

APL64 2026.0.1 Update

Table of Contents

Overview	3
Interpreter Improvements.....	3
Performance Improvements for these primitive functions and system functions.....	3
New ☐ FF and ☐ FL System Functions.....	3
New ☐ RNET system function: R-statistics & graphics tools	3
New ☐ ODBC System Function: Open Database Connectivity (ODBC) interface	3
☐ SQL and ☐ SQLite Improvements	3
☐ CSE Improvements.....	3
C# Script Engine Manual updated.....	3
Footer Date now represents CSE documentation version.....	3
The ☐ CSE GetValue and GetValueEx methods will return a .Net ValueTuple as an APL64 vector	4
Easily design ☐ CSE Scripts without repetitive concatenation	4
☐ ZIP: CreateFromDirectory action has new option to overwrite the target file.....	4
☐ EVAL: Added support for strings and arrays of Unicode characters.....	5
☐ UCASE and ☐ LCASE Improvements	5
APL64 strings: Additional ASCII control (escaped) characters supported	6
☐ FCREATE: File tie number already in use exception.....	6
☐ UCS: A bug in the UTF-8 encoding when used to output code values has been corrected	6
☐ STRING 'Split' avoids INDEX ERROR when RemoveEntryEmptyEntries argument is not present	7
APL64 Developer Version GUI Improvements.....	7
Editing and Syntax Coloring Improved.....	7
A blank row appeared between the executed statement and result in the History after unchecking 'Is Editable' option.....	7
New Menu item: Edit EDSS Column Header Is R1:C1 Cell Reference Style	7
New Menu: Edit Font for APL Editing.....	7
New Menu: Edit Enable Legacy Virtual Space in Text Editor	7
New Menu: Edit Enable Legacy Scroll Below Text in Text Editor	7
Added a text file import option to the Fetch Object (Ctrl+G) dialog	8
Menu Edit Add Inline Comment : Start/End keyboard shortcuts updated.....	8

Dragdown execution improved in the History pane.....	8
View License Password in the License Status dialog when activating on-line.....	9
Re-executing a statement in the history will now correctly return the caret to the APL command line.	9
Session Variable Editor	9
Improved Font Size and Stroke Weight for the Row and Column Numbers	9
Option to select Index Origin for Row and Column Numbers	9
Editing a Text Variable: Save text (rank < 3) to .txt file.....	10
Option to left align numeric cell values	10
User Documentation.....	12
New Menu Item: Help Developer Version GUI History and Edit Session Keyboard Shortcuts	12
Help Documentation Options ... Download dialog.....	12
Using ☐ SQLite	12
The default string data line continuation background color is now white	12
Replace All Tool performance is improved for most scenarios	12
Virtual Keyboard: Rendering of descender glyphs improved	12
Printing from the APL 64 Developer Version	13
Row numbers for history and variable editor.....	13
Printing from a floating variable editor pane is now possible	13
User Tool with a Ctrl+T shortcut will now override the default Ctrl+T action.....	14
Objects Editor Autosave Options: Selected autosaved object definition	14
The transpose glyph (⌵) is now available on the German (DE-DE) Keyboard Definition 6-key	14
The caret offset in History Pane changed to left margin from 6-space indent	15

Overview

This document describes the improvements in this APL64 2026.0.1 Update. Enjoy the new APL64 features. Thanks to the APL64 programmers who provide valuable feedback to the APL64 development team.

Interpreter Improvements

Performance Improvements for these primitive functions and system functions

- Reverse (Φ) primitive function for Φ BoolVector
- Find (ϵ) primitive function for Vector ϵ Scalar
- $\square$ DEB, $\square$ DLB, $\square$ DLTB, $\square$ DTB, $\square$ LJUST, $\square$ MATRIFY, $\square$ MATTOSS, $\square$ RJUST, $\square$ ROWFIND, $\square$ SSTOMAT, $\square$ WHERE, & $\square$ WORDREPL

New $\square$ FF and $\square$ FL System Functions

The $\square$ FF and $\square$ FL system functions find the first or last occurrence of a string contained in another string, respectively. To learn more about $\square$ FF and $\square$ FL, in the APL64 Developer version go to **Help | APL Language | System Functions**.

New $\square$ RNET system function: R-statistics & graphics tools

The APL64 system function $\square$ RNet is a .Net interface to the R statistical and graphics toolkit. $\square$ RNet is cross-platform and uses the x64 R-engine for greater capacity and performance. To learn more about $\square$ RNet, in the APL64 Developer version go to **Help | APL Language | Using $\square$ RNet**.

