

APL64 2026.1.5 Update

Table of Contents

Overview	3
Interpreter Improvements.....	3
New ☐ HTLC and ☐ HTLS: APL64 as Client and Server	3
New ☐ WSC system function: Save/Get Workspace Functions to/from Text file	3
New ☐ JS: Use JavaScript in an APL64 function.....	3
New Example: Use APL64 in html/JavaScript Web Pages.....	3
New ☐ PY: Use Python in an APL64 function	3
New ☐ CRC system function: Calculate CRC-32 value of an APL array.....	3
New ☐ DEMOTE system function: Demote APL array	3
Improved Performance: ☐ SS, ☐ WORDREPL, ☐ TEXTREPL, Index Assignment.....	4
Improved ☐ EDSS and ☐ EDSS.....	4
Passive processing by ☐ LJUST, ☐ RJUST and ☐ SSTOMAT system functions.....	4
Improved ☐ SQLx System Functions.....	4
Improved ☐ MATRIFY now supports a singleton character argument.....	4
New ☐ CSE Methods.....	4
LoadNuggetPackage.....	4
InitErr	4
Improved ☐ CSE Error Messages	4
Improved ☐ WSE 'SetSysVar' action.....	5
Improved Control Statement :ENDTRYALL immediately followed by :IF	5
Improved ☐ ROWFIND compatibility with APL+Win ROWFIND assembler function	5
Improved ☐ DR Supports empty text	5
Improved Dyadic Iota.....	5
Improved ☐ NA.....	5
Improved ☐ EVAL	5
Improved ☐ CPC.....	5
Stops in a multi-line string	5
Saving an APL Workspace will be completed before the APL instance ends	5
APL64 Developer Version GUI Modifications	5

Improved Options Colors Dialog	5
Improved APL Font Support: APL387 Font is installed with APL64 Developer version	7
Improved Debug Pane: Current suspended stop is now the last displayed line	7
Improved Function Editor: Remember the Collapsed Region State for a Function	7
Improved Printing in the APL654 Developer Version	7
Improved WRE and CPC Creation Utility.....	8
Improved]OUT User Command.....	8
Improved APL64 Developer Version Installer	8
Improved Floating Function/Variable Editor with Icon in the Taskbar.....	8
Objects Editors Pane Format Layout > Editors AutoHidden Option Removed	9
Documentation	9
APL64 Windows Runtime Executable	9
Additional Important Information	9
Important issues when using ☐ CSE	9

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

New [□HTLC](#) and [□HTLS](#): APL64 as Client and Server

The [□HTLC](#) and [□HTLS](#) system functions support HTTP communication between different instances of APL64, with the added capability to debug the server-side APL64 instance in a visible window.

[□HTLC](#) and [□HTLS](#) can be used as a cross-platform replacement for the Win32-only APL+Win ActiveX Engine.

For further details, refer to the **Help | Developer Version GUI | Using [□HTLC](#) and [□HTLS](#)** menu item.

New [□WSC](#) system function: Save/Get Workspace Functions to/from Text file

The [□WSC](#) system function can be used to prepare a plain-text representation of the function content of an APL64 workspace. This representation can be used in a code repository such as GitHub. For further details, refer to the **Help | APL Language | System Functions** menu item.

New [□JS](#): Use JavaScript in an APL64 function

The APL64 [□JS](#) system function is an interface to the JavaScript programming language. For further details, refer to the **Help | APL Language | Using [□JS](#)** menu item.

New Example: Use APL64 in html/JavaScript Web Pages

For an example that illustrates how APL4 public functions exposed as .Net methods in an APL64 cross-platform component can be used by html/JavaScript web pages, please refer to the **Help | Developer Version GUI | Cross-Platform Component | Cross-platform Component in JavaScript** menu item.

New [□PY](#): Use Python in an APL64 function

The [□PY](#) system function is an interface to the Python programming language. For further details, refer to the **Help | APL Language | Using [□PY](#)** menu item.

For an example creating the APL64 Cross-platform Component (CPC) using a Python programming language, please refer to the **Help | Developer Version GUI | Cross-Platform Component | Cross-platform Component in Python Program** menu item.

New [□CRC](#) system function: Calculate CRC-32 value of an APL array

This system function computes the CRC-32 of the array. For further details, refer to the **Help | APL Language | System Functions** menu item.

