

## Content

<b>Title</b>	<b>Page</b>
<b>Content</b>	<b>1</b>
<b>Project Description</b>	<b>2</b>
<b>Project Boundaries &amp; Constraints</b>	<b>3</b>
<b>Introduction</b>	<b>4</b>
<b>Evaluation of Internet Explorer</b>	<b>4</b>
<b>Managing Users: IE Administration Kit Profile Manager</b>	<b>5</b>
<b>Changing System Policies &amp; Restrictions</b>	<b>7</b>
<b>Using Custom Policy Templates</b>	<b>8</b>
<b>Performance Improvements</b>	<b>10</b>
<b>Dynamic HTML Behaviour Components</b>	<b>10</b>
<b>Client capabilities</b>	<b>11</b>
<b>Browserless Applications</b>	<b>13</b>
<b>Integration with Productivity Applications</b>	<b>13</b>
<b>Conclusions</b>	<b>14</b>
<b>Appendix A: Looking ahead with web-based Network Management</b>	<b>15</b>
<b>Appendix B: Bibliography</b>	<b>15</b>

## Project Title: Centralized Browser Support

Author: Farsheed Azad MSc MCSE, June 2000

### Project Descriptions

**Business Issue:**

The Ministry is deploying workstations to all staff. The workstations have an internet browser included as part of the standard image. In order to provide better management of the browser tool so that the Ministry can leverage off this technology to enable business applications, the Ministry would like recommendations for centralized management of the browser.

**Business Requirements:**

Provide recommendations for deploying and managing the Internet browser on the SDES workstations so that:

- Browser settings are standardized
- The cost and effort of managing the browser is minimized and contained (centralized management)
- The browser settings are optimized for the business applications that will be enabled through this tool
- The browser is positioned to be able to effectively use proxy servers and caching

**Proposed Technical Solution:**

Company will produce, test and document a Microsoft facility that can be used to control browser configuration. This facility will use an Intranet Web server to store and be the central reference point of browser configurations. Questions such as the frequency of updates, the time to download browser updates, etc., would be investigated in the test lab and the pilot.

The following steps will be executed as part of this SR:

1. Develop the modifications to the internet profile of the NT workstation needed to support centralized browsers
2. Test package with the Microsoft Internet Administration kit
3. Deposit the changes or control files in an Intranet Web server
4. Use system policies and/or create a login script to distribute the required registry modifications to point the browser to the central Configuration server. The intent is to use policies first, wherever possible.
5. Using a test population, distribute the changes and monitor the results
6. Document the usage of the packaging software and the update procedures of the central Configuration server.

The ongoing work of using a centralized browser facility that includes the development, negotiation and individual testing of new browser settings will be out of scope for this SR. The intent here is to provide the facility, methods and procedures.

It is suggested that any further work on creating the browser settings would be part of the ongoing image management tasks under existing SSR's. The magnitude of any individual tasks could vary greatly with the type of changes (i.e. a simple change to the list of recognized internal sites would be small while distributing a new plug-in could be quite large.)

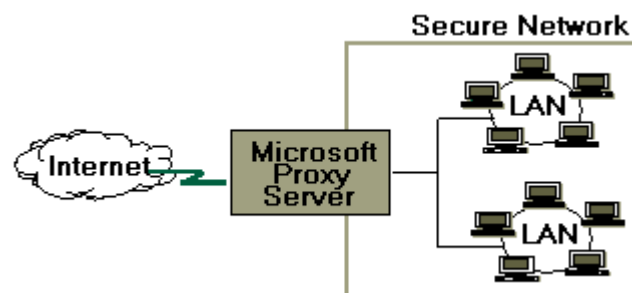
## Project Boundaries & Constraints

This work would illustrate to a large extent the way I would develop the project but the amount of details would usually depend on the following:

- Client's knowledge of the products (in this case Internet explorer, Proxy server and possibly Internet Information Server). This would only be understood by continuous liaisons with the client and possibly a sample of their end users.
- Result of tests coupled with the input from our support team who would be working on the Project.
- The clients existing set-up (in this case say, operating Systems on Servers and workstations, Internet Browsers, as well as software applications used. (This would in part provide the basis for our recommendations).

## Security

In the fast growing market for Internet related products and services, new offerings are made available on an increasingly frequent basis. It is critical to note that there is no one right answer to the question of what constitutes enough security. The answer is often a custom solution that depends on your needs for control and flexibility.



## Limitation of Testing

I have not been able to carry out any tests but many penetration attempts rely on long periods of time to analyze and understand targets, utilize intermediary nodes and networks, and employ automated techniques that may run unattended for extended periods of time. Such long term testing was beyond the scope of my work. In addition, penetration tests provide information regarding only very specific security vulnerabilities and should not be viewed as conclusive.

The results of any evaluation cannot disclose any weakness that may exist in prior or future modifications.