New $\square$ ODBC System Function: Open Database Connectivity (ODBC) interface

The APL64 system function $\square$ ODBC system function provides a structured query language (ODBC)-based interface to a relational database. The object model for $\square$ ODBC includes methods and properties to easily create a secure, high-performance application system data interface. To learn more about $\square$ ODBC, in the APL64 Developer version go to **Help | APL Language | SQL System Function | Using $\square$ ODBC**. The APL64 $\square$ ODBC system function can be used to access Microsoft Access database files using the no-cost [Microsoft Access Engine Redistributable](#), even if Microsoft Access is not installed on the target workstation.

$\square$ SQL and $\square$ SQLite Improvements

Exec, ExecDelete and ExecSelect actions now support value substitution and have short synonyms.

$\square$ CSE Improvements

C# Script Engine Manual updated

The **Help | APL Language | $\square$ CSE | C# Script Engine Manual** has been updated, and the source code for all examples is now contained within this document.

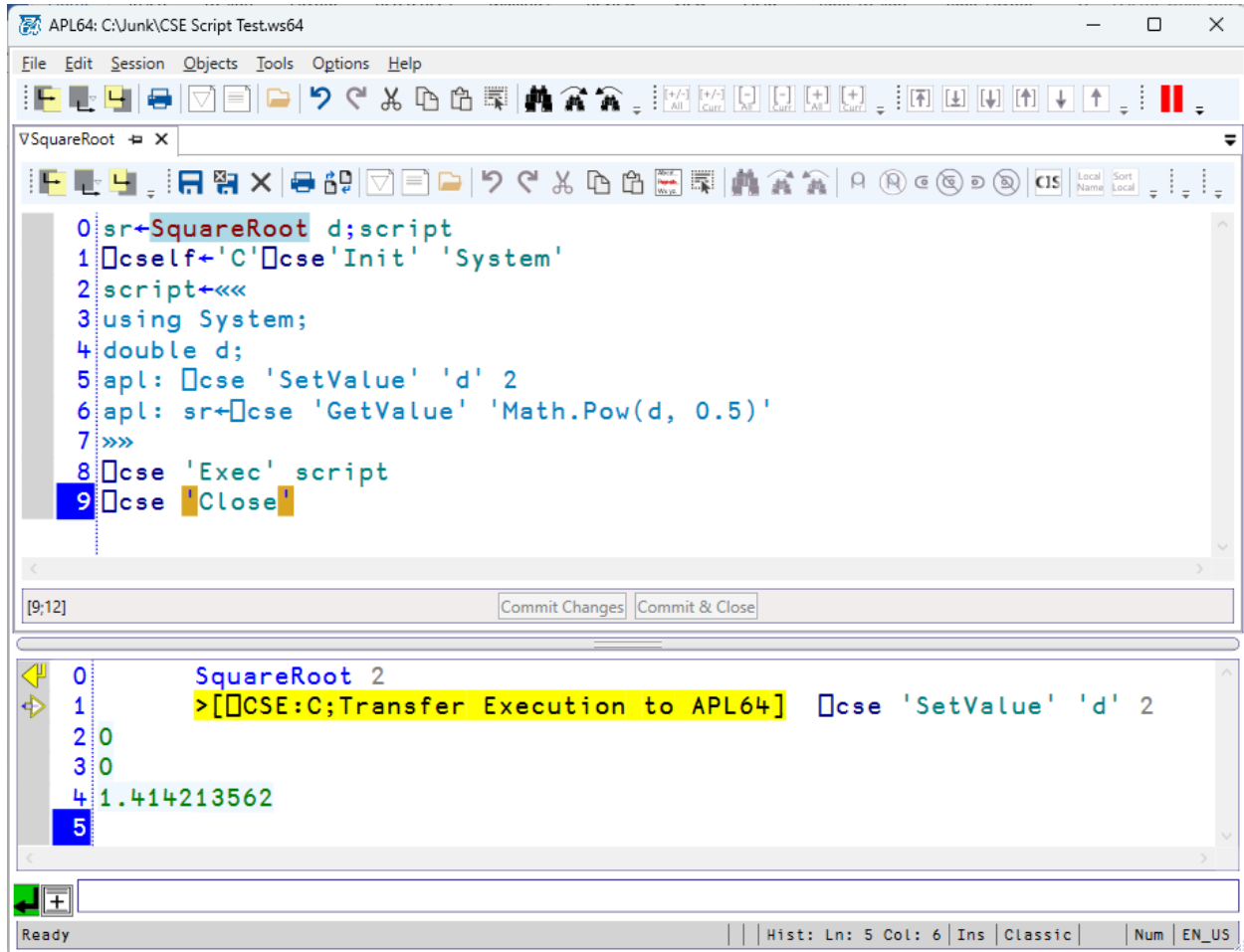
Footer Date now represents CSE documentation version.

