



高速運算於生物資訊之應用

HPC for Bioinformatics

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw



Powered by DRBL



PART 1 :

HPC 101

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw



Powered by DRBL

Brief History of Computing (1/5)



Source: <http://pinedakrch.files.wordpress.com/2007/07/>

1960 PDP-1

-
-
-

1965 PDP-7

-
-
-

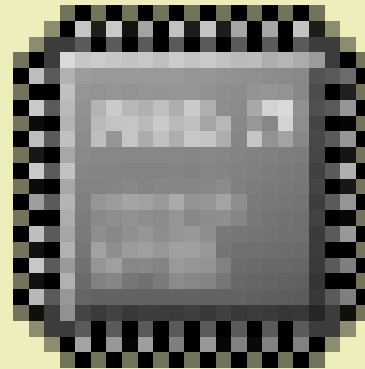
1969 1st Unix

**Mainframe
Super
Computer**

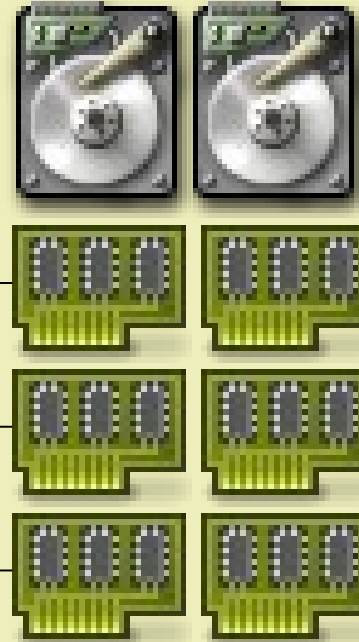
Evolution of Computing Architecture (1/5)



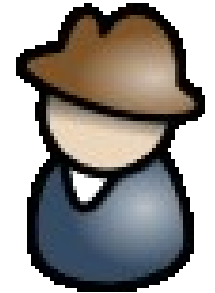
**Multiple
Users**



**Single
CPU**



**Shared
Memory**



**One
Admin.**

**Single
Super Computer**

**Mainframe
Super
Computer**

使用者心裡的『謎之聲』 (1/5)

等執行程式，要排隊排好久喔～

可惡，程式又死掉了，又得重排一次

真希望自己有一台電腦可以跑！！

超級電腦是有錢人才玩得起的玩具～

1977 Apple II

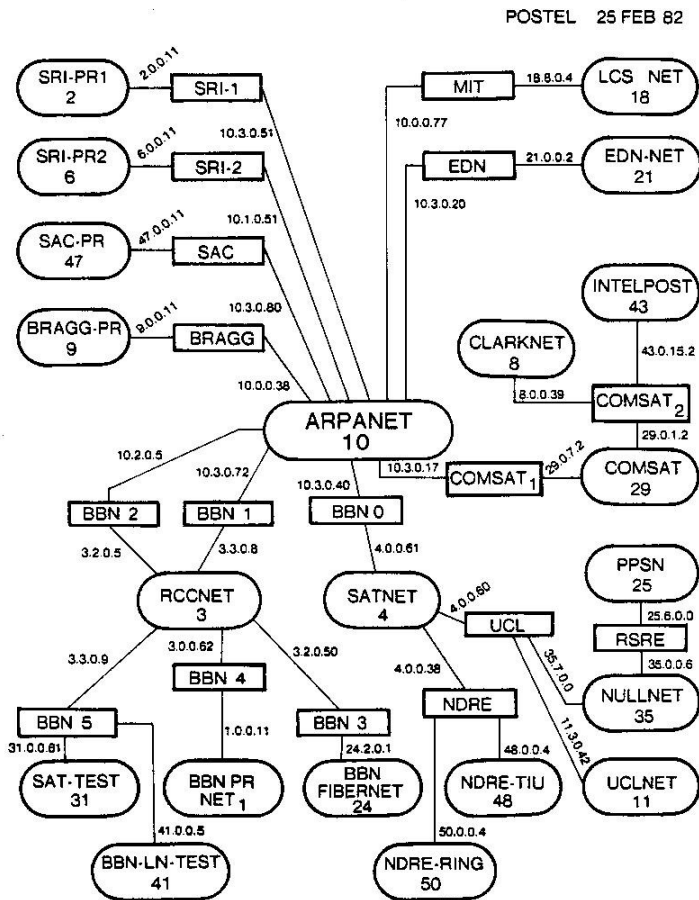


1981 IBM 1st PC 5150

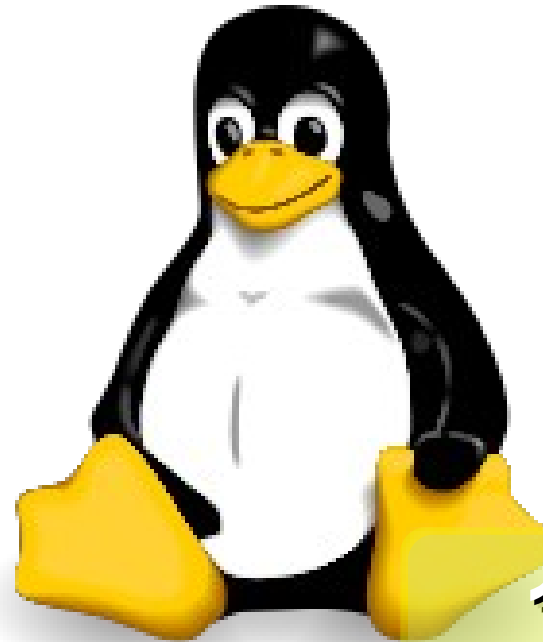


Back to Year 1970s ...

1982 TCP/IP



1983 GNU



1991 Linux

Back to Year 1980s ...

Brief History of Computing (2/5)



Source: <http://www.nchc.org.tw>

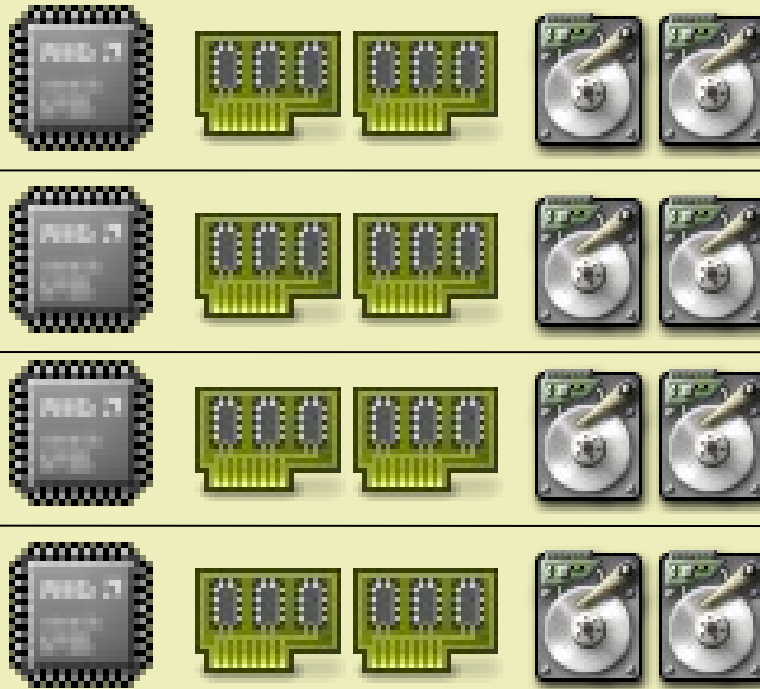
Mainframe
Super
Computer

PC / Linux
Cluster
Parallel

Evolution of Computing Architecture (2/5)

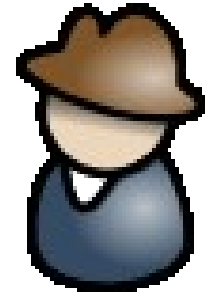


Multiple Users



Separate CPU

Separate Memory



One Admin.