## Disclaimer

Hackers and other criminals who are increasingly sophisticated and resourceful are continuously assaulting Internet security. Just as software virus protection programs are

hard put to remain a small step ahead of the virus perpetrators, security analyses and recommendations may unexpectedly become out-dated. While I believe that this material presents a fair and reasonable picture of Microsoft IE5 in conjunction with Microsoft Proxy Server security controls, nothing herein can substitute for clients own professional and business judgment, independent investigations, and research into the subjects covered. All information presented herein is of a general nature and may not be applicable to any particular situation. Nothing herein constitutes an endorsement of any products by the author.

## Introduction:

The main objective of this White Paper is to have a centralized Browser Support that would enable the Ministry's Systems Administrators better management of the Browser tools

### Assumptions:

Since I have not had the benefit of evaluating The Ministry's existing intranet Framework, I would make a general assumption that it is based on a mix of server side and client code, and is designed to provide benefit to its users, developers and administrators, such as:

- Developers can create sites quickly
- Administrators have a platform from which to extend Intranet services to the ministry
- Users get a standard look and feel

Internet Explorer 5 provides this centralized management by a feature called "Automatic Detection and Automatic configuration of Browser and Server settings."

When the network administrator defines the appropriate settings, the network can automatically detect and configure a browser's settings. This feature works with Dynamic Host Configuration Protocol (DHCP) servers that support the DHCPINFORM message and Domain Name System (DNS) servers.

In the following sections I will be discussing details of this along with Internet Explorer Administration Kit 5 and other features of the Internet Explorer 5 that will illustrate optimized browser settings that could easily be standardized.

## Evaluation of Internet Explorer 5:

I would begin the evaluation by looking at:

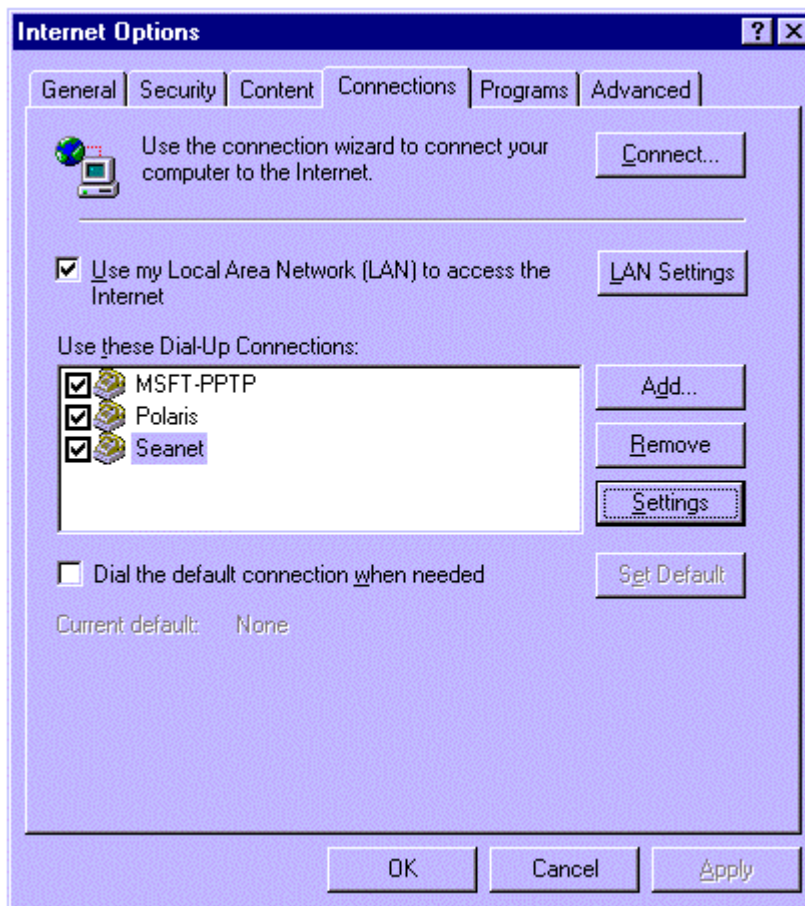
### Setting Up Automatic Configuration in Previously Deployed Browsers

Clearly it is advantageous to use the Internet Explorer Administration Kit to customize Internet Explorer 5 so that automatic configuration is enabled before the browser is deployed. If that is not possible, users will have to perform a simple task to enable automatic configuration. Once these steps are performed, the Internet Explorer Administration Kit Profile Manager can be used to lock these settings down so users will not be able to change them.

To set up Internet Explorer 5 for automatic configuration, users need to do the following:

1. From the Internet Explorer window, choose **Internet Options** on the **Tools** menu.
2. Select the **Connections** property page, then select a connection (such as LAN or a Dial-up).
3. Select either LAN Settings or Settings.
4. Type the *URL automatic configuration file name* in the Auto-

configuration URL section.

