



MBI

Molecular Biomedical Informatics

分子生醫資訊實驗室

W e b P r o g r a m m i n g

網 際 網 路 程 式 設 計

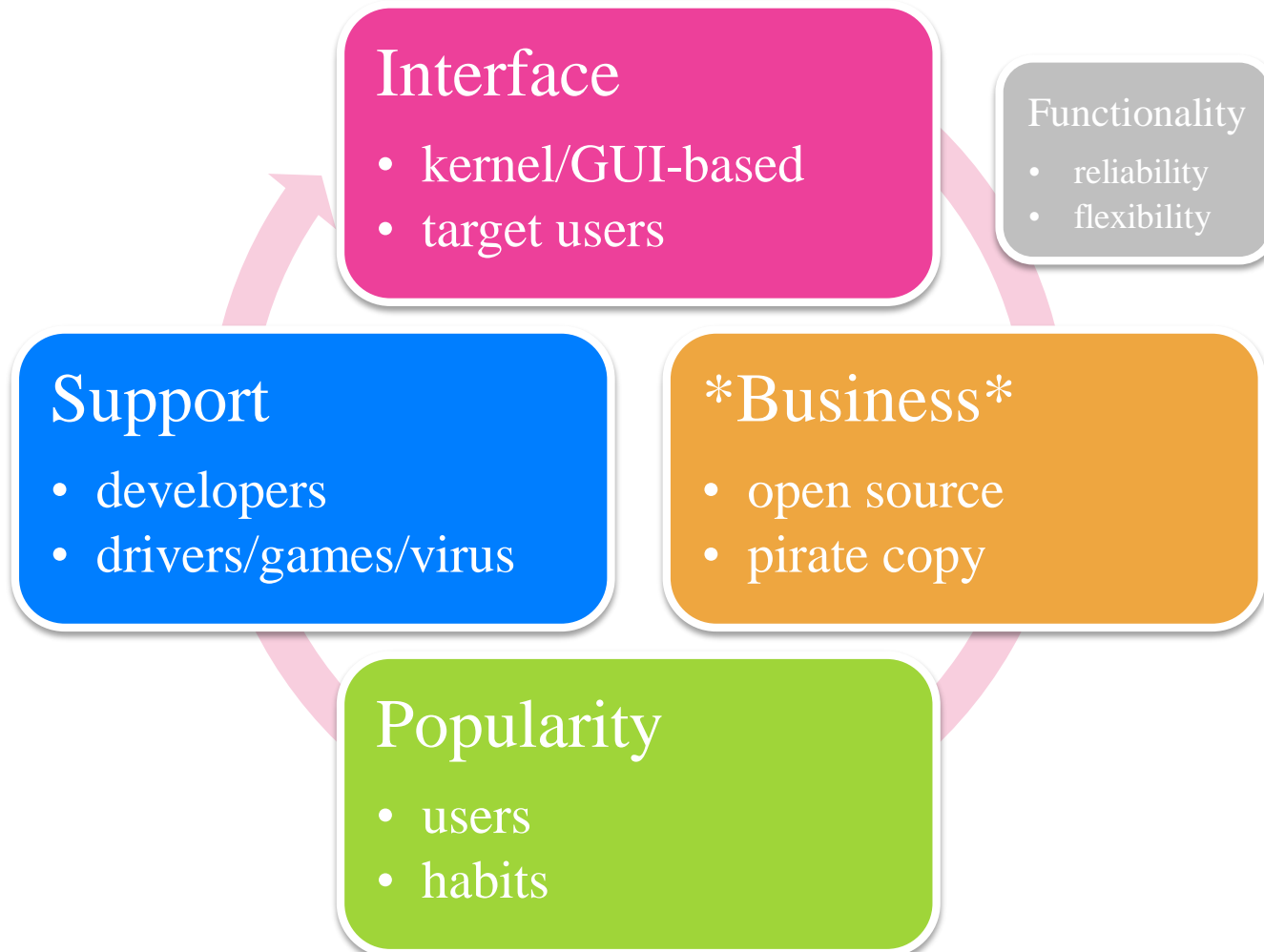
Environment 環境



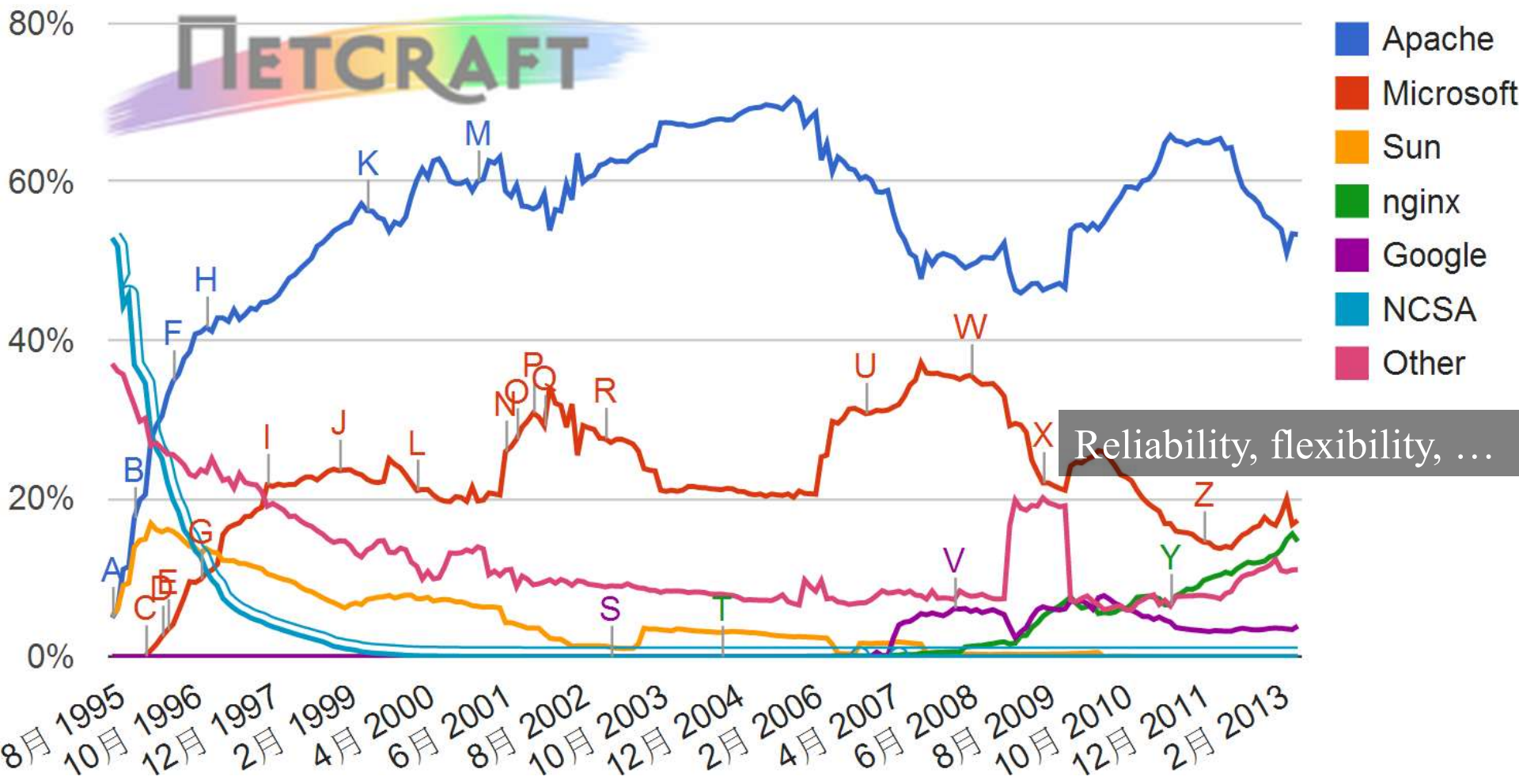
<http://www.junauza.com/2011/02/using-apache-web-servers-userdir.html>

Linux + Apache, of course
當然是 Linux + Apache

Linux vs. Windows



Web server developers: Market share of all sites



Various web servers on Linux

- Apache, nginx, Google, etc...
 - scale, flexibility, community support...
 - [Apache 2.4 Faster Than Nginx?](#)
 - you can use tools like [this](#) to test famous web sites
- They are usually easy to install
 - apt-get install nginx ← why you choose Windows?
- In this course, you don't have to worry about these details. However, if you are interested, welcome to discuss with me.