Game
er
uter

**PC / Linux
Cluster
Parallel**

**Multiple PC
in One Location**

使用者心裡的『謎之聲』 (2/5)

可惡，記憶體不夠大，程式又死掉了

奇怪，我的程式為什麼不能跑？

真希望自己有一組叢集可以跑！！

管理員老大，可以幫我裝LiBT嗎？

**1990 World Wide Web
by CERN**

...

...

**1993 Web Browser
Mosaic by NCSA**

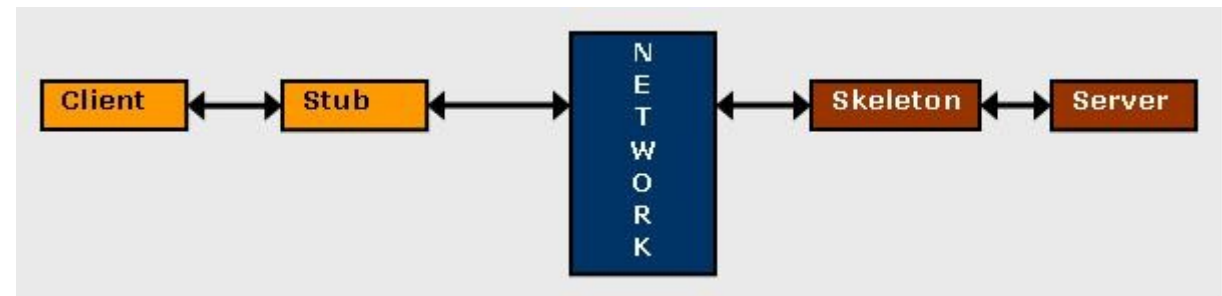


1991 CORBA

Java RMI

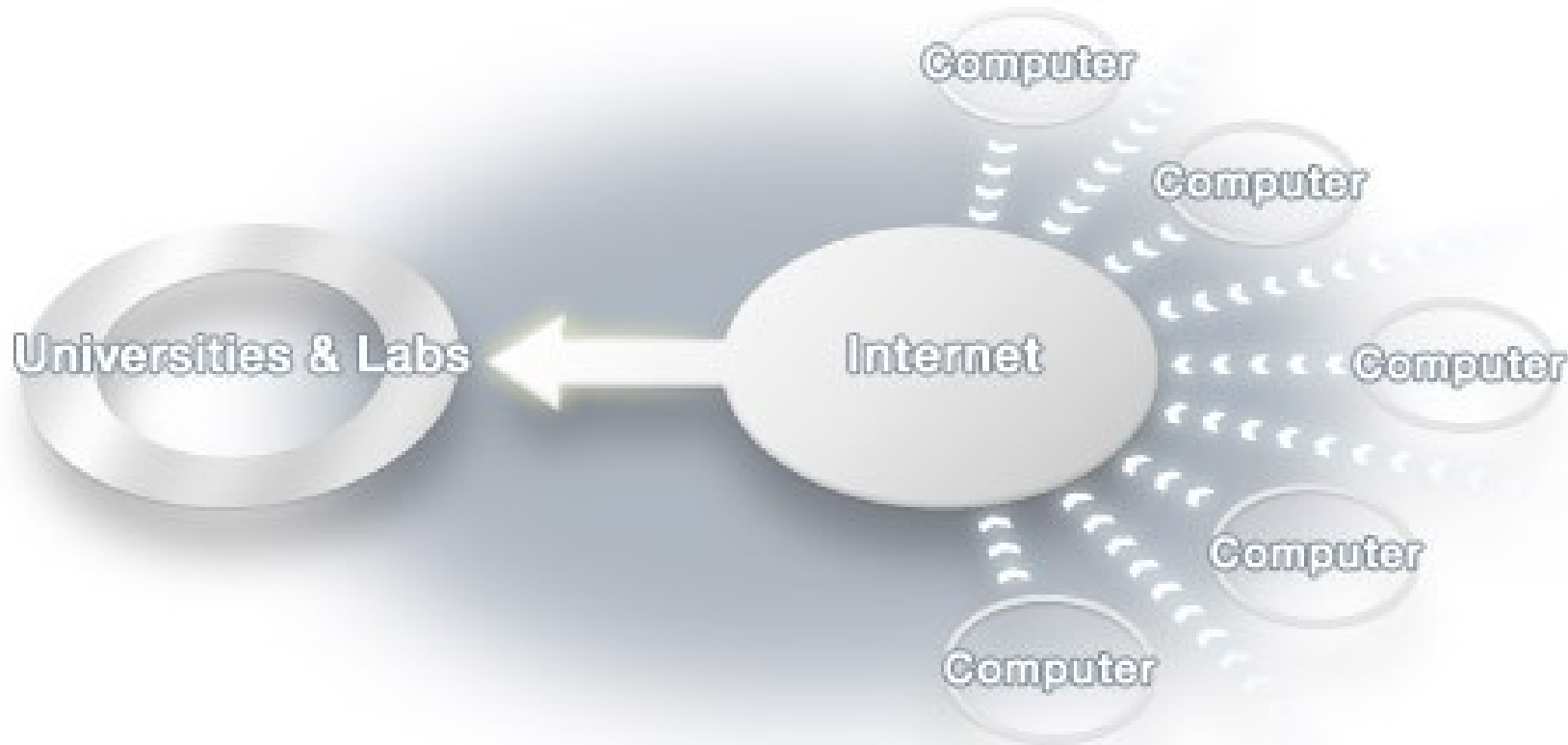
Microsoft DCOM

Distributed Objects



Back to Year 1990s ...

Brief History of Computing (3/5)



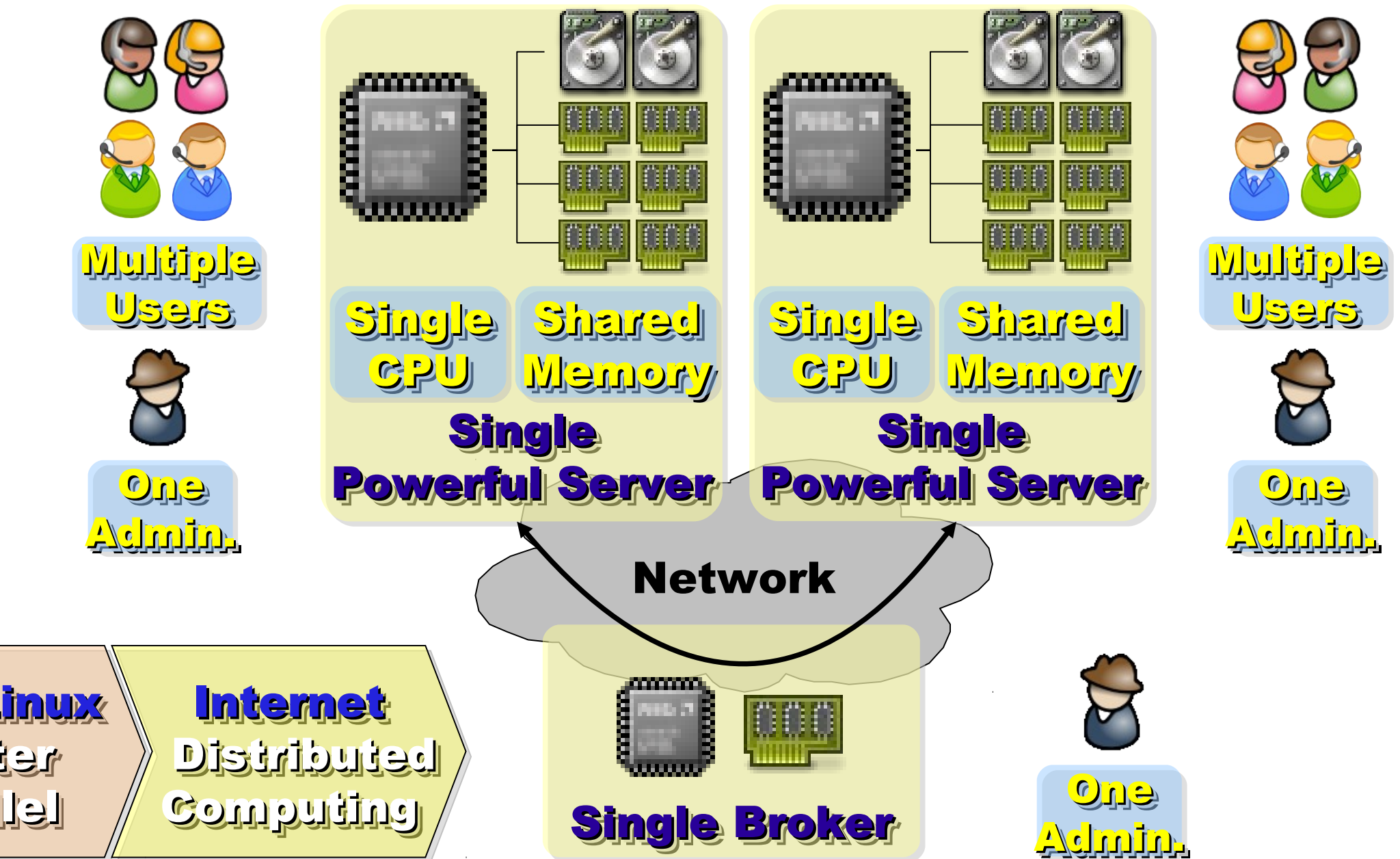
Source: <http://www.scei.co.jp/folding/en/dc.html>

