

README FOR

THE XYWWWB JUMBO ALMOST-ALL-INCLUSIVE U2 FILE

ROBERT J. HOLMGREN AND CARL L. DISTEFANO

[HTTP://XYWRITE.ORG/XYWWWB/](http://xywrite.org/xywwwb/)

The XyWWWb Jumbo U2 collects most of the enhancements to XyWrite 4, XyWrite for Windows, and Nota Bene for Windows available at the XyWWWb website in a single, easy-to-install User Command Set (U2) file, XYWWWB.U2. It includes a comprehensive Help facility and INformation (help) file, XYWWWB.INF. The Jumbo U2 is updated from time to time.

External Windows and DOS programs (all freeware) used by U2 routines are collected and updated in a separate ZIP archive, U2EXTRAS.ZIP (<http://xywrite.org/xywwwb/U2EXTRAS.ZIP>). Download it from time to time to ensure that you have all needed external programs – consult the online [changelog](#) for the latest information about updates and additions to external programs. Unzip U2EXTRAS.ZIP (all files) into either the directory with EDITOR.EXE (“Editor’s directory”) or a subdirectory named .\U2EXTRAS in Editor’s directory.

Note for Users of DOSBox-X or vDosPlus

Jumbo U2 routines that take advantage of the **vDosPlus** or **DOSBox-X** environment assume that the 4DOS shell is running; some of these routines will not work as expected under the default DOS shell. In vDosPlus, make sure that CONFIG.TXT (or applicable configuration file) includes the line “SHEL4DOS=ON” (no quotes). In DOSBox-X, the [config] section of the applicable configuration file (usually DOSBox-X.conf) should include the line “shell = 4DOS.COM” (no quotes).

Quick-Start Instructions

If you are running an earlier version of the Jumbo U2, back up your existing copies of XYWWWB.U2, XYWWWB.INF, and STARTUP.INT.

Step 1

Unzip XYWWWBU2.ZIP into the directory that contains EDITOR.EXE (“Editor's directory”). Overwrite all existing copies of the included files.

Step 2

Add the following two lines to STARTUP.INT; position them below the line that loads SETTINGS.DFL:

```
BX load XYWWWEB.U2Q2 ;*; Load U2
JM 2.XyWebU2.intQ2 ;*; Initialize U2
```

Note that **BX** , **JM** , and **Q2** are functions produced by commanding PFUNC BX, PFUNC JM and PFUNC Q2, respectively.

→ Save STARTUP.INT

Step 3

If you are new to U2, create a Helpkey in your keyboard file: you will use this key to launch U2 routines from the XyWrite command line. The Helpkey can be assigned to any key you choose. An unshifted key is recommended (I use <F11>, scan code 87=... under "TABLE=←" in the .KBD file), but is not required. Helpkey assignment:

```
nn=NO, JM, 2, ., P, r, s, C, M, l, i, n, e, Q2
- or, its equivalent -
nn=NO, JM(, 2, ., P, r, s, C, M, l, i, n, e, )
```

If a U2-compatible XY4.DLG is loaded (see below), either of the following assignments can be used for the Helpkey:

```
nn=$2 <== issues U2 commands only
nn=$X <== issues both U2 and native XyWrite commands
```

→ Save your keyboard file

Step 4

QUIT and re-start XyWrite.

Note on XY4.DLG (Dialog help file)

Make sure that you are running a U2-compatible version of XY4.DLG. One that is suitable for both XyWrite 4 and XyWrite for Windows is located here:

<http://xywrite.org/xywwwweb/XY4DLG.ZIP>

UnZIP it, quit XyWrite, replace your current DLG file (REName XY4.DLG to your current DLG filename, if necessary), then restart XyWrite. The DLG update includes bug fixes and new functionality, with no deprecation of original "factory" services.

Detailed Setup Instruction

Introduction -- for All Users

XYWWWEB.U2 provides the latest versions of code (source Holmgren and/or Distefano) described at the XyWWWWeb site. (A few XyWWWWeb programs are not included because they require machine-specific installation, or because they do not operate from U2.)

Note that, although XYWWWEB.U2 is just over 1 MegaByte, the actual consumption of XyWrite available memory is less than 20 KiloBytes, due to Help file indexing. Moreover, by merging into U2 your own personal programming (formerly loaded verbatim in a .SGT files by most users), you may release significant memory, for a net reduction of XyWrite memory overhead. XYWWWEB.U2 is extremely parsimonious about Save/Get and memory usage; although we reserve a large range of 200+ Save/Gets (600-799; and 997-999, 1700-1708, 1799, 1816, 1996-1999), at any moment very few of these are in use, and in most cases they are emptied (cleared) before routines terminate (see ClearSgt in XyWWWWeb.INF).

