Independent Study Proposal

The Practicality and Performance
of a
Fully-Dynamic Object-Oriented Microkernel

by
Frank Barrus

May, 1992

Goals:

To develop a fully-dynamic object-oriented microkernel, and, through comparison against a more static, monolithic system, show that the performance cost is minimal compared to the added flexibility offered, if the system is well designed.

By object-oriented, I mean that all elements of the system will be encapsulated into objects, which contain both the data they need, and the methods to access and manipulate that data. Multiple inheritence of methods from other object classes will be allowed, to minimize the overhead needed to create new objects, and to reduce redundancy in the system. Therefore the system will be built out of very general object-classes which will be inherited by more specialized, system dependent, objects.

By fully-dynamic, I mean that all objects and methods will be changeable on-the-fly, so that new ones can be added, or existing ones modified, at any time. This allows for adding extensions without bringing the system down, and allows for replacement of any of the primitive functions of the operating system while it's running. It also means that the core of the operating system needs only enough functionality to get it up and running, and that all other support can be loaded up afterwards, keeping the kernel size to a bare minimum.

Equipment:
I intend to carry out this project on some variant of the VAX architecture. On Computer Science House, we have several VAX 11/780's that have been taken out of service, one of which I have written a small operating system for two years ago. We also have many Kodak KIMS workstations, which are basically VAX machines. This will most likely be the machine I test the O.S. on.

Also, time permitting, I would like to try porting my O.S. to the Intel 386 architecture, to compare the performance of the two architectures, and test portability of the operating system.

Milestones:

1. A library of routines to keep track of objects, and all the necessary database support for looking up references to objects and methods, and handle all object-oriented functions such as a typing system, multiple inheritence, etc.

2. A core micro-kernel based on, and including the aforementioned library, which will perform all necessary tasks to get the system up and running, and allow patching in replacement objects at runtime for memory management, file systems, process control, etc.

3. Another more self-contained, static micro-kernel that is functionally similar to the dynamic one, but not flexible or expandable, for the purpose of testing the performace of the dynamic one compared to its static counterpart.

4. Extension of the dynamic micro-kernel, through the building of objects, into a fully-functional, useable operating system, including virtual memory, a reasonable process scheduler, support for standard devices, including either a local disk, or the ethernet, to support a filesystem. Also, if time permits, this includes porting some form of C compiler to run under the operating system.

I would like to get this independent study project approved for fall quarter, and registered, as I am currently signed up for only 13 credit hours for the fall. If you approve of it, could you either send in the necessary paperwork, or leave it in my mail folder.

Regarding my ability to complete this project, I completed Operating Systems Lab with Warren Carithers last fall quarter, and wrote the memory management, device-indepedent I/O routines, and serial I/O and buffering routines. Also, last year I did an independent project for Computer Science House, developing a small experimental operating system for the VAX 11/780.

Thank you for your consideration.

- Frank