User Interface - Feature #12058

replacement for the Gigasoft ProEssentials graph and pie chart ActiveX controls

10/20/2026 06:19 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 06:19 PM - Greg Shah

Replacement for the Gigasoft ProEssentials graph and pie chart ActiveX controls

Project: User Interface

Problem

Gigasoft ProEssentials v3 provides Pego32a.ocx (Graph, PEGOA) and Pepco32a.ocx (Pie Chart, PEPCOA). Applications draw bar and line graphs and pie charts of statistics, setting Subsets, Points, YData, SubsetLabels, PointLabels, titles, axis labels, PlottingMethod, GraphPlusTable, AllowZooming and AllowMaximization. Events: DataHotSpot, GraphHotSpot, PointHotSpot and DblClick. Many design-time properties are in the .wrx. v3 is long out of support, and a current version is not a drop-in replacement.

FWD has no chart widget. Its HTML-BROWSER widget hosts a live page with a postMessage bridge between the page and the converted code.

Work

1. Implement an HTML chart host: a replacement control built on HTML-BROWSER that renders the graph and pie types used with an established JavaScript charting library loaded into the page.
2. Map the property model above onto the library's configuration, sending it to the page through the bridge whenever the converted code changes a property.
3. Deliver the hot spot events (DataHotSpot, GraphHotSpot, PointHotSpot, DblClick) back through the bridge, with the subset and point indexes the control reports.
4. Register the control's CLSID in the OCX registry (ocx-registry.xml), mapping it to the target widget, as #11919 did for ctToolBar.
5. Decode the design-time state the control keeps in the .wrx storage at conversion time, so that it exists after conversion.
6. 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.
7. 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.
8. 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.