The present documentation provides directions for installation and use. For a discussion of the why's and wherefore's of our usage scheme, please read PARSEFRM.DOC and REORGaNIZ.DOC. These are old and deprecated documents, so *caveat emptor* -- but familiarity with these texts may enhance your enjoyment and use of the XyWWWWeb routines.

Before You Start

Make sure that Error Help and Command Override are both turned OFF. Since these features are inactive by default, many users will not need to make any change. You can verify your current values by commanding, respectively, VA/NV \$EH<cr> and VA/NV \$OO<cr>; in each case, the PRompt line should display "o" (zero). If either query returns "1", the default value has been changed, probably at startup. An XPL program may have toggled either EH or OO. SETTINGS.DFL (also XWSET.DFL, NB.DFL) and printer (*.PRN) files can also manipulate Error Help; SEARCH your DFL file for "df eh=1" and change the value to "o", or deactivate the line with a leading semi-colon. Scan your STARTUP.INT or equivalent file for statements such as "**BX d eh=0Q2 "** and/or "**BX ooQ2 "**, and remove them (or proactively enter this line:

```
BX d eh=0Q2 BX ooQ2 ;*;
```

at the end of STARTUP.INT). An XPL program could toggle either EH or OO. SETTINGS.DFL and printer.PRN) files can manipulate Error Help also; Search for "df eh=1" and change the value to 0 (zero), or deactivate the line with a leading semi-colon, thus:

```
;df eh=1
```

SETTINGS.DFL (XWSET.DFL | NB.DFL)

Additional suggestions for permanent values in SETTINGS.DFL (replace current values, if they exist -- don't just add what follows as a block!):

```
;1A=0 honors EOF marker 1Ah (Ascii-26); 1A=1 ignores it
df 1a=0
;
;BI sets Beep [Barf] Inhibit (1 turns off error beeps)
df bi=1
;
;D1 is the Delete Stack (Clipboard) size (# of entries,min # of
chars)
;df d1=30,4 <==default
df d1=200,1
;
;EH Error Help is an absolute no-no! (0=Off, 1=Enabled)
df eh=0
;
;FZ is Date in DIR listings, etc
df fz=MM-DD-YYYY
;
;If you use Windows:
;KS must be COMMENTED OUT because WinNT[2K|XP] balks at it
;KS is KeyboardSpeed: repeat rate, delay
;df ks=0,1
;
;LA sets the LAnguage (i.e. CodePage, or character set)
;In XyWrite|XyWin ONLY (do NOT adjust in NBWin!):
df la=437 <- XyWrite 4
df la=850 <- XyWin
;
;OD=1 - left and right OffsetDisplayed on screen; 0=not displayed
df od=0
;
```

```

;WA is WAit time for error messages (wa=18: ~1 second)
df wa=4
; or even
df wa=0
;
;WF=1 Wraps text to Fit window borders in draft and expanded views
;Doesn't wrap in Page-Line mode, only in WG mode
df wf=1
;
;XD=1 sets directory to read-only (& hilites current line)
;XD=0 makes directories editable/definable
df xd=0
;
;Preserve upper and lower case in spelling commands, e.g. Auto-
Expand
; ('Zero Capitalization')
df zc=1
;

```

PRiNter Files

Ensure that font name STANDARD is associated in your PRiNter file(s) (*.PRN) with a *monospace* display and printer font (e.g. COURIER10), in both the FA: and FP: tables (SEarch for “FA:**N**” and “FP:**N**”). The lines should say something like this:

```

FA:3
COURIER10=COURIER,1,66,154,201,211,240,250,260,280,290,LINEP
RINTER,STANDARD
...
FP:3
COURIER10=STANDARD,1,66,154,201,211,240,250,260,280,290
...

```

Nota Bene for Windows Users please Note Well: Compatibility with earlier versions of NBWin is considerable, though less than 100%; with recent versions, however, all bets are off. Earlier versions of NBWin have bugs, as well as “features” that need to be reckoned. (**Caution:** Some of the following advice may not apply to, or work with, recent versions of Nota Bene for Windows.)

One such feature is “simple sequential windows”, which (by default) automatically changes high window numbers to lower numbers as low windows are closed and become available for use. This confuses XyWWWeb routines that open and close

some windows but expect other windows to retain fixed window numbers. Candidly, we do not understand what useful purpose sequential windows serve. Sequential windows can, and must, be disabled, by means of a new NB.DFL VArIable:

```
;WN=1 turns off simple sequential windows
df wn=1
```

NBWin and also XyWin users must ensure that Message Boxes are not permitted to pop up when errors occur in the midst of a U2 routine. U2 routines do their own error handling, whereas Message Boxes require manual “OK” confirmation before processing can continue. You **MUST** compel error messages to be displayed on the PRompt line, and not in pop-up boxes:

```
;MB=0 turns off Message Boxes (forces display on PRompt
line)
df mb=0
```

Do you want the spell checker to beep as you type? Neither do we. Put func **AC** toward the end of NBSTART.INT (command “pfun AC<cr>”), to turn off Auto-Correction.

