User Interface - Feature #12078

reproduce OpenEdge behaviour for SYSTEM-HELP, the HELP key and CONTEXT-HELP

10/23/2026 04:34 PM - Greg Shah

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
billable:	No	**reviewer:**	
vendor_id:	GCD	**production:**	No
case_num:		**env_name:**	
version_reported:		**topics:**	
version_resolved:			

Description

Related issues:

Related to User Interface - Feature #12036: implement conversion support for ...	**New**

History

#1 - 10/23/2026 04:35 PM - Greg Shah

Reproduce OpenEdge behaviour for SYSTEM-HELP, the HELP key and CONTEXT-HELP

Project: User Interface

Problem

#12036 makes CONTEXT-HELP, CONTEXT-HELP-FILE and CONTEXT-HELP-ID convert, with runtime stubs: the attributes store and return their values, no help button is shown, and no help engine is called. This task replaces those stubs, and the current SYSTEM-HELP runtime, with OpenEdge's behaviour. It also covers the window form of CONTEXT-HELP, which #12036 does not.

In a Windows GUI session OpenEdge does not display help itself. It hands each help request to Windows, and what the user sees is decided by Windows: on Windows 8 and later, which ship only a stub winhlp32.exe, a "Windows Help and Support" window opens whether or not the help file exists. FWD currently does something different: SYSTEM-HELP opens <web server>/app/help (directory setting system-help/context) in the browser, with ?contextid=n for the CONTEXT form, and ignores the help file name.

Every behaviour is implemented only after a test has shown what OpenEdge does, and every test runs on both OpenEdge and FWD.

Known OpenEdge Behaviour

Proven on OpenEdge 11.6 64-bit GUI on Windows Server 2012:

- SYSTEM-HELP <file> CONTENTS never raises an error (ERROR-STATUS:ERROR = no, no messages) and returns at once, whether or not the file exists. Windows opens its help window.
- F1 with no ON HELP trigger runs applhelp.p from the PROPATH, with focus still on the widget. If no applhelp.p is found, OpenEdge shows "No application help is available" in the message area (its own applhelp.p, which also rings the bell).
- A dialog frame with CONTEXT-HELP has a "?" button in its title bar. "?" followed by a click on a field, and F1 in the dialog, both send a help request to Windows, with or without CONTEXT-HELP-FILE. F1 in such a dialog does **not** run applhelp.p.
- frame:CONTEXT-HELP returns yes; frame:CONTEXT-HELP-FILE returns the given string unchanged, even when it is not a file name, or unknown when none was given; SESSION:CONTEXT-HELP-FILE is unknown.

Work

1. **Window CONTEXT-HELP** (left out of #12036): convert CONTEXT-HELP in CREATE WINDOW ... ASSIGN and as a window attribute, and store it as for frames.
2. **The help request.** Define one client-side help request that carries what OpenEdge passes to Windows: the file name as given, the help command (contents, context id, key, and so on) and its argument. SYSTEM-HELP, F1 in a CONTEXT-HELP dialog and "What's This?" all go through it.

    1. A Swing client on Windows makes the same Windows help call with the same arguments, so Windows produces the same result.
    2. The web client, and a Swing client on another platform, cannot make that call. Decide its equivalent once the arguments are known; the existing system-help/context HTML help can be the target, extended to receive the file name and command.
    3. A directory setting turns the help request off. With it off, every help request does nothing and returns, as delivered by #12036; this is the default, so existing deployments do not change on upgrade.
3. **SYSTEM-HELP:** no error, no message, immediate return, and the request made the way OpenEdge makes it. CONTENTS is proven; the other options use the same path, and each one's arguments are verified on OpenEdge before it is changed.
4. **HELP key on ordinary screens:** FWD already runs applhelp.p from the PROPATH and falls back to the default message. Verify and fix the PROPATH search order, the bell, the state seen inside applhelp.p (LASTKEY, FOCUS, SELF, PROGRAM-NAME), the return of focus afterwards, the precedence of ON HELP triggers, and dialogs without CONTEXT-HELP.
5. **CONTEXT-HELP dialogs and windows:** the title-bar "?" button and its position, the "What's This?" mode (cursor, what each kind of click does, cancelling with ESC), F1 going straight to the help request without applhelp.p, whether a 4GL HELP event fires, and the CONTEXT-HELP-ID value sent with the request.
6. **Gap marking:** update rules/gaps/ to Full / Full for each item as it is completed.

