



---

**Poseidon House  
Castle Park  
Cambridge CB3 0RD  
United Kingdom**

TELEPHONE:  
INTERNATIONAL:  
FAX:  
E-MAIL:

**Cambridge (01223) 515010  
+44 1223 515010  
+44 1223 359779  
apm@ansa.co.uk**

---

## **Training**

# **ANSAwise - Distributed Object Systems in Action**

**Chris Mayers**

### **Abstract**

Organizations are aware that distributed object systems may offer technical benefits, but may be unsure exactly what is on offer, what the difficulties are, and whether they can actually deliver the promised benefits.

This is the first module of the ANSAwise training programme, and outlines the technical case for distributed systems.

This variant specifically focuses on the telecommunications market. Distributed systems other than CORBA are mentioned briefly.

[Unlike other variants of this module, the business case is mentioned only briefly. It also describes client/server systems briefly.]

---

APM.1742.01

**Approved**  
Briefing Note

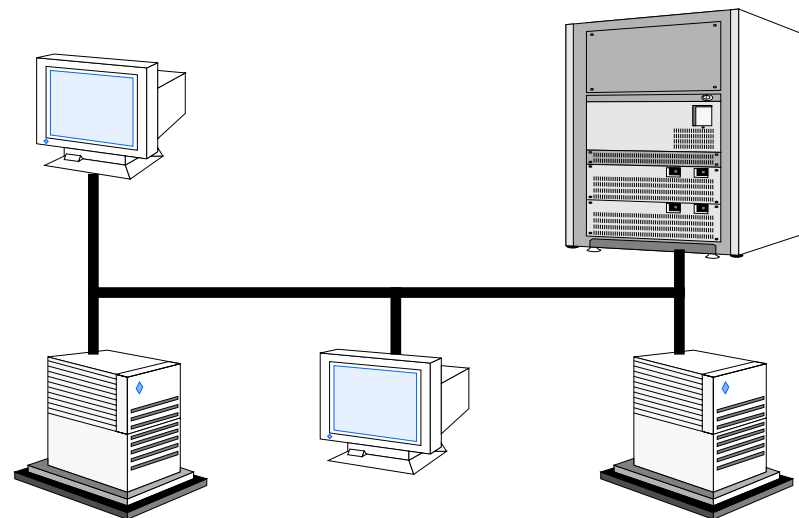
27th March 1996

---

**Distribution:**  
**Supersedes:**  
**Superseded by:**



## Distributed Object Systems in Action





## In this session

- Explain basic client-server approaches
- Explain in what ways distributed systems are different
- Examine some real applications



# What's the real business challenge?

## *Coping with change*

## The pressures for change

- **Political, economic, social, and technological...**

- **Globalization**



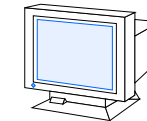
- **Rapid organizational change**



- **Increased customer expectations**



- **Inexpensive computing and telecommunications**





## Coping with change

- **The key is system evolution**
  - **not revolution**
  
- **Different parts of the system will evolve independently**
  - **this requires *interoperability***



## What is client/server computing?

- 'The splitting of an application into tasks that are performed on separate computers, one of which is a programmable workstation'
- 'User control of applications, IT control of infrastructure'
- 'The marriage of the usability of PCs to the infrastructure of the mainframe'
- 'Distributed data + distributed processing + graphical user interface'

...?

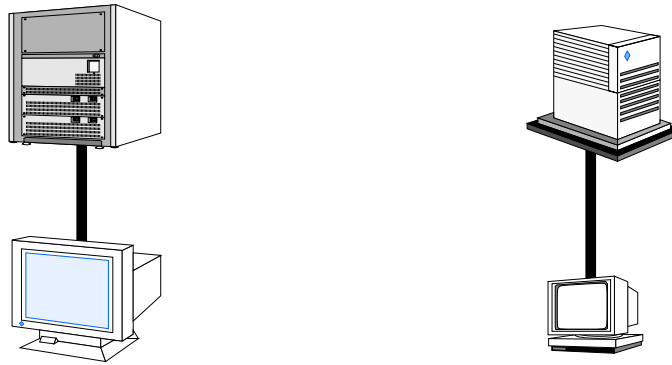




## Techniques for client/server computing

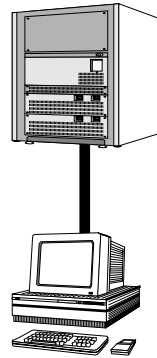
- **Terminal emulation**
- **Data download**
- **Remote data access**
- **Intelligent agents**