The necessity for other tweaks to the NB setup may become apparent with experience. We have built some NBWin fixes into XyWWWeb.U2, to relieve you of the need for action; but for now, the above fixes are required. *Repeat: they apply only to NBWin users.*

Installation

To use XYWWWEB.U2, you need four things: 1) the U2 file itself, LOAded in memory (usually via STARTUP.INT); 2) the INFormation file, XYWWWEB.INF; 3) the REGistry file, XYWWWEB.REG; and 4) a dedicated <Helpkey>, which you must configure in your KeyBoarD file.

1) Copy XYWWWEB.U2, XYWWWEB.INF, and XYWWEBU2.REG to your XyWrite subdirectory (the directory occupied by EDITOR.EXE, XWEXE.EXE, or NOTABENE.EXE). If XYWWWEB.REG does not already exist, REName XYWWEBU2.REG to XYWWWEB.REG.

2) Add two lines like the following to STARTUP.INT (Xy4), XWSTART.INT (XyWin), and/or NBSTART.INT (NBWin); in the first line, change the drive and path as necessary:

```
BX load d:\xy4\xywwweb.u2Q2 ;*;  
JM 2.XyWebU2.intQ2 ;*;
```

3) Create a “Helpkey” to launch programs from the command line: Call your .KBD file and assign the following to the key (nn) of your choice:

```
nn=NO, JM, 2, ., P, r, s, C, M, l, i, n, e, Q2  
- or, its equivalent -  
nn=NO, JM (, 2, ., P, r, s, C, M, l, i, n, e, )
```

If a U2-compatible XY4.DLG is loaded (see Note on XY4.DLG, above), either of the following assignments can be used for the Helpkey:

```
nn=$2 <== issues U2 commands only  
nn=$X <== issues both U2 and native XyWrite commands
```

4) STore these changes to disk.

5) QUIT and restart XyWrite.

6) To verify that XYWWWEB.U2 is properly installed, command:

```
VA/NV $U2<cr>
```

XyWrite|NB should report the full path and filename for the U2 file on the status (PROMPT) line. If that works, command:

```
HELP<Helpkey> or HELP ABOUT<Helpkey>
```

Note that you are using your newly-configured <Helpkey> to launch a routine instead of the carriage return (“<cr>”) or Enter key. XyWrite should call XYWWWEB.INF and report the installed XYWWWEB.U2 version and release date.

7) *XYWWWEB.REG Configuration*: XYWWWEB.REG has nothing whatsoever to do with the Windows Registry; it is a single, plain-text file, specific to the Jumbo U2, that adheres to the older .INI file format. XYWWWEB.REG contains “User Variables” (a/k/a keys) which point to user information or preferences (a/k/a values) that must be supplied for the routine to work properly. If required User Variable information has not been entered the first time you try to run such a routine, you will be prompted to do so. XYWWWEB.REG also contains numerous optional User Variables which need not be set but can be used to personalize various U2 routines. User Variables for specific routines are (usually) explained in the Help screens for

those routines. Command HELP routine_name<Helpkey>, or HELP user_variable_name<Helpkey>, or simply Search for the user variable name in XYWWWEB.INF.

If XYWWWEB.REG does not already exist in the directory with EDITOR.EXE, the Jumbo U2 will create it for you. (See above: Quick-Start Instructions, Step 3, first bullet point.) This “canned” REG file (which is created from the installation file XYWWWEBU2.REG) should allow all or most U2 routines to run without further user intervention. To change these pre-programmed settings to suit your personal preferences, you must edit XYWWWEB.REG. Before doing so, however, please read REG-READ.ME and follow all the instructions carefully.

How to Replace an Old Version of XYWWWEB.U2 with the Latest Version

Essentially, just overwrite the old XYWWWEB.U2 with the new one. Add your own U2-formatted programming, if any, in the space provided near the bottom of the file. **Avoid name clashes** between your personal U2 frames and those present in XYWWWEB.U2. REG-READ.ME contains further information.

External Files

A number of external Windows and DOS programs and other helper files are widely used in U2. Starting on 6 November 2016, they became available as a separate ZIP archive, [U2EXTRAS.ZIP](http://xywrite.org/xywwwweb/U2EXTRAS.ZIP), available at <http://xywrite.org/xywwwweb/U2EXTRAS.ZIP>. Unzip the entire contents of U2EXTRAS.ZIP into Editor’s directory, or, if you prefer, in a subdirectory of Editor’s directory named U2EXTRAS, overwriting any already-existing files.

