



Internet Explorer 7: Common Deployment Issues

“Tips and Tricks”

Draft 1.1

April 22, 2008

For the latest information, please see <http://www.microsoft.com/ie>

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, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT.

Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Microsoft Corporation.

Microsoft may have patents, patent applications, trademarks, copyrights, or other intellectual property rights covering subject matter in this document. Except as expressly provided in any written license agreement from Microsoft, the furnishing of this document does not give you any license to these patents, trademarks, copyrights, or other intellectual property.

© 2008 Microsoft Corporation. All rights reserved.

Microsoft, Windows, Windows XP, Windows Vista, Internet Explorer, IE7, the Windows logo, and the Internet Explorer logo are trademarks of the Microsoft group of companies.

All other trademarks are property of their respective owners.

The purpose of this document is to provide IT Professionals and Administrators with a quick solutions guide for commonly experienced problems when deploying Internet Explorer 7. This is not an exhaustive review of any and all potential issues. This document will focus on several of the most commonly observed technical issues and provide guidance for workarounds or problem resolutions.

Contents

I can't seem to get pages to come up or certain things aren't working quite right	5
I'm seeing messages and prompts about installing ActiveX controls.....	8
I'm having design and layout issues with Internet Explorer 7	13
I'm getting new certificate warnings and other security messages that I've never seen before.....	16
I'm having issue with our application trying to write to the local file system in Vista	19
I'm not even sure where to start figuring out what works and what doesn't.....	19
I'm having installation/deployment issues	21

User-Agent String Detection Errors

Many Web designers use browser-detection techniques to customize the experience for specific browser versions. It is important to ensure that the techniques chosen take into account future browser releases. Several typical browser detection techniques fail when they encounter Windows Internet Explorer 7. For example, some sites that look great when viewed in Internet Explorer 6 might display errors when viewed in Internet Explorer 7. This is not caused by any problems with Internet Explorer 7; it happens because the browser detection code (either on purpose or by accident) is not designed for newer versions of the browser.

In addition to the solutions addressed here, Microsoft recommends using feature detection, where the website logic detects if a browser is capable of supporting a specific feature. By using feature detection, website owners can ensure the experience they want their users to have, regardless of browser version and client upgrades. In the enterprise environment this can be especially helpful as well, where the need to support legacy applications makes it critical to ensure application compatibility. Feature detection logic could help enterprise applications age gracefully while they remain current in a parallel application development cycle.

Question:

I am trying to reach my website, but instead of loading the normal page I keep getting a message telling me to upgrade my browser version. I am using IE 7, so why do I see this error?

Answer: This is generally an issue with application logic on the server or code within a webpage that does not recognize the new 'MSIE 7.0' version number used in Internet Explorer 7. Prior to the release of IE 7 most web development practices accounted for versions up to, and including, 6.0. The logical operators they used did not account for future versions, and an unexpected result higher than 6 is causing the application to show an error.

Solution: Depending on where the application logic is being performed (in the server or within a webpage), that code would need to be modified to recognize the 'MSIE 7.0' string and accept it as a valid version number. Regardless of where the fix resides, it's a trivial bit of code to update and won't cause regression problems. If the application logic is contained within a webpage, it can be seen by right clicking in the browser window and selecting 'View Source'. If you do not see some kind of JavaScript browser version checking code in the page, the logic is most likely contained in a server side application. One common example of web page based browser checking logic is:

```
if ( ver = 6.0 )  
    msg = "You're using a recent copy of Internet Explorer."  
else  
    msg = "You should upgrade your copy of Internet Explorer.";
```

The problem with this version checking logic is that it specifically matches a version number, rather than looking for a minimum value. While this code is not incorrect, it causes unnecessary issues (and unintended errors) with newer browser versions. Simply changing the code to:

```
if ( ver >= 6.0 )  
    msg = "You're using a recent copy of Internet Explorer."  
else  
    msg = "You should upgrade your copy of Internet Explorer.";
```

fixes the issue and accommodates for any value greater than 6.0. By adding the > operator, the code will now be able to handle any future version values for IE.

Identifying application logic issues for server side applications requires looking for similarly written conditional statements. Given the vast number of possible server side application platforms and languages it is impractical to provide every server side example in this document. Microsoft has published a best practices document for effectively detecting Internet Explorer (covering both webpage and server side application logic), which is located at <http://msdn2.microsoft.com/en-us/library/ms537509.aspx>.