The document footer date now indicates the version of the **Help | APL Language | $\square$ CSE quick start guide** and manual documents. The file name number scheme in these file names is obsolete.

The ☐ CSE GetValue and GetValueEx methods will return a .Net ValueTuple as an APL64 vector

Easily design ☐ CSE Scripts without repetitive concatenation

The APL64 string line continuation structure (script←«« ... »») is supported for a multi-line ☐ CSE script.



☐ ZIP: CreateFromDirectory action has new option to overwrite the target file

zipFileName ☐ ZIP 'CreateFromDirectory' srcDirName compLevel inclBaseDir *[overWrite]*

```

0 zip '?'
1 ZIP Documentation Summary
2 zipFileName ZIP 'CreateFromDirectory' srcDirName compLevel inclBaseDir [overWrite]
3 zipFileName ZIP 'ExtractFile' zippedFilePath destFileName overWrite
4 zipFileName ZIP 'ExtractFiles' zippedFilePaths destFileNames overWrite
5 zipFileName ZIP 'ExtractToDirectory' destFolder overWrite
6 hetVector ← zipFileName ZIP 'FileProperties' zippedFilePath
7 String[] ← zipFileName ZIP 'GetAllFilePaths'
8 CharMat ← ZIP 'Help'
9 CharMat ← ZIP '?'
10 zipFileName ZIP 'InsertFile' zippedFilePath sourceFileName compLevel
11 zipFileName ZIP 'InsertFiles' zippedFilePaths sourceFileNames compLevel
12 zipFileName ZIP 'RemoveAll'
13 zipFileName ZIP 'RemoveFile' zippedFilePath
14 zipFileName ZIP 'RemoveFiles' zippedFilePaths
15
16 compLevel: Fastest NoCompression Optimal SmallestSize
17 string[] : zippedFilePaths, destFileNames, sourceFileNames
18

```

Ready | Hist: Ln: 18 Col: 6 Ins | Classic | EN_US

☐ EVAL: Added support for strings and arrays of Unicode characters

```

82 x←eval "(2p «abc») 'ΩM' «xyz» 'pq'"
83 x
84 abc abc ΩM xyz pq
85 [dr x[1]
86 326
87 [dr x[2]
88 326
89 [dr x[3]
90 164
91 [dr x[4]
92 326
93 [dr →x[1]
94 164
95 [dr →x[2]
96 162
97 [dr →x[3]
98 164
99 [dr →x[4]
100 82

```

☐ UCASE and ☐ LCASE Improvements

These system functions now support APL64 values of any type, rank and shape.

```
0 ⌐lcase 'This iS APL' «ABCDE»
1 ⌐lcase (2 3⌐a) 'ABCD' 'DEF'
2 (10 10⌐AL)≡⌐lcase 10 10⌐A
3 'ω'≡⌐lcase 'Ω'
4 'abc' (⌐10) 'xyz'≡⌐LCASE 'ABC' (⌐10) 'xyz'
5 this is apl abcde
6 abc abcd def
7 def
8 1
9 1
10 1
11 ⌐ucase 'This iS APL' «ABCDE»
12 ⌐ucase (2 3⌐a) 'abcd' 'def'
13 (10 10⌐A)≡⌐ucase 10 10⌐AL
14 'Ω'≡⌐ucase 'ω'
15 'ABC' (⌐10) 'XYZ'≡⌐UCASE 'ABC' (⌐10) 'xyz'
16 THIS IS APL ABCDE
17 ABC ABCD DEF
18 DEF
19 1
20 1
21 1
22
```

APL64 strings: Additional ASCII control (escaped) characters supported

- \a: Terminal bell
- \b: Backspace
- \f: Formfeed
- \v: Vertical Tab

☐ FCREATE: File tie number already in use exception

When the FILE TIE ERROR is presented, the file will no longer be created.

☐ UCS: A bug in the UTF-8 encoding when used to output code values has been corrected

□ STRING 'Split' avoids INDEX ERROR when RemoveEntryEmptyEntries argument is not present

APL64 Developer Version GUI Improvements

Editing and Syntax Coloring Improved

Editing and syntax coloring is vastly improved in the function editor.

A blank row appeared between the executed statement and result in the History after unchecking 'Is Editable' option

This occurred with the first statement executed in the APL command line.