Mainframe
Super
Computer

PC / Linux
Cluster
Parallel

Internet
Distributed
Computing

Evolution of Computing Architecture (3/5)



使用者心裡的『謎之聲』 (3/5)

分散式物件怎麼這麼抽象啊～XD

啊！網路斷線了～不能動了～

大家把閒置電腦都貢獻出來吧！！

給我網路遊戲，其餘免談！

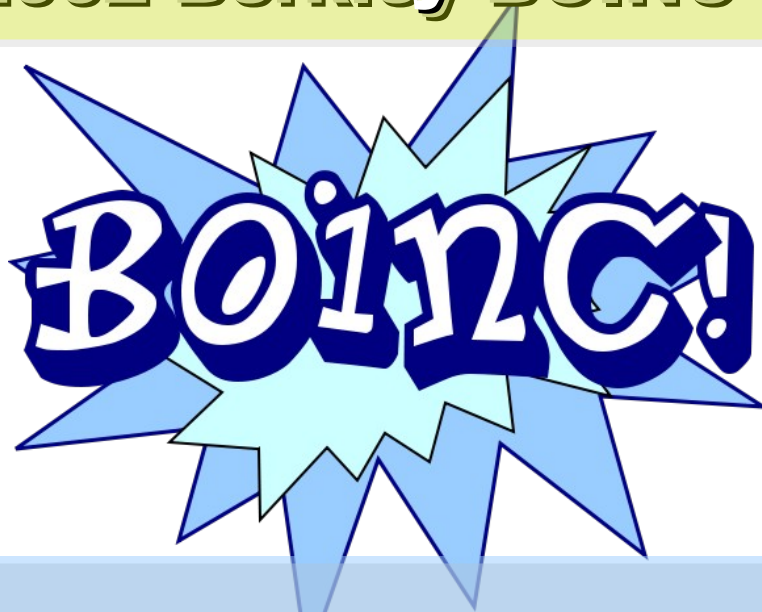
1997 Volunteer Computing
1999 SETI@HOME



2003 Globus Toolkit 2



2002 Berkley BOINC



2004 EGEE gLite



Back to Year 2000s ...

Brief History of Computing (4/5)



Source: <http://gridcafe.web.cern.ch/gridcafe/whatisgrid/whatis.html>

Mainframe
Super
Computer

PC / Linux
Cluster
Parallel

Internet
Distributed
Computing

Virtual Org.
Grid
Computing

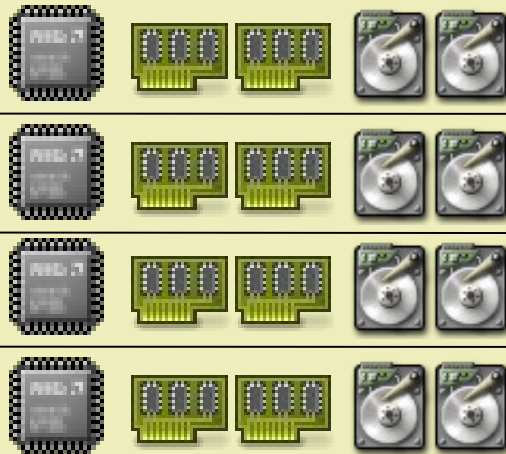
Evolution of Computing Architecture (4/5)



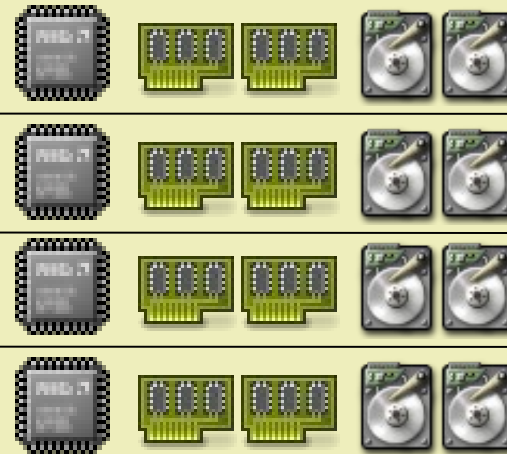
Multiple Users



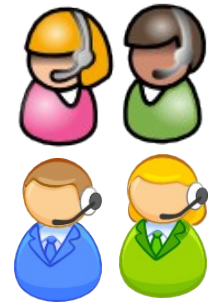
One Admin.



**Multiple PC
in one location**



**Multiple PC
in other location**



Multiple Users



One Admin.

Grid Middleware

Network

**Internet
distributed
computing**

**Virtual Org.
Grid
Computing**

Virtual Organization

Heterogeneous

CyberInfrastructure

使用者心裡的『謎之聲』 (4/5)

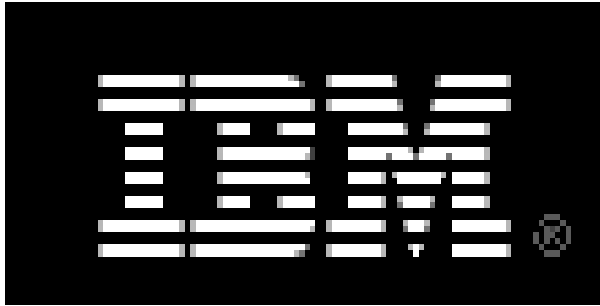
已給我認證了，為什麼要不到資源？

啥？可用資源在美國，慢慢搬檔案吧！

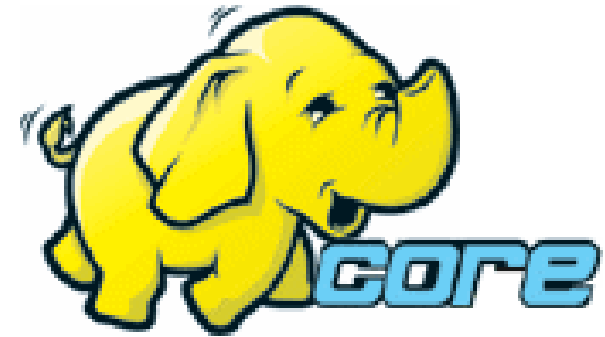
為什麼人家 Google 那麼會算?!

