

Netdocs Requirements for IE6

Timeline for NetDocs Ship Dates	
Milestone	End Date
NetDocs M3	Now
NetDocs M4	June-99
NetDocs M5	September-99
Office 10 feature complete	February 2000
Netdocs M7 -- feature complete	March 2000
NetDocs Full package ships	Q4 2000
Office 10 ships	Q2 2001

*← we need all of our Trident features complete in this time frame in order to be able to utilize them*

*← we are negotiating to become Outlook's default text editor in the Office 10 2000 timeframe, so this is feature complete for our universal canvas component*

*\* Indicates the Netdocs milestone we need this feature from Trident in*

Category	Feature Area	Problem	Suggested Solution	Dev Milestone*	Status
----------	--------------	---------	--------------------	----------------	--------

MS-CC-Sun 000001223230  
HIGHLY CONFIDENTIAL

Plaintiff's Exhibit  
8815  
Comes V. Microsoft

Drop dead (can't ship w/o)	Rich text editing experience	Our editing code requires the tree sync code that is in our private Trident source tree. Also, tree sync needs to work with DHTML (i.e. DHTML goes through markup services)	Check in and maintain our tree sync code in the IES code tree. Make DHTML go through markup services	AlexGo	M4	Preliminary tree sync code is active in official builds. Working on enhanced model with EricVas, et. al.
	General Stability	New builds of Trident severely destabilize NetDocs every time we take them	NetDocs BVTs are run against Trident checkins, commitment to fix NetDocs-related bugs, etc	AlexGo	M4	Agreed to; should happen week of 6/7.
	Editable print preview	We need to be able to accurately wrap text on screen the same way it is going to be wrapped on the printer	WYSIWYG text Trident's line layout code tells Line Services about both the printer resolution and the screen resolution	Sidda	M5	On Trident's dev schedule
		We need to be able to view multiple pages at the same time in print view (i.e. zoom out) and also need to be able to "Zoom" all HTML elements so that they are laid out correctly for the given printer resolution	WYSIWYG zoom for all HTML elements	Sidda	M5	Trident is implementing WYSIWYG layout. NetDocs is implementing zoom in inside Trident.
		NetDocs is creating an editable page view. In order to do this, the main Trident display code needs to be able to paginate text for an external layout	Layout block implementation (pagination of text) and page table-driven layout (necessary for efficient editable Print View) as an outside component.	Sidda	M5	Trident is implementing

MS-CC-Sun 000001223231  
HIGHLY CONFIDENTIAL

					support for multiple layouts per element. NetDocs will add support for chaining layouts together and will implement pagination and page-table driven layout inside Trident
		Printing appears consistent with print preview			NetDocs is implementing support for external printing in Trident. NetDocs will implement printing support in NetDocs.
		We have made a number of "App hacks" in our source tree for	Port or reimplement our app hacks in the IEB	Sidda	M5
					M5

MS-CC-Sun 000001223232  
HIGHLY CONFIDENTIAL

		Trident We need the functionality provided by these hacks to be incorporated into the main Trident tree (even if the implementation is different). For example, we have a hack that turns off Trident selection handling so that we can provide our own.	codebase			All our hacks have been enabled in core Trident code. Trident team will remove them.
	NetDocs Multimedia	Zooming: To implement "thumbnail" views (of documents, photographs, video clip, etc.) we need to have the ability to zoom an element and its content. Zooming will also be useful as an effect for photo editing.	New Transformation scheme based on the VML transform syntax.		ChrisF/MikeH/M4	
Must Have (cripping but we could possibly ship)	XSL Based Flexible presentation views	A NetDocs document consists of a number of structural elements such as a navbar and a title bar, plus a DIV where the document content lives. We want to immediately be able to show the first "page" of the DIV and allow the user to interact with the document while the rest is loading. We need to be able to download HTML content async into the DIV container while at the same time manipulating the tree. We want this container to appear as part of the larger document so that the user can select across the container boundaries, etc.	Implement an "inline frame" Give us a div/frame that is able to download a url async and the div/frame to be a "target". Ideally we would like it to be a iframe with an attribute so that it works on downlevel browsers like a normal iframe. This (requiring the tag to be iframe) is not an absolute requirement because we could write script to modify the div to an iframe in downlevel browsers.		Rejesh Jha M4	Talked about; presumably in the works (don't know for sure)