Question:

My webpage loads and the content renders, but some of the functionality (e.g. menus, buttons, rollovers, etc.) is not working. Is there something wrong with using IE7?

Answer: This issue is generally tied to browser identification logic errors within a webpage where the JavaScript code is not designed to properly handle the version of Internet Explorer you are using. There is nothing wrong with using IE7, nor is anything specifically wrong with the web application.

Solution: Since the webpage renders in the browser window, it appears the browser detection logic is likely happening in some JavaScript within the HTML content. In order to view the HTML source for the webpage (including the JavaScript programming code) simply right click in the browser window and select 'View Source'. In looking at the source code you should focus on content contained within <script></script> tags, especially looking for JavaScript variables that reference the *navigator* object. For example, you might see JavaScript code like:

```
this.ver=navigator.appVersion;  
this.agent=navigator.userAgent;
```

These lines are working with elements of the Document Object Model (DOM) that allow web developers to read information about the user and take actions based on the results. If you do not see any *navigator* object references in the webpage source code, you will need to find where the source code is being referenced. Some web developers prefer to use externally sourced JavaScript files rather than include all their code in one HTML page. For those situations you may need to load the URL indicated and open that .js file in a text editor. To find those files you would need to search for HTML tags that look like:

```
<script language="JavaScript1.2" src="inc/mycode.js">
```

Once you have found the JavaScript code relating to the browser logic activity, you need to look for instances where the code references versions of Internet Explorer. Specifically, you need to find any

items where the web developer may have limited the conditional operator to exclude new versions (or at least those beyond Internet Explorer 6). For example, you may see code which looks like this:

```
this.ie5=(this.ver.indexOf("MSIE 5")>-1 && this.dom)?1:0;
this.ie6=(this.ver.indexOf("MSIE 6")>-1 && this.dom)?1:0;
this.ie4=(document.all && !this.dom)?1:0;
this.ie=this.ie4 || this.ie5 || this.ie6
```

You can see that this JavaScript code has conditions to deal with versions up to 6, but in the case of IE 7 the code will simply exit without returning a result. Often this type of exit condition will cause a webpage to render properly – there is nothing wrong with the HTML or JavaScript code – but the functionality will be broken and you will not be able to use the application. The fix for this situation is to modify the JavaScript to include conditions which properly handle the IE 7 (and IE 8) version string. By modifying the code to look like:

```
this.ie5=(this.ver.indexOf("MSIE 5")>-1 && this.dom)?1:0;
this.ie6=(this.ver.indexOf("MSIE 6")>-1 && this.dom)?1:0;
this.ie7=(this.ver.indexOf("MSIE 7")>-1 && this.dom)?1:0;
this.ie7=(this.ver.indexOf("MSIE 8")>-1 && this.dom)?1:0;
this.ie4=(document.all && !this.dom)?1:0;
this.ie=this.ie4 || this.ie5 || this.ie6 || this.ie7 || this.ie8
```

You should immediately see the proper functionality with IE 7 that you experienced with IE 6.

Question:

I have too many applications to test in my organization, is there a quick way I can test the User-Agent string issue without digging through any code?

Answer: Yes, Microsoft recognized the need to have a simple solution for broad based web application testing.

Solution: Microsoft offers the User-Agent String Utility for system administrators, web developers, application owners and any other users to quickly and easily modify their browser settings to send the Internet Explorer 6 User-Agent string values. This tool, available at <http://www.microsoft.com/downloads/details.aspx?familyid=9517db9c-3c0d-47fe-bd04-fad82a9aac9f&displaylang=en>, enables you to launch an instance of Internet Explorer 7 that will send the modified User-Agent string (it will send the IE 6 UA String). You do not need to modify the system registry or make any other system modifications to use this tool. It is meant as a diagnostic tool to help you in evaluating deployment preparedness, and therefore does not permanently change any system values. If you find the testing is successful when using this tool, but your web applications fail to work properly when sending the Internet Explorer 7 User-Agent string, Microsoft recommends that you modify any applications to properly accommodate for the Internet Explorer 7 User-Agent string value. In the case of Commercial Off The Shelf (COTS), Enterprise Resource Planning, or Customer Relationship Management (CRM) software applications you should contact those vendors for availability of any upgrades or patches to their applications.

