The Niakwa Workbench

Windows Editor and Debugger tool for NPL Developers

Introduction to the Niakwa Workbench

The Niakwa Workbench introduces a powerful editor/debugger to the NPL development community. It is a visual project management tool that provides seamless access to Niakwa's 32-bit development tools.

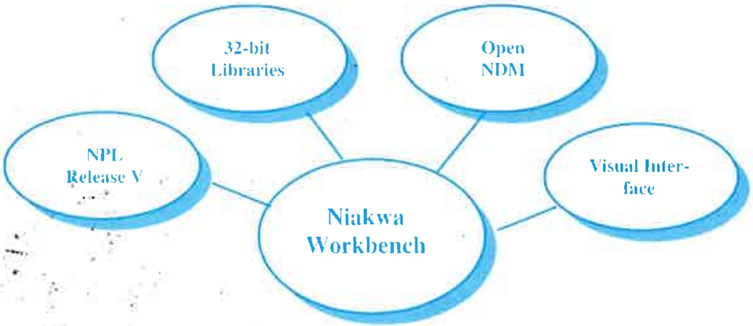
This user-friendly tool facilitates rapid application prototyping and development with features such as a drag-and-drop environment, full-screen editing, on-the-fly syntax checking, real-time "live data" debugging, and full project and library management.

Projects and libraries can be managed simply by dragging modules and data files between windows. Code is quickly accessed with a double-click of the mouse for easy full-screen editing.

Applications developed with the Niakwa Workbench can be prototyped and pre-tested before implementation, without affecting a production installation. With the user-friendly interface, both developer and end-user can work side-by-side to develop the application to suit the end-users needs.

With the integrated interpreter engine, the Niakwa Workbench allows execution of an application prototype exactly as it would operate in production, and with quick access to real-time debugging information and source code tools.

The Niakwa Workbench introduced in NPL Release V is the primary lens through which developers can view and interact with other IDE (Integrated Development Environment) components such as Open NDM for data management, the Niakwa Visual Interface Manager for user interface design and the Niakwa Programming Language for application programming.



The Niakwa Workbench is the lens through which developers will view and interact with the NPL language and development tools.

This data sheet provides specific details about the many features and benefits of this powerful development tool.

What is the Niakwa Workbench?

The Niakwa Workbench is a 32-bit Microsoft Windows based development environment in the tradition of the Borland and Microsoft IDE's It is a multi-window editing and debugging environment based on the concept of a project, which is a collection of files (libraries, object and source files) pertaining to a specific application which are grouped together.

The Niakwa Workbench is deeply integrated into the Windows 95/98 NT environment, particularly the Windows Explorer program. It associates several file types with itself and a set of icons via the system Registry. File names can be passed to the Workbench using any of the standard Windows mechanisms, including:

- drag and drop
- the clipboard
- command line parameters

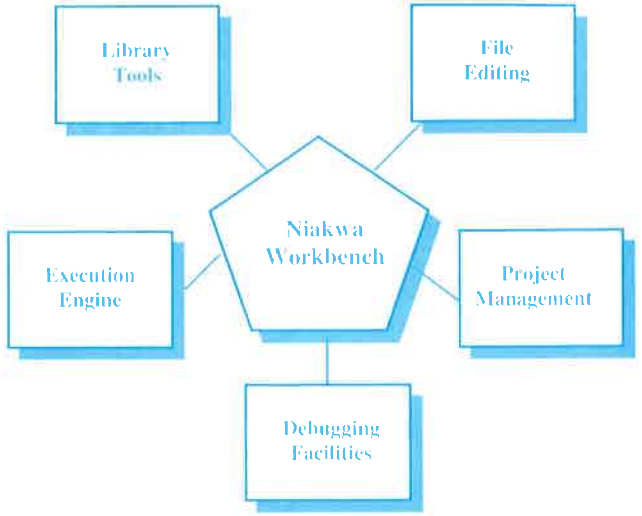
The Niakwa Workbench provides the ability to deal with source files and then use third-party source code control systems for team programming environments.

