

The Microsoft[®] ActiveX[™] Control Pad

Whitepaper

Microsoft Corporation
June 3, 1996

Table of Contents

INTRODUCTION	
ACTIVEX CONTROLS	
USING THE ACTIVEX CONTROL PAD	
THE TEXT EDITOR.....	
THE OBJECT EDITOR.....	
THE SCRIPT WIZARD.....	
AUTHORING FOR 2D LAYOUT: ADDING ADVANCED LAYOUT FEATURES.....	
<i>Features of the ActiveX Control Pad 2D Page Editor</i>	
<i>How is the HTML Layout Control Distributed?</i>	
<i>Compatibility and Future Support</i>	
DISTRIBUTION AND LICENSING OF THE ACTIVEX CONTROL PAD	
CONCLUSION	
FOR MORE INFORMATION	

Introduction

The Microsoft® ActiveX™ Control Pad is a utility that makes it easier to author Web pages that incorporate leading-edge, ActiveX content. Specifically, the ActiveX Control Pad makes it easy to add ActiveX Controls and ActiveX Scripting (Visual Basic® Script or JavaScript™) to HTML pages for viewing in the Microsoft Internet Explorer 3.0 or other Web browser supporting ActiveX Controls. The ActiveX Control Pad also includes WYSIWYG support for authoring 2D layout regions in conjunction with the Microsoft HTML Layout Control (please see *The Microsoft HTML Layout Control Whitepaper* for more information on 2D-style authoring using the HTML Layout Control).

The ActiveX Control Pad consists of the following core components:

- A Text Editor for editing HTML documents.
- An Object Editor for placing ActiveX Controls directly into an HTML document, and for visually setting properties on ActiveX Controls.
- A Script Editor for VB Script or JavaScript generation on actions and events.
- A WYSIWYG Page Editor for creating fixed, two-dimensional (2D) layout regions within an HTML document.
- A palette of ActiveX Controls that can be incorporated into Web pages.

ActiveX Controls

ActiveX Controls are interactive objects, created by programmers, that can be embedded in Web pages to enhance the experience of a Web site. For example, an ActiveX video control could be used to enhance a Web page with real-time video sequences.

The controls are language-independent, and can be programmed using programming languages such as C++, future versions of the Microsoft Visual Basic® programming system, or Java. Over 1,000 ActiveX Controls are available today from a wide variety of software vendors.

While the ActiveX Control Pad includes a number of Microsoft-supplied ActiveX Controls, the toolbox can also be customized with third party ActiveX Controls, such as the Macromedia® Shockwave for Director Control, the Adobe™ Acrobat™ Control or any other available ActiveX Control.

Using the ActiveX Control Pad

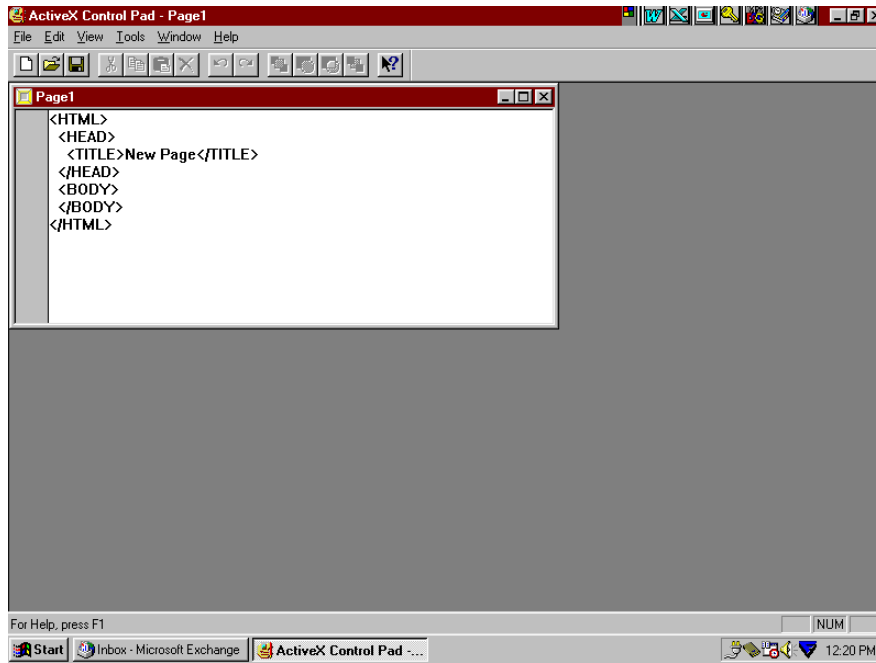
The ActiveX Control Pad is based on a document authoring metaphor, using document types of .htm, .html and .alx. Users create, open, edit and save these document types within the ActiveX Control Pad. The ActiveX Control Pad uses a multiple document interface so that authors can open and edit multiple documents within a single running instance of the application. The main starting window displays an HTML document, in text form, and includes a navigation bar along the left side which contains buttons that launch specific editors for adding the following features to the HTML document:

- ActiveX Controls
- ActiveX Scripts

- 2D Layout Regions

The Text Editor

Creating a new HTML file will open a window with HTML text which appears as follows:



The author can edit the raw HTML (which may have been created in any WYSIWYG HTML editor, such as the Internet Assistant for Microsoft Word, or Microsoft FrontPage™) within the text window. The text editor includes basic editing functions, including drag/drop of selected text blocks within the text document, multiple level undo/redo, cut, copy, paste, and basic file manipulation operations.

The Object Editor

Without a tool to aide in the insertion of ActiveX Controls into an HTML document, users must manually enter the HTML <OBJECT> syntax for each control, which includes:

- Locating and copying CLSIDs for selected ActiveX controls from the registry
- Adding syntax for <PARAM *property name* VALUE *property value*> to set control properties

Manually entering this data in a text editor can be time consuming and difficult, especially when one considers that each ActiveX Control has its own 128-bit unique identifier (CLSID) that must be entered. In addition, with no visual metaphor for setting control properties, the process is even more difficult. Listed below is a typical entry for a single ActiveX Control within a blank HTML document:

```
<HTML>
<BODY bgcolor="#804000">
<BGSOUND SRC="file:c:\winston\sounds\bbtrans.mid">

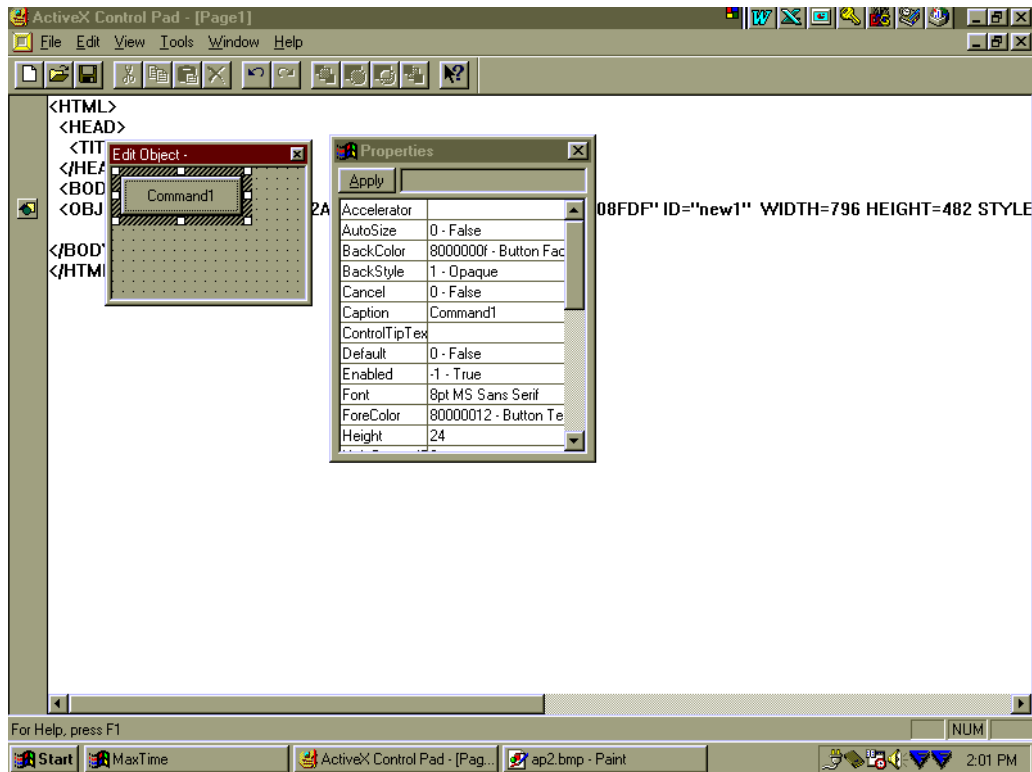
<CENTER>
<OBJECT CLASSID="CLSID:812AE312-8B8E-11CF-93C8-00AA00C08FDF" ID="winclub"
WIDTH=600 HEIGHT=380>
<PARAM NAME="ALXPath" VALUE="http://www.myserver.com/winclub.alx"> </OBJECT>
</CENTER>

</BODY>
</HTML>
```