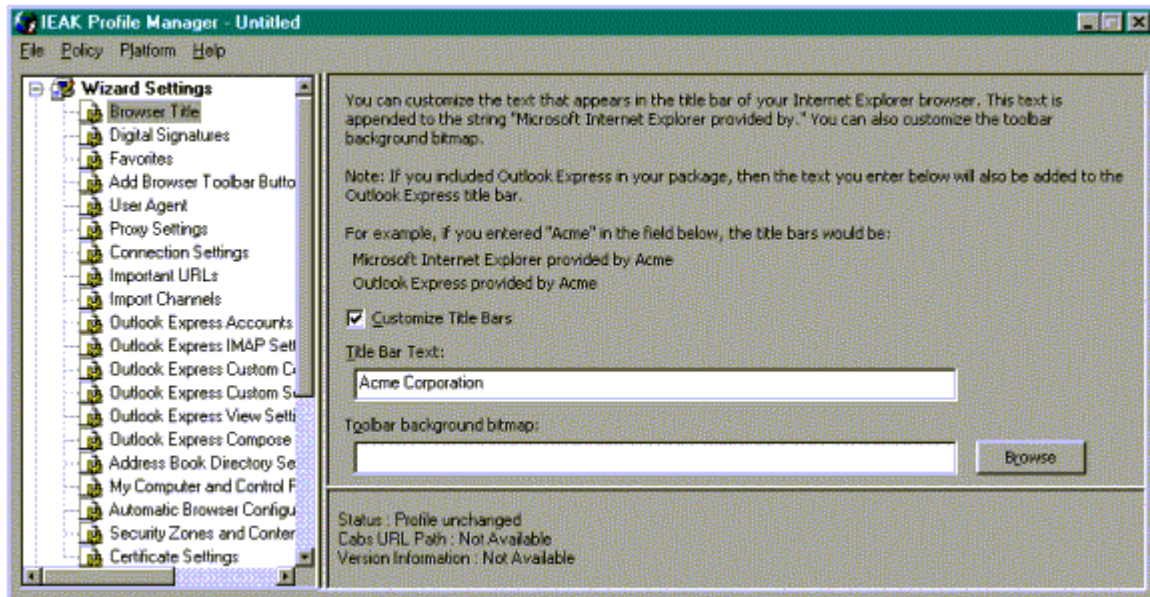


**Connections configuration tab in Internet Explorer.**

**Note** Some automatic configuration changes require the system to be restarted before changes to the system registry will take effect.

## **Managing Users: Internet Explorer Administration Kit Profile Manager**

The Internet Explorer Administration Kit Profile Manager can be used to create and maintain the automatic configuration files. With the Profile Manager, administrators can specify a wide range of browser, user, desktop, and system settings and restrictions. The Internet Explorer Administration Kit Profile Manager is used to maintain Internet Explorer 5 policies over time



Sets of Policies and Restrictions are saved as an .ins file. These .ins files are created into two places for each package during the Internet Explorer Administration Kit Customization wizard. The first location is in the INS subdirectory, for instance, "c:\builds\Corporate\INS\Win32\En\Install.ins." The other location is the CD subdirectory. Different .ins files can be created to reflect different administration and configuration requirements for an organization. Different packages can be created to distribute to each group in an organization, or the .ins file can be manually specified on the browser.

The Profile Manager is organized into a left-hand pane showing a hierarchical tree of objects and a right-hand pane showing the options. After creating a new .ins file or opening one of the install.ins files created by the Internet Explorer Administration Kit, an object in the tree in the left-hand pane is selected and the options and settings for that object appear in the right-hand pane. Administrators can change options or specify settings as necessary to manage automatic browser configurations.

The Internet Explorer Administration Kit Profile Manager provides two categories of settings that can be specified: **Wizard Settings** and **System Policies and Restrictions**.