	<p>Today any HTML generated by XSL is "pushed" into the the HTML tree but the tree isn't always ready to have stuff inserted into it. Therefore we have to wait a long period of time to be sure that the tree is ready, causing our app to appear sluggish.</p> <p>Also - for perf reasons (not to cause too many inserts and relayout in the Trident tree), we paste the xsl generated text as a block - so no async</p>	<p>Async insertion of XSL generated HTML. AnandRa has talked about a "pull model" where the tree asks a behavior for an async stream</p>	Rajesh Jha	M4	no progress; should be demoted to 'would like'
	<p>When are loading a page, our perf is slow because we don't know exactly when behaviors are created and ready to use. So we either wait too long so that we are sure the behaviors are ready, or we try to access the tree too early and crash. This kills our tree synchron code because we are trying to synchron items that are not in the tree yet</p>	<p>We need either  A) Synchronous creation of behaviors, plus a way to check if behavior creations are pending, plus a notification if all behaviors are created and accessible, or  B) Identity Peers that support synchronous creation, enter/exit tree notifications, and contentready notifications</p>	TamF	M4	
	<p>Today we have a problem where we insert stuff into the tree and then very non deterministically wait for the tree to tell us it is done computing its layout. Then we see if we need to insert more - etc</p>	<p>Pull model behaviors. Trident pulls in enough text from behavior and stops. Ideally the tree gives us a measurement surface or pulls in enough text to fit a div of a specified height.</p>	Rajesh Jha	M4	

	<p>Right now we don't know when we can calculate the size of an HTML object such as a DIV. We cannot tell when an object is finished laying out. For instance, our inbox is created using generated HTML. We take the size available for the inbox and then generate HTML to fill it. But we don't have a reliable way to find out when the div where the inbox is going is finished laying out and its size will not change anymore. This makes our perf suck.</p>	<p>A LayoutComplete notification and a way to check to see if layout calcs are still pending.</p>	TamF	M4 (at least the API defined)	
	<p>Perf is slow for performing multiple operations to the tree at one time. For example, when we change the orientation of our nav bars from left hand side to across the top, we relayout when we remove the nav bars, and then relayout when we add them to the top.</p>	<p>Prevent layout from calculating until multiple tree operations are complete.</p>	TamF	M5	
Rich text editing experience	<p>Currently, calling ParseString on table subelements (such as the string "blah", list elements and maybe others) results in the markup container that contains bogus garbage elements. This means we have to put things like "&lt;table&gt;" around our HTML, then sift through the resulting container to local the bit that we were really trying to insert. This causes our code to have a ton of special cases and makes the behavior unpredictable for extensions to our code.</p>	<p>ParseString to handle all elements in the way we would expect.</p>	TomLaw	M4	no progress

	In order for us to provide basic editing with in-doc UI elements such as table headers, we need to understand how the CP structure of the document changed during the course of the tree sync operations	Pre and post notifications for tree sync operations ?? We're open for suggestions on this one.	TomLaw	M4	discussed, but no agreed upon solution
	Inserting and deleting style sheets and script elements is not robust. We can add stylesheets but not delete them, and we can't add/delete scripts at all. This affects our work on creating a StyleMaster to globally keep track of stylesheets in the current document and update them as formatting occurs, and also makes our Themes not work properly	Need reliable methods for adding and deleting elements from the head (such as styles or scripts)	OliverF	M4	
	We need to be able to quickly compare and find the difference between style sheets. Right now the perf of our stylemaster is unacceptable.	On elements, styles and rule styles, be able to enumerate properties that are currently set in a fixed order	OliverF	M4	
	We need to be able to tell if a stylesheet/Rule/RuleStyle is actually in use (is it referenced by any elements in the current document). This is used by the stylemaster. Right now we use a fragile ref-counting system to see when we can delete a stylesheet/rule, but these are not robust in the case where the user has edited the HTML directly (because we are storing the refcount as an expando property on the stylesheet)	Some property that we can check to see if the stylesheet/rule is in use	OliverF	M4	

