NEXTSTEP DEVELOPER RELEASE 3

paste.tiff ¬

With NEXTSTEP's integrated set of development tools and extensive set of pre-built objects, you can produce rich, integrated application in a fraction of the time it takes on other platforms.

Many organizations have discovered that building custom business applications is one of their key competitive advantages. With NEXTSTEP Developer, the industry's leading object-oriented environment for building advanced client/server applications, these applications can be developed in significantly less time, with higher quality, resulting in faster deployment and a stronger competitive lead. NEXTSTEP Developer provides the complete development environment for NEXTSTEP, the acclaimed leader in object-oriented operating systems, running on the Intel 486 and Pentium microprocessor families, HP PA-RISC and SPARC workstations, and NeXT computers.

NEXTSTEP Developer includes a complete object framework and all of the tools needed to build advanced, easy-to-use applications. NEXTSTEP Developer starts with a complete set of Object Kits, including the Application Kit, a set of objects providing functionality common to all applications and a complete library of user interface objects, and objects that support information sharing between applications. Also included is Database Kit, which integrates popular client/server relational or other databases into any application. Built upon these object kits is a complete set of graphical application construction tools, such as Project Builder, which manages resources in an application; and Interface Builder, a tool for managing objects in an application. For developing custom objects, NEXTSTEP Developer provides compilers for the object-oriented languages of Objective C and C++, as well as ANSI C.

Because NEXTSTEP Developer and NEXTSTEP are completely object-oriented, complex and dynamic applications can be written with far fewer lines of code, five to ten times faster, and can be modified and maintained easily throughout their life cycle. This object-orientation enables an organization to build a library of reusable objects, improving programmer productivity and ensuring a higher degree of consistency, robustness and reliability across all applications.

DEVELOPMENT TOOLS

Project Builder
PB2.tiff ¬
The central application for managing, building, debugging, and maintaining software projects. Manages all files and resources

associated with an application, generates Makefiles. Provides a graphical interface to **make**, and allows you to go immediately from a compiler error to the line in the editor where the error occurred, simply by clicking on the error message.

Interface Builder
IB.tiff ¬
A rapid prototyping tool and object editor that enables you to design the relationship between objects in an application. Includes a user interface layout tool and facilities for subclassing existing objects. In addition to providing graphical palettes for NeXT Object Kits, Interface Builder allows you to load in third-party or custom-made palettes of objects not supplied by NeXT.

Edit
EditApp.tiff ¬
Powerful, integrated, mouse-oriented graphical editor for code and on-line help. Emacs key bindings; nesting of braces for C and LISP; access to ctags, UNIX commands, and manual pages; contracting/expanding based on code structure. Edit also can be used to create help files that contain graphics and hypertext-like links.

Graphical Debugger
Debugbutton.tiff ¬
Easy graphical access to debugging. Powerful GDB debugger includes such features as the ability to debug optimized code and data breakpoints. Fully integrated with Edit and Project Builder, includes mouse access to setting breakpoints, examining data, stopping and starting execution, and other typical graphical debugger features. Also includes a data browser that allows you to explore the current execution stack and its local variables, traversing complex data structures using the mouse.

MallocDebug
Mallocappicon_color.tiff ¬
A tool for measuring the dynamic memory usage of your application, useful for detecting memory leaks.

Header Viewer
Header.tiff ¬
A class browser for navigating object class hierarchies, providing access to class information and documentation. Header Viewer navigates the NeXT Object Kits, as well as your own code.

DBModeler

DBModeler.tiff ¬
A graphical tool for building data models. Specifically designed for use with Database Kit, this application enables you to build and customize data models based on the data dictionary of an underlying database.

Yap
Yapappicon_color.tiff ¬
An interactive PostScript language previewer for entering and executing PostScript code, making it easier to debug.

Icon Builder
IconBuildrappicon_color.tiff ¬
A simple pixel-based editor for creating icons.

GNU Emacs, vi, yacc, lex...All the popular UNIX tools.

OBJECT KITS

Application Kit
Appkit.tiff ¬
A rich set of standard objects that provide the framework for all applications, with a wide variety of customization options; Includes a complete set of user interface objects and controls; Objects that support data sharing and inter-application communication;

Look-and-feel is consistent across all NEXTSTEP applications. Application Kit objects provide a standard framework for handling details such as event management, window management, as well as printing and fax capability. Since all of NeXT's kits are composed of Objective C objects, you are free to modify or extend their behavior through subclassing.

Pre-built objects for simple controls such as buttons and scrollers, as well as complete support for common tasks such as selecting files, printers, fax machines, fonts, and colors.

Data sharing such as Cut, Copy, and Paste is automatically supported between all applications. More advanced features such as live-links and inter-application services can be easily integrated into your application. Distributed Objects make it easy to implement client/server applications.