長官，請幫我們去談好資源共享政策吧！

2001 Autonomic Computing
IBM



2006 Apache Hadoop



2005 Utility Computing
Amazon EC2 / S3

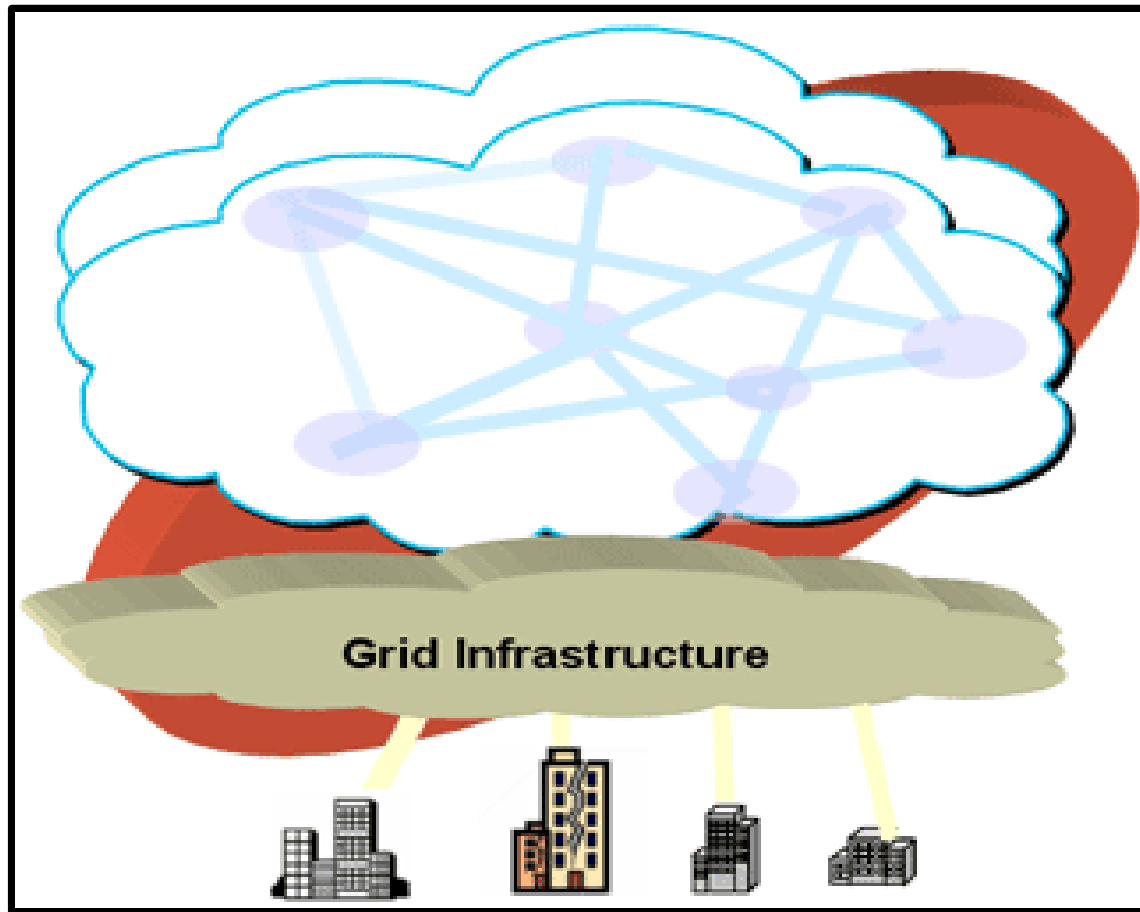


2007 Cloud Computing
Google + IBM



Back to Year 2007 ...

Brief History of Computing (5/5)



Source: <http://mmdays.com/2008/02/14/cloud-computing/>

mainframe
per
computer

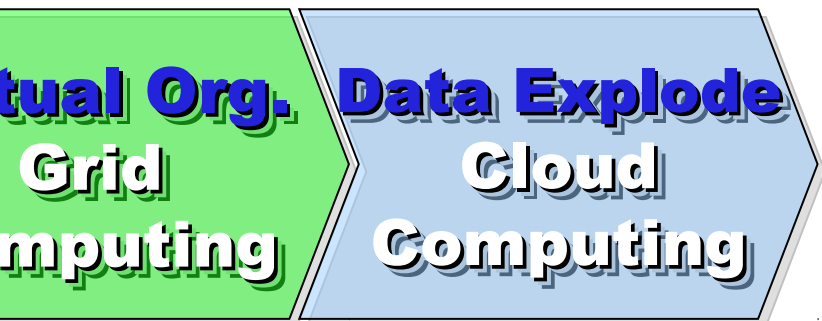
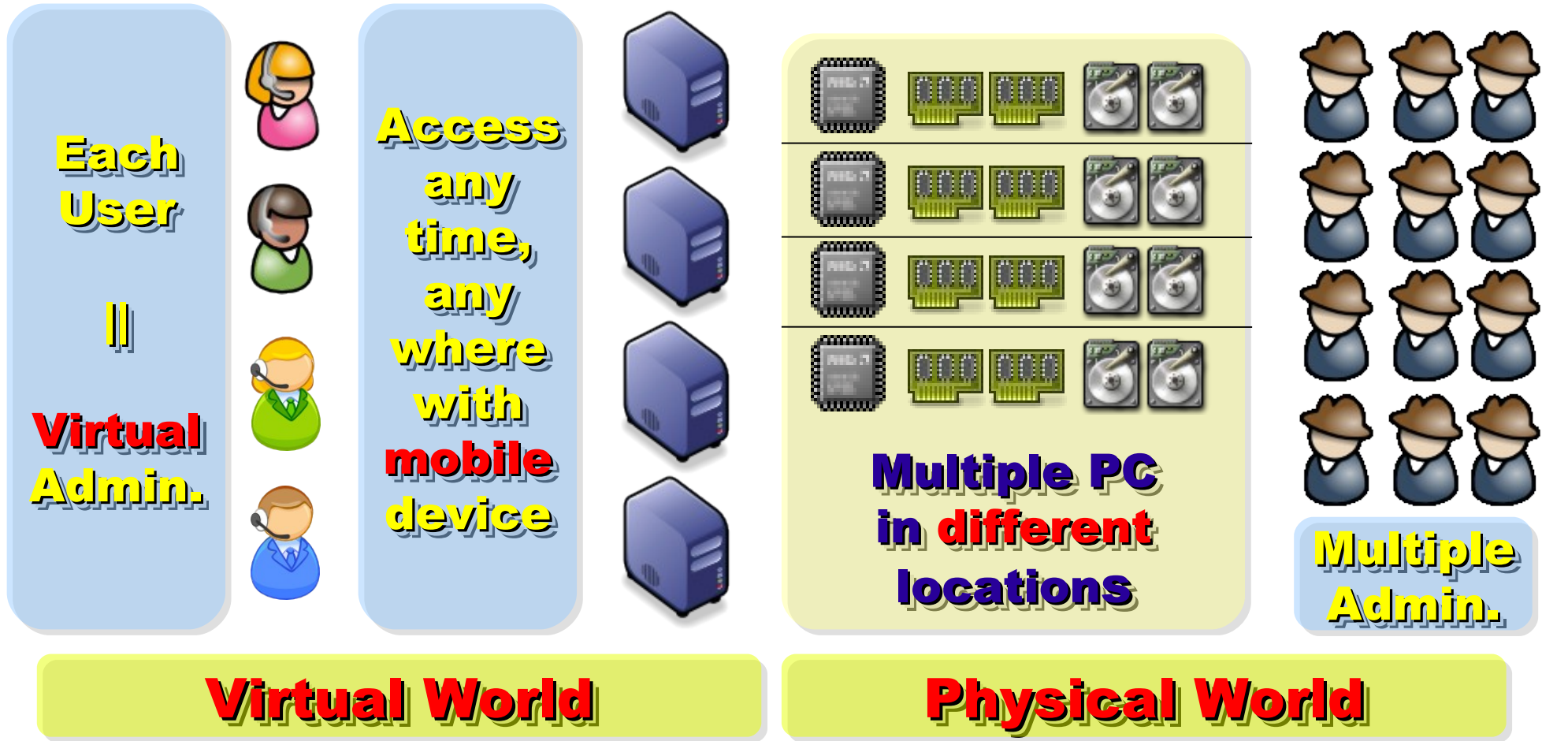
PC / Linux
Cluster
Parallel

Internet
Distributed
Computing

Virtual Org.
Grid
Computing

Data Explode
Cloud
Computing

Evolution of Computing Architecture (5/5)



What is NEXT ?!
Mobile Computing ?!