## Terminal emulation



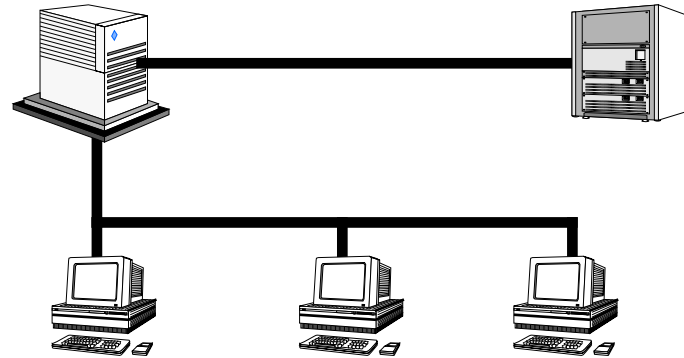
- In the 1970s, each computer was designed to use a particular type of terminal
- To use each host system, you had to use a different type of terminal
- So, manufacturers offered terminals that could also emulate other types of terminals

## Terminal emulation and the PC



- In the 1980s, the terminal emulation could be done by special PC software
- The PC was directly connected to the host computer, just like the original terminal

## Terminal emulation and the PC network



- With the rise of the PC local area network (LAN), not every PC needed a host connection
  - connection was indirect, via the LAN
- This was done with a communications gateway
  - funnelling traffic to the host computer



## Beyond terminal emulation

- **Terminal emulation is effective, but does not exploit the capabilities of the PC**
- **The user interface looks just as it would on a real terminal**
  - **very different from the PC's own user interface**
- **We can fit a PC-style 'look and feel' to the terminal emulation**
  - **this is called "screen-scraping"**



## “Screen Scraping”

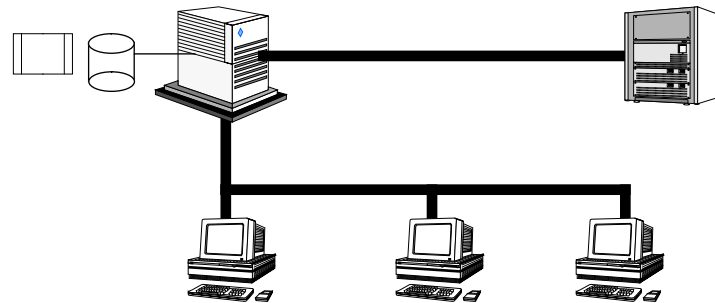
- Provides a ‘face-lift’ for terminal emulation
- Usually provides
  - short-cut keys for quick access
  - access to several host computers or applications at the same time
- May provide
  - integration into desktop productivity applications (spreadsheets,...)
  - continuous monitoring and display of information
  - a single point of login



## Screen Scraping - advantages and disadvantages

- **Advantages**
  - low risk
  - host computer continues to maintain data integrity
  - a technique that is always available
- **Disadvantages**
  - “Brittle” - there can be a high maintenance overhead
  - it's the same original application underneath
  - need to support and train users on old and new user interfaces
  - there are no open standards for screen scraping

## Data Download



- **Fetch the data from the host computer (possibly via a communications gateway)**
  - download data to individual machines, or a local LAN file server
  - download data from more than one source
- **Analyse the data with local applications**





## Data Download - building the local applications

- Use whichever software development tools you prefer
  - a standard application package
  - a standard database package
  - a 4GL
  - a graphical development tool