Database Kit

DBKit.tiff ¬

An object-oriented tool kit for creating database access applications;Database-independent implementation; Supports multimedia data types; Supports multiple database backends;

Extends the power of object-oriented programming to data-intensive applications. Dramatically reduces the effort required to design, implement, and maintain client-server database applications.

Allows applications to be migrated to a different vendor's database without requiring the recoding of user-interface and application logic. Enables the integration of data from multiple sources within a single, easy-to-use application.

Database Kit-based applications can easily incorporate a wide variety of data typesÐfrom simple text and numeric fields to rich text, images, and sound.

Adaptors for both ORACLE and SYBASE are included, providing everything necessary to access these databases in a client/server environment. Other databases are accessible through third party adaptors.

Indexing Kit

An object-oriented kit for efficiently storing, indexing, and retrieving both textual and record-based information;

Allows you to manage large amounts of text and structured information without the complexity of a commercial database.

3D Graphics Kit

3DKit.tiff ¬

Based on Pixar's RenderMan standard, a unified 3D imaging model, so the same code draws objects on both the screen and the printed page;

Lets developers easily add three-dimensional graphics to existing applications without being a 3D expert. Provides the necessary object and rendering framework for building sophisticated 3D applications.

Sound Kit

Objects for creating, saving, viewing, and playing audio data on all platforms equipped for sound;

Lets developers add sound to applications, without having to write code to support different sound devices on different hardware platforms.

BUNDLED LANGUAGES

Objective C
Simple and concise object-oriented extension to ANSI C; Run-time messaging facility, dynamic binding; Distributed Objects; Optimized native code compilation;

Provides many SmallTalk features with the efficiency of C. The Objective C language is easier to learn and produces clearer code than other language alternatives.

Provides the ability to reason about objects at run time, not merely at compile time, which is essential for programming flexible GUIs and enabling remote messaging.

Extends the object advantage to a distributed computing environment, by enabling applications to message objects in other applications (even on machines across the network)..

Source code is compiled into executable format, not translated, yielding highest performance.

C++
Extension to the C language, used for object-oriented programming on a wide variety of systems; Integration with Objective C; Optimized native code compilation;

Allows developers with code written in C++ to port their applications, using C++ objects, while still taking advantage of NeXT's rich suite of Object Kits written in the Objective C language.

Allows C++ modules to freely call (and be called by) Objective C objects.

Compiler produces executable code without translation, maximizing compiler performance and allowing for object-level debugging.

ANSI C
Industry-standard version of the popular C language;

Compatibility with the widest range of systems.

PostScript Level II
Industry-standard imaging language; Device-independent graphics model; Support for a wide variety of output devices;

Provides a ªunified imaging model.º Code to draw on the screen is the same code that's used to print.

Frees the programmer from dealing with device dependencies. Resulting applications can be deployed on different graphics platforms.

Compatible with the widest range of graphics output devices, from a desktop inkjet printer to a high-end imagesetter.

ADDITIONAL FEATURES

Source Code Compatibility Across All NEXTSTEP Computers

Simply recompile applications for use on any NEXTSTEP computer, or take advantage of NEXTSTEP Developer's cross-build capabilities.

Allows developers to port NEXTSTEP applications to any NEXTSTEP platform, simply by recompiling. Using the cross-build feature of Project Builder, applications can be built for any target platform, from any other platform, with the simple click of a switch.

Multi-Architecture Binary Support

Supports creation of a single binary for multiple CPU architectures. NEXTSTEP automatically determines the portions of the binary that relate to the local CPU type and runs the application.

Simply by selecting multiple architecture types in Project Builder's Build options, you can create a single application package that will

run on any NEXTSTEP platform, easing deployment in mixed architecture environments. Also allows ªshrink wrapº applications to be shipped, capable of running on any NEXTSTEP platform as well.

On-line Help
HelpCursor.tiff ¬
Context-sensitive help, with hypertext-like linking and graphics, available in key applications.

Helps you come up to speed in the NEXTSTEP development process and offers instant assistance with the task at hand.

Technical Documentation
TechDoc.tiff ¬
A complete set of technical documentation is included in both hardcopy form and on-line, covering NEXTSTEP tools, objects, and the Objective C language.

Provides you with the necessary reference materials to fully utilize the development environment. On-line documentation is indexed for use in NeXT's Digital Librarian¿application.

SYSTEM REQUIREMENTS

NEXTSTEP Developer Release 3.3 requires NEXTSTEP Release 3.3 for Intel Processors, SPARC or PA-RISC workstations, or NeXT Computers.

NEXTSTEP Developer requires a 330 MB hard disk or larger; 16 MB of RAM or more is recommended.

NEXTSTEP Developer comes on a single CD-ROM, and includes hardcopy technical documentation.

For additional information call **1-800-TRY-NeXT**.
NeXT Computer, Inc., 900 Chesapeake Drive, Redwood City, CA 94063 U.S.A.