New [□DEMOTE](#) system function: Demote APL array

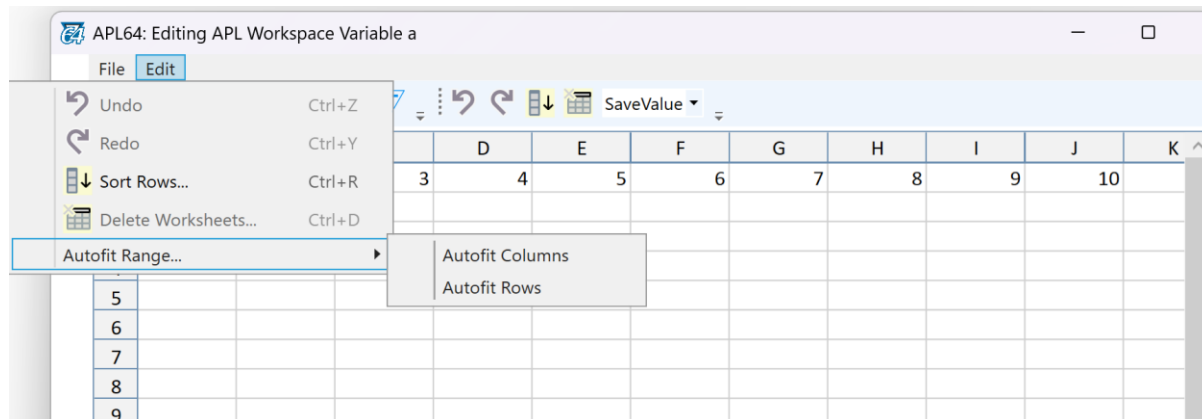
This system function demotes any array to its most compact representation. For further details, refer to the **Help | APL Language | System Functions** menu item.

Improved Performance: [□SS](#), [□WORDREPL](#), [□TEXTREPL](#), Index Assignment

These APL64 interpreter features have improved performance.

Improved [□EDSS](#) and [\)EDSS](#)

- New Autofit Columns and Autofit Rows actions implemented.
- Printing of wide columns in an EDSS editor is improved.



Passive processing by [□LJUST](#), [□RJUST](#) and [□SSTOMAT](#) system functions

These functions don't return a DOMAIN ERROR exception for right arguments that don't apply to these functions. Instead, appropriate right arguments are processed normally, and inappropriate right arguments are passed through without modification.

Improved [□SQLx](#) System Functions

- The new GetRecords action will return a matrix of the requested number of records from the record set, up to the number of remaining records which have not already been read from the record set.
- A new example shows how to write an SQL Delete or Select query with multiple lines and include comments. For further details, refer to the **Help | APL Language | SQL System Functions | Sql Delete and Select Scripts** menu item.

Improved [□MATRIFY](#) now supports a singleton character argument

For example, [□MATRIFY](#) 'a' is now supported.

New [□CSE](#) Methods

[LoadNugetPackage](#)

New LoadNugetPackage method will load a specified .Net Nuget package and its dependencies. For further details, refer to the **Help | APL Language | [□CSE](#) | C# Script Engine Manual** menu item.

[InitErr](#)

The Init and InitErr methods create a CSE instance. The key difference is when the methods aren't successful. If unsuccessful, the CSE 'Init' method returns a non-zero error code, whereas the CSE 'InitErr' method returns an explicit exception message.

Improved [□CSE](#) Error Messages

The Stack Trace information is now suppressed in CSE error messages. The Stack Trace information applies only to the APL64 development team.

Improved ☐WSE 'SetSysVar' action

The SetSysVar action for any system variable will return a domain exception when the programmer-provided value is out of range.

Improved Control Statement :ENDTRYALL immediately followed by :IF

An OUTER SYNTAX ERROR is avoided when the :if statement immediately followed the :endtryall statement in the function editor

Improved ☐ROWFIND compatibility with APL+Win ROWFIND assembler function

(3 2p'aa cc') ☐rowfind 1 0p' ' now returns 2 instead of 0

Improved ☐DR Supports empty text

A system error is avoided when 'inflate' ☐DR '' is executed.

Improved Dyadic Iota

A potential Rank Error is avoided for lArg ι**1.

Improved ☐NA

☐NA functions saved in APL+Win workspaces will now run in APL64.