Microsoft does not recommend using this tool as an enterprise deployment solution, it is meant for

testing and diagnostic purposes only. See the [registry based solution](#) for a suitable enterprise deployment and distribution solution.

Question:

I've checked out my applications and they work if I set the User-Agent string to version 6, but I need to deploy this to my entire organization (and the software vendor/developer does not offer a patch or upgrade to their application). How do I change it on every machine? And will it break anything else?

Answer: Making changes to the web application code is the recommended solution, but there can be logistical reasons that make it difficult or impossible. Sometimes the source code is not available (in the case of server side applications) or exists in too many places (in the case of embedded code in HTML pages) to be a viable solution in your environment. There are also many older (legacy) applications used in an organization and some applications vendors have chosen to only support the newest versions of their web based applications, and upgrades are either impractical or impossible. To facilitate migrations to Internet Explorer 7, you can modify system settings in the registry to force Internet Explorer 7 to send the Internet Explorer 6 User-Agent string. Fortunately, making this change will only skew the web statistic usage reports gathered by website administrators – it will not break any functionality of Internet Explorer 7, nor will it reduce any of the feature and security enhancements available to Internet Explorer 7 users.

Solution: The registry key is located at:

```
[HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\5.0\User Agent]
```

Under that key you will need to make the following entry:

```
"Version"="MSIE 6.0"
```

These changes can be made using Active Directory configuration changes or you can use login scripts to run a registry file update for installations that do not have AD infrastructures. The registry file needs to contain both of the lines above and have a .reg extension for it to properly import to the system registry. This registry file is supported on both Windows XP and Windows Vista. Administrator rights are required to install registry files, or they can be deployed via login scripting.

Microsoft recommends using this registry fix as a temporary solution while working to make your web applications compatible with a standard installation of Internet Explorer 7. When your web applications and software vendors provide compatible versions of their software, you can safely remove this registry modification and not lose any functionality.

ActiveX Controls

Internet Explorer 7 introduced a new feature designed to help prevent users from being potentially and unknowingly exposed to malicious attacks. Internet Explorer has long offered web developers the ability to build custom components and extensibility solutions using the ActiveX platform. As with any extensibility model, the more you allow people to do and the more you expose the users system, the more potential for abuse. To counter that potential issue, IE7 introduced the

ActiveX opt-in feature, which disables all previously unused controls by default. This helps in both case of upgrades and new installs, since ActiveX opt-in protects the user against controls that are installed along with non-web enabled applications. Prior to IE 7, those controls would be exposed to the Internet by default. Now the user is protected, and must give their approval before the control can be invoked. Any ActiveX control installed on the system must be on the approved list before it can be instantiated in the browser. A control which has never been used before (or previously called but not approved) will trigger IE7 to display the information bar to prompt the user to approve the control.

Question:

I am testing with Internet Explorer 7 and I keep getting prompted by the information bar to run an ActiveX control, why am I getting this warning?

Answer: One of the security improvements made in Internet Explorer 7 is the addition of a feature called 'ActiveX Opt-In'. The purpose of this feature was to restrict the use of ActiveX controls within Internet Explorer to those specifically intended by the user (or administrator). Unlike previous versions of Internet Explorer, users will now be prompted to run an ActiveX control if they have never used it before. Once approved, the user will never be prompted again for that specific control.