The Object Editor makes inserting ActiveX Controls significantly easier. The Object Editor lets users insert and edit individual ActiveX objects in an HTML file using a simple visual metaphor similar to Visual Basic forms. The Object Editor includes a modal form and a property table for the control to be inserted. To invoke the Object Editor, the user simply positions the text input cursor at the point in the HTML stream that they wish to insert an ActiveX Control. The user then can choose to "Insert Object" from the "Edit" menu. The Object Editor will then appear with a dialog from which the user can choose a control from a list of those controls registered on the user's system.

Inserting an ActiveX Control into an HTML Document

To insert an ActiveX Control, the user simply selects the appropriate control from the control dialog. The control will appear on the Object Editor Form, and can be resized appropriately. The user visually sets properties on the control using the visible property table which can be invoked/dismissed by double clicking on the control. Many of the available controls also support WYSIWYG, in-place editing for adding text. Upon closing the Object Editor, the Control Pad will automatically generate the HTML <OBJECT> syntax in the appropriate location in the HTML document, including locating and copying CLSIDs for selected ActiveX controls from the registry and generating syntax for <PARAM *property name* VALUE *property value*> based on the property values visually set by the user.



Because HTML is a stream-based metaphor, the form-based Object Editor is only appropriate for adding a single control at a time. HTML as specified today cannot recognize exact positioning information for multiple controls within a document (in other words, HTML has no mechanism for specifying exact x, y and z coordinates for objects). Thus, adding multiple controls to a single instance of the Object Editor is not allowed—instead, the user must invoke a separate instance of the Object Editor for each Active Control to be added to an HTML document. These controls can then be positioned within the HTML stream to the extent possible using available HTML formatting options.