Improved ☐EVAL

☐EVAL '2147490000 (2147490000)' result has been corrected

Improved ☐CPC

- ☐DR supported in CPC
- ☐CSE supported in CPC
- Create component files with user number of 1 instead of 0 in the file access matrix.
- PropValue action 'APPDESCSAMEASFILEDESC' property was always being set to 1.

Stops in a multi-line string

Function stops will now be effective only if they are set on the beginning line of a multi-line string, e.g. S←«...» AND s←««...»». This modification prevents a potential corrupted saved workspace.

Saving an APL Workspace will be completed before the APL instance ends

When a workspace is saved using ☐save,)save, etc., the APL instance cannot end until the workspace save process is completed.

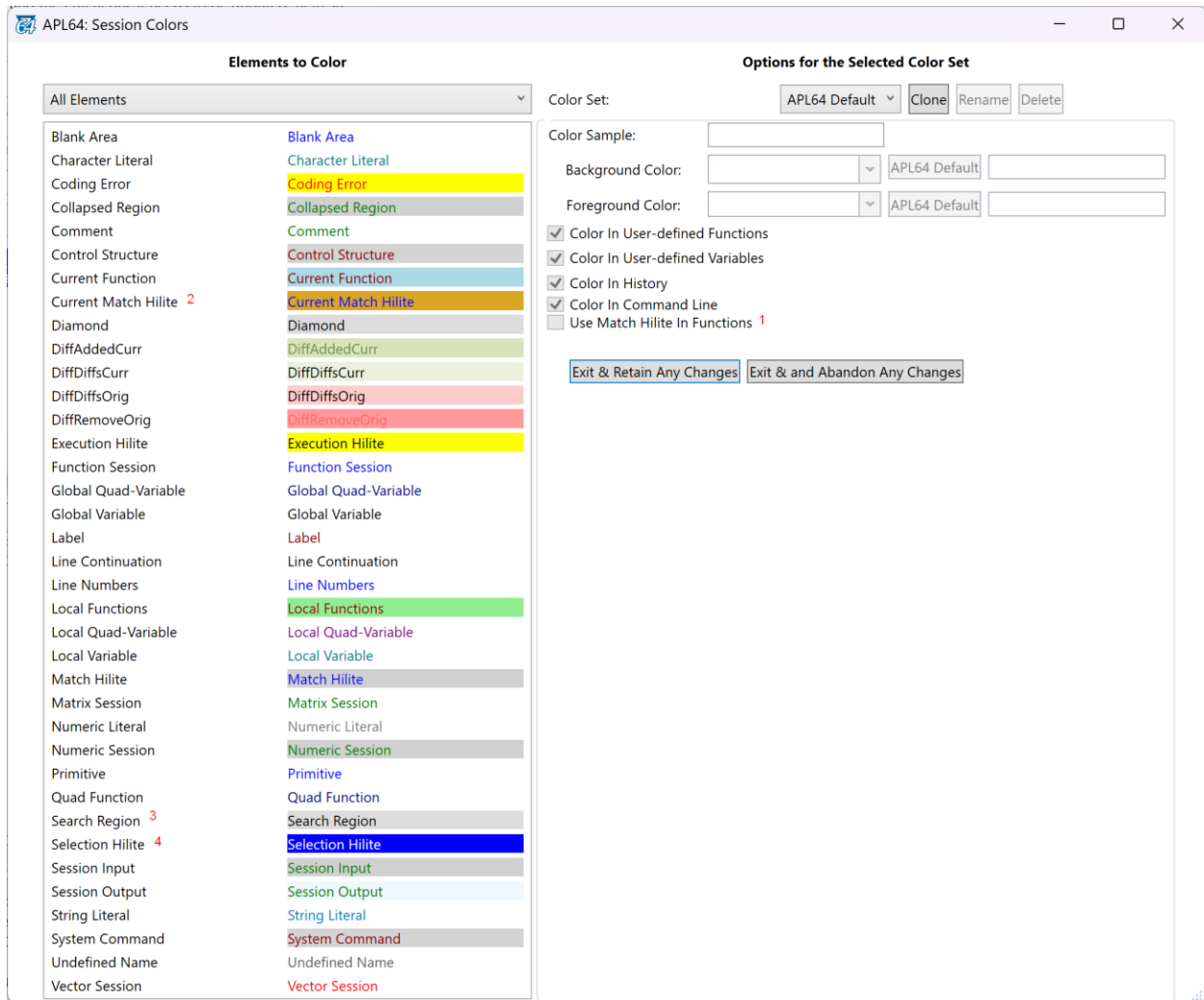
APL64 Developer Version GUI Modifications

