

The Microsoft[®] HTML Layout Control

Whitepaper

Microsoft Corporation
June 3, 1996

Table of Contents

INTRODUCTION.....	
DESIGNING 2D-STYLE PAGES USING THE HTML LAYOUT CONTROL.....	
HTML LAYOUT CONTROL - HOW DOES IT WORK?.....	
USING THE HTML LAYOUT CONTROL.....	
THE .ALX FILE.....	
HOW IS THE HTML LAYOUT CONTROL DISTRIBUTED?.....	
COMPATIBILITY AND FUTURE SUPPORT.....	
CONCLUSION.....	
FOR MORE INFORMATION.....	

Introduction

Until recently Web browsers have rendered HTML documents as single column scrolling windows. Authors have been limited in their ability to control the placement of features in their Web pages, which has limited their ability to create more sophisticated user interfaces for their Web sites. Such “2D layout” capabilities are standard features of most traditional tools used for the print media, such as Adobe™ Photoshop™, Quark Xpress®, or Microsoft® Publisher, as well as development tools such as the Microsoft Visual Basic® programming system. The advent of framesets for HTML, as supported in Web browsers such as the Microsoft Internet Explorer 3.0 and Netscape™ Navigator 2.0 has recently allowed authors to divide a single window into separate scrolling regions, each containing a single HTML document. However, framesets do not give authors precise, coordinate control over individual objects placed on a page; nor do they give authors the ability to overlap objects and frames.

Microsoft has been working closely with the Worldwide Web Consortium (W3C) in order to extend HTML capabilities to include full 2D layout capabilities. Recently, the W3C published a preliminary specification for incorporating 2D-style, fixed layout into the HTML standard by extending style sheets and framesets (see W3C Technical Report: *Frame-based layout via Style Sheets*, Bert Bos, David Raggett and Håkon W Lie, 5th April 1996 at <http://www.w3.org/pub/WWW/TR/WD-layout.html>)¹. Microsoft is providing a preliminary implementation of this specification for Web authors via a free, downloadable ActiveX™ Control, called the *HTML Layout Control*.

The HTML Layout Control allows publishers to create more compelling Web pages by providing exact, coordinate control over object layout, layering and transparency. Objects can be specifically placed within a fixed region with respect to the top and the left of the region, and also given specific height, width, and z-order attributes. The HTML Layout Control provides a preview implementation for incorporating these 2D regions within HTML documents for display in the Microsoft Internet Explorer 3.0 and other browsers supporting ActiveX Controls.

The HTML Layout Control creates a 2D layout format that hosts other ActiveX Controls. This Control can also take advantage of the ActiveX Control '96 specification for transparent and windowless controls. For example, consider the following web page as displayed in Microsoft Internet Explorer 3.0:

¹ Also see the W3C's Cascading Style Sheet specification at <http://www.w3.org/pub/TR/WD-CSS1.HTML>



The illustration depicts the HTML Layout Control as displayed in Microsoft Internet Explorer 3.0. The HTML Layout Control pictured above hosts a number of other controls to make up the complete page. In particular, the page includes:

- **Exact 2D placement.** Controls are placed exactly where the author intended them to be placed within the 2D region. On a typical HTML page, the client web browser determines the placement of each element; using 2D authoring and the ActiveX Layout Control, the author maintains this control, ensuring consistency of the user interface, and making it possible to create very rich images without having to use large bitmaps.
- **Overlapping regions.** The author can also specify the exact Z-order of each control on the page. In the example above, the box containing the “Breaking Grounds” text overlaps part of the larger box containing the “Good Vibrations” image. Both of these regions are overlapped by the “Volcano Coffee” logo image.
- **Transparency.** The text label “Good Vibrations” is overlapping both the background and the large shaded box in the middle of the page. The text control which implements this label is transparent, so that users can see through the text to objects underneath. Also, the cup (a Windows® metafile) is overlapped by the text label, and is itself a transparent control overlapping both the background and large shaded box. Any ActiveX Control implementing the ActiveX Control '96 specification (published on the Microsoft Web site) for windowless controls can be transparent and used in this manner within the HTML Layout Control.
- **Scripting.** The HTML Layout Control also fully supports scripting, including both Visual Basic Script and JavaScript™. Thus, any object contained in a 2D region can script other objects in that region. In this example page, clicking on various label controls will cause the images

within the region to change. Clicking on other objects on the page causes navigation—switching the browser to a different page. In this example, this is handled with VB Script.