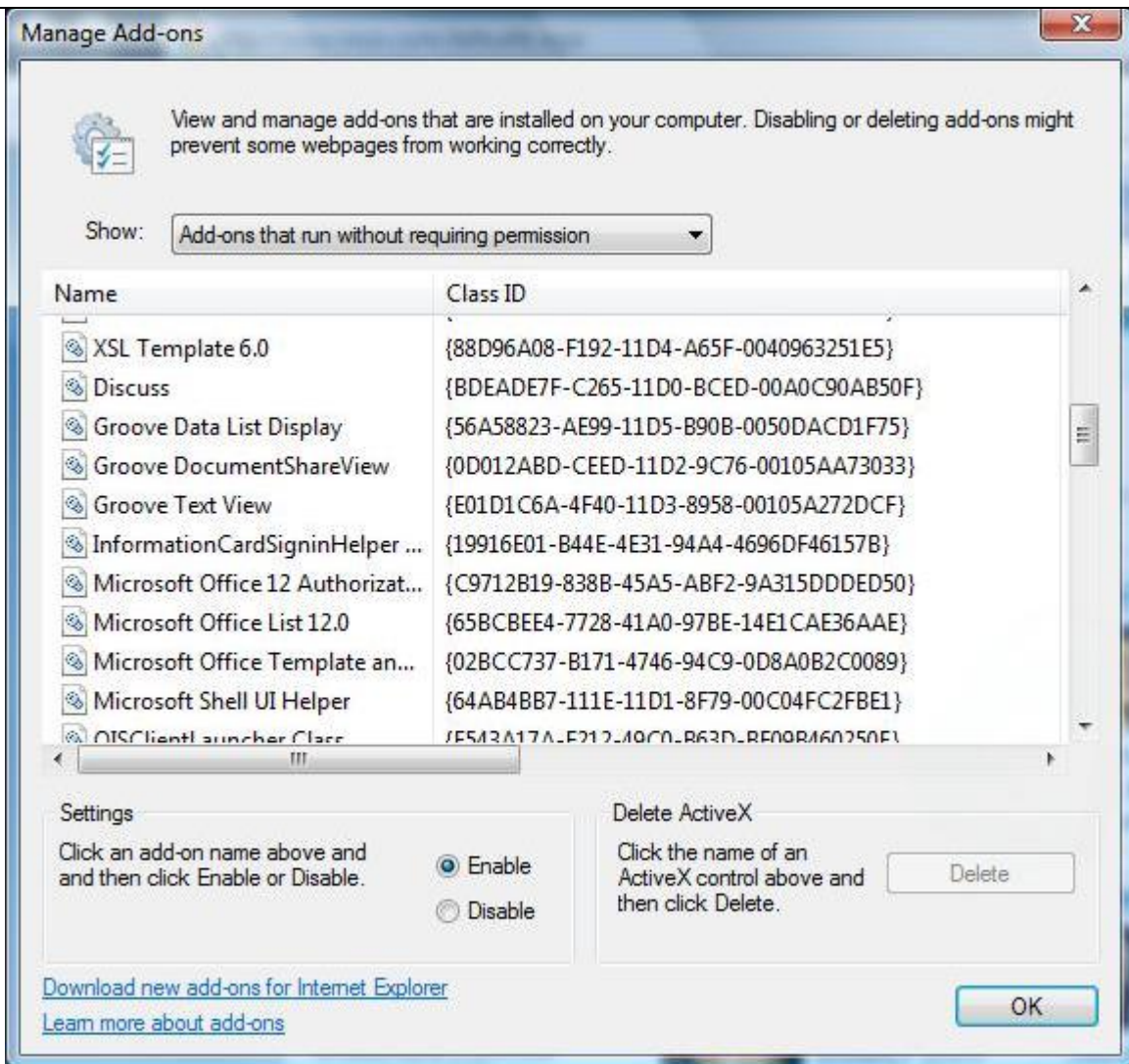
Solution: To best address this issue and provide for system stability and security in your enterprise, Microsoft recommends you create an internal list of approved ActiveX controls (both public and private in the case of internally created ActiveX controls). Centrally managing this list of approved controls will help ensure users are not blocked from certain actions and need helpdesk support to install/approve ActiveX controls for locked down systems. The Group Policy setting is found at:

Computer Configuration -> Administrative Templates -> Windows Components -> Internet Explorer -> Security Features -> Add-on Management -> Add-on List

Enter all approved ActiveX controls (listed by CLSID) in this object container. You need to enter two items, Name and Value. The Name is the CLSID and the Value needs to be one of:

- 0 – Deny this add-on
- 1 – Enable this add-on
- 2 – Permit user to use Manage Add-ons

To quickly and easily find your complete list of CLSIDs, once you have added all approved controls to a machine, open the Manage Add-ons dialog and add the 'Class ID' column.