The Niakwa Workbench accommodates **full-screen editing** of multiple files simultaneously. The **drag-and-drop** environment will greatly reduce the amount of keyboard entry involved in managing NPL files. The Workbench requires far less dependence on line numbers and automates many of the tasks involved in managing libraries with a simple mouse click.

The Workbench's functionality can be classified into five broad categories: library tools, file editing, project management, debugging facilities and execution engine.

Library Tools

The Workbench automates many of the tasks involved in managing libraries (diskimages). Simple mouse operations permit moving and copying files between libraries, eliminating the need to type long, complicated commands. In addition, the libraries dynamically resize themselves in order to accommodate changing file sizes.



The Niakwa Workbench contains an abundance of time saving tools for software developers.

File Editing

The Niakwa Workbench provides a full-featured multiple document interface (MDI) for full screen editing of multiple files simultaneously.

Project Management

The drag-and-drop graphical organization provides quick access to all necessary files contained within a project.

Debugging Facilities

The Workbench offers enhanced debugging capabilities through the use of multiple breakpoints, single-stepping, variable watch windows and immediate mode command window.

Execution Engine

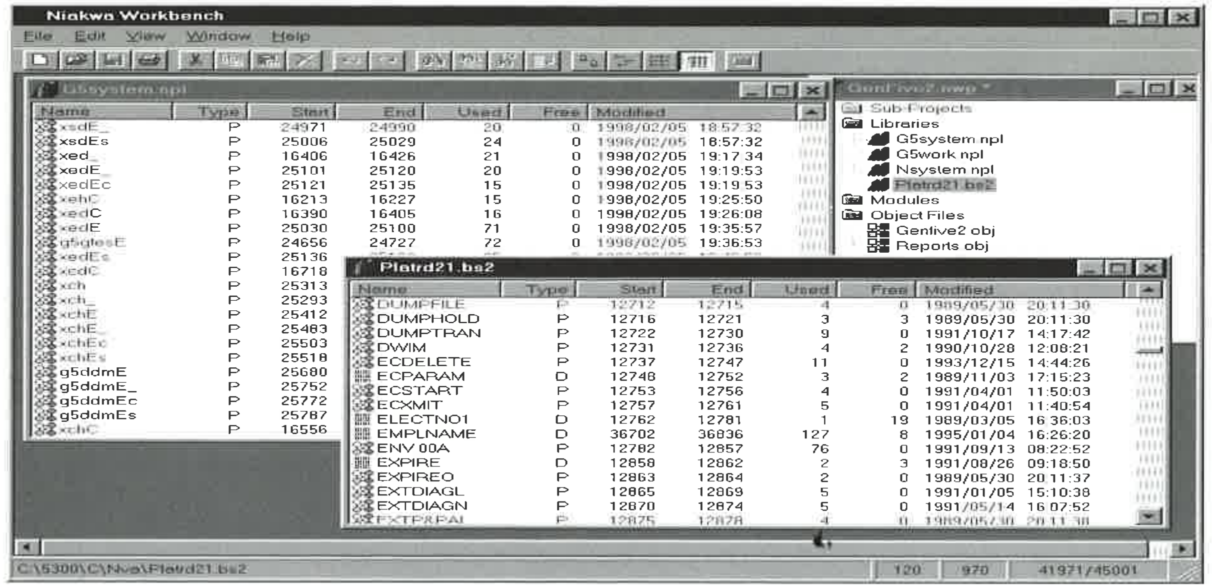
The Workbench gives you the ability to run one or more NPL programs without having to first start up their associated RunTimes.

Library Tools

Several additional tools are included with the Niakwa Workbench to make life easier for the developer. Beyond writing the application, there are many operations that a developer must perform in order to simply maintain a current application, and these tools help that effort.

