

Operating Systems Concepts

An introduction to the software that makes computers usable, ensures that systems can be made secure and provides the environment for software to be a marketable commodity

Lecturer:

- Julie A. McCann (~jamm)

Acknowledgement: Paul Kelly

© DOC, Imperial College.

What is an Operating System

- The OS is software - it's a program It “**virtualises**” your computer
- Your computer needs to be virtualised... **trust me!!**
- Because that's what makes it possible to buy and sell computer programs
- And that's a major economic activity

© DOC, Imperial College.

There are lots...

- Microsoft: [Windows ME](#), [Windows 2000](#), Windows CE
- [Linux](#)
- [BSD \(FreeBSD, NetBSD\)](#)
- Sun: Solaris
- IBM: AIX
- Compaq: Tru64 Unix, OpenVMS
- Hewlett-Packard: HP-UX
- Apple: MacOS-8, MacOS-X
- Symbian: EPOC
- [PalmOS](#)
- QNX
- VXWorks
- LynxOS
- [MVS](#), AS/400
- Many many more, mostly more-or-less Unix/POSIX compatible eg [MESHIX](#)
- Literally thousands of variants and research prototypes eg MACH, Plan9, Pebble, [Go!](#)
- Sun (again): the Java Virtual Machine is it??? JOS??

} Various kinds of Unix

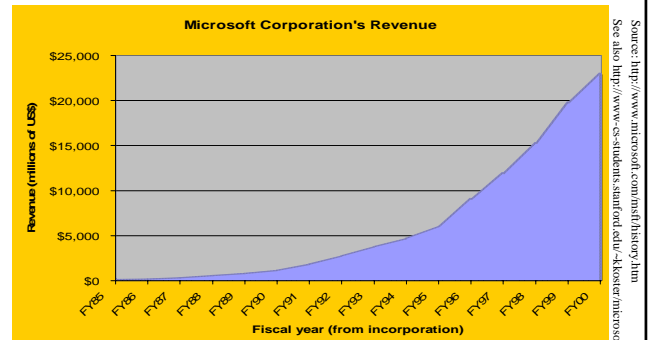
} Designed for handhels

} Embedded real-time

© DOC, Imperial College.

Opening up the software market

- More than 90% of all software products sold are designed for just one virtual machine



- This has been quite good for their business...

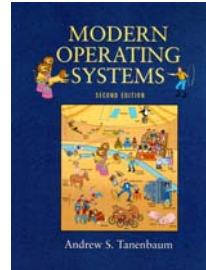
© DOC, Imperial College.

Overview of Course

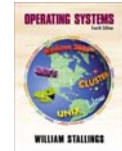
- What is an OS
- Terminology
- Kinds of OS
- How does the OS work
- What principles underpin OS Operation
- What does an OS look like 'under the bonnet'

© DOC, Imperial College.

Recommended textbooks



Modern Operating Systems. A Tanenbaum
Prentice-Hall



•Operating Systems (4th edition). William Stallings, Prentice-Hall 2001



•Operating Systems: a modern perspective (2nd edition). Gary Nutt, Addison Wesley, 2000

© DOC, Imperial College.

THERE IS NO SPOON

- **Read** books – course built on classic authoritative textbook
- **Exam** designed to test your understanding and ability to apply it to new ideas
- Some exam questions will introduce an idea **not covered** in the course
- But will ask you to explore an idea using what you've learned
- The **book** might have even covered that!

Three waves of computing

© DOC, Imperial College.

The first wave!

© DOC, Imperial College.

Chapter 1: - Introduction to OS

- What did the first operating system look like?

1945 to 1955

- No operating system
- human operators

© DOC, Imperial College.

1956 to 1965

- Transistors and batch systems ,
- Clear distinction between designers, builders, operators, programmers, and maintenance personnel,
- I/O channel,
- Read ahead / spooling,
- Interrupts/ exceptions,
- Minimal protection,
- Libraries / JCL



© DOC, Imperial College.

1965 to 1980

- ICs and Multiprogramming,
- System 360 and S/370 family of computers,
- Spooling (simultaneous peripheral operation on-line),
- Time sharing, On-line storage for System programs,
- User programs and data, Program libraries,
- Virtual memory,
- Multiprocessor configurations e.g. [MULTICS](#)



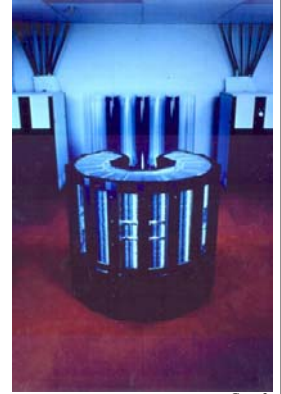
© DOC, Imperial College.

When people looked like [this](#), we saw the next generation
THE SECOND WAVE OF COMPUTING

© DOC, Imperial College.

