OpenStep On Solaris Readied For Beta

• By Frank Hayes

OpenStep on Solaris, the first implementation of the portable object-oriented environment jointly developed by SunSoft and Next, will go to beta users early next year, Sunsoft said last week.

Few users are expected to want the new object technology initially, Sunsoft said. But those who do can expect a fairly smooth transition from existing NextStep applications—with some potential incompatibilities, according to developers who have examined the OpenStep specification.

Both Sunsoft and Next have recently released new versions of their operating systems that lay the foundation for a later full implementation of the OpenStep APIs. In October, Sunsoft began shipping Solaris 2.4, which will form the base for OpenStep when it ships next year. And last week, Next began shipping NextStep 3.3, which includes key transition elements for users planning to migrate applications from NextStep to OpenStep.

NextStep 3.3 is available now for Intel and Motorola microprocessors and is slated for PA-RISC and SPARC in mid-1995. Pricing is $799 or $199 for upgrades. Enhancements include support for a broader range of Intel-based PCs; improved system administration, scalability, and E-mail; and support for enhanced-mode DOS and Windows applications via SoftPC.

Digital Equipment, Hewlett-Packard, and Data General have also announced plans to implement OpenStep-compatible

OBJECT-ORIENTED DEVELOPMENT

technology, and Next has discussed porting OpenStep to an operating system in Microsoft's Windows family.

"With OpenStep on all the different platforms, you're finally going to begin achieving open systems goals of the right hardware for each user, scalable from the office automation user to the back of the glass house," said Ted Shelton, president of systems integrator Information Technology Solutions in Chicago.

"I don't know if many people in the commercial software realm are thinking about it, but my customers in internal corporate development are certainly hoping that this is the case," he said.

But Sunsoft expects user uptake to be slow, at least among Solaris users.

"Only a small set of cus-tomers will be adopting objects now: those who find procedural tools inadequate to create or modify applications," said Bud Tribble, Sunsoft vice president of object products. Other Solaris users will continue to use conventional programming tools and languages, Tribble said.

Sunsoft's first OpenStep implementation, which will go to beta users in the first quarter of next year, will be part of Sunsoft's Distributed Object Environment (DOE), which will also include an object communication system complying with the Object Management Group's Common Object Request Broker Architecture, along with a C++ interface builder, object debuggers, and administrative tools.

Developers will be able to mix and match C++ and Objective C as needed using Sunsoft Workshop for C++, which is being enhanced to accept Objective C syntax, Tribble said. That should ease one transition problem for Solaris developers, whose greatest difficulty "mostly has to do with learning to program with objects," Tribble said.

Unlike the transition several years ago from SunOS to Solaris, the underlying operating system will remain the same with the addition of OpenStep and DOE, and there is no absolute requirement to rewrite any applications, he said.

Some rewriting of existing NextStep applications may be necessary, but Next will bundle much of the transition technology into the developer version of NextStep 3.3, said Eric Chu, product manager for deployment technologies at Next.

The developer version, which will begin beta-testing this month, won't support the OpenStep API, which will be implemented in NextStep 4.0, but will include FoundationKit, a more portable class library that replaces NextStep's AppKit in the transition to OpenStep, Chu said.

Porting existing NextStep apps to OpenStep on Solaris is complicated by both OpenStep's improvements over NextStep and the parts of NextStep not addressed in the OpenStep specification, Information Technology Solutions' Shelton said.

Apart from changes in the names of function calls and variables, OpenStep adds an enhanced text object. It's a great improvement over earlier versions, but it will break existing code that subclasses the old version of the string object, Shelton said.

STANDARD INTERFACES

OpenStep ◆ CORBA ◆ C++