使用者心裡的『謎之聲』 (5/5)

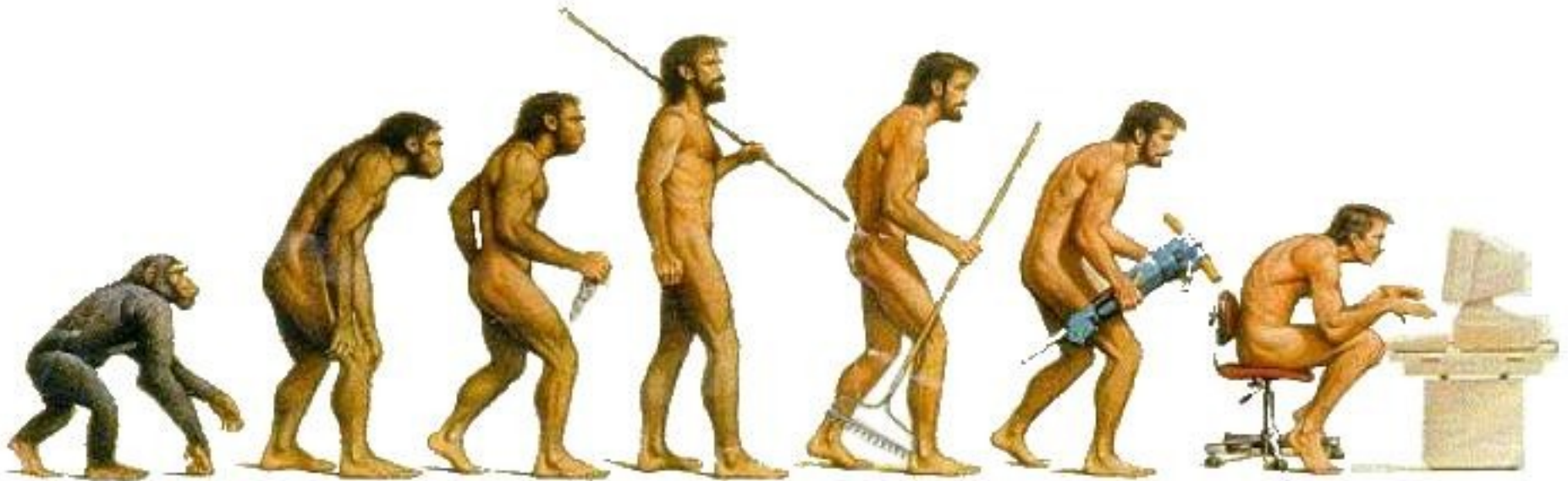
雲端運算合適我用嗎？

按使用時間計費，真的比較省？

Google 到底有沒有偷窺我的信？！

我們自己可以架雲端運算的環境嗎？

Evolution



(OR is it?)

Source: <http://cyberpingui.free.fr/humour/evolution-white.jpg>

**Which Type
of HPC is
the Right ONE
to solve
My Problem ?**

	執行程式 記憶體需求	執行程式 邏輯可分割	輸入資料 數量 / 大小	應用特性 計算特性
超級電腦 Mainframe	非常大 (比PC大)	不易分割	單一大檔 GB	即時性高 共享記憶體
叢集運算 Cluster Parallel	小於單一 計算節點 配置量	邏輯相近 可分割	一般數量 MB	即時性一般 共享檔案
分散式運算 Distributed	小於單一 計算節點 配置量	邏輯不同 偕同運作	一般數量 MB	即時性高 訊息傳遞
格網運算 Grid	小於單一 計算節點 配置量	邏輯相近 可分割	一般數量 MB	即時性較低 資料就計算
雲端運算 Cloud	小於單一 計算節點 配置量	邏輯相近 各自獨立	海量檔案 MB	即時性低 計算就資料



PART 2 :

HPC & Bioinformatics Application

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw



Powered by DRBL

BLAST (Basic Local Alignment Search Tool)



- <http://blast.ncbi.nlm.nih.gov/>
- National Center for Biotechnology Information
- BLAST is an algorithm for **comparing primary biological sequence** information. (BLAST 用來比對生物序列的主要結構)
 - the **amino-acid** sequences of different proteins
 - the **nucleotides** of DNA sequences
(例如：不同蛋白質的**氨基酸**序列 DNA 序列的**核甘酸**)
- 用途：搜尋其他物種 (如：老鼠) 未知基因，是否也存在人類基因中
- 優點：使用啟發式搜索來找出相關的序列，比動態規劃快上 **50** 倍。
- 缺點：不能夠保證搜尋到的序列和所要找的序列之間的相關性。
- 技術問題：**巨大的序列資料庫**需要進行比對，怎樣計算才快？
- Source: [http://zh.wikipedia.org/w/index.php?title=BLAST_\(生物資訊學\)&variant=zh-tw](http://zh.wikipedia.org/w/index.php?title=BLAST_(生物資訊學)&variant=zh-tw)



PART 2.1 :

Cluster 101 & mpiBLAST

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw



Powered by DRBL



At First, We have “4 + 1” PC Cluster

It'd better be
2ⁿ

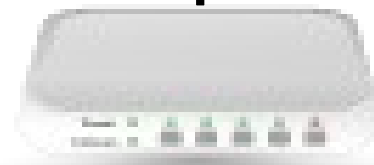


Manage
Scheduler

**Then, We connect 5 PCs with
Gigabit Ethernet Switch**



GiE Switch



**10/100/1000
MBps**



**Add 1 NIC
for WAN**



Compute Nodes

4 Compute Nodes will communicate via **LAN Switch**. Only **Manage Node** have **Internet Access** for Security!



WAN



Manage Node

Compute Nodes

Basic System Setup for Cluster

Messaging

MPICH

Account Mgmt.

SSHD

NIS

YP

GCC

GNU Libc

Bash



Perl

Kernel Module

Linux Kernel

Boot Loader

On Manage Node,
We need to install Scheduler and
Network File System for sharing
Files with Compute Node

Job Mgmt.

OpenPBS

File Sharing

NFS

Extra

Messaging

MPICH

GCC

Bash

Perl

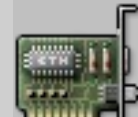
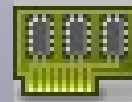
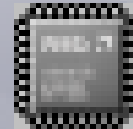
Account Mgmt.