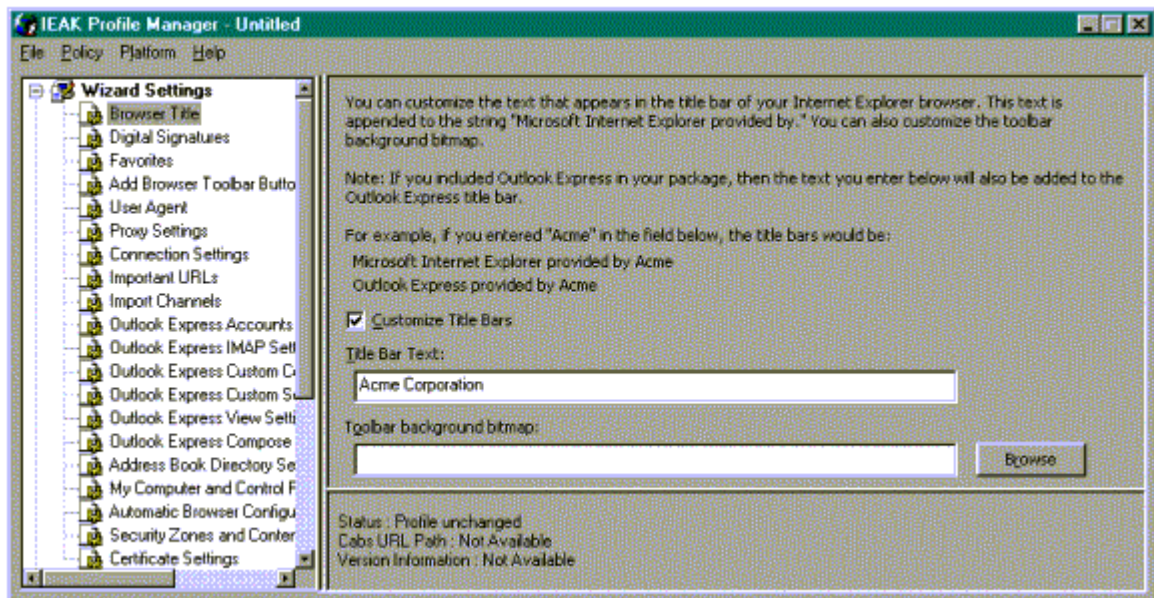
## How Profile Manager Works

The Internet Explorer Administration Kit Profile Manager creates and edits instruction (.ins) files. These .ins files are then specified in the browser under View/Internet Options/Connections/Automatic Configuration. The .ins files can be locally stored, i.e., c:\config.ins, or they can be stored in a Web location, such as <http://www.mywebsite.com/ie5/config/autoconfig.ins>. The .ins file gets checked either at a specified interval, such as every 720 minutes, or every time the browser is started if no interval is specified.

Because the location of this file will have to be specified before it is actually created, it is a good idea to plan out the name(s) of the configuration files and their locations in advance. The Internet Explorer Administration Kit lets administrators pre-configure browsers with the URL of an .ins file, **but this file is not created during the Internet Explorer Administration Kit process**. A default install.ins file is created in the path c:\builds\<build\_number>, but this file should be edited and renamed before distribution. It is usually a good idea to rename .ins files, as the Internet Explorer Administration Kit creates numerous instances of install.ins. Edits made to these files can easily get lost or confused unless the file is renamed. When the .ins file is edited, .cab files may also be created that will need to get copied to the Web distribution location.

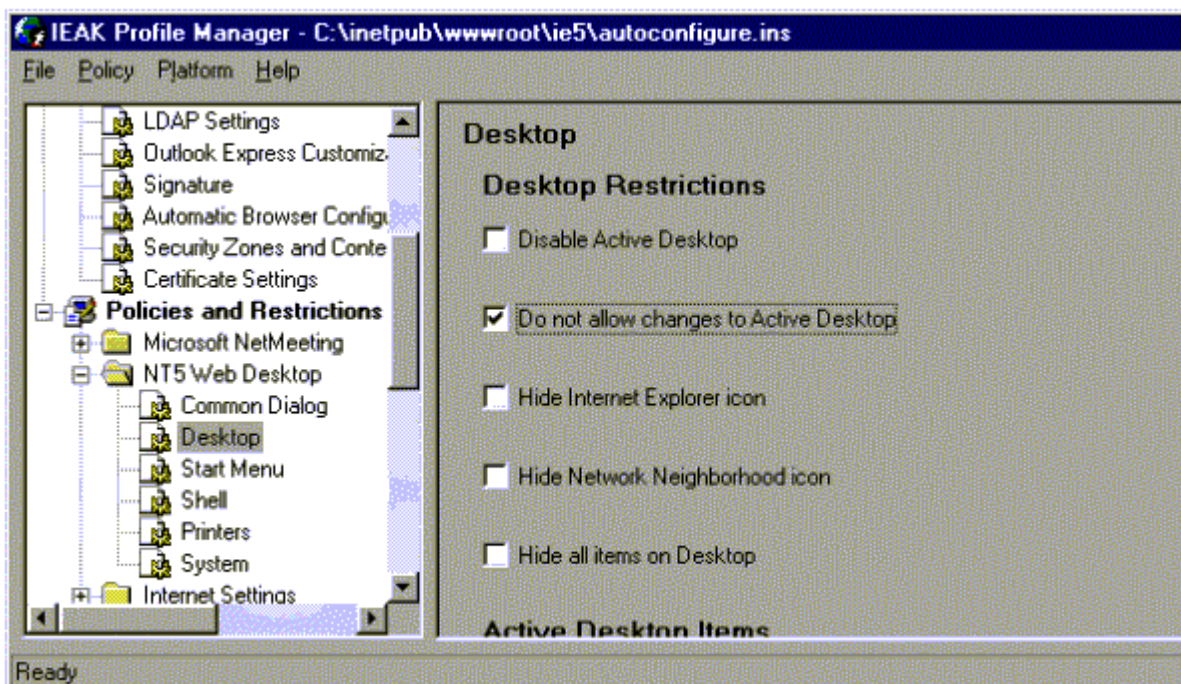
## Changing Wizard Settings

In the left-hand pane, the Wizard Settings object and its sub-objects correspond to settings that can also be specified using the Internet Explorer Administration Kit Customization wizard. An object can be selected and the corresponding options and settings can be changed. For example, the Browser Title object and Browser Title bar and Toolbar background bitmap options can be selected and changed.