Designing 2D-style Pages Using the HTML Layout Control

To assist in designing 2D regions, Microsoft is also providing a free editing environment, the ActiveX Control Pad. The ActiveX Control Pad is a WYSIWYG editing environment for embedding ActiveX Controls in a 2D region. Insertion of controls, including adjusting position, Z-order, and properties of each control can be accomplished using a simple, visual editor, similar in appearance to the form editor found in Microsoft Visual Basic. The ActiveX Control Pad supports automatic syntax generation of the <OBJECT> tag used to embed objects in HTML pages. It also automatically generates CLSIDs which identify each object in the registry, and property names and values for each object. A core collection of Microsoft ActiveX Controls are provided with the ActiveX Control Pad, and other ActiveX-compliant controls can be added to the toolbox. Many of the supplied controls fully support the ActiveX Control '96 windowless control specification, providing authors with a rich palette of tools for creating next-generation Web pages.

The ActiveX Control Pad also includes a Script Wizard for VB Script and JavaScript, and an Object Editor for placing ActiveX Controls directly in the HTML stream without using a 2D region. Please refer to the *ActiveX Control Pad Whitepaper* for more information on this tool. In addition, any text editor can be used to create pages using 2D-style layout. Authors can simply follow the 2D layout specification as provided in this document. Finally, Microsoft also plans to support WYSIWYG, 2D-style HTML authoring in a variety of other authoring and development tools over time.

Together, the HTML Layout Control and ActiveX Control Pad provide a set of tools for authors to build compelling Web pages. By itself, however, the HTML Layout Control presents a format for 2D layout that other authoring tools can also support.

HTML Layout Control - How Does it Work?

The HTML Layout Control represents a preview implementation of the draft W3C specification for 2D-style layout in HTML. As a preview implementation, the HTML Layout Control specifies a separate file for objects that are placed in the 2D region. This file is a simple text file that follows the preliminary W3C syntax. Objects inserted into the file are surrounded with the block tag, <DIV>, which is given a height and width. The file is stored on the server, and is read by the browser when the browser interprets the HTML Layout Control Object in the HTML stream.

Supporting the 2D-style layout format simply means following a few rules:

- 2D regions must be defined in a separate file. This file must be given an .alx extension and is defined as part of the HTML Layout Control object tag within the HTML stream.²
- Objects defined within the 2D region must be ActiveX-compliant objects.
- The 2D region must be enclosed within <DIV> tags as such: <DIV STYLE="LAYOUT:FIXED; HEIGHT *value* WIDTH *value*"> ... </DIV>

² After the W3C 2D layout specification is finalized, Microsoft will support 2D-style layout definitions directly within the HTML document, rather than in a separate file.

Supporting the 2D format defined by the HTML Layout Control is a preliminary mechanism to provide layout control over ActiveX Controls. Microsoft is committed to supporting the final specification natively within the Microsoft Internet Explorer browser after the W3C standard is formalized. Microsoft is also committed to ensuring that the HTML Layout Control format is compatible with our future 2D-layout support that will be incorporated directly into future releases of the Microsoft Internet Explorer.

Using the HTML Layout Control

To use the HTML Layout Control, the author creates an HTML page, and within the <BODY> of the page the HTML Layout Control is included in an <OBJECT> tag in the HTML stream. The ActiveX Control Pad can optionally be used to more easily insert an HTML Layout Control (2D region) within a new or existing HTML document.

Sample HTML file including an HTML Layout Control

```
<HTML>
<HEAD>
  <TITLE>Volcano Coffee</TITLE>
</HEAD>
  <BODY BGCOLOR=#a01e0c>
    <OBJECT CLASSID=" CLSID:812AE312-8B8E-11CF-93C8-00AA00C08FDF"
      ID="volcano" HEIGHT=444 WIDTH=635
      STYLE="LEFT:0;TOP:0;WIDTH:640;HEIGHT:480">
      <PARAM NAME="URL" VALUE="http://www.sample.com/sample.alx"></PARAM>
    </OBJECT>
  </BODY>
</HTML>
```

When instantiated, the HTML Layout Control defined by the <OBJECT> tag will create a 2D region within the HTML document. The 2D layout file (.alx file) used by this instantiation is referenced by the URL property with the VALUE "http://www.sample.com/sample.alx". This file contains the persisted layout for the 2D region designed by the author. The URL to this file is specified by the value of the URL property of the HTML Layout Control. This property is persisted as a <PARAM> attribute on the <OBJECT> tag of the Layout Control. The URL, <OBJECT> tag and <PARAM> attribute(s) are all automatically generated by the ActiveX Control Pad when inserting a 2D region.

The size of the region will be the height and the width (in pixels) specified as STYLE layout-attributes on the <DIV> tag which is defined within the 2D region (within the .alx file). If height and width are also defined on the <OBJECT> tag for the HTML Layout Control in the HTML stream, and the values for HEIGHT and WIDTH differ, the size of the layout will scale.

The <DIV> tag is defined by the W3C as a block tag for division containment and is currently parsed by IE 3.0. The HTML Layout Control uses a single <DIV> ... </DIV> block in the .alx file to define a 2D region.