## Data Download - advantages and disadvantages

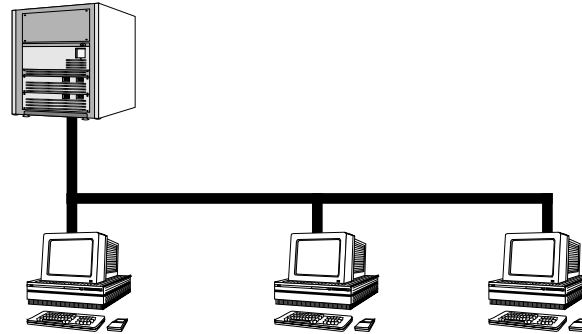
- **Advantages**

- little change needed to host system
- each data is internally consistent
- flexible information processing

- **Disadvantages**

- cannot easily be used to update information
- does not cope well with large volumes of data
- relies on understanding the format of the source data

## Remote Data Access



- Applications send and receive data directly from the host system
  - program-to-program communication, not terminal emulation
- Information processing is cooperative between the client and the server
  - the host system is the server



## Remote Data Access - advantages and disadvantages

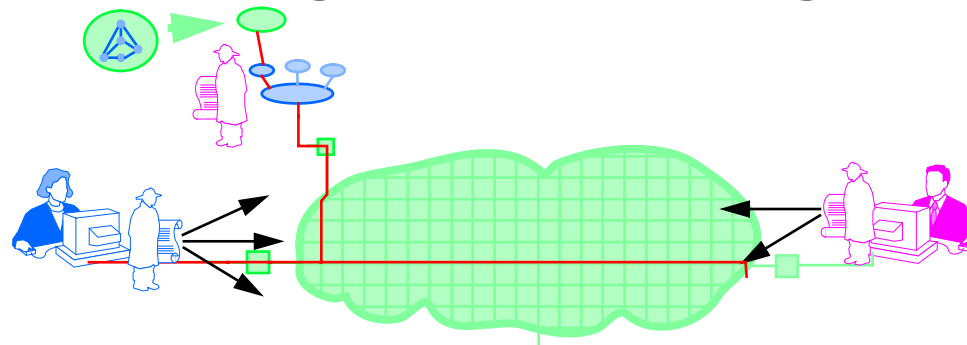
- **Advantages**

- most flexibility in developing new applications
- it may be possible to use standard software development tools

- **Disadvantages**

- local applications become responsible for data consistency
- impractical if the host system does not provide suitable interfaces
- it may place an unacceptable load on the host system

## Intelligent Software Agents



- **Intelligent agents will move around networks**
  - gathering and processing information, before returning to the user
- **It will be some years before agent technology is mainstream**
- **It will be used together with other client/server techniques**



---

## General Issues for Client/Server Systems

- **Scalability**
  - can the system expand as needed?
  - can the system be deployed in small and large configurations?
- **Interoperability**
  - can the system interwork with other systems?
- **Dependability**
  - can the system be made reliable?
  - can the system be made secure?
- **Internationalization**
  - can the system be deployed anywhere in the world?



## What People Say About Client/Server: Facts or Fallacies?

- “By the year 2000 there will be no mainframes”
- “70% of all commercial applications at the enterprise level will be Unix by 1996”
- “UK leads the world in client-server deployment”
- “Only 1 in 4 department client/server applications built today ever gets completed”
- “Client/server is more expensive than mainframes”



## Large-scale client/server

- Typically, large-scale client/server systems are distributed systems, rather than centralized ones
  - distributed systems being those which consist of interconnected cooperating components...
  - ...there being no central machine (or group of machines)





---

## Examples of distributed systems

- **Diverse business areas**
  - **Telecommunications**
  - **Airline reservations**
  - **Retail point-of-sale**
  - **Banking**
  - **Command and control**
  - **... and many more**
- **Built at the limits of the technology**



## **Distributed systems infrastructures**

- **CORBA from the Object Management Group (OMG)**
- **DCE from the Open Software Foundation (OSF)**
- **Distributed OLE from Microsoft**
- **Other specialized distributed systems technology**