SSHD

NIS

YP

GNU Libc



Kernel Module

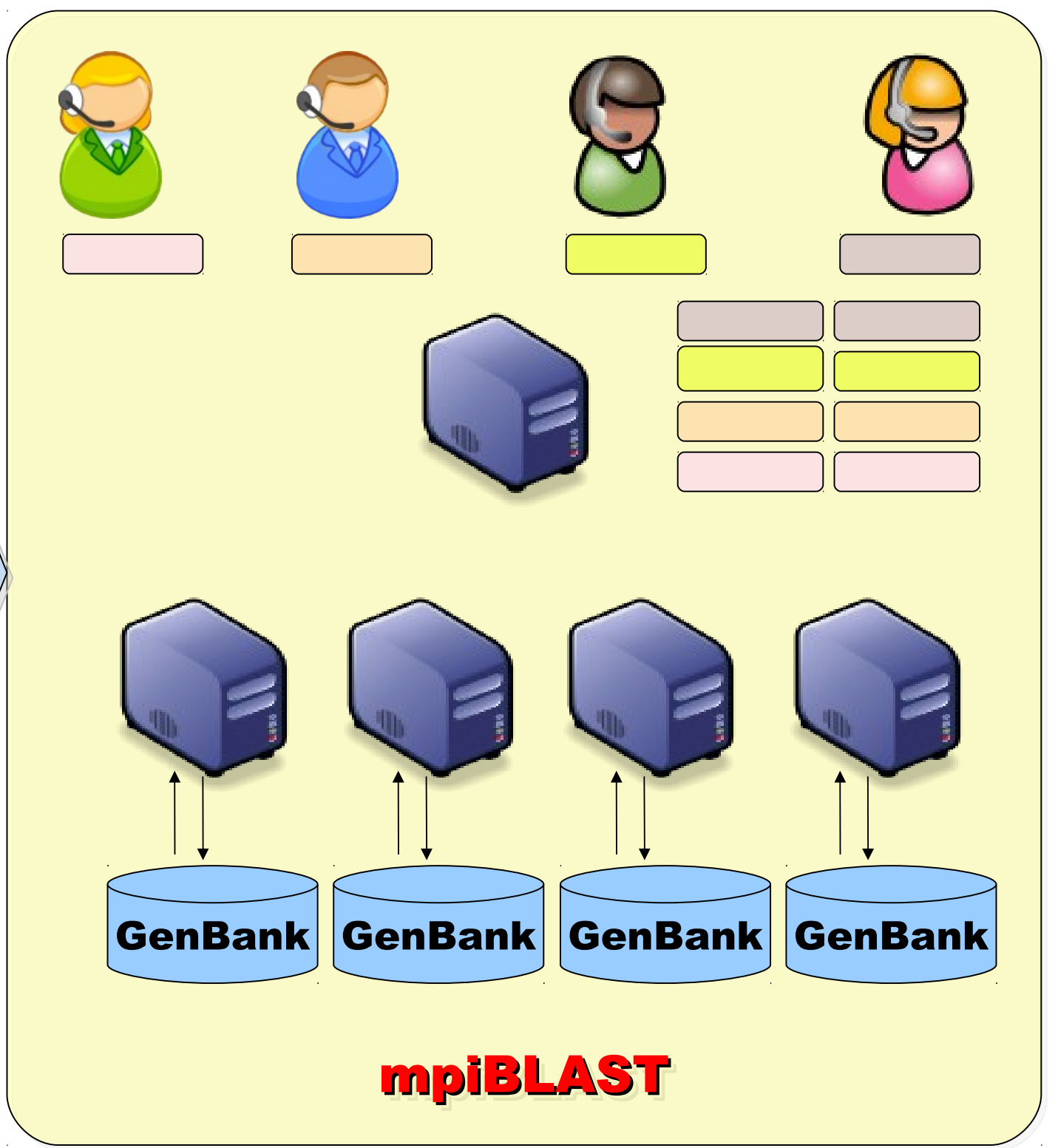
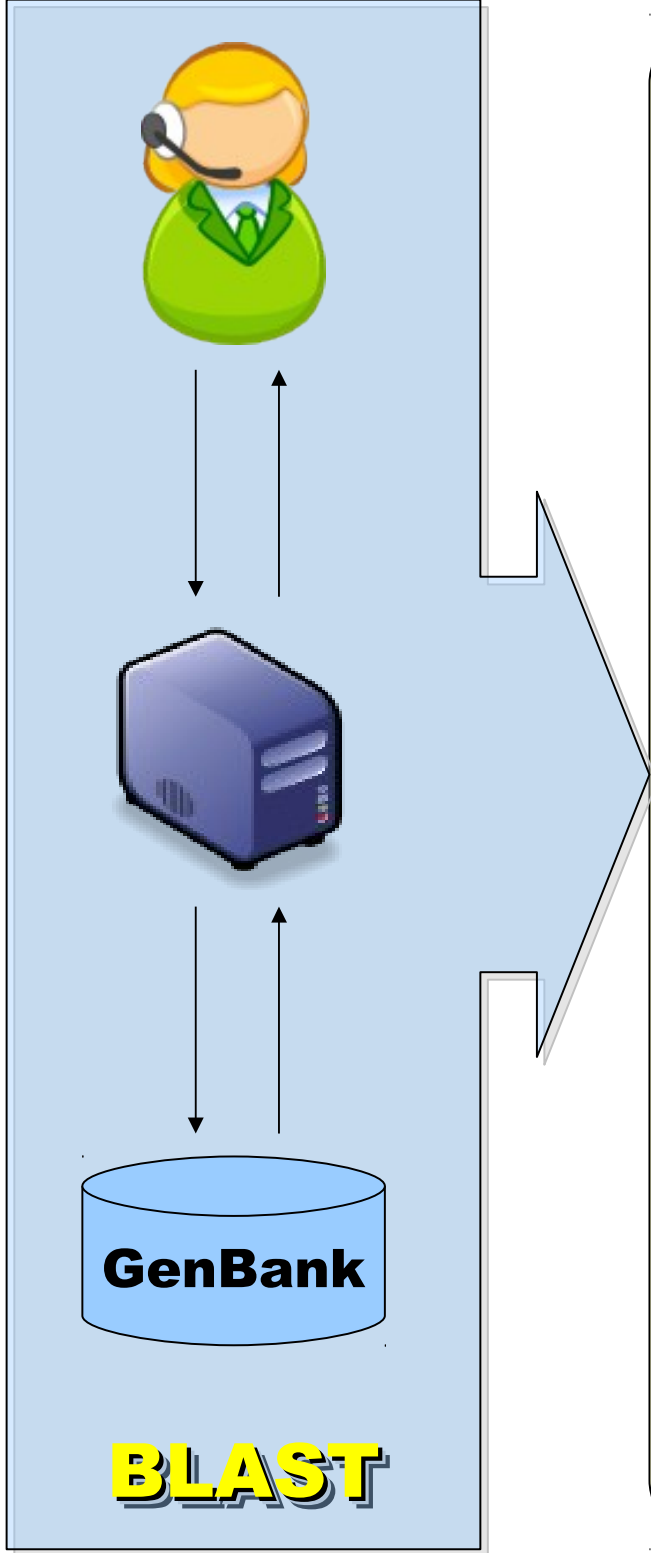
Linux Kernel

Boot Loader

mpiBLAST

- <http://www.mpiblast.org/>
- An **open-source, parallel implementation** of NCBI BLAST
- 特點：
 - **Database fragmentation**
 - **Query segmentation**
 - **Parallel input/output**
- 設計理念：
 - **The Design, Implementation, and Evaluation of mpiBLAST.**
 - <http://www.mpiblast.org/downloads/pubs/cwce03.pdf>
- 類似工具：
 - **TurboWorx TurboBLAST**
 - **Parallel BLAST by Caltech**







PART 2.2 :

