Introduction

Johan Montelius

KTH

2017
Choosing an operating system
Choosing an operating system
Choosing an operating system
Choosing an operating system
Choosing an operating system
Choosing an operating system
Choosing an operating system
Choosing an operating system
What is important when choosing an operating system?
Choosing an operating system

What is important when choosing an operating system?

- What everyone else is using.
What is important when choosing an operating system?

- What everyone else is using.
- Look and feel, ease of use.
Choosing an operating system

What is important when choosing an operating system?

- What everyone else is using.
- Look and feel, ease of use.
- Hardware
What is important when choosing an operating system?

- What everyone else is using.
- Look and feel, ease of use.
- Hardware
- Cost
Choosing an operating system

What is important when choosing an operating system?

- What everyone else is using.
- Look and feel, ease of use.
- Hardware
- Cost
- Availability of programs
Choosing an operating system

What is important when choosing an operating system?

- What everyone else is using.
- Look and feel, ease of use.
- Hardware
- Cost
- Availability of programs
- How it works under the hood.
What is important when choosing an operating system?

- What everyone else is using.
- Look and feel, ease of use.
- Hardware
- Cost
- Availability of programs

- 

- :
Choosing an operating system

What is important when choosing an operating system?

- What everyone else is using.
- Look and feel, ease of use.
- Hardware
- Cost
- Availability of programs
- How it works under the hood.
What is an operating system?

Abstraction, virtualisation and managing of resource.

Abstraction
How do we create an abstraction layer that provides an environment for programming of a process?

Virtualisation
How do we create the image of dedicated hardware while in fact we have several processes sharing the same hardware?

Resource management
Given that we have limited amount of resources, how do we share them in a fair way?
What is an operating system?

Abstraction, virtualisation and managing of resource.
What is an operating system?

Abstraction, virtualisation and managing of resource.
What is an operating system?

Abstraction, virtualisation and managing of resource.

- Abstraction
What is an operating system?

Abstraction, virtualisation and managing of resource.

- **Abstraction**
  - How do we create an abstraction layer that provides an environment for programming of a process?
What is an operating system?

Abstraction, virtualisation and managing of resource.

- Abstraction
  - How do we create an abstraction layer that provides an environment for programming of a process?

- Virtualisation
  - How do we create the image of dedicated hardware while in fact we have several process sharing the same hardware?
Abstraction, virtualisation and managing of resource.

- **Abstraction**
  - How do we create an abstraction layer that provides an environment for programming of a process?

- **Virtualisation**
  - How do we create the image of dedicated hardware while in fact we have several process sharing the same hardware?

- **Resource management**
  - Given that we have limited amount of resources, how do we share them in a fair way?
The Operating System
A well structured world

The Operating System
An operating system

A well structured world

a clean interface

The Operating System
An operating system

A well structured world

a clean interface

The Operating System

Hardware
An operating system

A well structured world

a clean interface

The Operating System

a complete mess

Hardware
to implement an operating system

Why is it hard to implement an operating system?
Why is it hard to implement an operating system?
layers of abstractions

OS A
layers of abstractions

Nice world

OS A
layers of abstractions

Nice world

a clean interface

OS A
layers of abstractions

Nice world

a clean interface

OS A

a complete mess

Hardware X
layers of abstractions

- Nice world
  - a clean interface
- OS A
- a complete mess
- Hardware X
- Hardware Y
layers of abstractions

Nice world
a clean interface

OS A

a complete mess

Hardware X

another mess

Hardware Y
layers of abstractions

Nice world

a clean interface

OS A

a complete mess

Hardware X

OS A

another mess

Hardware Y
layers of abstractions

Nice world

a clean interface

OS A

a complete mess

Hardware X

same interface

OS A

another mess

Hardware Y
layers of abstractions

- Nice world
  - a clean interface
  - OS A
    - a complete mess
  - Hardware X
- Same world
  - same interface
  - OS A
    - another mess
  - Hardware Y
layers of abstractions

Nice world
- a clean interface
- OS A
- a complete mess
- Hardware X

Same world
- same interface
- OS A
- another mess
- Hardware Y

Hardware Y
layers of abstractions

- Nice world
  - a clean interface
  - OS A
  - a complete mess
  - Hardware X

- Same world
  - same interface
  - OS A
  - another mess
  - Hardware Y

- Same world
  - same interface
  - OS A
  - another mess
  - Hardware Y

- Hardware Y
  - same mess
layers of abstractions

Nice world
a clean interface
OS A
a complete mess
Hardware X

Same world
same interface
OS A
another mess
Hardware Y

Windows
same mess
Hardware Y
layers of abstractions

Nice world
a clean interface
OS A
a complete mess
Hardware X

Same world
same interface
OS A
another mess
Hardware Y

Same world
same interface
Windows
same mess
Hardware Y

Same world
same interface
Hardware Y

Windows
same mess
Hardware Y

strange interface
layers of abstractions

- **Nice world**
  - a clean interface
  - OS A
  - a complete mess
  - Hardware X

