General Release Notes
for the Niakwa Programming Language Release 5.00
Report Date - August 10, 1998

I INTRODUCTION

Niakwa is pleased to announce the availability of NPL Release V, providing a new 32-bit MS-Windows RunTime which is compatible with the following operating systems and environments:

MS-Windows 95
MS-Windows 98
MS-Windows NT
Novell NetWare
MS-Windows NT Server
MS-Windows Terminal Server

Release V also introduces the Niakwa Integrated Development Environment (IDE) which is designed to simplify the process of obtaining and staying current with all Niakwa Development products.

The purpose of these release notes is to provide a complete overview of NPL Release V and its components.

NOTE: Refer to the NPL Release V Compendium for a complete discussion of platform specific issues related to NPL Release V and new language features.

Section II provides an overview of the new features which are available using Release V.

Section III contains general information about the online documentation included with Release V.

Section IV discusses "compatibility considerations" as they relate to compatibility with prior NPL releases, performance, and memory.

Section V discusses the product organization of Release V.

Section VI discusses known problems or bugs in the current Release V products that were discovered too late to be corrected on this release.

II FEATURES

Release V of the Niakwa Programming Language continues where NPL Release IV leaves off by improving programmer productivity while modernizing and maintaining the powerful NPL development environment. The major features introduced in Release V, described below, make NPL one of the most productive and modern languages available to serious application developers.

NPL Revision 4.30

- MS-Windows A:\SETUP standardized for all RunTime platforms
- Universal Upgrade enhancements introduced
- New COPYALL batch file introduced for MS-DOS and 386/DOS-Extender RunTime installation
- LIST PUBLIC and LIST DIM statements extended
- Enhanced Boolean expressions
- Improved Mixed Network Authentication implemented
- Default NIAKNETW value changed
- New /C command line option implemented for Instant Vinny support
- New RAW.DRV printer driver introduced
- Enhanced HALT key sequence in MS-Windows RunTime implemented
- $DECLARE enhanced to allow /POINTER parameter
- $OPTIONS byte 38 extended to allow more selective control of explicit variable declarations.
- New $LDATE statement implemented to fulfill Y2K compliance
- New DATA LOAD/SAVE BU statements implemented
- New REDIM statement implemented
- New LIST PROC statement implemented
- READ DC and LISTDC extended with new LDATE restriction

NPL Revision 5.0

- All of NPL Revision 4.30 enhancements included
- Windows 98 support introduced
- New NPL MS-Windows 32-bit RunTime introduced
- Niakwa Integrated Development Environment introduced
- Integrated 2227 asynchronous communications supported in the new MS-Windows 32-bit RunTime
- Support for variables > 64K introduced in the new MS-Windows 32-bit RunTime
- $DECLARE updated to support 32-bit calls from the new MS-Windows 32-bit RunTime
- RTIWIN.INI enhanced to handle both 16-bit and 32-bit specific sections.
- New native file handling statements added: OPEN #, READ #, WRITE #, SEEK #

III DOCUMENTATION

With Release V, Niakwa has introduced the Niakwa Integrated Development Environment (IDE) providing the NPL developer with the complete suite of Niakwa Development tools on a single CD-ROM. In addition to this, Niakwa has included on the new Niakwa IDE CD-ROM the popular Adobe Acrobat Reader along with the complete set of NPL documentation in PDF format.

To accompany the existing Release IV documentation, Niakwa has also provided an online version of the NPL Release V Compendium, which bring under one cover all post NPL Release IV documentation, along with the new features introduced in Revisions 4.30 and 5.00. We encourage you to review the new Release V Compendium to acquaint yourself with the many new features NPL has to offer. The following is a layout of the NPL documentation included on the Niakwa IDE CD-ROM. Refer to Section 2.6 of the Release V compendium for details on installing the Adobe Acrobat Reader. Assuming G: is your current CD-ROM spec (adjust accordingly):

