Poseidon House
Castle Park
Cambridge CB3 0RD
United Kingdom

TELEPHONE: **Cambridge (01223) 515010**
INTERNATIONAL: **+44 1223 515010**
FAX: **+44 1223 359779**
E-MAIL: **apm@ansa.co.uk**

Training

ANSAwise - Distributed and Networked Operating Systems

Chris Mayers

Abstract

This module of the ANSAwise training programme explains how operating systems fit within a distributed systems architecture.

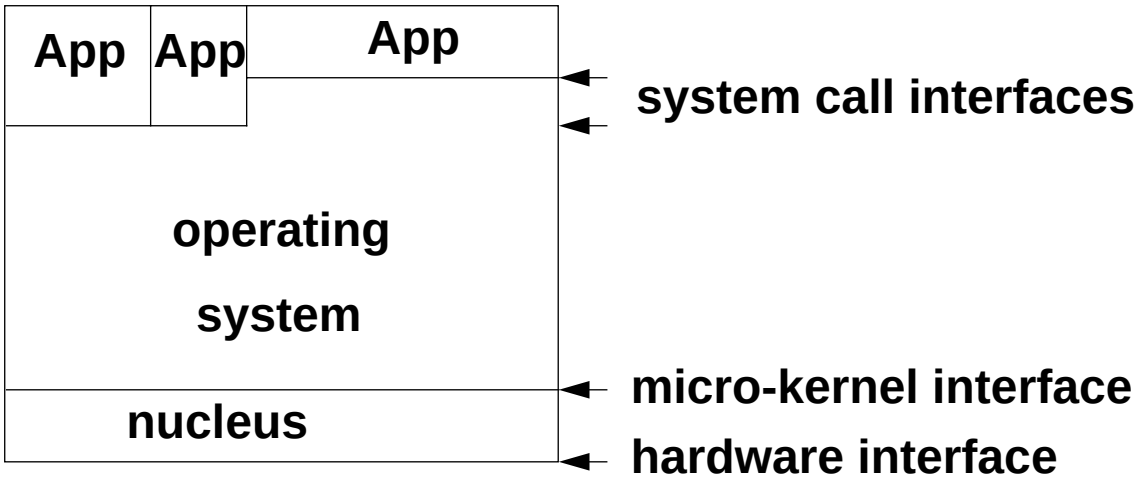
Networked operating systems might appear to offer all the benefits of distribution, but this is an illusion.

At the end of this module, participants will be able to explain the benefits and limitations of current operating systems, including networked operating systems.

APM.1317.02 **Approved** 20th November 1995
Briefing Note

Distribution:
Supersedes:
Superseded by:

Distributed and Networked Operating Systems



In this session

- *Explain the role of LANs in a distributed system*
- *Explain the trends in the new generation of operating systems*
- *Show how distributed systems can exploit these trends*
- *Discuss your views of these trends*

LAN/Network Operating Systems

- ***PC centralized networks***
  - **Microsoft/IBM (NT Advanced Server, LAN Server)**
  - **Novell NetWare**
  - **Banyan Vines**
- ***Unix decentralized networks***
- ***Mainframe networking***

Evolution of the PC network

- ***1985***
  - **Aim was *resource sharing* (printers, disks, files)**
- ***1990***
  - **Resource sharing still important**
  - **Additional aim was *information sharing* (databases, host systems)**
- ***1995***
  - **Resource and information sharing still important**
  - **Additional aim is *process sharing* (workflow, conferencing)**

Resource sharing

- *The earliest PC LANs did exactly that and no more*
  - disks and printers were shared as devices
  - programs accessed shared devices exactly as local devices
- *But when two users accessed a device at the same time...*
  - disk files would be corrupted
  - printout would be interleaved
  - ... so, partitioning schemes were needed; data sharing was impractical
- *Unsurprisingly, such systems did not last long....*
- *...They were superseded by server-based network operating systems*