The features of the Library Tools are as follows:

- Graphical display of library contents; can be sorted by any field
- Drag-and-drop capability between multiple libraries
- Library contents can be compressed to maximize physical disk space usage
- Libraries are automatically expanded when reaching full capacity
- Indexing method can be controlled for optimum compatibility



G5system.npl

```
Name      Type  Start   End    Used  Free  Modified
xsdE_      P    24971   24990   20    0    1998/02/05  18:57:32
xsdEs      P    25006   25029   24    0    1998/02/05  18:57:32
xed_       P    16406   16426   21    0    1998/02/05  19:17:34
xedE_      P    25101   25120   20    0    1998/02/05  19:19:53
xedEc      P    25121   25135   15    0    1998/02/05  19:19:53
xehC       P    16213   16227   15    0    1998/02/05  19:25:50
xedC       P    16390   16405   16    0    1998/02/05  19:26:08
xedE       P    25030   25100   71    0    1998/02/05  19:35:57
g5gtesE    P    24656   24727   72    0    1998/02/05  19:36:53
xedEs      P    25136
xedC       P    16718
xch        P    25313
xch_       P    25293
xchE       P    25412
xchE_      P    25483
xchEc      P    25503
xchEs      P    25518
g5ddmE     P    25680
g5ddmE_    P    25752
g5ddmEc    P    25772
g5ddmEs    P    25787
xchC       P    16556
```

GenFive2.nwp

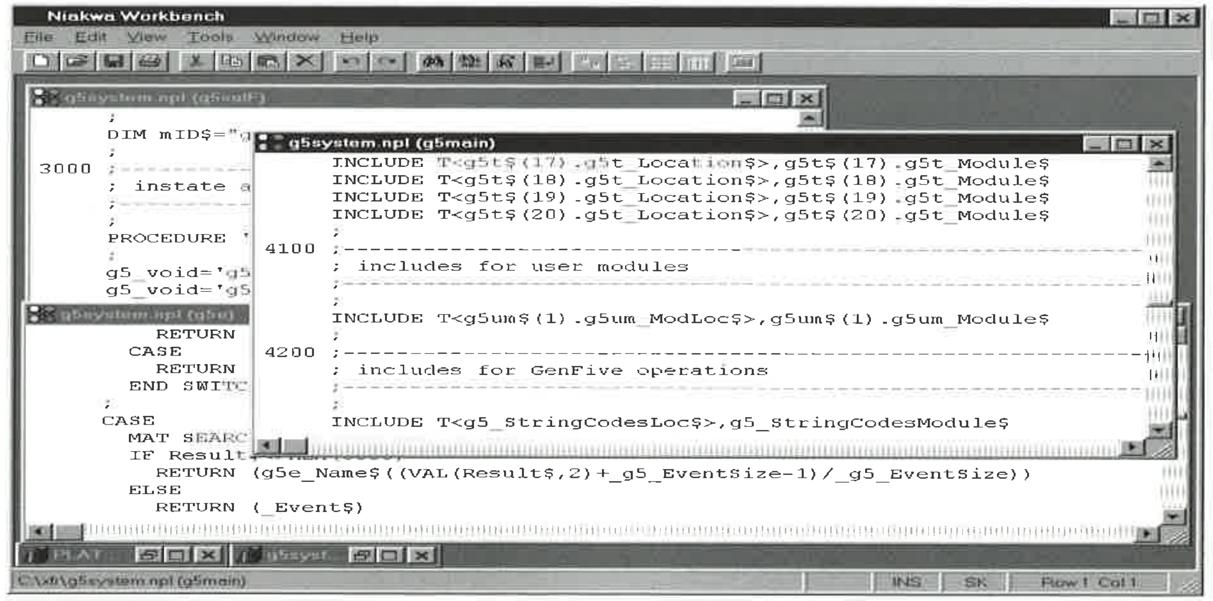
```
Sub-Projects
Libraries
    G5system npl
    G5work npl
    Nsystem npl
    Platrd21 bs2
Modules
Object Files
    Genfive2 obj
    Reports obj
```