Improved Options | Colors Dialog

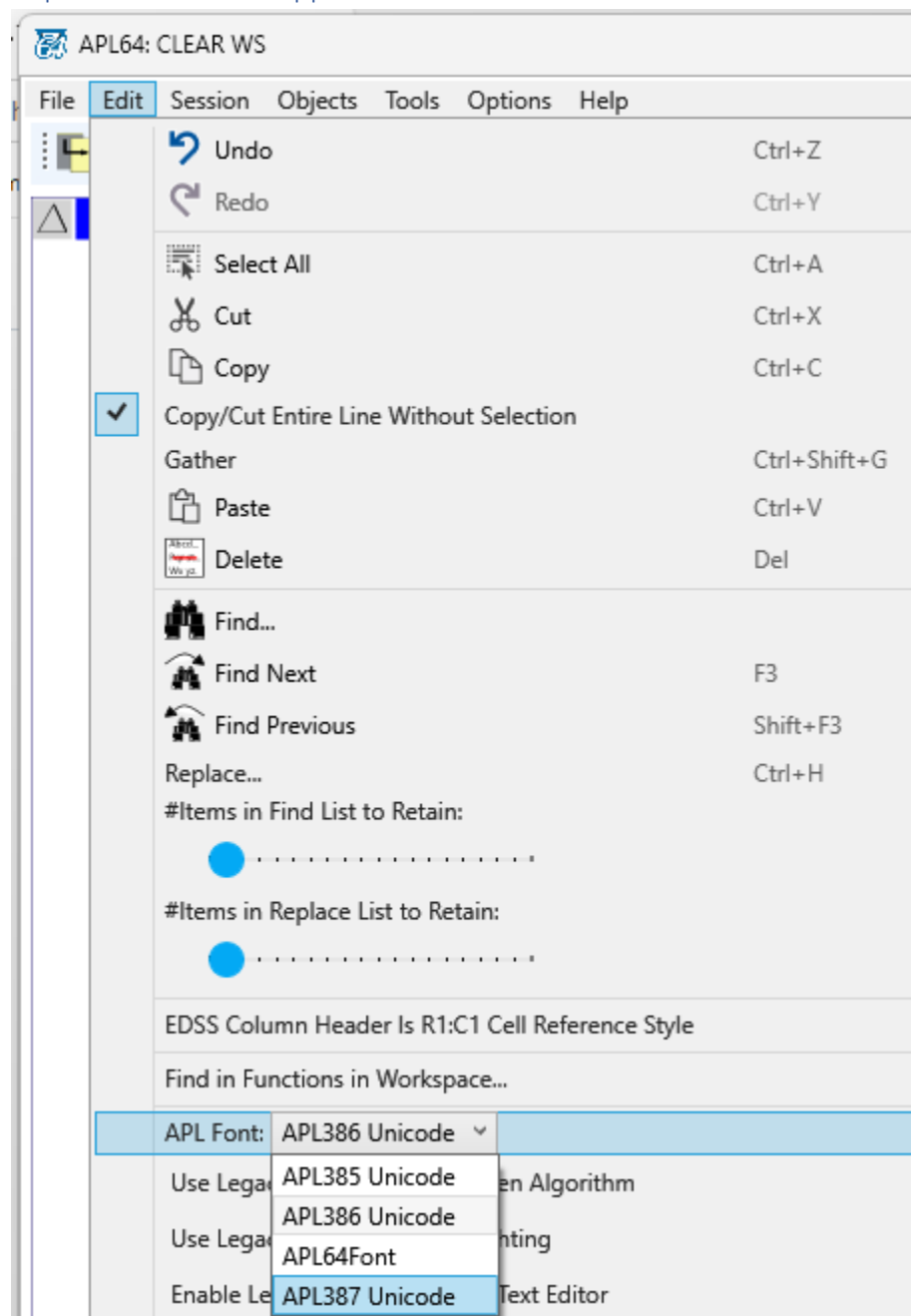
- The Elements to Color area is taller to view the entire list of elements.
- The checkbox 'Use Match Hilite in Functions' has been added to the Color Set section in the Session Colors tool. This parameter controls the syntax highlighting of matching elements in the function

editor. The default value is false. When this setting is enabled, the Match Hilite color element will be in effect. Note: The behavior is now the same as in APL+Win.