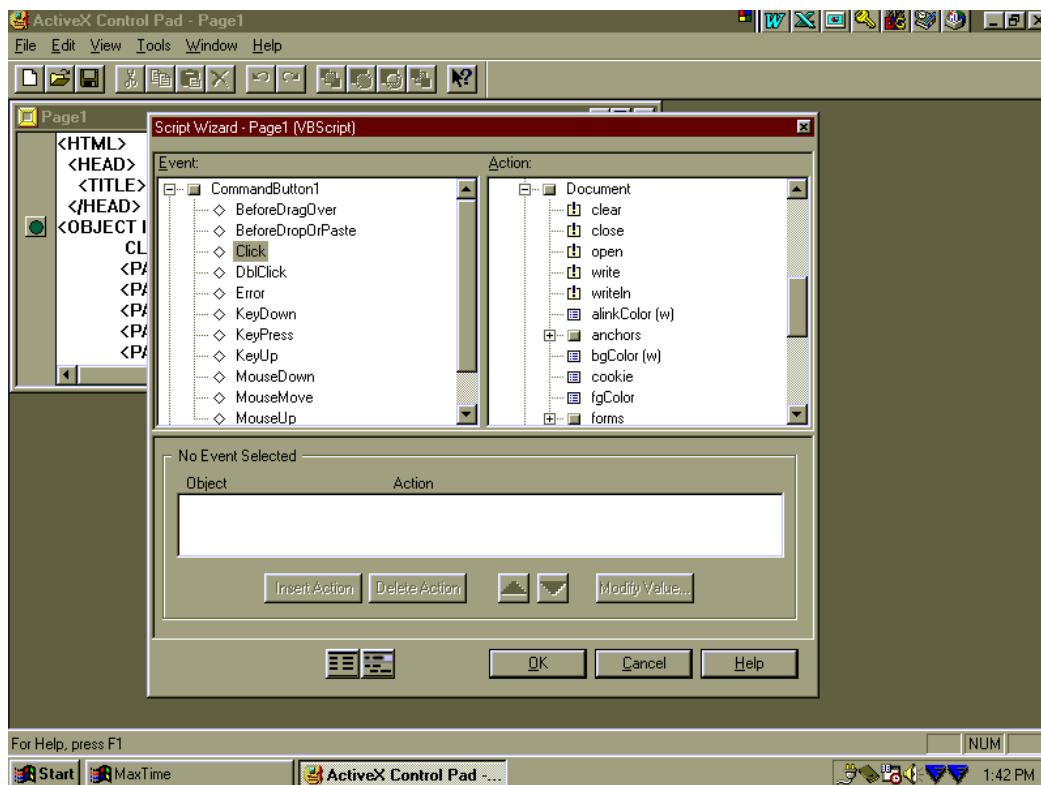
Microsoft has also provided an early implementation of exact, 2D-style layout for HTML, similar to Visual Basic forms and popular desktop publishing tools. This early implementation is based on an early draft specification by the W3C for a future version of HTML. Using 2D-style layout, authors and developers can use the ActiveX Control Pad to create forms that contain multiple controls, exactly placed within the Web browser (including x and y coordinate placement and z-ordering for object layering). When inserting a 2D region using the Control Pad, authors are allowed to place and position multiple ActiveX Controls on a form, to be displayed using the Microsoft HTML Layout Control within the Web browser. For information on 2D-style authoring with the ActiveX Control Pad, read the section entitled “Authoring for 2D Layout: Adding Advanced Layout Features.”

Tracking Active Controls within the HTML Document

To aide in tracking objects added to an HTML document, and to allow users to easily invoke the Object Editor on previously added objects in order to make changes, the text editor contains a vertical left pane that provides a visual icon for each ActiveX Control placed in the document. Upon pressing the icon from the text editor, the Object Editor will be invoked, displaying the Active Control for further editing and property-setting by the user.

The Script Wizard

The purpose of the Script Wizard is to make it easy to add interactivity to Web pages, based on actions and events associated with ActiveX Controls. The Script Wizard includes a user interface for easily connecting an event (such as a mouse click) with an action (such as playing a video clip). The interface is designed to be both approachable for designers and at the same time powerful enough for developers. The Script Wizard supports both VB Script and JavaScript. To invoke the Script Wizard, the user simply selects “Script Wizard” from the main “Tools” menu. The following window will appear:



In *list view*, the Script Wizard lists on the left-hand pane all of the events associated with a given object in the HTML document. By clicking an event in the left pane, the user can then select an associated action in the right hand pane without any programming. Multiple actions can be added in this manner to a single event. In *coder view*, actions are shown as actual lines of VB Scripting or JavaScript code. In code view, the user can use the text entry window to insert any number of lines of custom scripting. In both modes, the Script Wizard automatically adds the appropriate scripting information into the HTML document.