PC LAN network operating systems

- *These use a centralized machine (a 'server'), supporting*
  - file system
  - printing
  - database
  - other specialized services
- *The server base operating system varies...*
  - Novell NetWare: dedicated NetWare OS
  - Microsoft NT Advanced Server: Windows NT
  - Banyan Vines: Unix variant
- *... So does the network protocol (IPX/SPX, NETBIOS, modified IP)*
- *Client workstations are usually DOS/Windows*
  - but gateways allow any client operating system

Peer-to-peer network operating systems

- *In a peer-to-peer network, any machine can run a server process*
- *For example:*
  - Mac Appletalk
  - Unix NFS
  - Microsoft Windows for Workgroups
- *Again, the network protocol varies (Appletalk, NFS, NETBIOS)*

What are the disadvantages of a centralized network OS?

- ***Disadvantages include***
  -
  -
  -

What are the disadvantages of a peer-to-peer network OS?

- ***Disadvantages include***
  -
  -
  -
  -

A radical view - “the LAN is not a network”

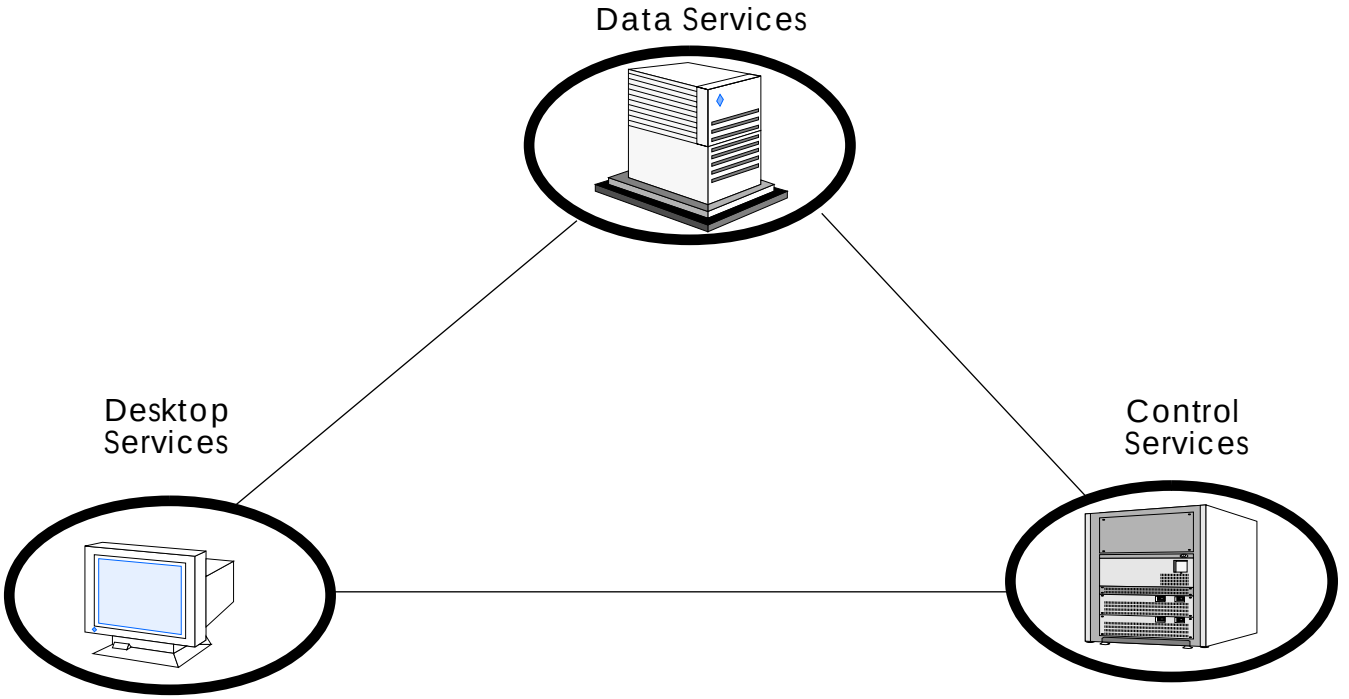
- *By 1993:*
  - More than half the PCs were connected to a LAN
  - LANs were generally reliable
  - Resource sharing (e.g. printers) was universal, and often transparent to the end-user
- *So if you can’t tell the LAN is there, it isn’t really a network!*

Do you agree?