- The 'Current Match Hilite' color element is used to color text matching the text in the Find textbox of the Search/Replace window.
- The 'Search Region' color element was improved to use the selected region in the search scope.
- The new 'Selection Hilite' color element sets the background and foreground colors of user-selected text in the Find textbox of the Search/Replace window.



Improved APL Font Support: APL387 Font is installed with APL64 Developer version



Improved Debug Pane: Current suspended stop is now the last displayed line

A stop set on a line is now the last visible line (at bottom) displayed in the Debug pane.

Improved Function Editor: Remember the Collapsed Region State for a Function

A region will remain in the closed or expended state when opened the next time in the function editor.

Improved Printing in the APL654 Developer Version

- A potential exception is avoided when printing a function saved in the APL64 January 2026 version.

- Improved Printing of the History Pane and Variable Editor when the number of rows of text is greater than 9999. The closing square bracket in the row number column number is now on the same row as the row number.

Improved WRE and CPC Creation Utility

- New Excluded Components option in the WRE and CPC creation utility. Selecting one or more of these options will reduce the size of the resulting WRE executable or cross-platform component by the amount displayed for each exclusion option. Exclude a component when the WRE application or cross-platform component will not use the excluded component. For further details, refer to the **Help | Developer Version GUI | Create Windows Runtime Executable (WRE)** and **Help | Developer Version GUI | Cross-platform Component (CPC) | Create Cross-platform Component Creation Utility** menu items.

Excluded components:

☐ ☐ XL (5.769MB)	☐ ☐ CSE (2.263MB)
☐ ☐ RNet (0.194MB)	
☐ ☐ SQLx/ ☐ ODBC (103.369MB)	

- New Nuget Packages option in the WRE and CPC creation utility. Use this optional entry field to add Nuget packages to the WRE and cross-platform component project file. If present, this Nuget package will be included in the WRE or CPCP .Net project file. For further details, refer to the **Help | Developer Version GUI | Create Windows Runtime Executable (WRE)** and **Help | Developer Version GUI | Cross-platform Component (CPC) | Create Cross-platform Component Creation Utility** menu items.

NuGet Packages:

- The typo “ujcommitted” is corrected in the WRE creation message text in the WRE creation utility.
- Resolved the issue responsible for the error: “CSC : error CS5001: The program does not contain a static “Main” method suitable as an entry point...” when creating a WRE Output Type of "Single EXE file including All Dependencies".
- The APL64 xml-format Configuration File: option in the CPC creation utility is not applicable. An APL64 CPC is a .Net library assembly and not a .Net application. A CPC is designed to be consumed (used) by a .Net application in Windows, Linux or Mac OS environments. Therefore, there was no need for the APL64 XML-format configuration file used by an APL64 CPC. Similarly, there are no command line arguments supported by an APL64 CPC.

Improved JOUT User Command

The Jout user command now supports empty elements.

Improved APL64 Developer Version Installer

APL64 has been upgraded to .Net 10 SDK version 10.0.300 which must be installed when APL64 is installed.

Improved Floating Function/Variable Editor with Icon in the Taskbar

Changing the object’s name is now in sync with the name appearing in the editor’s icon in the Windows taskbar when the editor pane is floated.

This option was removed so that the object's name is synchronized with its floating editor pane name in the Windows taskbar.

Documentation

[APL64 Windows Runtime Executable](#)

The type "WreOutputType" updated to WreOutType in the ☐WRE 'PropValue' action.

Additional Important Information

[Important issues when using ☐CSE](#)

1. Identify dependencies (additional .Net assemblies) of the primary .Net assembly which is being used. All dependencies must also be loaded in the ☐CSE instance
2. Identify the .Net version each .Net assembly is targeting compared to the required .Net version of ☐CSE (2⇒☐sysinit)
3. Consider the .Net assembly location: Loading an assembly in ☐CSE can be done by:
 - LoadAssemblyBy name which initiates a search locally and eventually the global assembly cache on the workstation
 - LoadAssembly which requires a correct relative or full path to the .Net assembly