New Menu item: Edit | EDSS Column Header Is R1:C1 Cell Reference Style

This option is used to apply the R1:C1 or A1 column header to EDSS worksheet. The default is not checked. If checked, the EDSS column headers will be numbered (1 ...). If not checked, the EDSS column headers will be letters (A ...). Changing this setting affects EDSS editors created after the change. Note: This setting does not affect the cell references in the EDSS formula bar.

New Menu: Edit | Font for APL Editing

APL64 now supports additional fixed width (non-proportional) fonts. This **Edit | Font for APL Editing** menu item is used to specify the APL font for the History and editor. The fonts available include the current **APL385 Unicode** and new **APL386 Unicode** and **APL64Font** fonts. The implementation of fonts in APL64 is modified to use the installed fonts rather than previously where fonts were embedded within the APL64.exe application. These fonts differ in line spacing and glyph styling. The user-selected font is also used when printing from the APL64 developer version (**File | Print**).

These are the characteristics for each of the fonts:

- The APL64 developer version default font is APL385 Unicode. The APL385 Unicode font is monospaced for all current APL64 glyphs, except for these glyphs:
 $\text{E}_L, \text{U}_L, \text{S}, \text{r}, \text{f}, \text{c}_R, \text{U}_L$
- The APL386 Unicode font is monospace for all current APL64 glyphs, except for these glyphs:
 $\text{J}, \text{L}, \text{r}, \text{r}, \cong, \cong, \text{E}_L, \text{U}_L, \text{S}, \text{c}_R, \text{U}_L$, but it has no representation for these glyphs
 $\text{r}, \text{f}, \text{f}$
- The APL64Font is monospaced for all current APL64 glyphs.

New Menu: Edit | Enable Legacy Virtual Space in Text Editor

When you select Enable Legacy Virtual Space in Text Editor, the caret may be placed to the right of existing text in 'virtual space'. Entering text in virtual space will convert the virtual space on the associated line to physical text. This option applies to the classic history pane, the command line, the function editor and pure text elements in the variable editor. The default is not checked.

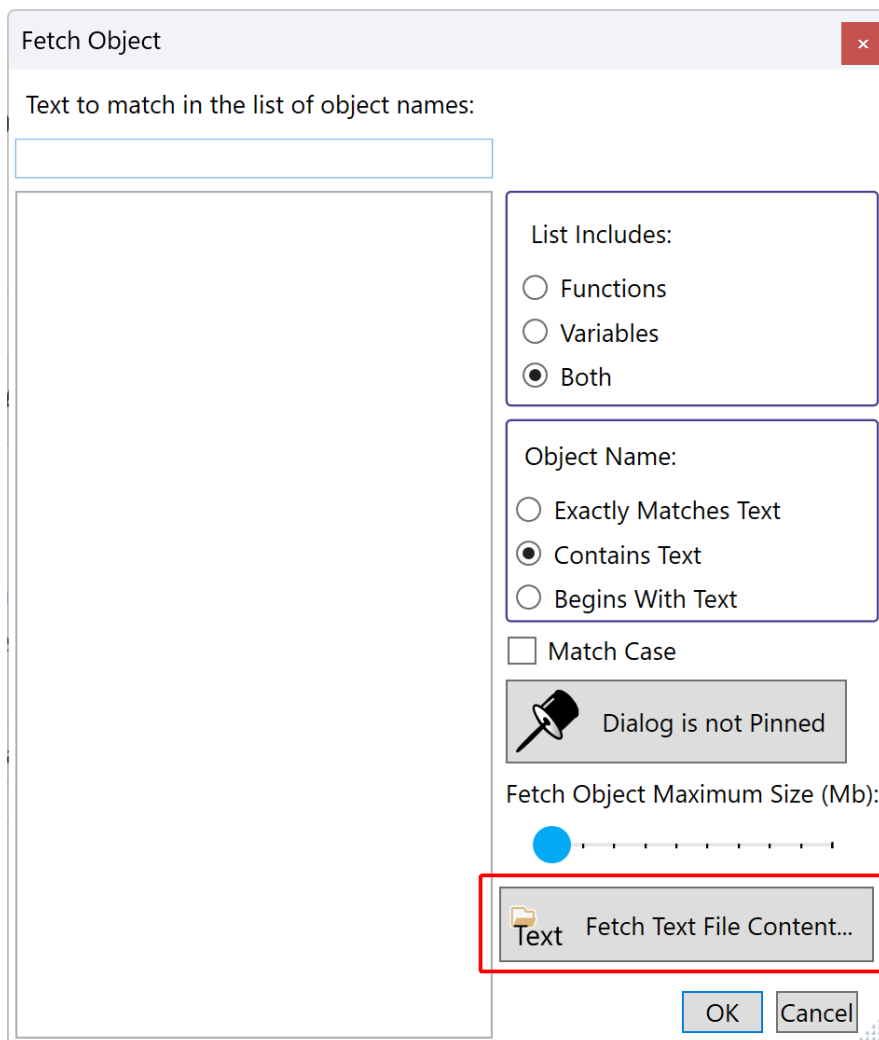
New Menu: Edit | Enable Legacy Scroll Below Text in Text Editor