## Changing System Policies and Restrictions:

In the left-hand pane, the Systems Policies and Restrictions object and its sub-objects correspond to the same settings available on the Internet Explorer Administration Kit Customization wizard **System Policies and Restrictions** page. Administrators can select an object and change the corresponding policies and restrictions. For example, the Web Desktop tree can be expanded, and the Desktop object can be selected to specify Desktop options such as **Do not allow changes to Active Desktop**.



The **System Policies and Restrictions** folder contains nine default policy template (.adm) files to specify policies and restrictions. These are saved to information (.inf) files, which are packaged into the automatic configuration companion cabinet (.cab) files for download to a user's system. When unpacked, the .inf files are used to change policies and restrictions on users' systems.

The first time either the Internet Explorer Administration Kit Profile Manager or the Internet Explorer Administration Kit Customization wizard is run, it creates the nine default policy template (.adm) files in the C:\Program Files\leak\Policies directory. When changes are saved to system policies and restrictions, the Internet Explorer Administration Kit Policy Manager saves the changes to the .inf files and packages them in cabinet (.cab) files. The two possible cabinet files are branding.cab and desktop.cab. If these .cab files are created, then they need to be copied to the Web distribution point where the automatic configuration .ins file resides.

Administrators can specify desktop, shell, and security settings across an organization. Numerous settings can be customized, ranging from whether users can delete printers to whether they can add items to the **Windows Desktop Update**.

Administrators can control or "lock down" features and functions. For example, use the **System** options under the **Shell** category to prevent Windows 95 users from restarting their systems in MS-DOS® mode, or use the **Security** option under **Internet Properties** to prevent users from changing any of the security settings on the **Security** property page in Internet Explorer. When features are locked down, they either don't appear or they appear in gray type on the user's desktop.

Before changing system policies and restrictions, administrators should understand the impact of the security settings on users, especially if roaming users will be sharing computers with other users. Consider for example, the implications of removing icons from the desktop, or not allowing users to change their security settings. These policies should be clearly communicated to the users before deploying any changes to the organization.

## Using Custom Policy Templates:

Administrators familiar with Windows policy template (.adm) files can create their own templates to define additional restrictions. Policies and restrictions, and policy template (.adm) files are a standard feature of Windows 95 and Windows NT operating systems. For more information, see the Windows documentation.

Administrators can choose **Import** on the **Policy** menu to import custom policy template (.adm) files. Choose **Check Duplicate Keys** on the **Policy** menu to check for duplicate registry keys in the templates and to delete any duplicates from the templates. Always test templates thoroughly in the lab before using them to make changes to users' systems.

When administrators use custom policy templates, the Internet Explorer Administration Kit Profile Manager generates an .inf file using the file prefix for the imported custom template. For example, when custom.adm is imported, a custom.inf file will be generated and added to the companion .cab files.

## Maintaining Automatic Configuration Files

To maintain automatic configuration files, administrators can open the .ins file in the Profile Manager, change settings, and save the .ins file. The Profile Manager keeps the companion files current each time the .ins file is saved.

**Caution:** Files should not be maintained on the production server. Administrators should copy the .ins file) and their companion files from the production location (such as c:\inetpub\wwwroot\ie5 to a working directory, modify the files as necessary, and test automatic configuration in a lab to verify that automatic configuration changes work as

intended. After changes have been validated, update the automatic configuration files on the production server.

## Network Connections Automatically Detect

Internet Explorer can automatically detect the network connections that are available on your computer. If you are not already connected to the network when Internet Explorer 5 is started, the browser evaluates the available connections (for example, LAN, RAS, and ISDN) and automatically uses the highest speed connection available.

## Intelligently Detect Editing Programs

Many productivity applications, such as Microsoft Office, rely on the browser as the primary means of viewing HTML content. In the past, it was difficult to switch back and forth between creating a Web document with the productivity application, viewing the document with the browser, and then editing it with the productivity application. Now, you can perform all of these functions from within the browser window.

When you view an HTML document that was created by using an application on your computer, Internet Explorer recognizes the application based on a metatag in the HTML code. The **Edit** button in Internet Explorer then automatically changes to the application icon. When you click the **Edit** button, Internet Explorer automatically opens the file in the appropriate application.

## Developing and Authoring for the Web

Internet Explorer 5 offers a feature-rich platform for building Web-based applications and developing informative content for users. The browser provides enhanced support for standards-based Internet technologies and improves the ease and speed with which developers can take advantage of these technologies. Improvements to the browser-programming model, such as drag-and-drop capabilities across frames and applications, further enrich the Web development platform.

The following sections describe the new and enhanced features that can help you create Web-based applications.

### Enhanced Support for Dynamic HTML Standards

Enhanced support for Dynamic HTML standards provides you with a comprehensive standards-based platform for authoring interactive content. You can create content that is compatible with other browsers that also support these standards.