MS-CC-Sun 000001223236  
HIGHLY CONFIDENTIAL

	<p>We want to provide all of the typical document navigation commands such as arrow up/down, pageup/down, etc. This functionality is very hard to do in Trident currently. In order to do this, we rely on the undocumented interface, HTMLViewServices. This is not a stable interface for situations such as HTML elements with padding, margins, etc, and also does not provide enough functionality to implement pageup/pagedown robustly.</p>	<p>Stabilize HTMLViewServices and add the ability to move in page increments (or a way for us to extend your functionality to achieve this)</p>	<p>AlexGil</p>	<p>M4</p>	
	<p>We need to be able to tell if a stylesheetRule/RuleStyle is actually in use (is it referenced by any elements in the current document). This is used by the stylemaster. Right now we use a fragile reflowing system to see when we can delete a stylesheet/rule, but these are not robust in the case where the user has edited the HTML directly (because we are storing the reflow as an expando property on the stylesheet).</p>	<p>Some property that we can check to see if the stylesheet/rule is in use.</p>	<p>OliverF</p>	<p>M5</p>	



	<p>We cannot implement in-doc UI elements that are "pinned" to other HTML elements in a robust way. We don't know how to make them update if the objects they are trying to be pinned to move or resize. Adorners don't work for us because they don't provide rich enough notifications (i.e. scrolling is missing) and they are too slow. Also, during notifications, we need to be able to update the tree, which we can't do w/ adorners</p>	<p>Notification that an HTML element's size or position has changed (including the entire Trident window)</p>	Zhenya Sigal	M5	
Using Trident as a GDI to render Netdocs UI	<p>We can't completely control the mouse cursor right now. For instance if you disable draggin an element and then you click and drag on the element, you will always get the ghostbusters icon</p>	<p>More complete control over the mouse cursor at the host level</p>	Mhop	M4	
	<p>We can't make a positioned DIV span across the boundary of a frame or any other object implemented as an ActiveX control (the ActiveX control will always be on top in the Z-order). This makes it impossible to implement robust pop-up controls.</p>	<p>Ability to make a positioned DIV have a greater Z-order than an ActiveX control.</p>	Mhop	M5	
	<p>Because we are using our own pluggable protocol, the images in our UI don't get cached and are reloaded every time you view the same page</p>	<p>Support for caching with pluggable protocols</p>	Paul Erickson	M8	

MS-CC-Sun 000001223238  
HIGHLY CONFIDENTIAL



	Table operations such as insert, delete, selection, etc are slow and buggy right now because we have to create a hash table for every HTML table in order to map column number to actual table cells (because merged cells complicate this transformation)	Interface and methods to get correct Table cell from column number (they're not the same in case of merged cells). Something like GetCellAtRowColumn(long row, long column, HTMLTableCell*). Sasha says Trident already has this information internally	SashaT	M5	
	Because table selection is not supported in Trident, we have to fake it by setting background color on cells individually. This has very bad perf for both column and rectangular table selections (ex. A 13x13 region needs 269 calls to the object model in order to appear selected).	Implement table selection rendering natively in Trident	SashaT	M5	
	Perf for finding a specific column is too slow (also the table object model seems incomplete)	Implement COLS collection in object model	SashaT	M6	
WebOC Problems: Allowing the NetDocs shell to have the same browsing functionality that the RE shell does	There are various assumptions in shdocvw right now which prevent the weboc from behaving consistently regardless of the way it was created etc. E.g. it does not fire notifications (even if clients have advised) if it is not the top level browser. For instance, we don't get enough information from the webOC to be able to properly enable/disable our back/forward buttons. Also, we can't host frames correctly if we aren't inside shdocvw	The ability to tell a webbrowser OC that we want it to be the top level browser	Harip	M5	