Platrd21.bs2

```
Name        Type  Start   End    Used  Free  Modified
DUMPFILE     P    12712   12715    4    0    1989/05/30  20:11:30
DUMPHOLD     P    12716   12721    3    3    1989/05/30  20:11:30
DUMPTRAN     P    12722   12730    9    0    1991/10/17  14:17:42
DWIM         P    12731   12736    4    2    1990/10/28  12:08:21
ECDELETE     P    12737   12747   11    0    1993/12/15  14:44:26
ECPARAM      D    12748   12752    3    2    1989/11/03  17:15:23
ECSTART      P    12753   12756    4    0    1991/04/01  11:50:03
ECXMIT       P    12757   12761    5    0    1991/04/01  11:40:54
ELECTNO1     D    12762   12781    1   19    1989/03/05  16:36:03
EMPLNAME     D    36702   36836  127    8    1995/01/04  16:26:20
ENV 00A      P    12782   12857   76    0    1991/09/13  08:22:52
EXPIRE       D    12858   12862    2    3    1991/08/26  09:18:50
EXPIREO      P    12863   12864    2    0    1989/05/30  20:11:37
EXTDIAGL     P    12865   12869    5    0    1991/01/05  15:10:38
EXTDIAGN     P    12870   12874    5    0    1991/05/14  16:07:52
EXTP&PAI     P    12875   12878    4    0    1989/05/30  20:11:30
```

C:\5300\C\Nvo\Platrd21.bs2 | 120 | 970 | 41971/45001

Figure 1: This illustration shows a project window and two libraries (diskimages) open at the same time. Notice that the library in front is sorted by module program name, while the library in back is sorted by modification date/time. Libraries may be displayed from either the same or different projects. The Explorer-style shows the nested hierarchy of the elements of the project.



g5system.npl (g5sulF)

```
          ;
          DIM mID$="g
          ;
3000      ;-----------
          ; instate a
          ;-----------
          ;
          PROCEDURE '
          ;
          g5_void='g5
          g5_void='g5
```

g5system.npl (g5main)

```
          INCLUDE T<g5t$(17).g5t_Location$>,g5t$(17).g5t_Module$
          INCLUDE T<g5t$(18).g5t_Location$>,g5t$(18).g5t_Module$
          INCLUDE T<g5t$(19).g5t_Location$>,g5t$(19).g5t_Module$
          INCLUDE T<g5t$(20).g5t_Location$>,g5t$(20).g5t_Module$
          ;
4100      ;------------------------------------------------------------
          ; includes for user modules
          ;------------------------------------------------------------
          ;
          INCLUDE T<g5um$(1).g5um_ModLoc$>,g5um$(1).g5um_Module$
          ;
4200      ;------------------------------------------------------------
          ; includes for GenFive operations
          ;------------------------------------------------------------
          ;
          INCLUDE T<g5_StringCodesLoc$>,g5_StringCodesModule$
```

g5system.npl (g5e)

```
        RETURN
      CASE
        RETURN
      END SWITC
    ;
    CASE
      MAT SEARC
      IF Result
        RETURN (g5e_Name$((VAL(Result$,2)+_g5_EventSize-1)/_g5_EventSize))
      ELSE
        RETURN (_Event$)
```

C:\xfi\g5system.npl (g5main) | INS | SK | Row 1 Col 1

Figure 2: This illustration shows three program file windows open with the original library windows still open in the background. The minimized icons for the two library windows are displayed at the bottom of the screen.

Editor

As a code editor, the Workbench allows you to have multiple files open (and displayed on the screen) simultaneously. Each window contains its own unique display, menu items, toolbar buttons and status bar information.

The types of windows available within the Niakwa Workbench are:

Library Window

The Library Window is an Explorer-like interface that facilitates drag-and-drop and right mouse click functionality to speed common library tasks.