Internet Explorer 5 includes enhanced support for the following standards:

- HTML 4.0
- Cascading style sheets (CSS) 1.0 and CSS positioning
- Scripting and support for the Document Object Model (DOM)
- Extensible Markup Language (XML)
- Extensible Stylesheet Language (XSL)
- ECMA-262 scripting (JavaScript standard)
- HTTP

### Enhanced Support for XML

XML provides you with a standard data-interchange format for building data-based applications. Internet Explorer 5 includes the following enhancements to XML support:

- Full support for the W3C XML Document Object Model (DOM) offers extensive programmatic control of XML document content, structure, and formats.
- You can embed XML in a document as data or metadata. The full XML DOM is then exposed for the element, but the elements inside the XML tag are not rendered on the page and are not included in the HTML DOM.

- You can intermix XML tags with HTML tags in a Web document. You can also apply CSS properties to these elements to control how they are displayed.
- Using native XML support, you can view XML like a regular document within the browser.
- With XSL support, you can apply style sheets to native XML documents and display the data dynamically.

## Performance Improvements:

With enhancements to the browser programming model, both developers and users will experience improved performance in the following areas:

- **A more efficient DHTML rendering engine**—Optimization of the DHTML rendering engine (and, in particular, the internal algorithms) has increased the performance of basic browser functions, allowing significantly faster and more efficient display of content than in earlier versions of the browser. These improvements are most noticeable when users are viewing very large documents built with Dynamic HTML and pages that include databinding technology.
- **Fixed-layout tables**—Internet Explorer 5 includes support for fixed-layout tables, which allow you to specify table-column sizes, while the content size inside the table cells does not affect the table layout. By not calculating the minimum and maximum size of each cell in the table, fixed-layout tables are progressively rendered, so the browser displays each table row as it downloads.
- **Render first for the HTTP-Expires header**—Internet Explorer 5 now supports the HTTP-Expires header on both pages and images. The browser does not automatically check the network first when an object from the cache has not yet expired. For a period before the expiration time (specified in the cache control header), Internet Explorer renders from the cache even if a newer version is available. This ability reduces network traffic and improves browser performance.

## Dynamic HTML Behavior Components:

Dynamic HTML behavior components are simple, lightweight components that, when applied to standard HTML elements on a page by using CSS, can enhance the element's default behavior. Additionally, behavior components separate the script from the content on a page, making it easy to reuse code across multiple pages and improving the overall manageability of the page. Just as CSS enables Web-site developers to separate the content of a page from its format, dynamic HTML behavior components extend that idea to separate scripted behaviors as well.

You can now build reusable scripts and custom XML tags that Web pages can reference. For example, you could write a script that specifies which text or pictures fly in from the right side of the page. This script can be referenced from any page simply by calling it. You can then change content without affecting the script.

For example, to build the mask-entry field by using Internet Explorer 4.0, you would have included an **INPUT** tag on the page, then added script that monitored the focus and keyboard interaction with the control. You would have needed to create the script for every page that required the new mask-entry type. Now, with the new Dynamic HTML behavior components, you can encapsulate all the script inside a component that can quickly and easily be referenced from any page, even by someone who has no knowledge of scripting.

## Dynamic Properties

Instead of creating long, complicated scripts to perform relatively simple activities, you can define any property on a page as a function of any other property. This feature can be useful for pages that use CSS positioning. You can now set up very complex screen

layouts that are simple to author, don't require a line of script, and respond to screen changes dynamically.

For example, the font size of a section of text can be set to 20 percent of the width of a table, and the font size will change dynamically if the page is resized without being refreshed from the Web server. As the document changes size, a recalculation engine (similar to technology used in spreadsheet applications, such as Microsoft Excel) determines a dependency and resets the property.

## Full Drag-and-Drop Object Model Support

The Internet Explorer 5 object model now includes full support for drag-and-drop from Web pages to the desktop or any other application. Full control over cursors, drag initialization, and Clipboard support has also been added to the object model. You can build Web-based applications that enable users to drag content between frames and even to other applications.

## Retaining Persistence

Any element on a Web page, such as a collapsible outline, can remain in its current state, even when a user leaves the page and returns later. While in the past you have been able to do this in a limited way with cookies, the new persistence technology of Internet Explorer 5 provides an XML-based method for persisting data.

You can persist form data, dynamic positions and content, styles, and script variables, which increases the speed of navigation and content authoring. For example, you can specify that a collapsible list of links within a table of contents on a page remain in the same expanded state until the user returns to the page. You can preserve documents exactly as they are displayed on the screen rather than relying on settings for a document maintained on a remote server.

Internet Explorer 5 provides a local store for retaining persistence that is protected from unauthorized cross-domain access and is not affected by the 4 KB limit imposed by cookies. Properties can be stored hierarchically by using name and value pair combinations, and you can control the storage and retrieval of this information. By allowing information to safely reside on the client computer, fewer server transactions are required.

