

Operating System Structure

Monolithic kernels (the “big mess” according to Tanenbaum) led to interest in more structured operating systems, such as layering and better structuring.

Layered operating system example is Xinu—“a small, elegant operating system”.

From there work began on taking functionality out of the kernels and making them much smaller—“micro” versions.

Microkernels

Philosophy of microkernel is to have the bare essentials in the kernel.

Benefits of microkernels:

- uniform interfaces—same for kernel and user-level services
- extensibility—can add new services
- flexibility—can subtract/modify services
- portability—easier to port operating system because os-specific part is smaller
- reliability—smaller kernel makes its implementation likely to be more reliable. Also more well-defined APIs.
- distributed system support—pieces can be on another machine.
- object-oriented operating system—architecture works in OO context.

Performance of such microkernels (Mach and Chorus) has been the concern. Approaches to improve performance (co-location of services in the kernel) go against microkernel idea.

Mach Operating System

One such system is Mach, which was developed at CMU. Mach was used as the basis for work on OSF/1, which was adopted by some companies such as Digital. The version of OSF/1 running on Digital Alpha machines is currently known as Digital Unix.

Look at “Microkernel Operating System Architecture and Mach” paper (1992).

Windows NT Operating System and Beyond

Single-user, multitasking operating system designed to run on a variety of PCs and workstations.

Uses a modified microkernel architecture (Figure 2.13)—some system services are in the kernel for performance reasons.

Hardware Abstraction Layer (HAL) contains hardware-specific code.

Windows NT works with objects.

Paging is done in the kernel as well as all I/O.

Provides kernel level threads.

Uses *local procedure call* (optimized message-passing mechanism) for communication.

The server processes at the user level basically provide different interfaces. Most common is Win32 subsystem—in fact other user-level services must go through Win32 subsystem.

Second Generation Microkernels

Look at “The Performance of Micro-Kernel-Based Systems” paper. Slides:
<http://www.cs.wpi.edu/~cs535/f05/hartig:sosp97/>

Exokernel Operating Systems

Look at “Application Performance and Flexibility on Exokernel Systems” paper. Slides:
<http://www.cs.wpi.edu/~cs535/f05/kaashoek:sosp97/>

Virtualization: Xen Operating System

Look at “Xen and the Art of Virtualization” paper. Slides:
<http://www.cs.wpi.edu/~cs535/f05/barham:sosp03.pdf>

Planet Lab

Look at “Operating System Support for Planetary-Scale Network Services” paper. Slides:
<http://www.cs.wpi.edu/~cs535/f05/bavier:nsdi04.pdf>

Operating Systems Support for Server Applications

Look at “Better Operating System Features for Faster Network Servers” paper. Slides:
<http://www.cs.wpi.edu/~cs535/f05/banga:wisp98/>

Support for Secure Applications

Look at “Sub-Operating Systems: A New Approach to Application Security” paper. Slides:
<http://www.cs.wpi.edu/~cs535/f05/ioannidis:ew02/>

Support for Internet Services

“Ninja: A Framework for Network Services” paper.

Language-Based Operating Systems

“Techniques for the Design of Java Operating Systems” paper.

Object-Oriented Systems

Look at Choices and Clouds papers

Alternate Approaches

Examine Amoeba and Sprite comparison paper.

Summary

Many types of structures. The microkernel approach, although with flaws has moved forward in the marketplace with Windows NT and Digital Unix.

Researchers are still looking to find operating system structures that support new types of applications and needs of networking.