Authoring for 2D Layout: Adding Advanced Layout Features

The ActiveX Control Pad is the first tool that makes it possible to author two dimensional (2D) layout regions using new style sheet attributes specified in a W3C draft specification for HTML. Authoring 2D regions with the ActiveX Control Pad provides developers with a WYSIWYG form upon which they can place, and exactly position, any number of ActiveX Controls using true frame-based layout. Scripting within these 2D regions is also possible using the Script Wizard. This 2D authoring support makes it possible to build rich, interactive user interfaces for Web pages. In this early preview implementation of 2D authoring, 2D regions are displayed using a special runtime control, called the Microsoft HTML Layout Control. As the W3C finalizes the specification for 2D layout in HTML, support for browsing 2D regions will become native to the Microsoft Internet Explorer and other Web browsers. Please refer to the *HTML Layout Control Whitepaper* for more technical information on Microsoft's preview implementation of 2D-style layout for HTML.

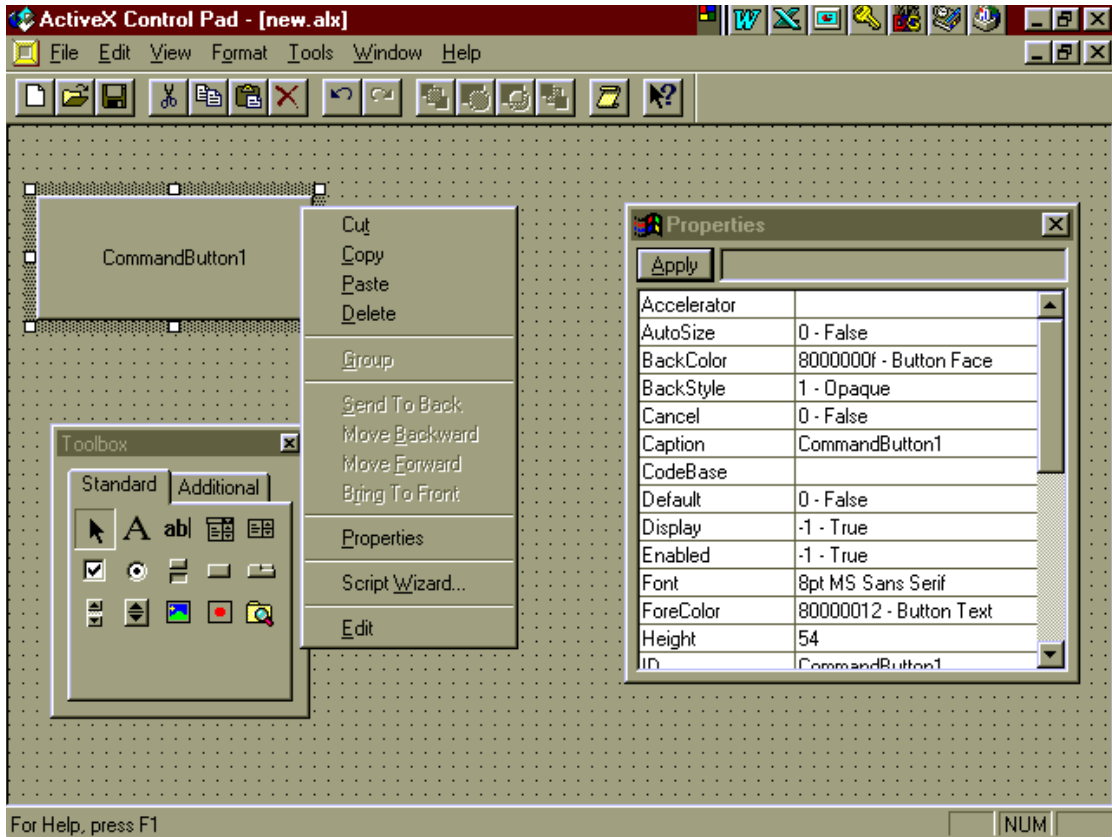
To insert a 2D region into an HTML document, users simply choose "Insert," "HTML Layout Control Region" from the "Edit" menu. This will cause an instance of the HTML Layout Control to be inserted into the HTML document based on the W3C <Object> tag syntax. Next, the user must identify a file that will contain the 2D layout information (in text form, as specified in the draft W3C specification). The HTML Layout Control will use this file to read and display the 2D layout region within the Web browser. The 2D region layout information is stored in a file that has a ".alx" extension.

