# User Interface - Feature #12060

## replacement for the rmpHTML (nBit HTML Editor) ActiveX control

10/20/2026 06:23 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 06:23 PM - Greg Shah**

# Replacement for the rmpHTML (nBit HTML Editor) ActiveX control

**Project:** User Interface

## Problem

nBit HTML Editor ActiveX 3.02 (rmpHTML.ocx, type HTMLed, CLSID {B068AFB5-AC36-49F9-B6A4-06C0178602A6}) is a WYSIWYG HTML editor built on the IE6 MSHTML engine, end of life. Applications use it to edit stored HTML templates: DocumentHTML get and set, EditorEnabled, SelectText, InsertAtCursor, Insert_Image_NoFTP, Open_Document, Save_Document, Show_Code, CSSHttp and CSSText, toolbar visibility and custom toolbar buttons. Events: BeforeCodeViewChange and ToolButtonClick. The toolbar layout, CSS and flags are in the .wrx. The FTP client and the spell checker are not needed, and setLicenceKey becomes a no-op. The vendor's Developers Reference Guide matches the version and is the specification.

## Work

1. Implement an HTML editor widget, hosting an established JavaScript editor in HTML-BROWSER on the web client, with the API above.
2. Settle the data compatibility question first: the HTML already stored was produced by IE6 and a modern editor will not round-trip it identically. Measure the differences on real stored documents and decide what normalisation is acceptable.
3. Register the control's CLSID in the OCX registry (ocx-registry.xml), mapping it to the target widget, as #11919 did for ctToolBar.
4. Decode the design-time state the control keeps in the .wrx storage at conversion time, so that it exists after conversion.
5. 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.
6. 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.
7. 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.