Do you

need to learn Linux?

Don't resist because of resistance
不要為反而反

Then, what to worry about
那麼，要煩惱什麼呢

This is I using Linux



It is not that scary

- Download [PieTTY](#) and login
 - just like any BBS client (even journalists can use them) with [secure shell \(SSH\)](#)
 - actually, this is the remote desktop (遠端桌面) of Linux
- Make the directory, or folder, to store your web site
 - `$ mkdir public_html` ← this name is just a configuration of Apache
- Write a HTML file
 - `$ vi public_html/index.html`
 - vi is a legend text editor, see the appendix
 - you may use any FTP client with SFTP support, such as [WinSCP](#)
- Now you can see your web site via browsers
 - [ttp://merry.ee.ncku.edu.tw/~xxx/](http://merry.ee.ncku.edu.tw/~xxx/)



Any Questions?

about our environment

Today's assignment

今天的任務

Design an about me page

- Request an account and put a web site online
 - contact [TA Kuo](#) via email or Facebook
- Reference
 - see [the 11th slide](#)
 - passwd ← remember to change your password / 別忘了先改密碼
 - `cd public_html; mkdir ex1; ln -s ex1 cur`
 - [Showcase of Inspiring “About Me” Pages](#)
 - [The Essence Of About Us Page With 12 Captivating Showcases](#)
 - [Great Examples of ‘About Us’ Page Design: A showcase for Inspiration](#)
- Your web site (<http://merry.ee.ncku.edu.tw/~xxx/cur/>) will be checked not before 23:59 12/2 (Tue). You may send a report (such as some important modifications) to [me](#) in case I did not notice your features.



In class demo

I'll demo a few (depending on the time) exercises in each class. However, you can see others' exercises anytime. If you found some techniques interesting, welcome to propose them in class.