The syntax for the <OBJECT> tag conforms to the W3C HTML standard which is implemented by Microsoft Internet Explorer 3.0. See the W3C specification for the <OBJECT> tag, *"Inserting Objects"* located at <http://www.w3.org/pub/WWW/TR/WD-object.html>.

More than one HTML Layout Control can be contained on an HTML page. Each 2D region defined by a single instance of the HTML Layout Control behaves like other objects within an HTML page—they can be placed in tables, aligned, centered, etc. In this preliminary release, each instance of an HTML Layout Control, however, is autonomous: no event or scripting within an HTML Layout Control can

have an effect on objects outside this region, and no event or scripting outside this region can have an effect on the HTML Layout Control.

The .alx file

The .alx file defines the fixed region and includes any scripts that would act on events and controls defined within this region. The region itself is defined with a <DIV> tag. Script blocks can be defined before or after the <DIV> tag within the .alx file. This implementation does not allow script tags inside the <DIV> tag to help ensure graceful forward compatibility with HTML standards and implementations.

The following outlines the attributes for the <DIV> tag as used in this implementation. While the .alx file format does not support CSS (cascading style sheets), the attributes specified on the <DIV> tag are CSS style attributes that Microsoft anticipates, at the time of this writing, will be forward compatible with the future implementation of standard W3C HTML CSS. These CSS style attributes are specified with the STYLE="LAYOUT:FIXED" attribute.

The bare-boned structure of a .alx file looks like:

```
<DIV [ID=name] STYLE = "layout-style-attributes">  
    object-blocks  
</DIV>
```

Where direct attributes on the <DIV> tag are as follows:

<DIV> attributes	Description
ID	an optional ID attribute to identify the fixed region to scripting
STYLE	an inline style for <DIV> tag. See Style Attributes below.

And attributes on the STYLE attribute are as follows:

STYLE attributes	Description
LAYOUT	This must be defined as FIXED for a 2D region.
HEIGHT	Specifies the height of the layout region in pixels.
WIDTH	Specifies the width of the layout region in pixels
BACKGROUND	Specifies the background color of the layout region in HEX digits.

Any ActiveX Control may be used within an .alx file. This includes, but is not limited to, controls that implement the ActiveX Control '96 specification for windowless, transparent controls. A limitation of the HTML Layout Control is that no other type of OBJECT may be specified with the <OBJECT> tag within an .alx file. <OBJECT> tags defined outside the 2D region are parsed by the browser, and in the case of Microsoft Internet Explorer 3.0 would conform to the <OBJECT> defined within the W3C HTML specification.

Sample .alx file

```
<!--Define A 2D Division-->
<DIV STYLE="LAYOUT:FIXED;WIDTH:635;HEIGHT:444;">
<!--Add Object blocks-->
  <OBJECT ID="MyButton" STYLE="TOP:0;LEFT:0;WIDTH:808;HEIGHT:552;"
    CLASSID="CLSID:978C9E23-D4B0-11CE-BF2D-00AA003F40D0">
    <PARAM NAME="BackColor" VALUE="794272">
    <PARAM NAME="Size" VALUE="21378;14605">
  </OBJECT>
  <OBJECT ID="Image1" STYLE="TOP:144;LEFT:16;WIDTH:200;HEIGHT:192;"
    CLASSID="CLSID:D4A97620-8E8F-11CF-93CD-00AA00C08FDF">
    <PARAM NAME="PicturePath" VALUE=" http\\www.mywebserver\\drawncup.wmf">
    <PARAM NAME="BorderStyle" VALUE="0">
    <PARAM NAME="Size" VALUE="5292;5080">
    <PARAM NAME="VariousPropertyBits" VALUE="19">
  </OBJECT>
</DIV>

<SCRIPT LANGUAGE="vbscript">
sub MyButton_click
  Image1.zorder(0)
end sub
sub Image1_click
  window.location.href = "http:\\www.mywebserver\\gv.htm"
end sub
</SCRIPT>
```

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.

Conclusion

The HTML Layout Control provides a preview implementation of 2D-style layout for HTML based on an early draft specification by the W3C. 2D-style layout allows page developers to create more sophisticated page designs and user interfaces by providing precise control over object layout. Authors can exactly control x, y and z-ordering of individual objects and collections of objects. In this way, 2D-style authoring significantly enhances HTML.

For More Information

For more information or to obtain the beta software with sample pages, visit the Internet Explorer 3.0 Web site at <http://www.microsoft.com/ie/>.

Companies, names, and/or data used in screens and sample output are fictitious, unless otherwise noted.

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, Visual Basic, and Windows are registered trademarks and ActiveX is a trademark of Microsoft Corporation.

Adobe and Photoshop are trademarks of Adobe Systems, Inc. Netscape is a trademark of Netscape Communications Corporation. Quark Xpress is a registered trademark of Quark, Inc. JavaScript is a trademark of Sun Microsystems, Inc.