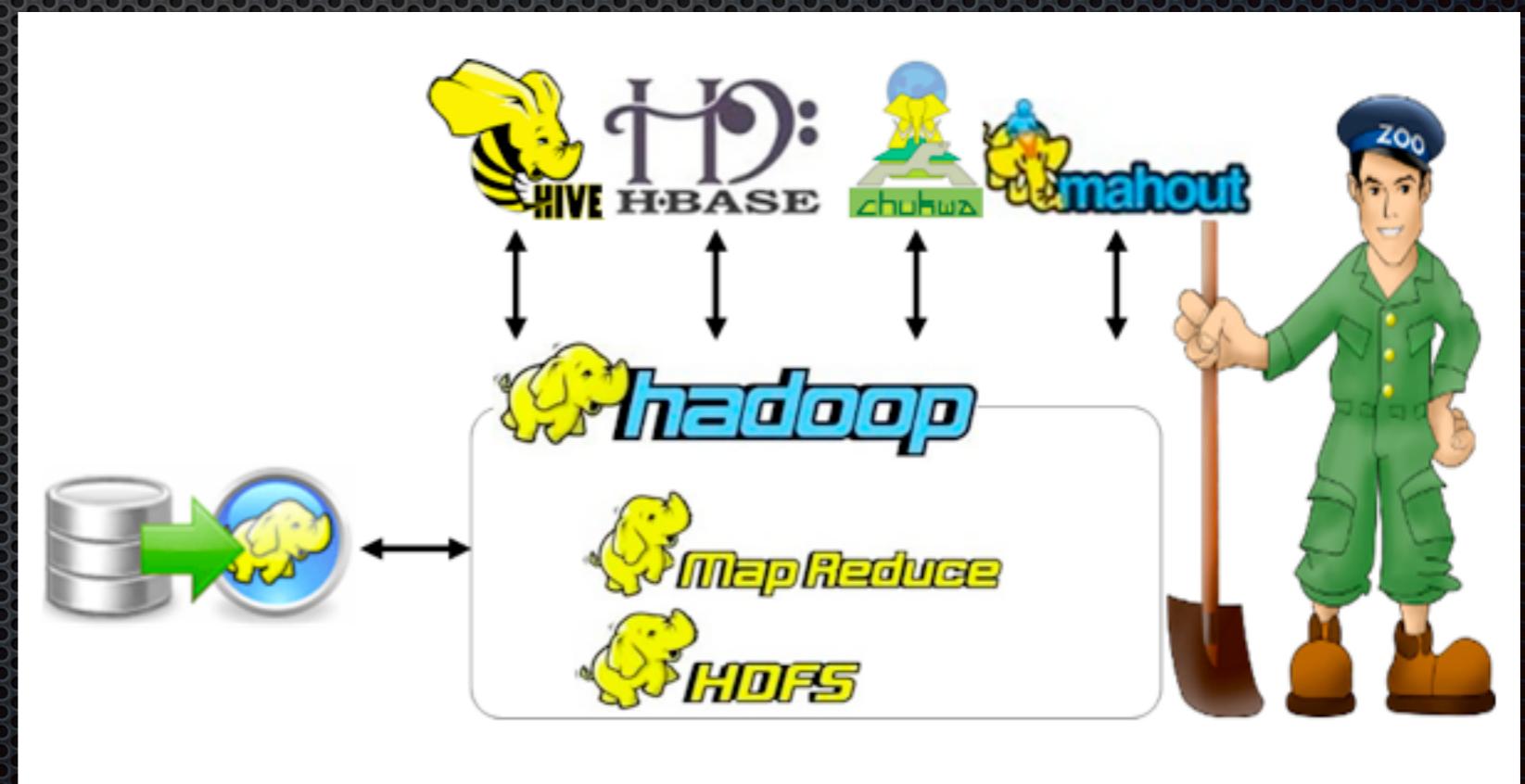
Hadoop is bad for

- Pi estimation
- Dependency calculation (Recurrent relation)
- DBMS replacement / transaction

A diagram showing a circle inscribed within a square. The circle touches all four sides of the square. A radius of the circle is drawn from the center to the bottom side, labeled 'r'. The width of the square is indicated by a red double-headed arrow at the bottom, labeled '2r'. To the right of the square, the formula πr^2 is written in red, representing the area of the circle.

Easy ways to MapReduce

- Use high-level analysis tools
 - Hive (SQL style)
 - Gnu R
 - Spark
 - H2O



தொல்வி data நிலைமை mapReduce மாதிரிகள்

Thank you
Question?