G:\Manuals				
	\ACROREAD			The Acrobat Viewer 16-bit and 32-bit Setup programs
	\PDF_Files			
		\NPL_Docs		
			\R4DOSSUP	Release IV MS-DOS Supplement
			\R4NOVADD	Release IV Novell Addendum
			\R4WINADD	Release IV MS-Windows Addendum
			\R4386ADD	Release IV 386/DOS-Extender Addendum
			\R4NETADD	Release IV NetBIOS Addendum
			\R4UNXADD	Release IV SCO UNIX Addendum
			\R4AIXADD	Release IV IBM RS/6000 Addendum
			\R4AS4ADD	Release IV AS/400 Addendum
			\R4SUPADD	Release IV SuperDOS Addendum
			\R4PGUIDE	Release IV Programmers Guide
			\R4SGUIDE	Release IV Statements Guide
			\R5COMP	Release V Compendium
			\R5Notes	Release V General Release Notes (This document)
		\NDM_Docs		
			\NDMGUIDE	NDM Developers Guide
			\NDM2DOS	NDM MS/DOS/Btrieve Addendum (Rev 1.20)
			\NDM2UNIX	NDM SCO UNIX/C-ISAM Addendum (Rev 1.20)
			\NDM2AIX	NDM IBM AIX Addendum (Rev 1.20)
			\NDM2AS4	NDM IBM AS/400 Addendum (Rev 1.20)
			\NDM2ADD	NDM Revision 1.20 Addendum
			\NDM2NOTE	NDM Revision 1.20 Release Notes
			\NDM5ADD	**NDM Revision 5.0 Addendum**
		\VIM_Docs		
			\VN2GUIDE	Visual NPL Developer’s Guide (Rev. 2.00)
			\VN2NOTE	Visual NPL Release Notes (Rev. 2.00
			\IV1GUIDE	Instant Visual NPL Developer’s Guide (Rev. 1.00)
			\VIM5ADD	**Visual Interface Manager Revision 5.0 Addendum**
		\ODB_Docs		
			\ODBGUIDE	**NPL Gateway to ODBC Programmer’s Guide (Rev. 5.0)**
	\MISC			
		\SCDNOTES		Niakwa SCD Release Notes (Rev. 2.4)

NOTE: Items denoted above in bold are new documents to Release V and are included on the Niakwa IDE CD-ROM only. If hardcopy is required, these documents may be printed directly from the Adobe Acrobat Reader or purchased directly from Niakwa.

IV COMPATIBILITY CONSIDERATIONS

The following section discusses the compatibility issues to be considered with NPL Release IV.

Upward Compatibility

Applications developed using NPL Release III (or earlier) versions of NPL are fully upwardly compatible to Release V and require no changes or recompilation to execute.

Downward Compatibility

Applications developed or modified with NPL Release V operate with prior NPL release RunTime as long as no Release V or Revision 4.30 specific syntax is used.

V PRODUCT ORGANIZATION (SOFTWARE)

The following section details the product organization for NPL Release V.

Niakwa Integrated Development Environment (IDE)

This section introduces all development components distributed as part of the Niakwa IDE Package. The Niakwa IDE includes four new core components in addition to the standard NPL Compiler, Utility, Terminal Support and BESDK files provided in past releases. All components of the Niakwa IDE are distributed on a single CD-ROM in the following directory structure:

```
\NPLV_IDE
        WB              <=== New Niakwa Workbench
        VIM             <=== New 32-bit Visual Information Manager (Vinny)
        OPEN_NDM        <=== New 32-bit Open Niakwa Data Manager (Open NDM)
        ODBC            <=== New 32-bit NPL Gateway to ODBC
        W32BESDK        <=== New 32-bit Windows External Subroutine Development Files
\NPL430
        COMPILER
        UTILITY
        TERMINAL
        DOSBESDK
        W16BESDK
        386BESDK
        SCD             <=== Niakwa Scientific and Communications Drivers
\MANUALS
        ACROREAD        <=== Adobe Acrobat Reader
        PDF_FILES       <=== NPL Online Documentation Files
        MISC            <=== Miscellaneous Release Notes
```

Refer to the NPL Release V Compendium for a complete discussion of all of the above components.

RunTime Package

The NPL Release V RunTime Package contains an NPL RunTime Package Installation Guide and an NPL Release V Gold Key containing a set of Windows Setup diskettes. All NPL Release V products are licensed for the new NPL 32-bit MS-Windows RunTime. In addition, NPL Release V RunTimes continue to include all NPL Revision 4.30 16-bit RunTime products (i.e., MS-DOS, 16-bit MS-Windows and 386/DOS-Extender) to provide continued support for end-user sites which have not yet migrated to 32-bit operating environments.