Adding content to 2D regions is accomplished using ActiveX Controls. Each open .alx file has its own toolbox that presents a standard set of ActiveX Controls that allow the author to add text, graphics, user interface elements, and other multimedia objects to the region (see the table below for a list of these controls). The toolbox can be customized with other controls registered on the user's machine with a "Custom Control" function from the context menu on the toolbox. The user can simply right click on the toolbox to accomplish this customization.

The following ActiveX Controls are provided with the Microsoft ActiveX Control Pad, and the HTML Layout Control:

Controls on toolbar include:	
Option button	Textbox
Tab strip	Listbox
ScrollBar	Combobox
Spinner	Checkbox
Label	Command button
Image	Toggle button
Hot Spot	Web browser control (for displaying HTML documents)

Note that many of the controls support in-place editing.



A property table can be displayed for any selected control placed in the 2D region by right clicking on the control and choosing “Properties” from the context menu. Values for object properties are persisted as <PARAM> tags in the <OBJECT> tag within the .alx file. The illustration above shows an example property table displayed in the 2D Page Editor.

The object code generated within the .alx file for the Command Button illustrated above will look like:

```
<OBJECT ID="CommandButton1" STYLE="TOP:240;LEFT:160;WIDTH:176;HEIGHT:32;"
CLASSID="CLSID:978C9E23-D4B0-11CE-BF2D-00AA003F40D0">
  <PARAM NAME="Caption" VALUE="New Caption">
</OBJECT>
```

Each control hosted by the ActiveX Control Pad for inclusion in a 2D region inherits properties, methods and events from the HTML Layout Control. The tables below give these possible values:

Property	Description
ID	The identifier for the control - this is what is used to reference the control in scripting.
Width	Width in pixels of the object
Height	Height in pixels of the object
Top	Position from Top edge of object in 2D region (in pixels)
Left	Position from Left edge of object in 2D region (in pixels)

Visible	A Boolean property -> True = visible; False=hidden
TabStop	A Boolean property -> True = user can tab into
TabIndex	Position of object in tab order stack (integer)
CodeBase	string value which allows user to specify in the form of a URL where the code lives which embodies the ActiveX control.

Method	Description
Zorder(level)	(level=0,1) enumerated to mean (Top, Bottom)

Events	Description
AfterUpdate	Occurs after data in a control is changed through the user interface
BeforeUpdate	Occurs before data in a control is changed
Enter	Occurs before a control actually receives the focus from a control on the same form
Exit	Occurs immediately before a control loses the focus to another control on the same form

The HTML Layout Control also exposes a navigation object via a Window.Location.href object which takes the value of a URL as a string. When read, it returns the name of the HTML file (using the complete URL) which hosts the HTML Layout Control. When written (set), it changes the URL which the browser is pointing to, replacing the HTML page which is currently being viewed. This object can thus be used with the Script Wizard to create navigation functionality within a 2D region.

For an example of how 2D, frame-based layout can be used to create compelling, interactive designs, please refer to the samples included on the Microsoft ActiveX Control Pad Web site (<http://www.microsoft.com/intdev/author/cpad/>). These samples can be browsed using Microsoft Internet Explorer 3.0 after installing the ActiveX Control Pad or HTML Layout Control.

Features of the ActiveX Control Pad 2D Page Editor

The Page Editor included with the ActiveX Control Pad also has a number of other useful features for designing 2D regions. These include:

- **Toolbox control customization.** Right clicking on the toolbox allows users to add and delete individual controls.
- **Toolbox custom tabs.** New toolbox tabs can be added using the toolbox context menu.
- **Toolbox object templates.** Users can drag objects from the form back onto the toolbox, creating an object template. The template captures all of the properties set by the user, and can be used to easily incorporate objects with custom properties onto new pages. Groups of controls can also be used as object templates in this manner.