- **Same world**
  - same interface
  - OS A
  - another mess
  - Hardware Y

- **Different world**
  - strange interface
  - Windows
  - same mess
  - Hardware Y
yet another layer
yet another layer

nice B

OS B
yet another layer

nice B

OS B

standard C

OS C
yet another layer

- clean A
  - OS A
- nice B
  - OS B
- standard C
  - OS C
yet another layer

Java API

Java VM

clean A

OS A

nice B

OS B

standard C

OS C
yet another layer

Java VM

Java API

clean A

OS A

Java VM

Java API

nice B

OS B

standard C

OS C
yet another layer

Java API
Java VM
clean A
OS A

Java API
Java VM
nice B
OS B

Java API
Java VM
standard C
OS C
yet another layer

Java world

Java API
Java VM
clean A
OS A

Java API
Java VM
nice B
OS B

Java API
Java VM
standard C
OS C
virtual operating systems

clean A

OS A
virtual operating systems

clean A

OS A

nice B

OS B
<table>
<thead>
<tr>
<th>clean A</th>
<th>nice B</th>
<th>standard C</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS A</td>
<td>OS B</td>
<td>OS C</td>
</tr>
</tbody>
</table>
virtual operating systems

World A
  clean A
  OS A

nice B
  OS B

standard C
  OS C
virtual operating systems

World A

OS A

clean A

Virtual OS A

nice B

OS B

standard C

OS C
virtual operating systems

World A
  clean A
  OS A

Virtual OS A
  clean A
  nice B
  OS B

Virtual OS A
  clean A
  standard C
  OS C
provide isolation and control

clean A

OS A
provide isolation and control

Virtual OS A

clean A

OS A
provide isolation and control

Virtual OS A
Virtual OS A

clean A

OS A
provide isolation and control

Virtual OS A  Virtual OS A  Virtual OS B

clean A

OS A
provide isolation and control

Virtual OS A

Virtual OS A

Virtual OS B

mean and lean

— Hypervisor —
Some history - way back

- 1960
- Large single job machines, batch processing
- Operating system often only libraries to handle hardware.

BESK - KTH 1953, 512 word memory
also long time ago

- 1970 - 1980
- Multiuser systems, minicomputers (very large)
- Time sharing, virtual memory, hard drives, ...
- Birth of Unix and C

PDP-10 - KTH 1970 ca, 256 Kiword memory
before you were born

Mac or IBM PC?

- 1980 - 1990
- The personal computer.
- Manual switching between programs.
- MS-DOS, Mac OS, ..
The *nix war

AIX, HP-UX, Solaris, Ultrix ...

- 1980 - 2000
- Unix flavors become the leading operating systems for everything but personal computers.
- Everyone wants a standard - their own.
at the same time

Gnu is Not Unix

- 1983 -
- Providing a free (as in speech, not beer) operating system with everything you could need.
- Everything was in place ... the kernel will soon be ready.
Hello everybody out there using minix -
I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu):

PS. Yes - it's free of any minix code, and it has a multi-threaded fs. It is NOT portable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that's all I have :-(.

- Linus Torvalds

1991
Hello everybody out there using minix -

I’m doing a (free) operating system (just a hobby, won’t be big and professional like gnu) for 386(486) AT clones. : : :

PS. Yes - it’s free of any minix code, and it has a multi-threaded fs. It is NOT portable (uses 386 task switching etc), and it probably never will support anything other than AT-harddisks, as that’s all I have :-(.  

- Linus Torvalds
Linux - the kernel for GNU

- 1993 -
- Linus Torvalds
- A monolithic system targeting the Intel 386 CPU.
- Linux was born and became the kernel for GNU.
Share of Top500 Super Computers
Linux - give it a try

Three options:
Linux - give it a try

Three options:

- Install a *virtual machine* on top of your regular system.
Three options:

- Install a *virtual machine* on top of your regular system.
- Take an old laptop and boot it from scratch.
Three options:

- Install a *virtual machine* on top of your regular system.
- Take an old laptop and boot it from scratch.
- Join the good side.
What’s next