MS-CC-Sun 000001223240  
HIGHLY CONFIDENTIAL

		We can't get the address for the webOC's current location back with get_LocationURL() in all cases (specifically "server" type locations don't work). This causes our homgrown address bar to be flaky	Should be able to get the address of the webOC's current location back with get_LocationURL() in all cases					
				HariP	M6			
		Scripted window events such as window close shul down the WebOC wio any warning to NetDocs	Need notifications on scripted window events especially window.close, so that the containing application can shut down in a top down manner. Something like onWindowClose(BOOL *pContinue).	HariP	M6			
		We can't provide inplace folder browsing functionality using the webOC	Enable this for cases where the WebOC is not inside of IE	HariP	M6			
		We can't use the "Search when URL fails" feature in NetDocs because the webOC doesn't support it outside of IE	Enable this for cases where the WebOC is not inside of IE	HariP	M6			
Editable print preview		We need to be able to control how the pages are laid out on the screen for page view (i.e. when showing multiple pages, how are they arranged). Also, we want to be able to control actual page structure for items such as margins, columns, headers and footers.	Page structure and arrangement is externally defined					Trident is implementing layout templates for us.
		We need to be able to intelligently break tables and other elements across pages	Tables, DIVs and other block elements can break vertically across pages	Sidda	M6			

MS-CC-Sun 000001223241  
HIGHLY CONFIDENTIAL

						<b>NetDocs will implement table and DIV breaking inside Trident</b>
		We need to be able to reasonably map all HTML elements to the printed page and display this in our print view	Good model for all HTML elements, (for example absolute & relative positioned objects, page backgrounds, filters, animated GIFs, other dynamic content) in layout block model	Sidda	M6	<b>NetDocs is implementing support for positioned objects in Print View inside Trident</b>
<b>Architecture:</b>	<b>using Trident as a platform</b>	Because Trident does things async it is VERY difficult to track down bugs because we are not sure which of our calls caused it	Support a command we can send Trident that says "Do all your calculations and don't return until you are done" We would turn this on when we were debugging	Bruce Leban	M4	

MS-CC-Sun 000001223242  
HIGHLY CONFIDENTIAL

Bugs	When table rows or cells are inserted or removed using markup services (for instance, via MarkupServices.Move or MarkupServices.RemoveElement), the table does not update cached information regarding the number of rows or columns, or other properties. The cache isn't even invalidated at all. This causes subsequent calls to produce wrong results or AV, making manipulating tables through markup services rather non-functional.	Update Table cache information after MarkupServices are used			
			tomlaw	M5	no progress
	We can't use CSS imports because it causes a leak when there is an OBJECT on the page	Fix this leak			
			Paul Erickson	M6	
NetDocs Multimedia	Need behaviors to be able to have efficient access to DD surface internally used by Trident rendering	New Behavior rendering scheme based on DC surfaces as opposed to DCs	ChrisF/MikeHy	M4	Waiting for spec from DMassy
	Need to be able to write behaviors that have "filter-like" functionality	New behavior rendering scheme allowing a behavior to provide a DD surface for Trident to render the content of the associated element into.	ChrisF/MikeHy	M4	Waiting for spec from DMassy
	Rotation and other transforms: critical photo editing effect. Need to be able to apply rotation, shifts, inversion, etc., to any element and its content. Should include decent anti-aliasing of the transformed bits	New Transformation scheme based on the VGX transform syntax			
			ChrisF/MikeHy	M4	