When you select Enable Legacy Scroll Below Text in Text Editor, it is possible to vertically scroll the existing text in the pane to display empty, non-editable, white space below the existing text. If not checked, the

visibility of the existing text will be maximized. This option applies to the classic history pane, function editor and pure text elements in the variable editor. The default is not checked.

Added a text file import option to the Fetch Object (Ctrl+G) dialog

This option will import a text file into the History or editor that has the focus at the caret position at the time the text file is imported with clicking the OK button in the Fetch Text File Content dialog.



Menu **Edit** | **Add Inline Comment**: Start/End keyboard shortcuts updated

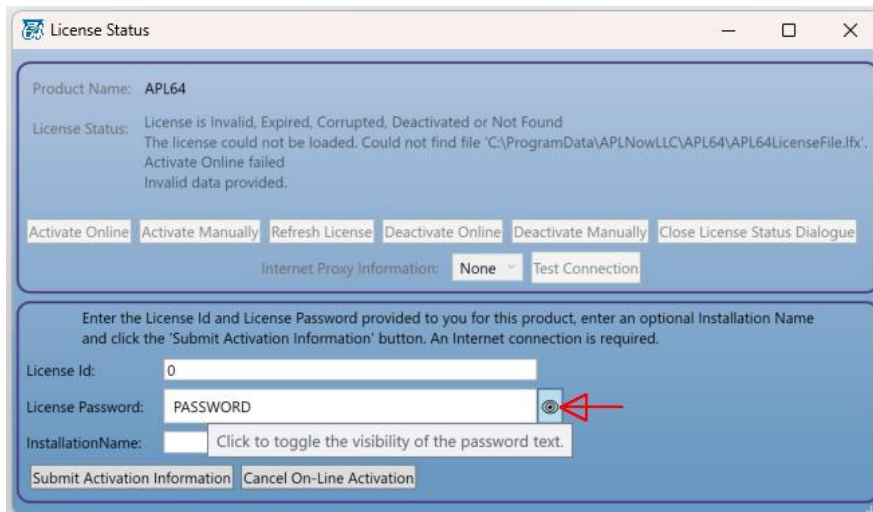
Replaced 'Ctrl' with 'Shift' for inserting starting and ending inline comment glyphs.

Start Ⓐ	Alt+Shift+Z	Add Inline Comment	▶
End Ⓐ	Alt+Shift+X	Delete Inline Comment	▶

Dragdown execution improved in the History pane

View License Password in the License Status dialog when activating on-line

A button has been added to the License Password edit field to toggle the visibility of the activation license password.



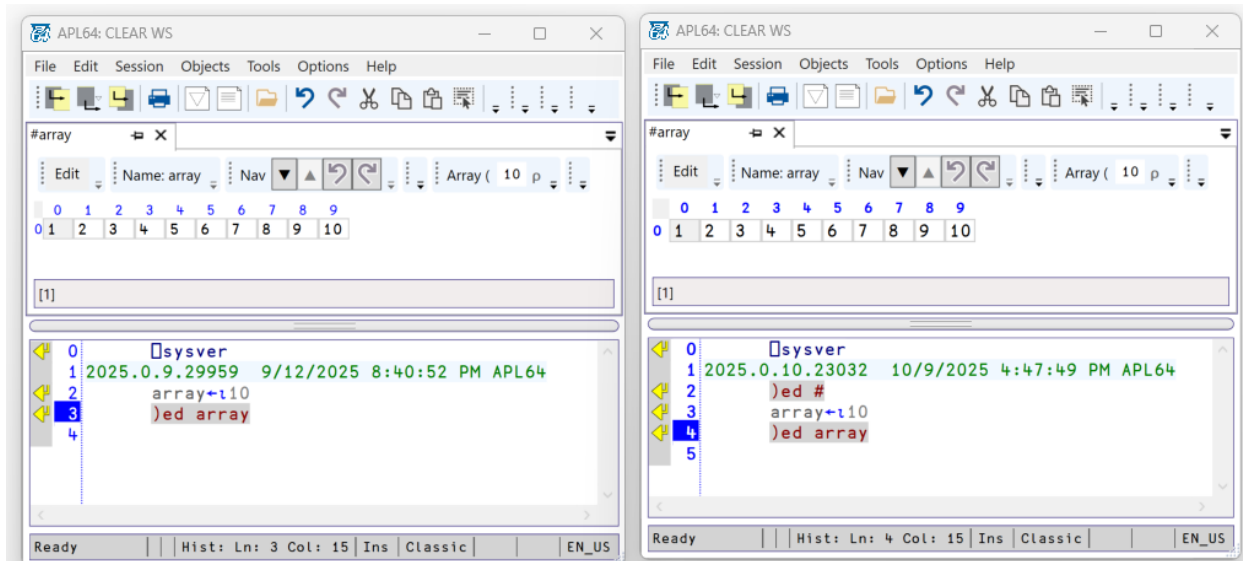
Re-executing a statement in the history will now correctly return the caret to the APL command line