Refer to the NPL Release V Compendium for a complete discussion of the NPL Release V RunTime Package.

Upgrade RunTime Package

The NPL Universal Upgrade Package is a special registration-based RunTime used to upgrade an old NPL Gold Key to the most current NPL Revision. The upgrade procedure is fully automated, replaces all previous Gold Key diskettes and may be used to increase the capabilities of the previous Gold Key.

Refer to the NPL Release V Compendium for a complete discussion of the NPL Release V RunTime Package.

VI ADDITIONAL INFORMATION

The following items were too late to incorporate into the NPL Release V documentation.

Niakwa IDE Updates

The Niakwa IDE has been updated to include the Niakwa Scientific and Communication Divers package which was previously sold separately. The Niakwa SCD can be found in the following directory on the Niakwa IDE CD-ROM.

```
G:\NPL430
        \SCD
```

Workbench Updates

As of this release, the “ | ” character in a text window of the workbench must always be followed by 3 hex digits. The |hh sequence is the equivalent of an escape sequence representing HEX(hh) embedded in the code.

Some programs that interface with DOS / Windows code may want an ASCII “ | “ character (hex 7C) inside a string, i.e. the equivalent of a British Pound Symbol Alt-156 (£) or as the explicit hex escape sequence |7C.

Incorrect hex escape sequences are not diagnosed as syntax errors at entry time, when syntax checking is turned on, but will be flagged when the program is saved.

Workbench Known Problems

This section discusses known bugs which were discovered too late to incorporate into the Release V final cut of the Niakwa Workbench.

1. Programs with very large long identifier name tables (LIN’s in excess of 32K) cannot be decompiled or produced by B2C of any kind. However, RTI can save programs with LIN’s larger than this. This means that sometimes a program which is valid cannot be viewed by the workbench (which uses B2CWIN32 to get source code.)

   The result of this problem is a fatal error message displayed in the current source window when Workbench attempts to load the file:

   ```
   Fatal  - Invalid long identifier table
   ```

   As a work around, programs or modules with large declarative sections should be broken into two or more modules.

2. If a source file containing invalid escape sequences (i.e., a pipe character not followed by 2 hex digits) is saved, an error diagnostic is issued but the saved program can be invalid (with missing lines etc...) if reloaded.

NPL Known Problems

This section discusses known bugs which were discovered too late to incorporate into the NPL Release V final cut.

1. Constant array-names should be permitted for the sort-array name in MAT SORT but are not. $SOURCE should accept any constant string valued expression for the LIN table argument but did not.

```
0000 STR(rtext$(),,Text)=$SOURCE(_rPCode$,_rLinTable$())
                                               ^
ERR S29 Missing Alpha-Variable

0100 MAT SORT _rArray$() TO mTemp1$(), mTemp2$()
               ^

ERR 28 Missing Alpha-Array
```

2. Logical expressions which contain both boolean (i.e., X meaning X<>0) and parentheses with at least 2 AND or OR operators will LIST incorrectly, but EXECUTE correctly. For example:

```
0010 IF A OR (B AND C)
:LIST
0010 IF A OR (B OR C)
:                 ^    Incorrect operand listed back.
```

This problem can be worked around by using the explicit <>0. For example:

```
0010 IF A<>0 OR (B<>0 AND C<>0)
```

3. If a variable address is used in a SELECT#x with more than 3 characters, the string function $SELECT(#x) returns more that 3 characters. On other platforms, and on previous releases, exactly 3 characters are always returned by $SELECT. For example:

```
0010    A$=”D13AND SOME JUNK”
        SELECT #1<A$>
        B$=$SELECT(#1)
        PRINT B$
    :RUN
    D13ANDSOMEJUNK
```

previous releases displayed “D13"

As a work around, any code that uses the $SELECT(#x) function should store the value in a 3-character string.

4. When STEPping through code, if the current line is overlaid from a boot or program file without doing a CLEAR P and then the program is re-started, breakpoints may be incorrectly determined. This can cause a program to operate incorrectly or fault.

   As a work around, before loading programs during a debugging session use a CLEAR P command.