User Interface - Feature #12045

replacement for the DBI ctSpin ActiveX control (spin button)

10/20/2026 05:46 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

History

#1 - 10/20/2026 05:46 PM - Greg Shah

Replacement for the DBI ctSpin ActiveX control (spin button)

Project: User Interface

Problem

ctSpin (ctSpin.ocx 3.0) is a bare up and down button pair. Applications handle its Click event, read the button argument, and increment or decrement a companion fill-in or a time editor. The control's own value, minimum and maximum are not used. Apart from LoadControls, only ENABLED is used.

FWD has no spin button widget.

Work

1. Implement a small spin button widget that emits Click with the direction argument the control passes, or convert the control into two image buttons with the same Click contract. Choose whichever reproduces the layout and keyboard behaviour better.
2. Register the control's CLSID in the OCX registry (ocx-registry.xml), mapping it to the target widget, as #11919 did for ctToolBar.
3. Decode the design-time state the control keeps in the .wrx storage at conversion time, so that it exists after conversion.
4. Implement the members and events listed above on the target, with the vendor's semantics: argument types, return values, defaults, and the order in which events fire. Members of the type library that are not used become no-ops with a conversion warning, as the OCX pipeline already does.
5. Follow the OCX emulation playbook attached to #11919: an OpenEdge test program with goldens for each member and event, recorded input sequences for the interactive parts, and Lyra automation.
6. Update the gap marking and the OCX registry documentation.

Testcases

Each member and event above, exercised on Windows OpenEdge with the real control to produce goldens, then run against the FWD replacement on the web and Swing clients. The vendor help files are the reference where behaviour cannot be observed directly.