This just applied to the Sequential, Grouped and Separated History Formats.

Session Variable Editor

Improved Font Size and Stroke Weight for the Row and Column Numbers

Side-by-side comparison between APL64 2025.0.9 (left) and this update (right):



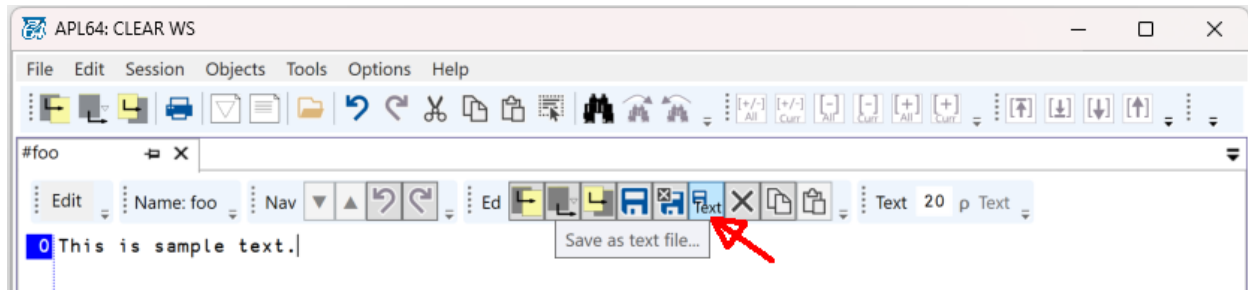
Option to select Index Origin for Row and Column Numbers

The new menu item Objects | Variable Editor | Row Numbers | Row & Column Numbers Index Origin provides this option. The selected value (one, zero, or current ☐ IO) will be used as the row and column

index origin for a new editor. Changing the row and column index origin will not affect APL64 variable editors which were created prior to the change.

Editing a Text Variable: Save text (rank < 3) to .txt file

A new option was added to the variable editor toolbar to save the text data of rank < 3 to a txt file, when possible. See below for the location of the new toolbar button:

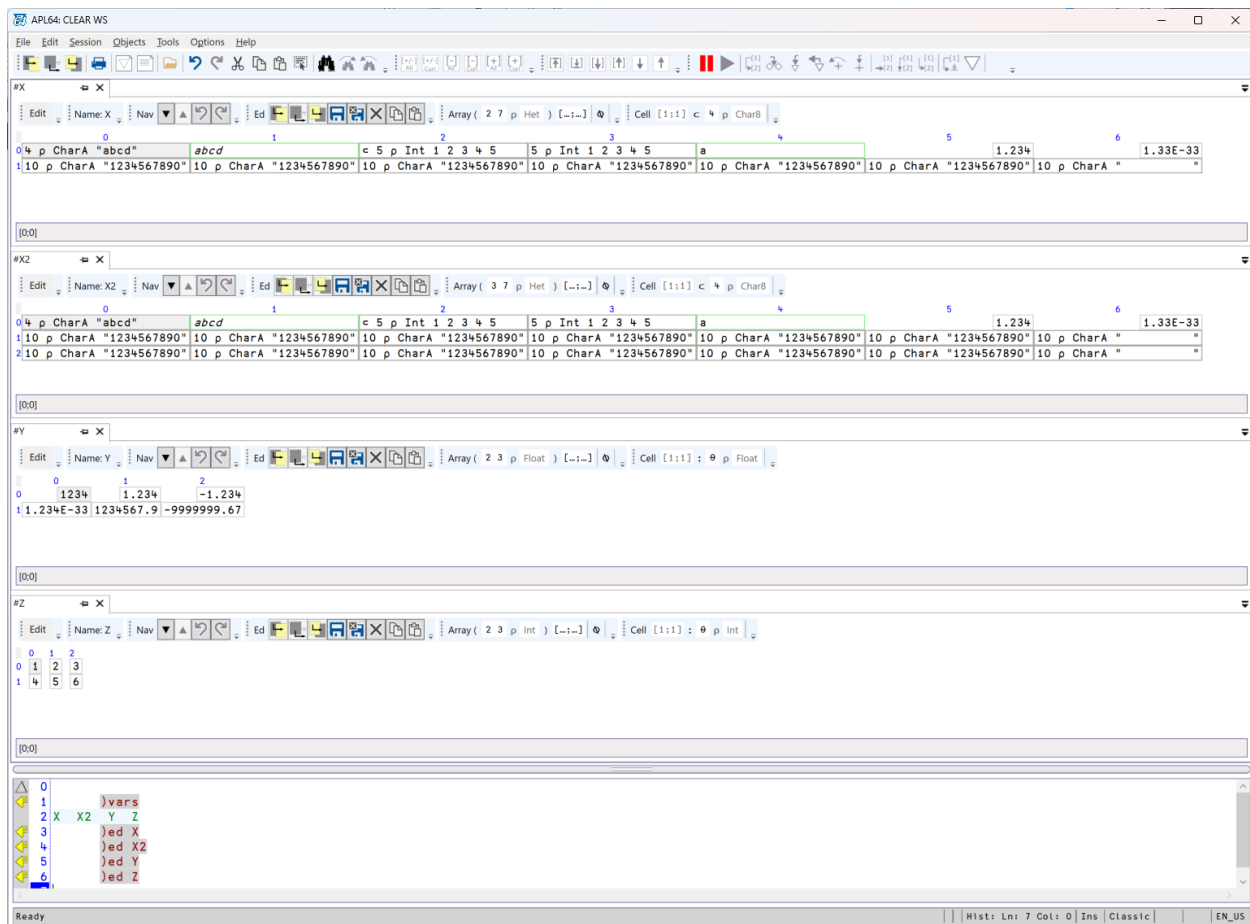


Option to left align numeric cell values

When the **Objects | Variable Editor | Align Cell Values** menu item is unchecked, the default, numeric values will be left aligned and right aligned when checked. Note: The setting is applied to any variable editors that are opened after the change.

Objects | Variable Editor | Align Cell Values: Not Checked





User Documentation

New Menu Item: [Help | Developer Version GUI | History and Edit Session Keyboard Shortcuts](#)
Documents the keyboard shortcut available in the History and Edit Sessions.

[Help | Documentation Options ... | Download dialog](#)

The 'Select Non-Existing' files button will not remain depressed during the file download process

Using ☐ SQLite

The documentation includes the new section, *Get Record Set Column Titles*, that provides information and examples for retrieving record set column titles in a table. Refer to **Help | APL Language | SQL System Functions | Using ☐ SQLite** for the details.

The default string data line continuation background color is now white

When creating a string scalar using «« ... »» or « ... », the line continuation character default background color is now white instead of blue.

[Replace All Tool](#) performance is improved for most scenarios

[Virtual Keyboard: Rendering of descender glyphs improved](#)

The rendering of the rho (ρ), epsilon underscore (ϵ), delta underscore (Δ), and lower case (g,p,q) is improved in the virtual keyboard (**Session | APL Virtual Keyboard**).

Before

Click on a quadrant of a virtual key to enter its corresponding glyph. Click on a modifier key button (Base, Shift, Alt or Alt+Shift) to highlight the associated glyphs. AltGr glyphs are controlled by the user-selected Windows Language Preference and are not illustrated in the APL64 virtual keyboard. Current Keyboard Definition: EN-US

Now

Click on a quadrant of a virtual key to enter its corresponding glyph. Click on a modifier key button (Base, Shift, Alt or Alt+Shift) to highlight the associated glyphs. AltGr glyphs are controlled by the user-selected Windows Language Preference and are not illustrated in the APL64 virtual keyboard. Current Keyboard Definition: EN-US

Printing from the APL 64 Developer Version

Row numbers for history and variable editor

When printing text in the history pane or variable editor, the output will now include or exclude row numbers depending on the user selection:

Session | History Row Numbers | Display Row Numbers in History

Objects | Variable Editor | Row Numbers | Display Row Numbers

Printing from a floating variable editor pane is now possible

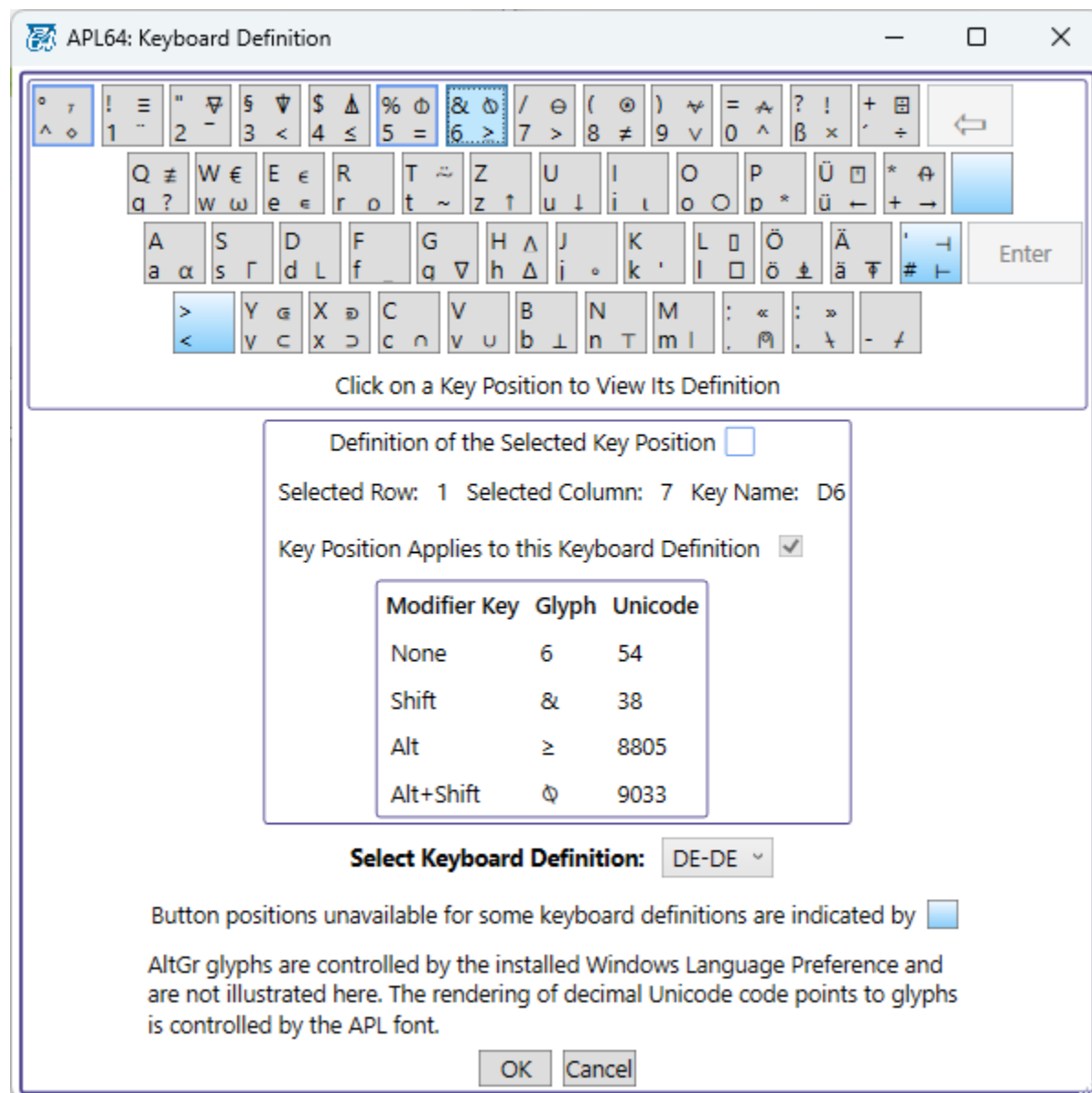
User Tool with a Ctrl+T shortcut will now override the default Ctrl+T action

The default Ctrl+T shortcut action toggles the keyboard focus between the current editor pane and the history pane. If a user tool specifies a Ctrl+T shortcut, that the user tool's shortcut action will override the default Ctrl+T action.

Objects | Editor Autosave Options: Selected autosaved object definition

The selected autosaved object definition will now be properly displayed in this dialog.

The transpose glyph (⌘) is now available on the German (DE-DE) Keyboard Definition 6-key



The caret offset in History Pane changed to left margin from 6-space indent

This occurred after using **Open Editor using the Objects | Recent Objects Name List** menu to edit an object then using Active Panes via the Ctrl+Tab shortcut to toggle to the History.