Distributed Operating Systems

- *The research community has been studying distributed operating systems since the late 1970's*
  - The ANSA project started looking at this in 1983
- *Attempts have been made to produce fully-transparent distributed operating systems*
  - location transparency for files, processes, memory, resources

One distributed operating system?



The problem with true distributed operating systems

- *Complete location transparency can be achieved...*
- *... but even this is impractical*
  - it places far too many overheads on the system (at several levels)
  - it won't scale beyond a small group of machines
  - it won't scale over a wide-area network
  - it makes it difficult to support a diverse collection of hardware
  - it doesn't allow for interworking with other operating systems
- *Research continues, but full transparency is no longer a goal*

Three kinds of systems

- ***Centralized***
  - **single location**
- ***Decentralized (peer-to-peer)***
  - **multiple location, un-coordinated components**
- ***Distributed***
  - **multiple location, coordinated components**

New generation operating systems

- *They have common aims...*
  - Dependability, to support mission-critical applications
  - Portability, to support new and old hardware platforms
  - Modularity, a component-based architecture
  - Scalability, from low-cost to high-performance
  - Ease of use, having a familiar graphical user interface
- *Applications must inherit these benefits*
  - but they do not come automatically

IBM OS/2, Workplace OS

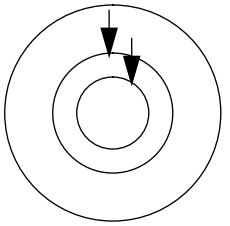
- *"A better DOS than DOS, a better Windows than Windows"*
- *Development started in 1985 in conjunction with Microsoft*
- *OS/2 1.0 shipped in 1988*
  - IBM have continued the development ever since

Microsoft Windows NT

- *Development started in 1988*
- *Runs PC applications in compatibility mode*
  - PC applications under emulation (hardware emulation on non-80x86)
- *... "The definitive 1970s operating system"*
  - Choice of internal mechanisms based on long experience
- *Shipped in 1993; sales were slow at first*
  - NT 3.1 required too much memory

Unix

- ***The traditional Unix kernel is monolithic***
  - **Typically 2 megabytes or more**
  -
  -
  -
- ***Often uses hardware-supported ring protection***