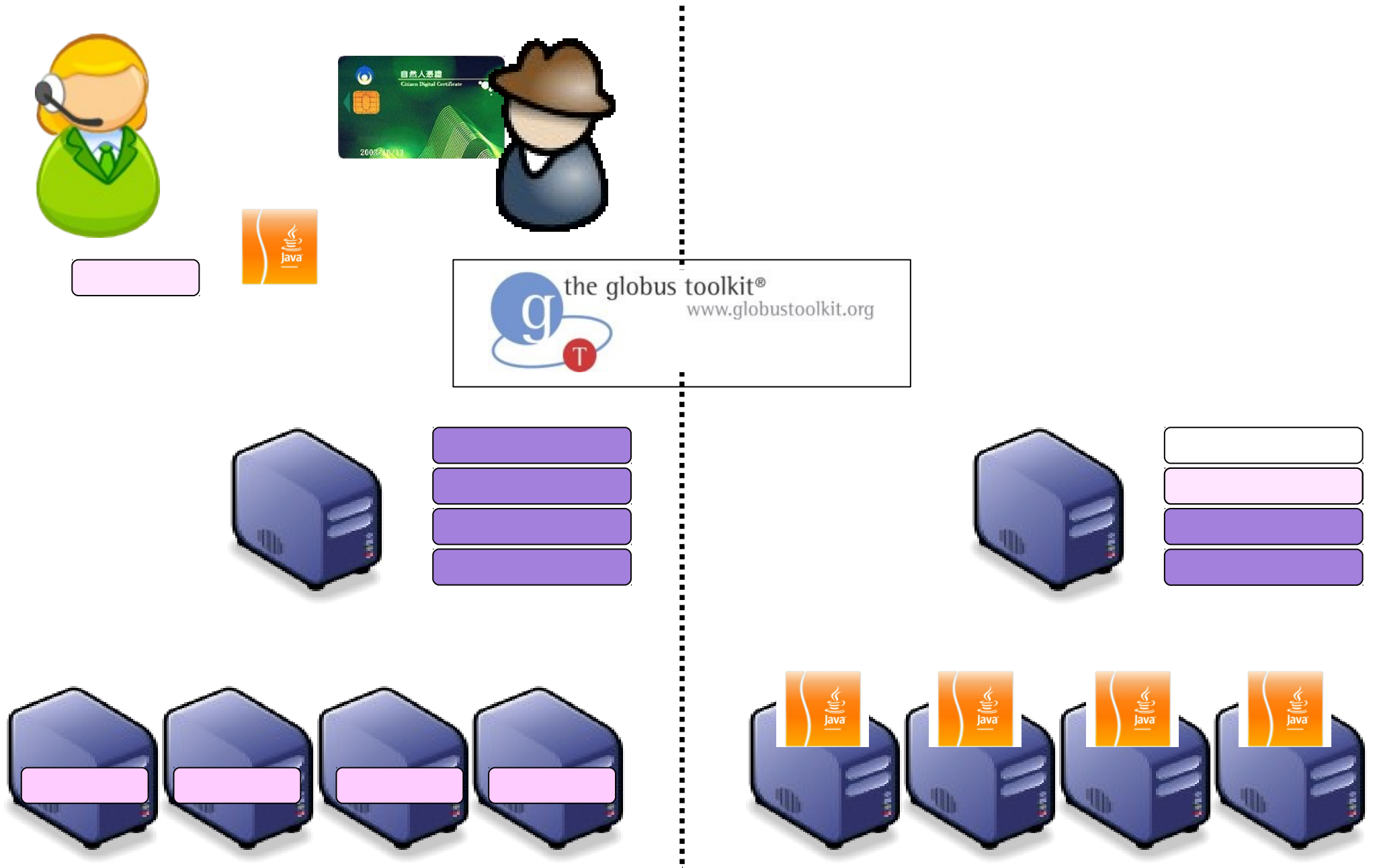
Grid 101 & mpiBLAST-G2

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw



Powered by DRBL

Grid \approx Cluster of Cluster



mpiBLAST-G2

- **mpiBLAST-G2 is an enhanced parallel program of LANL's mpiBLAST. It is based on **Globus Toolkit 2.x** and **MPICH-g2**.**
- **Bioinformatics Technology and Service (BITS) team of **Academia Sinica Computing Centre (ASCC), Taiwan****
- 參考：
 - **The MPIBLAST-g2 Introduction**
 - **MPIBLAST-g2 Example**
 - **mpiBlast-G2 with GT4**





PART 2.3 :

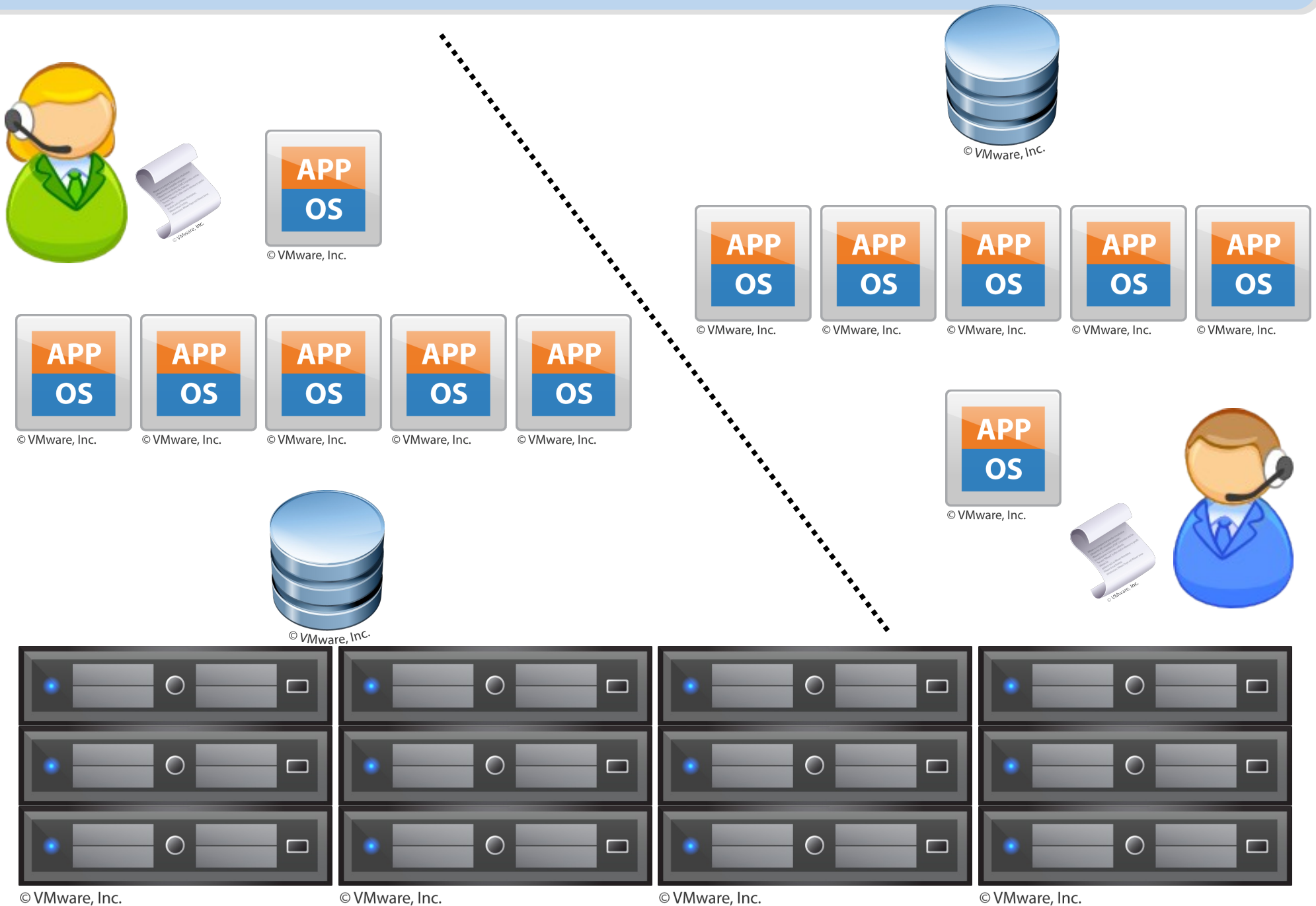
Cloud 101 & CloudBLAST

Jazz Wang
Yao-Tsung Wang
jazz@nchc.org.tw



Powered by DRBL

Cloud \approx Virtualization + Cluster



RunBLAST : mpiBLAST in Amazon EC2

The screenshot shows the RunBlast.com web interface. At the top, there are input fields for 'AWS Access Key' (054F2VTX3A67EW2MG802) and 'AWS Secret Key' (masked with asterisks). Below these are buttons for 'Add Machine', 'Refresh Table', 'Stop All', 'Reset Login Keys', and 'About'. A table displays the status of four EC2 instances, with the first instance highlighted. The table has columns for 'Status', 'Public Name', and 'Internal Name'. Below the table is a 'Worker Threads: 0' indicator and a log window showing repeated messages: 'machines status table updated'. The video player controls at the bottom show a progress bar and a timestamp of 00:18 / 06:13.

