



# Simple Hyperlink Navigation

December 1995

Distribution: Public

© Microsoft Corporation 1995. All Rights Reserved.

DRAFT

**Executive Summary:** This document contains the specification for Simple Hyperlink Navigation, a set of APIs that allow *new and existing applications* to integrate with each other and with Web browsers. Simple Hyperlink Navigation is a subset of the OLE Hyperlinking specification, a more complete model for how applications, documents, and document containers interact during hyperlink navigation. Applications that use Simple Hyperlink Navigation APIs will automatically integrate with the more complete OLE Hyperlinking interfaces (still in review). These “simple” APIs will provide seamless hyperlinking navigation when used within OLE Document Objects and OLE Controls hosted in Microsoft® Internet Explorer®, and they will also provide hyperlinking functionality for OLE-unaware applications.

## 1. Simple Hyperlink Navigation

- 1.1. Introduction.....
- 1.2. Technical Details.....

**NOTE:** THIS DOCUMENT IS AN EARLY RELEASE OF THE FINAL SPECIFICATION. IT IS MEANT TO SPECIFY AND ACCOMPANY SOFTWARE THAT IS STILL IN DEVELOPMENT. SOME OF THE INFORMATION IN THIS DOCUMENTATION MAY BE INACCURATE OR MAY NOT BE AN ACCURATE REPRESENTATION OF THE FUNCTIONALITY OF THE FINAL SPECIFICATION OR SOFTWARE. MICROSOFT ASSUMES NO RESPONSIBILITY FOR ANY DAMAGES THAT MIGHT OCCUR EITHER DIRECTLY OR INDIRECTLY FROM THESE INACCURACIES. MICROSOFT MAY HAVE TRADEMARKS, COPYRIGHTS, PATENTS OR PENDING PATENT APPLICATIONS, OR OTHER INTELLECTUAL PROPERTY RIGHTS COVERING SUBJECT MATTER IN THIS DOCUMENT. THE FURNISHING OF THIS DOCUMENT DOES NOT GIVE YOU A LICENSE TO THESE TRADEMARKS, COPYRIGHTS, PATENTS, OR OTHER INTELLECTUAL PROPERTY RIGHTS.

---

---

# 1 Simple Hyperlink Navigation

---

## Introduction

One of the most compelling ease-of-use features in World Wide Web applications is the navigation-based user-interface model including *point-and-click hyperlink navigation*, a *history* list (with commands like *Go Back* and *Go Forward*), and a *favorites* list. Like other OLE-based application integration technologies, OLE Hyperlinks enable you to offer this same user-interface and navigation model within your documents and application, allowing you to integrate them seamlessly with other hyperlinking applications including web browsers.

This document provides an overview of the Simple Hyperlink Navigation API—a subset of the more complete OLE Hyperlinking specification that allows developers to add hyperlink navigation support to an application or document. When using “simple hyperlinking”, the frame takes over responsibility for providing a navigation stack and integrated history/favorites. For OLE Controls or OLE Document Objects hosted within a browser frame such as Microsoft® Internet Explorer, this API allows full integration in the browser navigation stack and history list. For OLE-unaware application, the API uses ShellExecute to launch the necessary viewer to accomplish hyperlink navigation. The Simple Hyperlink Navigation API provides seamless hyperlinking integration with the more complete OLE Hyperlinking specification<sup>1</sup>, and will allow hyperlinking to and from documents hosted in any hyperlink frame.

Related Documents	Location
HREF="hlink.doc	hlink.doc

---

## Technical Details

```
typedef enum tagHLNF {
    HLNF_INTERNALJUMP,
    HLNF_OPENINNEWWINDOW
} HLNF;

HRESULT HlinkSimpleNavigateToString ([in] LPCWSTR szTarget, [in] LPCWSTR szLocation, [in] LPCWSTR szTargetFrameName,
    [in] IUnknown* punk, [in] IBindCtx* pbc, [in] IBindStatusCallback* pbsc, [in] DWORD grfHLNF, [in] DWORD dwReserved);
HRESULT HlinkSimpleNavigateToMoniker ([in] IMoniker* pmkTarget, [in] LPCWSTR szLocation, [in] LPCWSTR
    szTargetFrameName, [in] IUnknown* punk, [in] IBindCtx* pbc, [in] IBindStatusCallback* pbsc, [in] DWORD grfHLNF, [in] DWORD
    dwReserved);
HRESULT HlinkNavigateString ([in] IUnknown* punk, [in] LPCWSTR szTarget);
HRESULT HlinkNavigateMoniker ([in] IUnknown* punk, [in] IMoniker* pmkTarget);
HRESULT HlinkGoBack([in] IUnknown* punk);
HRESULT HlinkGoForward([in] IUnknown* punk);
```

### HLNF Enumeration

Values from the HLNF enumeration are used to indicate how hyperlink navigation is to proceed, and also convey contextual information about the navigation from each of the objects participating in the navigation protocol to the others.

---

<sup>1</sup> This specification is still under review. A complete implementation matching the specification will be available in the future.