Application

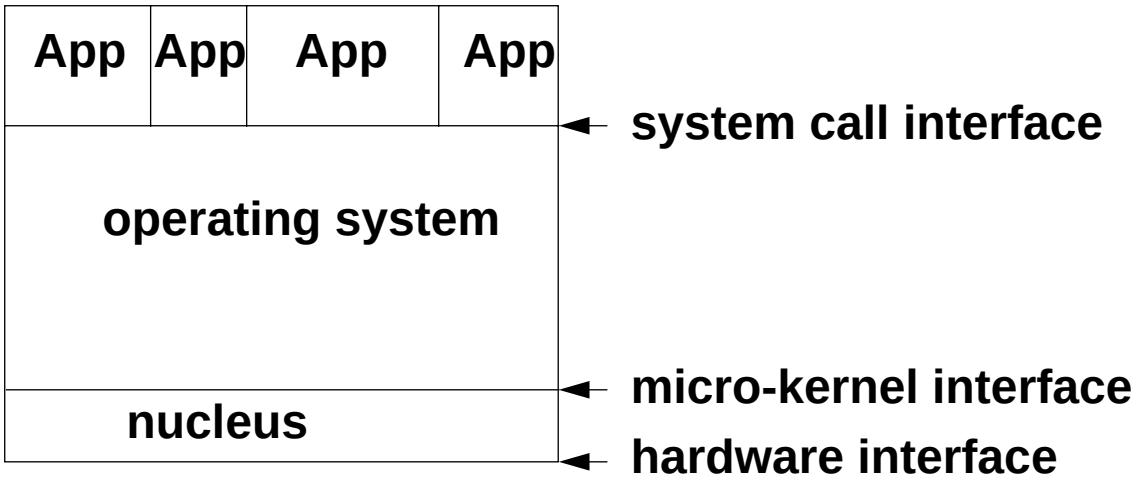
System

Kernel

- ***New-generation Unix uses a micro-kernel***

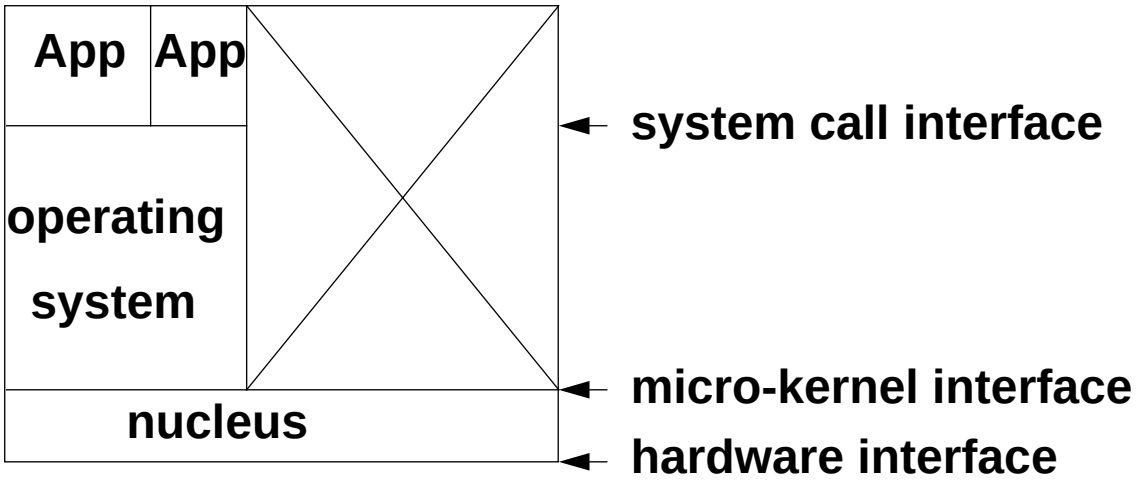
The structure of a micro-kernel operating system

- *From a complete micro-kernel Unix system...*



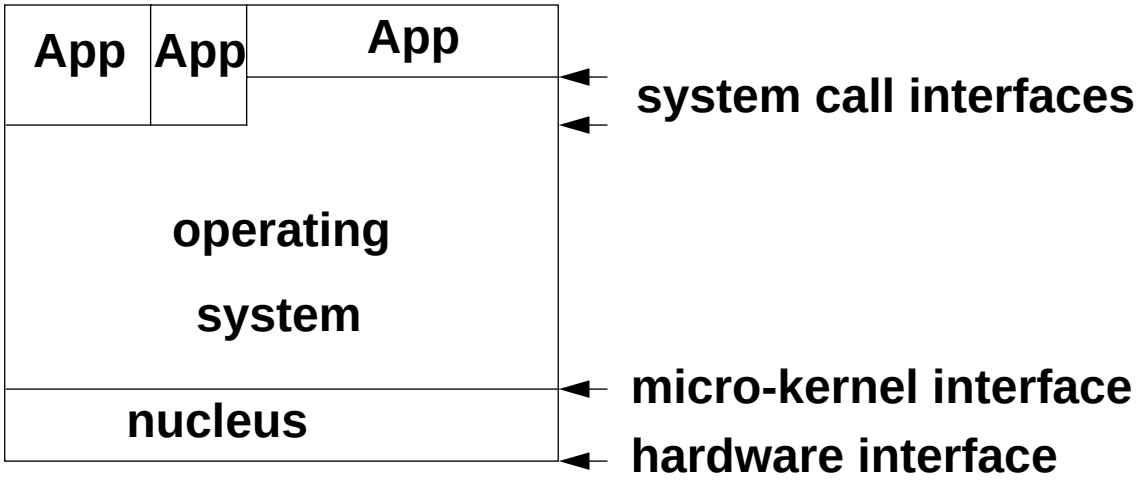
Micro-kernel for modular operating systems