- **Multiple level undo/redo.** Edit operations can easily be undone.
- **Control Alignment/spacing/sizing.** Controls can be aligned and automatically spaced and sized for easier layout.
- **Control drag/drop.** Controls can be exactly positioned via drag/drop. Also, controls can be dragged from one page to another page.
- **Control z-ordering.** Z-ordering can be set by right clicking on any control.
- **Script Wizard.** The Script Wizard can easily be invoked from the menu to add interactivity to a page.

How is the HTML Layout Control Distributed?

The final release of the HTML Layout Control will be fully integrated into the final release of the Microsoft Internet Explorer 3.0 Web browser. The first HTML Layout Control beta release is being supplied as a free, downloadable control by Microsoft. For user to view 2D regions, they must have the HTML Layout Control installed on their computer, along with a browser such as the Microsoft Internet Explorer 3.0 that supports ActiveX Controls. The control can be freely re-distributed by any customer or software vendor. Once installed, the HTML Layout Control does not need to be re-installed as users navigate to subsequent pages that use 2D layouts.

Compatibility and Future Support

As the W3C finalizes its specification for 2D-style layout in HTML, Microsoft plans to incorporate support for 2D layout directly into future versions of the Microsoft Internet Explorer, as well as a variety of authoring and development tools. Once this support becomes native, the need for a separate .alx file and HTML Layout Control will be eliminated. Our early, preview implementation of 2D layout for HTML follows the current W3C draft specification, and we will continue to work closely with the W3C as they move forward with this specification. If required based on the final, adopted W3C specification for 2D-style layout, Microsoft plans to provide a conversion utility for the file format of the HTML Layout Control. This utility would convert Web pages created with the ActiveX Control Pad (or other editor) to be rendered in future browsers (like IE) that will support 2D layout with approved W3C standards. The utility will also integrate existing .alx files directly into the HTML stream. This conversion utility will be free to the public.

Distribution and Licensing of the ActiveX Control Pad

Microsoft intends to make the ActiveX Control Pad available to all users, free-of-charge. The ActiveX Control Pad will be available for download within the Microsoft Developer Toolbox Web site at <http://www.microsoft.com/intdev/author/cpad>. A beta release of the ActiveX Control Pad will be available in early June, 1996, followed by a final release several weeks later.

Conclusion

The ActiveX Control Pad is a tool that makes it easier to author Web pages that incorporate leading-edge, ActiveX content.

The ActiveX Control Pad consists of the following core components:

- A simple text editor for editing HTML documents.
- An Object Editor for placing ActiveX Controls directly into an HTML document, and for visually setting properties on ActiveX Controls.
- A Script Editor for Visual Basic Scripting or Java Script generation on actions and events.
- A WYSIWYG Page Editor for creating fixed, 2D-style layout regions within an HTML document in conjunction with the HTML Layout Control.
- A palette of ActiveX Controls that can be incorporated into Web pages.

The HTML Layout Control will be available from Microsoft for free download, and will enable a wide variety of Web designers and developers to create compelling Web pages with exciting, ActiveX content.

For More Information

For more information or to obtain the beta software with sample pages, visit the ActiveX Control Pad Web site at <http://www.microsoft.com/intdev/author/cpad/>.

The information contained in this document represents the current view of Microsoft Corporation on the issues discussed as of the date of publication. Because Microsoft must respond to changing market conditions, it should not be interpreted to be a commitment on the part of Microsoft, and Microsoft cannot guarantee the accuracy of any information presented after the date of publication.

This document is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.

© 1996 Microsoft Corporation. All rights reserved.

Microsoft and Visual Basic are registered trademarks and ActiveX and FrontPage are trademarks of Microsoft Corporation.

Adobe and Acrobat are trademarks of Adobe Systems, Inc. Macromedia is a registered trademark of Macromedia, Inc. JavaScript is a trademark of Sun Microsystems, Inc.