Value	Description
HLNF_INTERNALJUMP	The navigation is an internal jump within the current hyperlink target.
HLNF_OPENINNEWWINDOW	Open the hyperlink target in a new window after navigating to it

### HlinkSimpleNavigateToString

HRESULT HlinkSimpleNavigateToString (szTarget, szLocation, szTargetFrameName, punk, pbc, pbsc, grfHLNF, dwReserved);

This function will execute a hyperlink jump to a new document or object (specified in szTarget). Most of the parameters in this function call are optional. Example code for an OLE control jumping to a hyperlink location:

```
// do the navigation
HlinkSimpleNavigateToString( "http://www.microsoft.com/default.htm", NULL, NULL, NULL, punkMe, pbc, pbsc, 0, 0);
```

Argument	Type	Description
szTarget	LPCWSTR	String which helps identify the hyperlink target. This string is resolved into a moniker for underlying binding operations via MkParseDisplayNameEx or via IBindHost::CreateMoniker. If NULL, then the navigation is within a document.
szLocation	LPCWSTR	Optional string representing the location within the hyperlink target for the new hyperlink. May be NULL.
szTargetFrameName	LPCWSTR	Optional string naming the target frame for the hyperlink navigation. This argument only affects navigation within a document container that understands frame-sets. May be NULL.
pUnk	IUnknown*	The IUnknown pointer to the document or object that is initiating the hyperlink. If NULL, it is assumed the hyperlink originates from an OLE-unaware application.
pbc	IBindCtx*	The bind context to use for any moniker binding performed during the navigation.
pbsc	IBindStatusCallback*	The bind-status-callback to use for any asynchronous moniker binding performed during the navigation. May be NULL, in which case the caller is not interested in progress notification, cancellation, pausing, or low-level binding information.
grfHLNF	DWORD	Values taken from the HLNF enumeration.
dwReserved	DWORD	Reserved for future use, must be set to zero.
<i>Returns</i>	S_OK	Success.
	E_INVALIDARG	One or more arguments are invalid.
	<i>other</i>	Various error return values.

### HlinkSimpleNavigateToMoniker

HRESULT HlinkSimpleNavigateToString (pmkTarget, szLocation, szTargetFrameName, punk, pbc, pbsc, grfHLNF, dwReserved);

This function will execute a hyperlink jump to a new document or object (specified in mkTarget). Most of the parameters in this function call are optional. Example code for an OLE control jumping to a hyperlink location:

```
// do the navigation
HlinkSimpleNavigateToMoniker( pmkHome, NULL, NULL, NULL, punkMe, pbc, pbsc, 0, 0);
```

Argument	Type	Description
pmkTarget	IMoniker *	Moniker which identifies the hyperlink target. If NULL, then the navigation is within a document.
szLocation	LPCWSTR	Optional string representing the location within the hyperlink target for the new hyperlink. May be NULL.
szTargetFrameName	LPCWSTR	Optional string naming the target frame for the hyperlink navigation. This argument only affects navigation within a document container that understands frame-sets. May be NULL.
pUnk	IUnknown*	The IUnknown pointer to the document or object that is initiating the hyperlink. If NULL, it is assumed the hyperlink originates from an OLE-unaware application.
pbcb	IBindCtx*	The bind context to use for any moniker binding performed during the navigation.
pbcb	IBindStatusCallback*	The bind-status-callback to use for any asynchronous moniker binding performed during the navigation. May be NULL, in which case the caller is not interested in progress notification, cancellation, pausing, or low-level binding information.
grfHLNF	DWORD	Values taken from the HLNF enumeration.
dwReserved	DWORD	Reserved for future use, must be set to zero.
<i>Returns</i>	S_OK	Success.
	E_INVALIDARG	One or more arguments are invalid.
	<i>other</i>	IHlink::Navigate error return values.

### **HlinkNavigateString**

HRESULT HlinkNavigateString (punk, szTarget);

This function will execute a hyperlink jump to a new document or object (specified in szTarget). This function is implemented as a simple macro that calls HlinkSimpleNavigateToString passing in NULL for most parameters.

Argument	Type	Description
pUnk	IUnknown*	The IUnknown pointer to the document or object that is initiating the hyperlink. If NULL, it is assumed the hyperlink originates from an OLE-unaware application.
szTarget	LPCWSTR	String which helps identify the hyperlink target. This string is resolved into a moniker for underlying binding operations via wEx or via IBindHost::CreateMoniker.
<i>Returns</i>	S_OK	Success.
	E_INVALIDARG	One or more arguments are invalid.
	<i>other</i>	Various error return values.

### **HlinkNavigateMoniker**

HRESULT HlinkNavigateString (punk, pmkTarget);

This function will execute a hyperlink jump to a new document or object (specified in mkTarget). This function is implemented as a simple macro that calls HlinkSimpleNavigateToMoniker passing in NULL for most parameters.

Argument	Type	Description
pUnk	IUnknown*	The IUnknown pointer to the document or object that is initiating the hyperlink. If NULL, it is assumed the hyperlink originates from an OLE-unaware application.
pmkTarget	IMoniker *	Moniker which identifies the hyperlink target.
<i>Returns</i>	S_OK	Success.
	E_INVALIDARG	One or more arguments are invalid.
	<i>other</i>	IHlink::Navigate error return values.

### **HlinkGoBack**

HRESULT HlinkGoBack(punk);

This function will execute a hyperlink jump backwards within the navigation stack. **Note:** this function will only work if the specified pUnk points to an object that is hosted within a hyperlink-aware container, such as Microsoft Internet Explorer or an OLE Hyperlink Frame.

Argument	Type	Description
pUnk	IUnknown*	The IUnknown pointer to the document or object that is initiating the hyperlink. May not be NULL.
<i>Returns</i>	S_OK	Success.
	E_INVALIDARG	One or more arguments are invalid.
	<i>other</i>	Various error return values.

### **HlinkGoForward**

HRESULT HlinkGoForward(punk);

This function will execute a hyperlink jump forwards within the navigation stack. **Note:** this function will only work if the specified pUnk points to an object that is hosted within a hyperlink-aware container, such as Microsoft Internet Explorer or an OLE Hyperlink Frame.

Argument	Type	Description
pUnk	IUnknown*	The IUnknown pointer to the document or object that is initiating the hyperlink. May not be NULL.
<i>Returns</i>	S_OK	Success.
	E_INVALIDARG	One or more arguments are invalid.
<i>other</i>		Various error return values.