Source Code Window

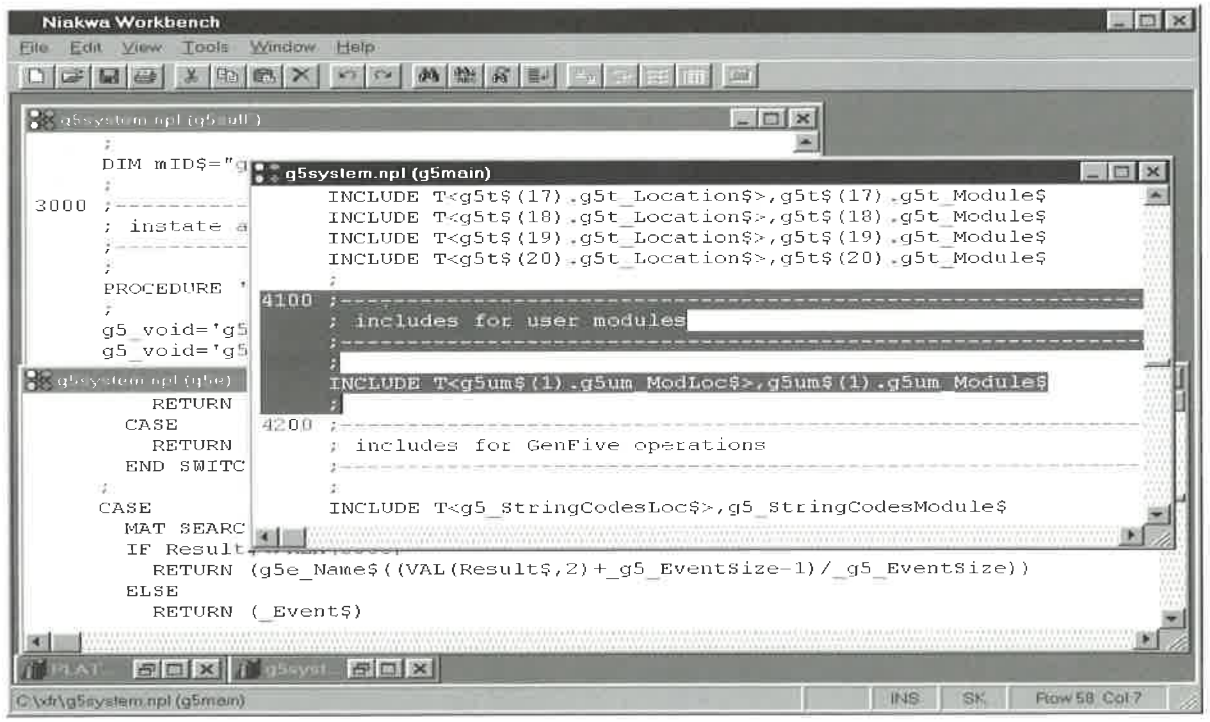
The Source Code Window is used for editing program source code in native NPL with full Windows Clipboard editing features. The window can also be used for editing non-NPL source code, such as VB or text files.

Error Window

The Error Window is used for native NPL code only. It provides a clickable listing of syntax errors within the code. The Editor functions as a syntax highlighting editor that provides significant improvements over normal editing. For example, it provides the tools necessary for removing free space, setting the date and time and renumbering the lines in an NPL program.

Project Window

The Project Window graphically organizes the files selected for a particular project, which may include libraries, modules, source files and other files. It provides direct access to any file contained within the window via right mouse click.


Niakwa Workbench
File Edit View Tools Window Help
g5system.npl (g5main)
C:\xfr\g5system.npl (g5main)
INS
SK
Row 58 Col 7