Testcases

Saved as a repeatable set in the Testcases project. 4GL probes with ABLUnit assertions cover everything observable inside the session (ERROR-STATUS, whether and when applhelp.p ran and with what focus, attribute values, trigger order), and run unchanged on OpenEdge and FWD. Sikuli covers what is only visible on screen (the help window, the message-area text, the "?" button, the "What's This?" cursor), against both prowin.exe and FWD, with few, coarse checks. Lyra covers FWD-only regression once the behaviour is fixed.

"Proven" means shown on OpenEdge 11.6 as above; "to write" means the test must be written and run on OpenEdge before the behaviour is implemented.

SYSTEM-HELP

ID	Test	Status
SH-01	CHOOSE of a button runs SYSTEM-HELP <missing file> CONTENTS: no error, no messages, immediate return, help window opens	proven
SH-02	The same with an existing .hlp file: identical result	proven
SH-03	The same trigger reached by APPLY "CHOOSE" from a key handler	to write
SH-04	Keyboard activation of the button: its mnemonic, and Tab to the button then Enter or Space	to write
SH-05	Focus and events after the call: which window is active, whether LEAVE / ENTRY / focus events fire, whether the screen stays responsive	to write
SH-06	The exact request OpenEdge makes (file name as passed, help command), for example by monitoring the Windows help call or with the WinHelp viewer installed	to write
SH-07	A second request while the help window is still open: a new window or the same one	to write
SH-08	A converted program with a help button, end to end	to write
SH-09	SH-01, SH-02 and SH-06 repeated on Windows 11	to write

HELP key

ID	Test	Status
HK-01	F1 in a fill-in, no trigger, no applhelp.p on the PROPATH: "No application help is available"	proven
HK-02	The same with an applhelp.p that only returns: nothing visible	proven
HK-03	applhelp.p runs on F1, with focus on the fill-in	proven
HK-04	Which applhelp OpenEdge used in HK-01 (the one under the OpenEdge install, which rings the bell), and the PROPATH search order	to write

HK-05	F1 on a button, browse, editor, combo-box and toggle	to write
HK-06	F1 in a dialog **without** CONTEXT-HELP: is applhelp.p run?	to write
HK-07	F1 where an ON HELP trigger exists: the trigger runs and applhelp.p does not; RETURN NO-APPLY in the trigger	to write
HK-08	The state inside applhelp.p: LASTKEY, FOCUS, SELF, PROGRAM-NAME; F1 returns to the same field afterwards	to write

CONTEXT-HELP

ID	Test	Status
CH-01	A dialog with CONTEXT-HELP (with or without CONTEXT-HELP-FILE) has a "?" button in its title bar	proven
CH-02	"?" then click on a fill-in: help window opens	proven
CH-03	F1 in the dialog: help window opens	proven
CH-04	F1 in the dialog does not run applhelp.p	proven
CH-05	Attribute values: CONTEXT-HELP = yes; CONTEXT-HELP-FILE = the given string, or unknown; SESSION:CONTEXT-HELP-FILE unknown	proven
CH-06	"What's This?" mode: cursor shape; clicking a button, a label, empty frame space or outside the dialog; cancelling with ESC	to write
CH-07	Whether a 4GL HELP event fires (an ON HELP trigger in the dialog) for F1 and for "?" + click	to write
CH-08	F1 on buttons with CONTEXT-HELP-ID 0 in a window without CONTEXT-HELP	to write
CH-09	The exact request for "?" + click and for F1: which file (including a CONTEXT-HELP-FILE that is not a file name), which command, which context id	to write
CH-10	The "?" button's position and the title bar in dialogs of different sizes and fonts, with and without CANCEL-BUTTON	to write
CH-11	CONTEXT-HELP on a window (CREATE WINDOW ... CONTEXT-HELP = yes) and its interaction with MIN-BUTTON / MAX-BUTTON	to write
CH-12	CH-02, CH-03 and CH-09 repeated on Windows 11	to write

Every test also runs with the help request turned off, and must then give the #12036 result.

Character-mode clients are out of scope: SYSTEM-HELP and CONTEXT-HELP are Windows GUI features. The same probes can confirm what _progres does if needed.

#2 - 10/23/2026 04:35 PM - Greg Shah

- Related to Feature #12036: implement conversion support for CONTEXT-HELP, CONTEXT-HELP-FILE and CONTEXT-HELP-ID added