MS-CC-Sun 000001223243  
HIGHLY CONFIDENTIAL



	<p>We need to be able to get enough information about an element in the tree to be able to reconstruct it in another tree, or to rebuild it in the same tree for undo cases. Currently there is no way to get attributes for an element other than inserting it into a container, getting outerHTML, and stripping out the unwanted parts of the resulting string, which is very slow, fragile, and causes spurious behavior and script failures if any of that is associated with an element</p>	<p>An efficient way of getting all the attributes for an element and setting them, including for elements not currently in a container. This does not necessarily have to be in the serialization format (i.e. "a=x b=y c=z") though it would be useful</p>	TomLaw	M6	they've committed to it
	<p>We don't have a good way to find out if an element is still in the main doc or not (i.e. has it been deleted) Currently, asking the element for document causes a container to be created for the element if it isn't already in one. This is too slow, since we need to do this often. (We do it because we maintain a map from the data tree to the view tree and need to know if the element is still in the view tree. it might not be, but we don't know because the behavior's Detach() method is called asynch, long after the fact</p>	<p>Fast way of determining if an element is in a container or not.</p>	TomLaw	M6	no progress

MS-CC-Sun 000001223245  
HIGHLY CONFIDENTIAL



	Currently, QueryServices are immediately delegated to the container, which is a WebOC, so we have no opportunity to service the request ourselves. This would be used for exposing our per-document and per-view NetDocs specific services (such as in-doc UI, edit manager, extension manager, etc)	Extensible query-service support. The ability to register specific services that can be reached via a QueryService within the DHTML object model (e.g. by calling QueryService on a Element)	TomLaw	M6	no progress
	Because we can't tell when the mouse is over some parts of the scrollbar (esp. the lower right corner where the two scrollbars meet), autoscrolling does not work in cases where the mouse is in these regions	Hitesting/capturing mouse processing for them	AlexGil	M6	
	If someone has authored their own HTML, we want to be able to make our edits look as close as possible to what they wrote (figuring out where formatting has been applied in the stylesheet hierarchy)	On IHTMLCurrentStyle, being able to find out where an attribute comes from (is it inline, inherited from a parent, inherited from a stylesheet, etc)	OliverF	M6	
	Netdocs needs to support real TABs, especially inside of tables	Implement TABs per the CSS2 spec	alexgil	M6	
	Need to be able to manipulate multiple class names in the Trident object model. Currently these are all returned as a string and we have to parse them ourselves.	Implement a class names collection	OliverF	M6	
	Need to be able to save tree locations as CPs, save the HTML, load it again, and have those CPs still refer to the same location. We want this so we can implement persistent undo	Round-trip parsing with respect to the CP structure of the loaded representation	TomLaw	M6	

MS-CC-Sun 000001223246  
HIGHLY CONFIDENTIAL

	We want to do our own custom selection UI (outlining the objects instead of changing their background color)	Extensible selection UI support including table selection	AlexGil	M6	
	In-doc UI (such as delta decoration markers) do not show up correctly near tables because tables return incorrect coordinates from GetLineInfo()	Using IHTMLViewServices->GetLineInfo(), the correct baseline coordinates need to be returned for all elements—currently they are only correct for elements such as span, div, and k	Zhenya Sigal	M6	
	We can't change the IP to reflect italics, rotated text, etc	API to be able to change the IP	AlexGil	M6	
	Different APIs in Trident use different coordinate systems, which is super confusing	Use one consistent coordinate system (global)	Zhenya Sigal	M6	
	We want to be able to specify how drag and drop targets are rendered	Add the ability to hook in our UI to this feature	AlexGil	M6	
Using Trident as a GDI to render Netdocs UI	We get mouseout events when the mouse goes from a container into one of its children	Ability to set a property so that these types of events don't fire when you go from parent to child.	a-grecol	M6	
	Comparing two trees is too slow for us	Built-in Trident methods to compute the hash of an HTML tree so that we can quickly be able to see if two trees are likely to be the same. This would be a lot faster if it was inside Trident than if we do it.	Bruce Leban	M6	
	We can't specify sizing for UI elements in the way that we would like	More CSS attributes, being able to specify size in line units (may a box 3 lines tall for a particular font)	Bruce Leban	M6	