- *.... unnecessary services can be removed...*



Micro-kernel for modular operating systems

- *.... or different operating systems emulated simultaneously*



- *for example, different variants of Unix*

The importance of micro-kernel

- *A micro-kernel provides the minimum necessary facilities*
  - interprocess communication
  - low-level scheduling and process management
  - basic memory management
  - low-level I/O
- *Once again, the kernel can be small (less than 100 Kbytes)*
  - generic low-cost embedded systems can be built
  - the micro-kernel is isolated; it can be developed for reliability

Operating System Dependability

- ***Micro-kernels help here***
  - **they help contain the effects of faults**
- ***Operating system design and engineering has improved***
- ***Product maturity helps too***
  - **...every application exercises the operating system**
  - **...bugs get fixed eventually (“time cures all wounds”)**
- ***Fault tolerance is still an issue***
- ***Application dependability is still an issue***

Operating System Portability

- ***Operating system suppliers gain a wider market from portability...***
  - **... but what do users gain?**
- ***Basically, investment protection***
  - **so you can move your application if you must**
- ***Unix is portable, but...***
  -
  -
  -

Operating System Modularity

- *Micro-kernels have simplified the operating system itself*
  - but now what about the applications?
- *We need to split applications into smaller pieces*
  - for just the same reasons
- *Component-based (object-based) operating systems may help...*
  - Microsoft Windows OLE, and later Cairo
  - Taligent
  - NextStep
  - OpenDoc
- *... but they aim mainly for a modular user interface*
  - not at open distributed processing

Multiprocessors and Multicomputers

- ***Multiprocessors***
  - **have a single address space**
  - **communicate via shared memory**
- ***Multicomputers***
  - **have multiple address spaces (one per machine)**
  - **communicate via message passing**
  - **.... so LANs are a particular kind of multicomputer**
- ***A distributed system combines both as needed***

Multiprocessors today

- *Small-scale multiprocessors: up to 4 processors, say*
  - typically used for high-throughput LAN servers
- *Medium-scale multiprocessors: up to 16 processors, say*
  - typically used for on-line transaction processing (OLTP)
- *Large-scale multiprocessors: hundreds of processors*
  - typically used for parallel database query (data mining/OLAP)

The significance of multiprocessors

- *The hardware offers scalable performance upwards*
  - but what application would need such a wide range of performance on a single machine?
- *The straightforward benefit is incremental (by adding processors)*
  - to speed up one application
- *Integration within a distributed system is what matters*
  - and this may not be difficult to achieve

Your views?

- ***What benefits do you seek?***
- ***Which operating systems and platforms do you expect to deliver them?***
- ***What else do you think is needed?***

Summary

- ***The new generation of operating systems and hardware platforms provides useful technology***
  - **we must be able to exploit them...**
  - **... but they don't provide any assistance in building distributed systems**
- ***The applications developer's job is no simpler***
  - **Operating systems developers have solved *their* problems, not yours**
- ***To build distributed systems, we need something extra***

Background information

- ***For more information***

  - **on LAN operating systems, see *Distributed Computing: A Practical Synthesis*, by Amjad Umar (Prentice Hall)**

  - **on distributed operating systems, see *Distributed Systems Concepts and Design*, by Coulouris, Dollimore, and Kindberg (Addison-Wesley)**