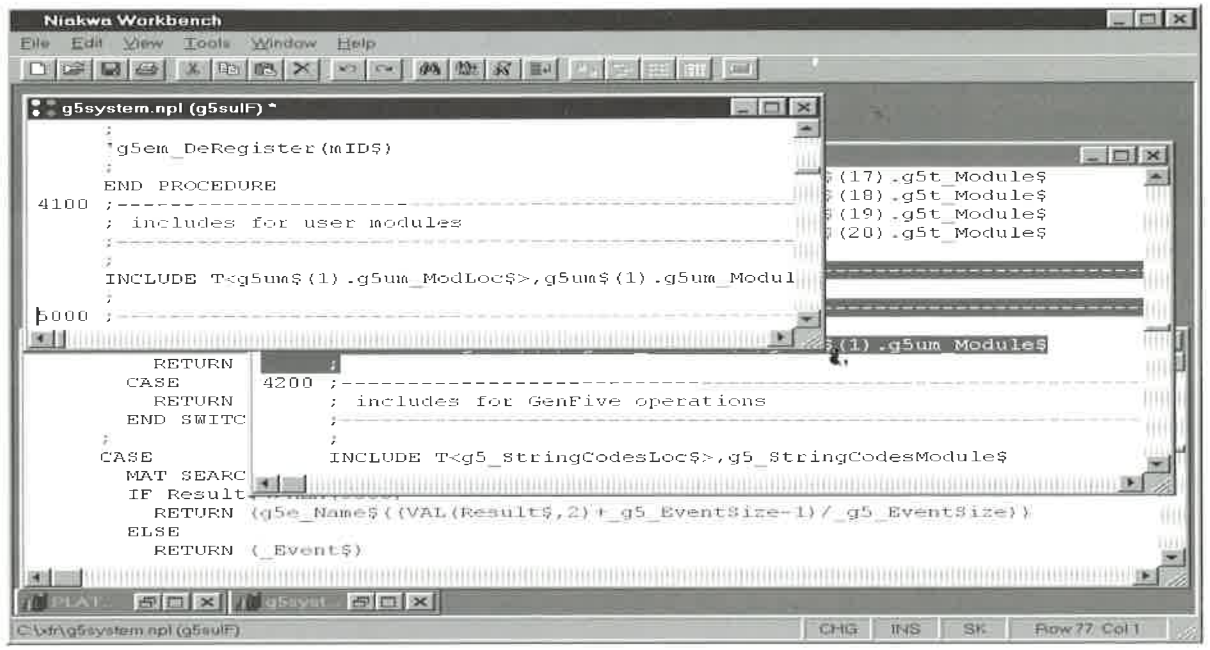
```
       INCLUDE T<g5t$(17).g5t_Location$>,g5t$(17).g5t_Module$
       INCLUDE T<g5t$(18).g5t_Location$>,g5t$(18).g5t_Module$
       INCLUDE T<g5t$(19).g5t_Location$>,g5t$(19).g5t_Module$
       INCLUDE T<g5t$(20).g5t_Location$>,g5t$(20).g5t_Module$
       ;
4100   ;-------------------------------------------------------
       ; includes for user modules
       ;-------------------------------------------------------
       ;
       INCLUDE T<g5um$(1).g5um_ModLoc$>,g5um$(1).g5um_Module$
       ;
4200   ;-------------------------------------------------------
       ; includes for GenFive operations
       ;-------------------------------------------------------
       ;
       INCLUDE T<g5_StringCodesLoc$>,g5_StringCodesModule$
```

```
       RETURN
     CASE
       RETURN
     END SWITC
   ;
   CASE
     MAT SEARC
     IF Result
       RETURN (g5e_Name$((VAL(Result$,2)+_g5_EventSize-1)/_g5_EventSize))
     ELSE
       RETURN (_Event$)
```

Figure 3: This illustrates five lines of program text that are marked for copy to another module. The Workbench uses the Windows Clipboard for cut/paste/copy operations.

Features of the Editor

- Fully graphical environment for working with NPL code
- Simple mouse operations replace many NPL commands
- Multiple document interface (MDI) support for up to 64 separate programs open for simultaneous editing
- Full "editing" commands like **copy, paste, drag-and-drop**
- Global **search/replace/edit** with string search patterns
- Common **windows, menus, toolbars** and **status bars**
- Work with multiple projects simultaneously
- Use of NPL terminology
- Complete **on-line documentation** (help files)
- On-the-fly syntax checking
- Many of the current NPL commands for functions such as retrieving, saving, renaming and deleting files replaced with Windows-like GUI interface
- New **utility commands** previously available only in 3rd party products
- **Colors** syntax for easy detection
- Renumbering commands
- Does not require RunTime to execute
- Ability to open and edit additional text file formats


Niakwa Workbench
File Edit View Tools Window Help
g5system.npl (g5sulF) *
C:\xfr\g5system.npl (g5sulF)
CHG
INS
SK
Row 77 Col 1

```
       ;
       'g5em_DeRegister(mID$)
       ;
       END PROCEDURE
4100   ;-------------------------------------------------------
       ; includes for user modules
       ;-------------------------------------------------------
       ;
       INCLUDE T<g5um$(1).g5um_ModLoc$>,g5um$(1).g5um_Modul
       ;
5000   ;-------------------------------------------------------
```

Figure 4: This illustrates the five lines of copied text from figure 3 pasted into another module.

Project Management

The Niakwa Workbench provides more than just a program editor or debugger since project management tools are built right in. These tools allow the developer to store and maintain multiple files, libraries, data files, and other non-NPL files together as a single project. Projects may also be nested into other larger projects providing ultimate flexibility for the most complex applications.