Usage

To launch a routine (or “frame”), type its name on the CMLine and hit your Helpkey:
FRAMENAME<Helpkey>

Supplementary instructions may be passed to a frame that accepts command-line arguments. Some frames require one or more arguments; in others, arguments are optional (or nonexistent). Arguments are set apart from the framename by a space; multiple arguments normally are separated from each other by commas (but consult the Help screen for each routine):

```
FRAMENAME argument1, argument2, ...<Helpkey>
```

An important example of a routine that accepts an argument is HELP, which provides on-line usage advice for most XyWWWWeb routines. This Help INFORMATION is contained in a freestanding text file named XYWWWEB.INF, which is automatically accessed by routine HELP, but also can be opened

and freely searched or browsed. Indicate the information you desire by passing a framename to HELP as an argument:

```
HELP framename<Helpkey>
```

e.g.:

```
HELP ViewURL<Helpkey>
```

```
HELP Help<Helpkey>
```

Keyboard-File Usage

A routine called PrsKbdArg is used to assign a U2 routine and any associated arguments to a key. Case is sensitive! The basic usage (no arguments) is:

```
nn=JM, (, 2, ., P, r, s, K, b, d, A, r, g, ) , f, r, a, m, e, n, a, m, e, NO
```

Example:

```
19=JM, (, 2, ., P, r, s, K, b, d, A, r, g, ) , R, u, n, c, o, d, e, NO
```

The concluding function NO (NoOperation) is mandatory.

A single argument is coded thus:

```
nn=JM, (, 2, ., P, r, s, K, b, d, A, r, g, ) , f, r, a, m, e, n, a, m, e, , a, r, g, NO
```

Example:

```
25=JM, (, 2, ., P, r, s, K, b, d, A, r, g, ) , d, i, a, l, , C, O, M, 2, NO
```

Multiple arguments to a single frame are separated from each other by func CO:

```
nn=JM, (, 2, ., P, r, s, K, b, d, A, r, g, ) , f, r, a, m, e, , a, r, g, 1, CO, a, r, g, 2, CO, a, r, g, 3, NO
```

Multiple frames may be assigned to a single key:

```
nn=JM, (, 2, ., f, r, a, m, e, 1, ) , JM, (, 2, ., f, r, a, m, e, 2, )
```

which may also be written as:

```
nn=JM2, ., f, r, a, m, e, 1, Q2JM, 2, ., f, r, a, m, e, 2, Q2
```

Getting Started

The following commands will help you become acquainted with the array of powerful features in XyWWWWeb:

```
INDEX<Helpkey>
```

displays a categorical list, with brief descriptions, of all routines in the XyWWWWeb Jumbo U2 file. You can search the Index for specific Help topics using one or more arguments, for example:

```
INDEX ansi<Helpkey>
```

```
INDEX defined text<Helpkey>
```

```
INDEX search<Helpkey>
```

The Index also can be summoned with commands

```
HELP IDX<Helpkey>  
HELP INFO[RMATION]<Helpkey>  
HELP CONTENTS<Helpkey>  
TABLE INDEX<Helpkey>
```

HLIST<Helpkey> produces a sorted listing, with brief descriptions, of all routines in the U2 file, including all routines added by users (HLIST requires two helper files, AI3.EXE and REGEX.A3X, to be present in Editor's directory; both files are included in [U2EXTRAS.ZIP](#)).

CITE UserVar<Helpkey> (issued against HLIST output) produces a list of all routines that include User Variables

HELP [framename]<Helpkey> summons usage information for any routine

Save/Get Usage

The use of extended (memory-resident) Save/Gets in XYWWWEB.U2 is mostly confined to the range 600-799. Additional memory-resident Save/Gets used by some routines are: 997-999, 1700-1708, 1799, 1816, 1996-1999. In the interest of keeping initial installation and future upgrades hassle-free, we ask that users reserve these SaveGet ranges for U2's exclusive use and RENUMBER third-party programs if they intrude on this range, to avoid conflicts with XyWWWWeb routines.

Reminder: Use XyperLinks! They are handy, and fast. They work just like browser hyperlinks, except that they do much more: fetch files from the Internet, jump to a REGistry variable so you can fill in data, open a Web browser on a particular page at a remote website, run XPL and Windows programs, etc. etc. Put your cursor on the XyperLink, and hit <Helpkey>.

Feedback

Your comments and suggestions are welcome, and often encourage further development. We are always delighted to hear from users.

Enjoy!

C.L.Distefano <cld (at) acedsl (dot) com>

R.J.Holmgren <holmgren (at) acedsl (dot) com>

LastRev. 5 January 2023