MS-CC-Sun 00001223247  
HIGHLY CONFIDENTIAL

		Our perf is slow when we load a page to edit because we have to create our own table of stylesheet rules indexed by selector text	Access to the style sheet rules by their selector text, not just index	OliverF	M6	
		We can't override the arrow key behavior for our own custom controls such as a listbox	Mechanism to override the default behavior for special keys (such as arrow keys)	Mhop	M6	
		We can't specify the formatting for different INPUT elements in an external stylesheet, so that we could change their look by switching stylesheets	Ability to format all INPUT elements separately from stylesheets	OliverF	M6	
		When using images in our UI, we never want to see image placeholders. Currently we can't turn them off	Ability to turn off image placeholders altogether	Bruce Leban	M6	
		Right now we can't give our scrollbars a "non-Win32" look (make them look flat, etc)	Control style for scrollbars	a-grecol	M6	
		Support for more Windows-like layout operations. We cannot achieve the kind of complex windowing behavior expected from an Office app. We should discuss the specifics about what can be done, but the type of features we need include: the ability to specify a hierarchy of nested rectangles that have anchors (N, W, S, E, etc), expands (true/false—whether the object expands to fill extra space), fill (x,y, both, how an object is stretched), has custom clipping behavior (shrink or clip).		Bruce Leban, Jon Perlow	M6	
			Break out Trident parser so that we can kerate over tokens	Prakash Sikchi	M6	

MS-CC-Sun 000001223248  
HIGHLY CONFIDENTIAL

			HTC files should be cached. Right now accessing the same file causes it to be loaded over and over again.	Paul Erickson	M6	
			HTC files are huge in memory--these need to be optimized.	Paul Erickson	M6	
		Because Trident is on the UI thread, our UI is not responsive when Trident is doing a lot of calculations.	Trident needs to get off the UI thread.	Bruce Leban	M6	
		We want to be able to retain mouse capture even if the mouse moves outside of the Trident window.	Trident enables this functionality.	Mhop	M6	
Allowing NetDocs functionality from within the IE shell	NetDocs documents use a lot of custom binary behaviors. Inside of IE, we would still like to be able to handle this behaviors.	Ability to create binary behaviors inside of IE that are not known to IE (i.e. to be able to specify that NetDocs will handle a binary behavior instead of having it be always passed off to IE to handle).		Paul Erickson	M6	
	We want to be able to add NetDocs functionality directly to IE if NetDocs is installed on the machine.	Allow us to hook in our DOM when encountering an XML island in a document in IE, and override Trident's DOM.		Rajesh Jha	M6	
Editable print preview	Many HTML tables are larger horizontally than a page and there is currently no way to print them all out. This is particularly important since we are targeting some	Tables can break horizontally across pages.		Sidda	M6	

	spreadsheet scenarios				None of these features are scheduled to be implemented yet.
	Printer-independent layout—like Word where changing the printer d/n cause the page to relayout		Sidda	M6	
	Efficient memory usage automatically discard unused display data structures & regenerate them on demand		Sidda	M6	
	widow/orphan control, keep together/keep with next	Support the CSS printing improvements for these	Sidda	M6	

Feature request Additions:

One additional item that I didn't see in the list, which I thought was supposed to be there, is pullInnerHTML() support for elements inside tables. Right now Trident does not handle such insertions inside of tables. Because of this our xsl application needs to scan what it is about to insert, and if it sees table elements, insert them using markup services or the table OM — a major perf problem

Raman/Prakash  
- the etag request should probably be here as well

MS-CC-Sun 000001223250  
HIGHLY CONFIDENTIAL

Scalability /  
Virtualization  
Issues.

MS-CC-Sun 000001223251  
HIGHLY CONFIDENTIAL