The Project Management features are as follows:

- Support for any program types: module, object, source file
- Support for libraries (diskimages)
- Support for non-NPL files (anything associated in Windows Explorer)
- Support for other project files
- Drag-and-drop functionality between projects and libraries
- Hierarchical or list views of project contents
- Quick access to project item contents by double-clicking on the item

Debugger

(Currently in development. Delivery date to be announced.)

The Niakwa Workbench includes a debugger which is an interactive monitoring and testing tool to optimize execution and performance of NPL applications developed within the Workbench. The debugger is tightly integrated with the execution engine and editor to provide real-time analysis of data as the application is executed. This is handled through the use of breakpoints and the ability to view variable contents.

Features of the Debugger

The Niakwa Workbench contains the following debugging facilities:

Multiple breakpoints
Individual statements in the editing window can be set, marked or removed as a breakpoint anytime during the debug or editing process. At a set breakpoint, program execution stops, allowing developers the ability to test or direct control of the application by hand.

Single-stepping
Developers may trace through an application line by line as it executes, allowing the behavior of screens or variable contents to be observed.

Variable watch windows
Any variable or constant within the application may be monitored. The value of a selected variable will be displayed and automatically updated whenever a change in value is detected. A value may also be changed by the developer in order to observe an alternate program outcome.

Immediate mode command window
Traditional NPL interpretive mode commands may be entered through this window to perform tasks, just as if it were within an NPL runtime. This window provides a familiar interface for the "command-line" developer, with newer GUI-style operations still available.

Other debugging features include:

- Global search and clear of breakpoints
- Continue-Next operation allows skipping long or repetitive subroutines or constructs
- Error Window provides a clickable list of all program errors
- Syntax errors are automatically highlighted within the code editor
- A standard NPL error message file, which is customizable
- Screen output may be monitored in a separate window

Execution Engine

The “behind the scenes” execution engine within the Niakwa Workbench provides the means to interpret and execute source code and display the output. The core of the execution engine is the same as that for a traditional NPL runtime installation.
The engine can be used to run the application as a complete prototype, or just for a specific module. The engine is also controlled with the debugger features, such as breakpoints and stepping to stop and start program execution.

Although the engine runs in the background of the Niakwa Workbench, it produces standard program output which is observable in its own window.

Features of the Execution Engine

The execution engine has the following features:

- Full compatibility with NPL p-code up to revision 5.00
- Standard compatible output with the MS-Windows platform
- Most system variables and constants can be used
- External library support, including Visual Interface Manager (Vinny) and Open NDM

Compatibility

The Niakwa Workbench is a true 32-bit Windows application, thus the developer must be using Microsoft Windows 95/98 or Windows NT. Although the Niakwa Workbench operates in a Windows environment, the developer is free to edit and manage NPL applications written for any platform in any revision that supports NPL RunTimes. This includes the platforms of DOS, Windows, UNIX (Intel), AIX (RS/6000), HP-UX (HP 9000), and VMS (VAX) and all supported revisions on those platforms.

Developer Knowledge

NPL developers and non-NPL developers alike will find the Niakwa Workbench extremely easy to use with its common graphical user interface, universal editing features and drag-and-drop functionality.
The visual interface and syntax help assists developers of all caliber in achieving a more rapid application development cycle. The seasoned NPL veteran will spend more time programming and less time dealing with previously complex file and library management.

Hardware Requirements

The Niakwa Workbench enhances any currently installed Windows development environment with no additional requirements. However, to fully appreciate the multiple window-interface, a 17” or larger monitor is recommended. In addition, a high performance video card can provide additional performance improvement when developing in a graphical environment.

Operating System Requirements

Windows 95
Windows 98
Windows NT

Availability

The Niakwa Workbench is being introduced initially in conjunction with the release of NPL V, the 32-bit NPL RunTime for Windows environments. The first release includes the program editor and project management features. The debugger will be added in a subsequent near-term revision.

Niakwa Inc.
23600 North Milwaukee Avenue
Vernon Hills, Illinois 60061
(847) 634-8700, Fax (847) 634-8718