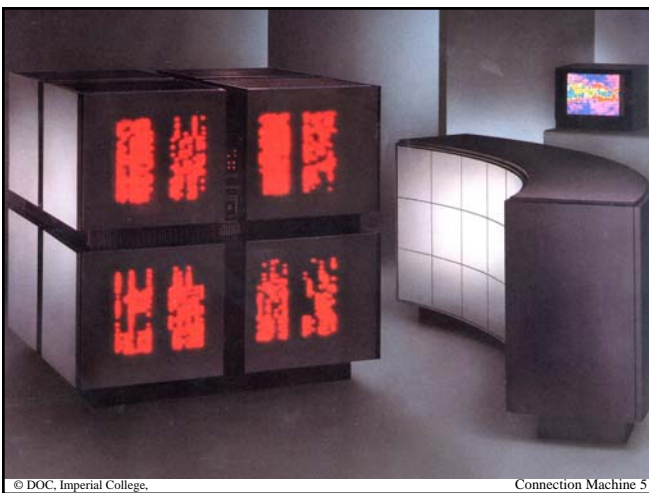
1980-now

- Personal computers and workstations
- MS-DOS and Unix
- Massively parallel systems
- Pipelining
- Array processing / SIMD
- General multiprocessing / MIMD
- Symmetric multiprocessing / SMD
- Any [process](#) and any [thread](#) can run on any available processor
- Computer networks (communication aspect) -- network operating systems
- Distributed computing -- distributed operating systems



Cray 2

© DOC, Imperial College.



Connection Machine 5

© DOC, Imperial College.

What is an OS

- A [program](#), C or C++
- Loads and runs other programs
- Application programs might wish to:
 - Draw on the screen
 - Interact via a keyboard, mouse etc
 - Access the hard disk (files)
 - Communicate with other application programs
- The OS should provide a [consistent](#) way to do this
 - Works on whatever hardware you have.

© DOC, Imperial College.

Virtual or Abstract Machines

- Effectively wraps the hardware to provide a 'virtual' machine
- Consistent interface to devices and services
- What are the limits/difficulties?

© DOC, Imperial College.

Challenges to the VM approach

- Performance
 - machine too slow to run application,
 - I want to safely run lots of applications!
- Resources - Machine too little memory

© DOC, Imperial College.

Overcoming VM challenges

- Most OSs solve this problem
- Uses virtual memory – paging
 - Application or program requiring 100MBytes of memory can run on just 10 Mbytes of physical memory using VM
 - Slowly mind!
- What if I want to run more than one program?
 - Can you read mail and eat? – you have one brain
 - Processor switch from one process to another
 - Regularly, use a timer to interrupt the processor
 - Most do this fairly (according to application needs)
 - Time-slicing – scheduling and priority

© DOC, Imperial College.

Overcoming VM challenges

- What if the other programs behave maliciously e.g. try to steal my secrets?
 - OS has privileged control over 'address translation's hardware
 - Each application allowed access to only own data
 - Data is authenticated (i.e. user is allowed to access it)
- What if two applications try to access the same device at the same time eg printer?
 - Fix with mutual exclusion – deadlock!

© DOC, Imperial College.

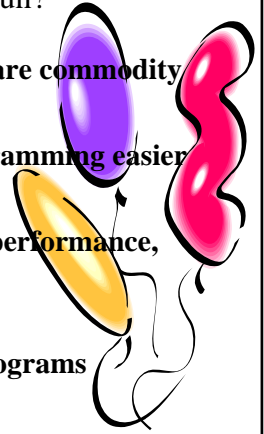
Lets look at an Operating System at work

- Julie put some program running on system
- Do ctrl/alt/del
- Start app
- Start another instance of application

© DOC, Imperial College.

Why are OSs fun?

- **Make application software commodity**
- **Abstraction make programming easier**
- **Challenging - security, performance, reliability**
- **Do fancy things with programs**



© DOC, Imperial College.

<http://www.doc.ic.ac.uk/~jamm/teaching/xpsucks.html>

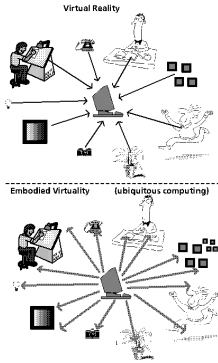
© DOC, Imperial College.

So what's the third wave?

- [Mark Weiser](#)

© DOC, Imperial College.

Ubiquitous Computing?



- Involves many disciplines.
- Inspired by the social scientists, philosophers, and anthropologists
- paradigm shift?
 - currently we expect the **user to find ways** to use the computer
 - however we **currently do not emphasise how the computer can find its own way** to serve the user
 - focus on HCI
 - focus on security, privacy → big brother?

© DOC, Imperial College.

Current Technology



Laptop

Mobile phone / PDA

Best friend

Personal digital assistant (PDA)

DoCoMo video phone

Thanks to M. Sloman for slides

© DOC, Imperial College.

Wearable



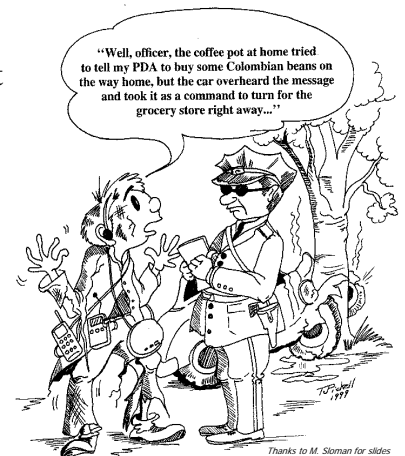
Designer Gear

Thanks to M. Sloman for slides

© DOC, Imperial College.

Intelligent Environment

- Fridge and cupboards tracks consumption and reorder your groceries
- Your car computer reminds you to pick up your order on the way home when you are near the supermarket.



Thanks to M. Sloman for slides

© DOC, Imperial College.

Intelligent Environment - 2

- Lights, air conditioning, TV automatically switch on and off when you enter or leave rooms
- Sit on your favourite chair and TV switches on to the program you usually watch at this time of the day
- Use communicator/pda for phone, remote control, keys payments, passport, health records, authenticator.
- Route input from 'virtual' keyboard to nearest suitable display.
- *Automatic* detection of new items to control and physical layout in a room or office, using computer vision.