## Client Capabilities

The Internet Explorer 5 platform introduces a new feature that enables the server to request the specific capabilities of the client computer. The server creates a special HTTP request (449) that includes a script for the client computer to execute. The script queries the client computer about its system capabilities, as well as the availability of browser features. The client computer executes the script, and then sends the information back to the server. This approach is secure—the client computer executes a script that is similar to any other script the browser can run.

Client capabilities consist of information about the browsing environment, including screen resolution, screen dimensions, color depth, CPU, and connection speed. Internet Explorer 5 also detects the components installed on the system. The server is specific about the capabilities that it needs from the client computer, so only the required information is generated and transferred.

You can customize content to provide the best possible user experience based on this client information. For example, if the user has an Intel® Pentium 266 computer, you might want to provide more sophisticated content; if the user has an Intel 386 computer, you might want to provide less sophisticated content, regardless of the browser used.

## Cache Control

For specific Web pages where you do not want Internet Explorer to cache content, such as the user's password, you can disable the AutoComplete feature. You would add the following tags to a Web page to prevent AutoComplete from storing information:

- pragma:no-cache
- cache-control:no-cache
- cache-control:no-store
- cache-control:private (when not using Windows NT with per user-cache)

## Multiple Cascading Style Sheet (CSS) Class Support

Internet Explorer 5 includes the ability to add multiple CSS classes to an element. You simply apply a list of CSS classes to the element's class property. Any element can accept a list of CSS classes, which makes it much simpler to write script when different actions can occur for a single element. For example, by adding multiple CSS classes to an element, you can easily write the code for a picture that changes when the mouse moves over it and changes again when it is clicked.

## CSS Positioning

You can use CSS positioning to gain more control over the position and layout of elements on your Web pages. You can benefit from the following enhancements to CSS positioning:

- Every HTML element can now be positioned, either absolutely or relatively.
- Elements can be placed on a page with relative positioning, but can revert to absolute positioning on the fly.
- Elements can now change from positioned to non-positioned, or vice-versa, at any time.

## The `currentStyle` Object

Internet Explorer 5 introduces the **currentStyle** object (`element.currentStyle`), which exposes the current value that each element is using for all of its CSS properties (not just the ones that have been explicitly placed on that object). The **currentStyle** object represents the cascaded format and style of the object that is specified by global style sheets, inline styles, and HTML attributes.

Through the **currentStyle** object, cascaded style values of an object can be retrieved. Reading the **currentStyle** object differs from reading the **style** object, because **style** is not set inline on an object. For example, if the color property is set on a paragraph only through a linked or embedded style sheet and not inline, then `object.currentStyle.color` will return the color, whereas `object.style.color` will not return a value. If, however, you specify `<P STYLE="color:red">`, both the **currentStyle** and **style** objects will return a value of red.

The **currentStyle** object reflects the following CSS order of style precedence:

1. Inline styles
2. Style sheet rules
3. Attributes on HTML tags
4. Intrinsic definition of the HTML tag

The **currentStyle** object supports user-defined properties in style rules. It returns values that reflect the applied style settings for the page and may not reflect what is currently rendering at the time a value is retrieved. For example, an object that has "color:red; display:none" will return a `currentStyle.color` of red even though the object is not being rendered on the page. The **currentStyle** object, then, is not affected by the rendering constraints. Disabled style sheets also do not affect **currentStyle** objects.

## Developer Mode

Internet Explorer 5 includes a Developer Mode, which enables developers and site designers to view all scripting and site error messages for debugging purposes. This option is turned off by default for most users who are simply browsing the Web. The error messages in Internet Explorer have also been improved to provide more detailed information about scripting errors, HTML structure errors, and other useful information for diagnosing application errors.

## Compatibility Mode

Internet Explorer 5 has an Internet Explorer 4.0-compatibility mode. Using this special compatibility mode, you can set up a single computer to test your Web sites using the rendering capabilities of both versions of Internet Explorer.

## Browserless Applications:

You can create browserless (.hta) applications that are built using Internet technologies, but that do not run within the browser window. You author an .hta application by using Dynamic HTML and scripting, the same way you author Web pages for Internet Explorer 5. However, an .hta application can run in its own window, which is controlled from corner to corner, instead of running within the browser frame. The .hta applications are not subject to the security constraints imposed on Web pages; like executable (.exe) files, they can run without browser security restrictions.

An .hta application includes extensions for special behaviors and permissions that are not available to HTML pages. You can take any HTML page, set the “application/hta” MIME type, and then run it as an application. To make the HTML page into a full-featured application, you can add a few special .hta application declarations to specify the application icon, window size, window border, system menu, and other settings.

## Integration with Productivity Applications:

Internet Explorer 5 enables you to specify default applications for e-mail, calendar, contacts, Internet dial-up, newsgroups, and HTML editing to use with the browser. When you try to perform a task that requires one of these applications, Internet Explorer automatically starts the selected application. For example, if you click the **Edit** button in Internet Explorer, the browser opens the Web page in the selected HTML editing application.

The Internet Explorer user interface and commands are also more consistent with Microsoft productivity applications, such as Microsoft Office 2000.