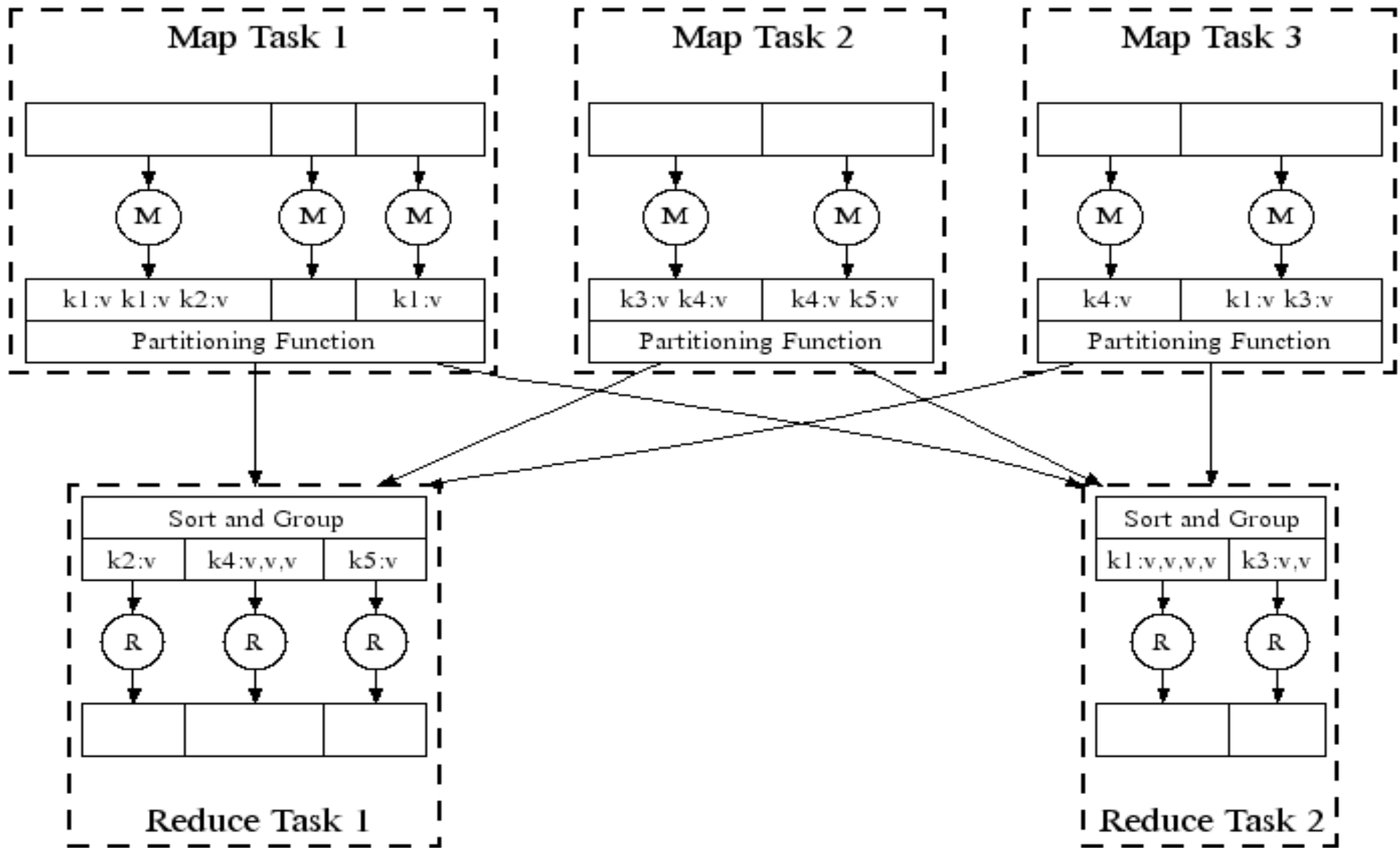
	Status	Public Name	Internal Name
1	running	ec2-67-202-36-153.compute-1.amazonaws.com	domU-12-31-38-00-6C-C8.compute-1.internal
2	running	ec2-67-202-24-53.compute-1.amazonaws.com	domU-12-31-38-00-29-B1.compute-1.internal
3	running	ec2-72-44-46-12.z-2.compute-1.amazonaws.com	domU-12-31-35-00-35-12.z-2.compute-1.internal
4	running	ec2-72-44-47-251.z-2.compute-1.amazonaws.com	domU-12-31-35-00-40-71.z-2.compute-1.internal
5			
6			

Worker Threads: 0

```
machines status table updated
machines status table updated
machines status table updated
machines status table updated
machines status table updated
machines status table updated
machines status table updated
machines status table updated
machines status table updated
machines status table updated
```

Video: <http://www.runblast.com/videos/runblast-blastwizard.swf>

Map/Reduce



CloudBLAST

- “**CloudBLAST: Combining MapReduce and Virtualization on Distributed Resources for Bioinformatics Applications**”, eScience 2008
- 特點：採用 **MapReduce** 演算法進行 **BLAST** 運算

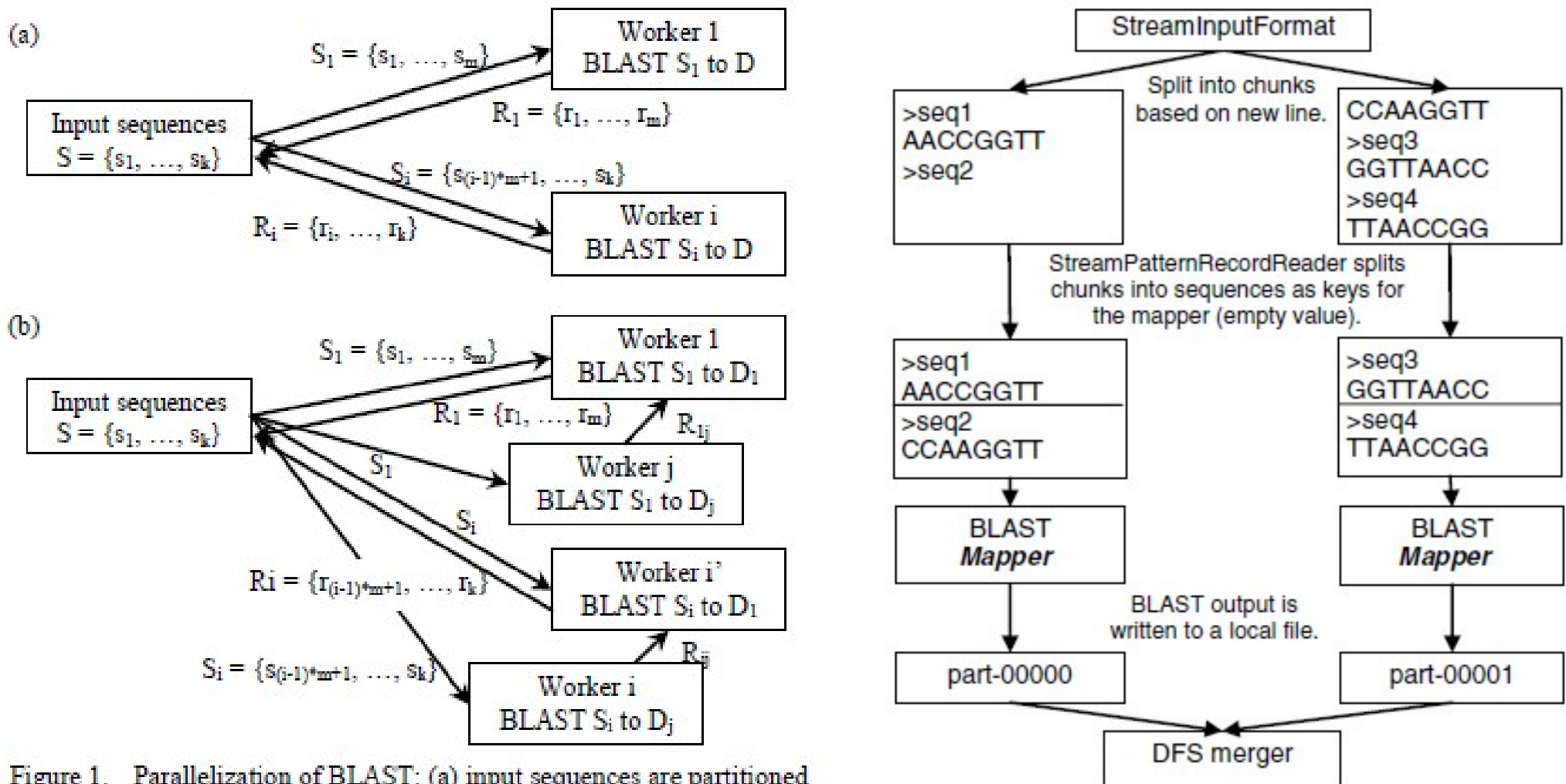


Figure 1. Parallelization of BLAST: (a) input sequences are partitioned