The NeXTstep®
Development Environment

What is NeXTstep?

The graphical user interface (GUI) and associated development environment bundled with NeXT™ computers is called NeXTstep. This graphical system was created to solve two major problems with applications: without good user interfaces they are difficult to use, and good user interfaces are difficult to develop. This paper describes the benefits of NeXTstep from an application developer's perspective. A companion paper called *The NeXTstep User Interface* describes the features and benefits of NeXTstep from a user's point of view.

There are four layers in NeXTstep (see figure): the Window Server, Application Kit™, Interface Builder™, and Workspace Manager™. Conceptually, NeXTstep is all of the software above the operating system but below the applications. This paper begins by describing Application Kit and the integrated technologies that make it possible. It then discusses the remarkable Interface Builder tool. Finally, it finishes with a discussion of the NeXTstep Window Server. Together, these components carry the software developer to new levels of productivity.

Object-oriented programming and AppKit™

Just as great architectural achievements begin with solid foundations, the heart of the NeXTstep development environment is the most powerful and flexible programming paradigm available. It is called *object-oriented programming,* and although the concept has been available for over a decade, NeXT is the first general purpose computer vendor to build the foundation of its entire development environment on the concept.

At the core of object-oriented programming is the same idea that led electrical engineers to use integrated circuits comprised of thousands of transistors instead of just individual transistors. A complex circuit board is much easier to design, build, and modify if an engineer can begin with building blocks of encapsulated functionality instead of single transistors. Object-oriented programming applies this idea to software by recognizing that a large software application is much easier to design, build, and maintain when a software engineer begins with a building block larger and more functional than a single line of code. An *object* is exactly that building block.

Objects are capsules of functionality that have a well defined interface. The abstract description of an object and its external interface is called a *class*. An actual object in an application is called an *instance* of a class. A running application comprises many objects, and objects work together to accomplish tasks by sending messages to each other.

Because all object interaction, and therefore all work, is accomplished through the external interface of objects, there is an *encapsulation* of data and functionality that allows the developer to work at a higher level than individual lines of code. Now the programmer can think in terms of objects and their interactions, rather than thinking about how to write, for example, a loop termination condition. Object-oriented programming allows the developer to take a building block approach to an application, and thus applications can be developed much more quickly and efficiently.

The NeXTstep development environment includes a rich set of object classes called Application Kit (AppKit) that define the appearance and functionality of the NeXTstep GUI. Examples of classes that the AppKit defines are *Window, Menu, Text Field, Button, Scroll Bar,* and *View.* Together, these and the other classes in the AppKit provide developers with the building blocks needed to build a fully functional and consistent GUI for their applications. In addition, through an object-oriented concept called *inheritance,* a developer can modify and extend the classes in the AppKit to suit a specific need.

Of course, building blocks are only raw material for a mason, who must use mortar, bricks, skill, and artistry to create a finished structure. The same is true for the AppKit and other custom objects that

must be assembled into a NeXTstep application. Fortunately for software engineers NeXT provides a powerful construction tool called Interface Builder.

Interface Builder

Interface Builder is a developer's tool that exploits the object-oriented paradigm and AppKit to make user interface design and construction simple. Interface Builder is far more than a code generation tool. It is an entire application construction tool that significantly reduces the effort required to design, implement, and maintain a GUI based application.

When constructing a new application, Interface Builder presents the developer with a set of palettes, each of which contains several associated classes. The developer constructs an application by dragging classes from the palettes into the application, thereby creating object instances that will comprise the finished application. For example, when developers want to add a button to an application, they simply drag the button off a palette and place it in the application they're building. They can size, resize, and move the button until it is exactly what they want.

As the application begins to take shape, Interface Builder allows developers to establish communications between the objects. If you want one object to send a message to another object as the result of some user action, you can establish a graphical connection between the two objects and specify the message that is to be sent. Simple applications can be built without ever having to write a line of code. Developers can add their own custom objects and class palettes to the already rich environment, which significantly reduces the effort required to build even the most complex applications.

Interface Builder keeps track of everything for the developer, from the objects (both AppKit and custom), to the connections between objects, to the code files associated with the application. When asked to do so, Interface Builder produces source code that the developer needs to extend and customize, but hides the source code for standard objects so that the developer doesn't have to be concerned with it.

Interface Builder is truly the ultimate application construction tool available in the computer industry. By reducing the actual amount of code that the developer must write (and for that matter, come in contact with), it eliminates the most time consuming part of building GUI applications and it offers unlimited flexibility through the creation of custom object classes and palettes.

Of course, there is still another area of application construction that developers dread: getting their applications to print. NeXTstep removes this thorn in the side by once again applying the most advanced technology available.

The Window Server and Display PostScript

A core technology of all GUIs is the imaging model— the language and paradigm used to put images on the screen and the printer. The NeXTstep Window Server does what no GUI has ever done before: unifies the imaging model for the screen and the printer.

Until NeXTstep, drawing to a computer's display was accomplished with an entirely different set of instructions than was used to draw on the printed page. This dichotomy not only caused inconsistencies between what you saw on the screen and what you saw on the printed page, but it often required developers to write two different versions of code to accomplish screen display and printing. The NeXTstep Window Server, by incorporating the powerful PostScript® language from Adobe, unifies the imaging model for the screen and the printer.

The PostScript language is an advanced graphics and page description language that has been used on high-quality laser printers and image setters for many years. NeXT worked closely with Adobe to help bring to market a version of PostScript for the screen called Display PostScript®, thus unifying both display and print graphics.

What this means to the developer is simple: you write one set of instructions that are used for both the screen and the printer. And because the instructions are written in the powerful PostScript language, you have tremendous freedom of expression (not to mention the fact that PostScript has quickly become the imaging model of choice throughout the computer industry). Once again, the NeXTstep development environment puts emphasis on productivity and flexibility.

The bottom line

What determines the greatness of a software development environment is the productivity of developers, and NeXTstep takes their productivity to new heights. NeXT is the only general purpose computer vendor to integrate the object-oriented paradigm into the entire development environment. Application Kit and Interface Builder build upon this foundation to greatly reduce the time and effort required to create high quality graphical user interfaces. Finally, the unification of the screen and printer makes it easier to get an application to print.

Welcome to NeXTstep and the NeXT world.

NeXTstep includes the components listed below:

- Window Server (with Display PostScript)
- Application Kit
- Interface Builder
- Workspace Manager

The NeXT development environment includes the software listed below:

- Motorola 68030 and 68040 Assembler
- Motorola 56001 Assembler
- Objective-C® Language Compiler
- C++ Language Compiler
- Sound Kit™ and Music Kit™
- Terminal
- GNU Emacs
- PostScript Tools
- AppInspector™ Object Browser
- GDB Symbolic Debugger
- MallocDebug Memory Debugger
- Ariel Bug-56™ Symbolic Debugger
- Code Examples
- On-line Technical Documentation

© 1990 NeXT Computer, Inc. All Rights Reserved.

The NeXT logo and NeXTstep are registered trademarks, NeXT, NeXTmail, AppInspector, Application Kit, AppKit, Interface Builder, Music Kit, and Sound Kit are trademarks of NeXT Computer, Inc. PostScript and Display PostScript are registered trademarks of Adobe Systems Incorporated. Ariel and BUG-56 are trademarks of Ariel Corporation. Objective-C is a registered trademark of The Stepstone Corporation. WriteNow is a registered trademark of T/Maker Company. All other trademarks mentioned belong to their respective owners.

NeXT Computer, Inc., 900 Chesapeake Drive, Redwood City, California 94063