The following user-interface enhancements provide additional consistency between the browser and productivity applications:

- Tools menu**—More advanced commands, such as **Internet Options**, are now available on the new **Tools** menu, just as they are in Office applications.
- Toolbar appearance**—You can select smaller toolbar buttons and icons for the browser, similar to those used in Office applications.
- Toolbar commands**—You can add popular Office commands, such as **Print**, **Cut**, **Copy**, and **Paste**, to the browser toolbar.

## Per-Connection Proxy Settings

With Internet Explorer 5, you can set up multiple types of connections, each with different proxy servers, and switch automatically from one connection to another. If you need

multiple Internet and intranet connections (for example, a connection to your office LAN and a connection to your personal Internet account through an Internet service provider), you no longer have to change your proxy-server settings manually each time a different connection is made. When you choose a different connection, Internet Explorer automatically uses the appropriate proxy-server settings.

## Conclusions:

Internet Explorer 5 can play a key role in the Ministry's information technology infrastructure. The Ministry could create an implementation of its key Intranet applications to take advantage of the new features and capabilities of Internet Explorer 5.

By moving to Internet Explorer 5 the ministry could substantially reduce Intranet application development and maintenance time and costs and could enable its users to move quickly and easily to access business-critical information services

The Ministry's developers could initially achieve a substantial reduction in development time due to the ability to reuse Dynamic HTML (DHTML) behaviors in Internet Explorer 5. They could also improve their productivity by utilizing the browser's capabilities and their IT administrators would be able to reduce the costs of deploying and administering the browser.

Internet Explorer 5. They could also reduce the size of their Intranet applications because of the ability to integrate XML tags based on DHTML behaviors in place of additional code. Because the Ministry's developers do not have to reinstate dynamic scripting / HTML functionality in updated Internet Explorer 5 Intranet applications, they could achieve a sizeable reduction in the time spent developing updated applications.

The Ministry's developers would rate Internet Explorer 5 higher than the previously used browsers in the ease of creating customized extensions, and in its application integration. They would see that IE 5 is more extensible and would provide far richer application integration than their existing browser does.

Installation costs would also come down because administrators could control how IE 5 installation will interact with each client computer.

By using the Internet Explorer Administration kit 5, the Ministry would benefit, often substantially, for tasks such as adding a component automatically, changing proxy servers, adding security certificates, updating the content of browsers, and using the Wizard. The ability to customize the browser's tool bar and components with the Wizard could result in savings in administrative customization effort, while the ability to lock down changes to the browser in IE 5 could result in reduction in the administrative efforts for this function.

Finally the browser is a strategic tool and the Ministry could use Internet Explorer 5 to provide business-critical information to users and the their developers could

## Appendix A: Looking Ahead with Web-Based Network Management

Web-based network management is the ability to monitor and actively manage a network, regardless of the location of the network or the network manager, using the Internet as an access vehicle.

A practical definition might be that Web-based management is any system that allows an administrator to manage the same things that are usually handled from the comfort of a Web browser inside of a data center.

Web-based system is a system that allows the management Information Systems (MIS) department to leverage Web technology to extend the function of its network management system. It lets an administrator to extend the existing monitoring, configuration, and troubleshooting functions of the corporate network by taking advantage of the natural efficiencies of the browser.

There are at least four good reasons for any network manager to look into Web-based systems:

### **1: Ease of access**

Web-based systems provide the ability to get to key management sites or components from anywhere in the world where one can hookup a laptop, so it is possible to carry a network management access node along under one's arm.

### **2: Platform Freedom**

there is no need to scatter sun workstations or X-terminals in locations that someone considers to be strategic sites, the goal is to bring network access to the user rather than force the user to move or travel to the network access mode, instead Web-based management systems generally provide access once a user finds a computer link.

### **3: Ease of use**

Simply by adding a bookmark to a computer browser, a network administrator can hop to the management component of the system under consideration. Almost every Internet tool contains a spot to list bookmarks or favorite places. These are the addresses that a user frequently accesses

### **4: Cost Savings:**

There are real advantages to being able to see a problem before actually heading off to the user's site to fix it. Clicking on a browser and being transported across the internet to the device's management screen allows an administrator the current situation of the device before making any attempts to repair it.

**In summary**, the Internet is evolving into the new public network, and those who take advantage of this trend could achieve a fundamental business head start. For the Internet to progress from its strength in connectivity and become a fundamentally reliable in infrastructure, advances will have to come both from business and technology leaders to solve the problems involved in growing networks at unprecedented rates. Though the Internet is recent, the phenomenon of service adoption of communication services is not, and as users reach critical mass, the service must have a transition from accommodating early adopters to mainstream users.

## **Appendix B: Bibliography**

- 1: TechNet (November 1999): Internet Explorer Technical notes.
- 2: TechNet (November 1999): Internet Explorer 5 Resource Kit.
- 3: Web-based Network Management by Kurt Harler, published by Wiley Computer publishing ISBN 0-471-32739-5