Appendix

附錄

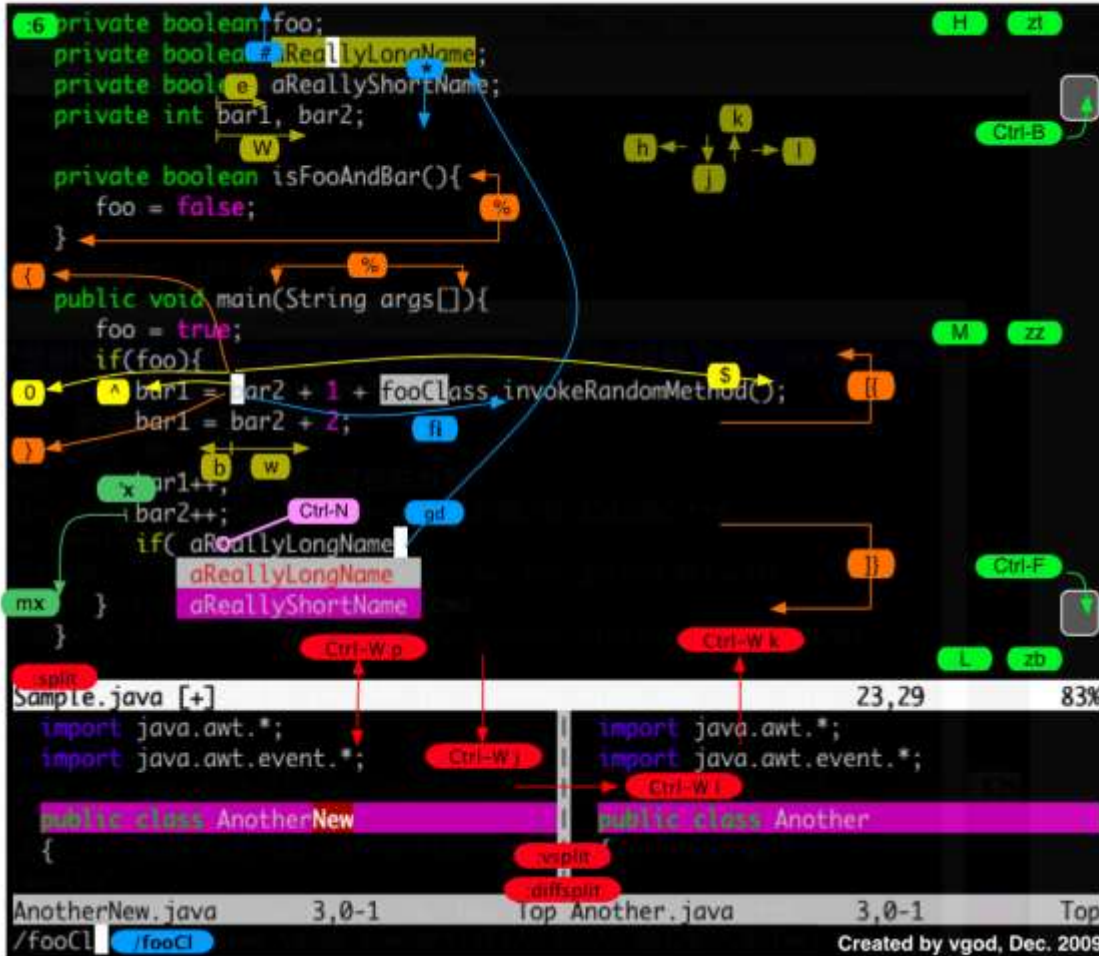
vi

- Regular text editors such as Notepad (記事本)
 - cursor movement/selection (mouse)
 - add/delete/edit text (keyboard)
 - copy/paste (menu or hotkey)
 - easy to learn (learnability)
 - hard to forget (memorability)
 - won't annoy users (satisfaction)
 - Everything looks well? The only problem is the efficiency.
- vi
 - Two modes, like gears in car. The most frustration for rookies is mode error.
 - Regular editors actually use 'temporary' modes.
 - Split a command into small units, such as character, word, line, screen...
 - w (jump to the next word)
 - } (jump to the next paragraph)
 - Too complicated! What's the benefit?

More vi

- vi separates actions and units
 - dw delete a word
 - y} yank (copy) a paragraph
 - d2w delete two words
 - think about how many hotkeys are required for regular text editors for all these combinations...
- vi has all should-have functionalities
 - split screen, search (regular expression, find previous/next), diff

Vim Visual Cheat Sheet



Movement/Range

Character

h j k l
← ↓ ↑ →

word, WORD(all non-blank ch)

w b next/prev word
W B next/prev WORD
e E end of word/WORD

Line

O \$ begin/end of line
^ begin (non-blank) of line

Paragraph, Block

{ } prev/ next paragraph
[[]] begin/end of block
% matching parenthesis

Window, File

H top of win zt scroll to top
M mid of win zz scroll to middle
L btm of win zb scroll to bottom
C-B C-F prev/next page
gg G begin/end of file
mx x mark/jump to x

Search

* g find current word backward/forward
fx to character x to right
gd to definition of current word
/xxx search xxx
n N next/prev search result

Mode Commands

ESC C-~~~~~ enter normal mode
v enter visual mode
V enter visual line mode
C-v enter visual block mode
i enter insert mode
R enter replace mode
a append
A append at end of line

General Commands

y yank/copy (range)
d delete/cut (range)
c modify (range)
x delete/cut (character)
D delete to end of line
C modify to end of line
p paste after cursor
J join lines
r replace (character)
> indent
< indent leftward
r redo
u undo

EX Commands

:w save(:wq save and quit)
:q quit(:q! quit anyway)
:e x edit file x
:n new window
:h vim help
:xx jump to line #xx

Auto-completion (insert mode)

C-N C-P auto-complete
next/prev keyword
C-X C-F auto-complete file name

Split window

vsp hsp vertically/horizontally split
diffs split and diff
C-W p to last accessed window
C-W w to next window

Of course you cannot remember all of them right now



Reference for vi

- [打造你的專屬VIM Editor](#)
- [\[HCI\] 淺談模式 "mode" 與文字編輯的技術與學習](#)
- [給程式設計師的Vim入門圖解說明](#)
- [Commanding Your Text Editor](#)
- [VIM Adventures](#)
- [Vimcasts - free screencasts about the text editor Vim](#)