## Motorola Iridium Project

- **66 satellites in low orbit providing a Global Cellular Network**
- **First launch in 1996, open for service in 1998**
- **\$3.4 billion investment**
- **Distributed system controls the satellites**
- **Various ground control stations need to exchange “route maps”**
- **Uses IONA’s Orbix**



## Telefónica PUEN Project

- **Management system for distributed application services**
  - **operating system access services**
  - **distributed data management services**
  - **application monitoring**
  - **fault management**
- **In the near future will integrate with TMN Q3 and IETF SNMP management interfaces**
- **Uses IONA's Orbix and SunSoft's NEO**



## Other telecommunications projects using CORBA

- **Oracle: Media Server Framework**
- **Orlando Full Service Network**
- **BNR(Nortel): Magellan Concorde System Manager**
- **Vodafone: Security Administrator**
- **BellSouth: SNET management system**
- **Ericsson: Cellular Management Operations System**



## Inherent features of distributed systems

- **Separation: physical and logical dispersal**
- **Diversity: many types of machines in the same system**
- **Legacy: evolution and interworking of existing systems**
- **Scalability: low cost of computing per machine**
- **Decentralization: no single point of control**
- **.... these differences are fundamental**

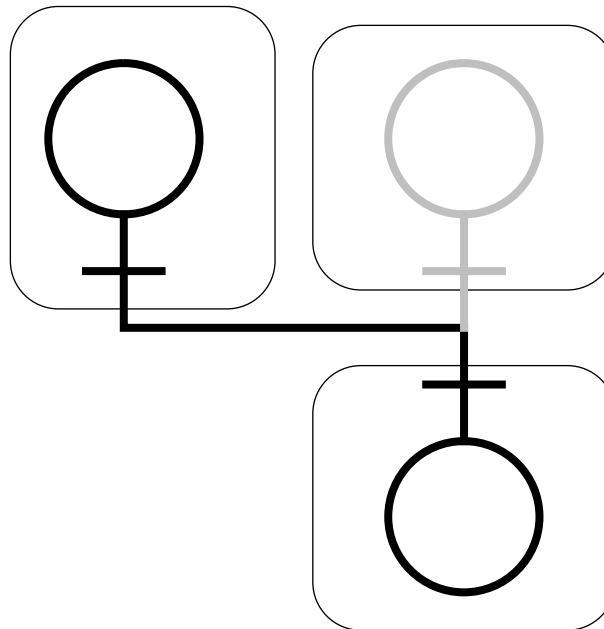


## Transparency for distributed systems

- **These inherent features would make it complex for the application programmer**
  - unnecessary complexity should be masked from the applications
- **The distributed systems infrastructure should make these features transparent**
  - using special *transparency* mechanisms

## Example Transparency - Migration

- **Migration Transparency**
  - application need not know where the object has moved to







## Different policies for different applications

- *Availability versus Consistency*
- *Autonomy versus Uniformity*
- *Security versus Convenience*
- **... and many other unavoidable trade-offs**



## “Openness”

- Which of these would you class an “open system”?

	Yes	No	Don't know
IBM PC			
Apple Macintosh			
Unix			
Microsoft Windows			
The worldwide telephone network			
Novell NetWare			
A 4GL that you know			

- Which system most closely fits your idea of an open system (it may not be listed above)?



## Thinking about openness

- Looking at your answers, try to write down what you think defines “openness”. (A list of keywords is fine)
- Try out your definition with some other systems you have heard of

	Yes	No	Don't know

**Get ready to discuss this**



## Your notes



## Summary

- **The major business challenge is coping with change**
- **Distributed object technology is here today**
  - **there are real applications being built with industrial-strength products**
- **The inherent features of distributed systems make application programming difficult**
  - **unless the infrastructure provides transparency mechanisms**
- **Take care when assessing “openness”**



## Where next?

- **In this course we'll be exploring distributed systems, focusing on CORBA from the Object Management Group (OMG)**
  - **...the specifications**
  - **...the products**
  - **...the challenges**