Once this list is complete, you can use Group Policy to create and propagate the approved list to systems in your AD domain. If you want to lock down this list of controls to only those you have approved for your organization, 'enable' the Group Policy Setting at:

Computer Configuration -> Administrative Templates -> Windows Components -> Internet Explorer -> Security Features -> Add-on Management -> Deny all add-ons unless specifically allowed in the Add-on list

To provide additional control over the user experience, and remove the information bar notifications entirely, you can 'disable' the Group Policy Object found at:

Computer Configuration -> Administrative Templates -> Windows Components -> Internet Explorer -> Security Features -> Information Bar -> Internet Explorer Processes

Question:

Internet Explorer 7 is blocking users from running critical ActiveX controls so they can't use our web applications. How can I unblock those controls?

Answer: By default, Internet Explorer 7 is designed to block all previously unused ActiveX controls. This change was made to increase overall security and limit potential system exposure due to ActiveX controls that were never intended to be exposed to the Internet. Certain well known and tested ActiveX controls are 'pre-approved' and will not prompt users even if they have never previously used the ActiveX control. More information about the ActiveX Opt-In feature can be found at <http://msdn2.microsoft.com/en-us/library/bb250471.aspx>.

Solution: You can use Group Policy to specifically approve certain controls, and/or block all but those approved for your organization. The Group Policy setting for managing the list is found at:

Computer Configuration -> Administrative Templates -> Windows Components -> Internet Explorer -> Security Features -> Add-on Management -> Add-on List

Once your list is complete, you can use Group Policy to create and propagate the approved list to systems in your AD domain. If you want to lock down this list of controls to only those you have approved for your organization, 'enable' the Group Policy Setting at:

Computer Configuration -> Administrative Templates -> Windows Components -> Internet Explorer -> Security Features -> Add-on Management -> Deny all add-ons unless specifically allowed in the Add-on list

In addition, users running Internet Explorer 7 on Windows Vista can take advantage of the ActiveX Installation Service to install pre-approved controls without the need for administrator privileges. More information about the ActiveX Installation Service can be found at <http://blogs.msdn.com/uac/archive/2006/09/13/752248.aspx>.

Question:

I'm running the 64 bit version of IE and it keeps crashing when I go to various websites, what's the problem with Internet Explorer?

Answer: You are most likely seeing an issue where Internet Explorer is experiencing a situation with add-ons that are not 64 bit compatible and Internet Explorer is unable to properly load and use the 32 bit versions.

Solution: At this time we are working directly with many of the top add-on software vendors to make their ActiveX controls available in 64 bit versions, but most have not yet made version of their applications available. Until they make their software available, you would need to run Internet Explorer in 32 bit mode in order to access sites which use those controls. For browsing sites that do not use those controls you can continue to run the standard 64 bit version and take advantage of the increased speed and performance of that version.

Question:

We are testing Internet Explorer 7 on both Windows Vista and Windows XP SP2. We get a red 'X' icon when trying to use Outlook Web Access with Internet Explorer 7 on Windows Vista, but Windows XP SP 2 works fine. What's wrong with Windows Vista?

Answer: There is nothing wrong with Windows Vista, or Internet Explorer 7. Among many of the changes made in Windows Vista, the DHTML editing control was deprecated. In that process, any websites which made use of the control will no longer be able to access it. Outlook Web Access is one popular application that used the DHTML control.

Solution: For Windows Vista users the only solution is to modify the control which the web application is attempting to use. The Exchange Server team issued a patch (Knowledge Base article 911829 <http://support.microsoft.com/kb/911829>) to resolve this issue. Applying this patch will resolve the issue and restore proper functionality for all users. You will need to contact your other software vendors to inquire about and obtain their software fixes if they use this deprecated control.

Be advised this control was deprecated for security purposes and Microsoft recommends it should not be installed on Windows Vista machines as a workaround to this issue. The only recommended workaround is to install the Exchange Server patch, there is no client side fix for this issue.

Question:

Our systems create PDF documents and we keep having issues loading a document, closing the tab and opening a new document. After we close the first one, we keep getting blank pages. What's wrong with our installation?

Answer: There is nothing wrong with your installation. We are aware of this issue being reported by some users, and are working with Adobe to provide an updated ActiveX control to resolve the issue.

Solution: The issue appears to be caused when the Adobe Acrobat ActiveX control fails to properly register and react to a 'tab close' event. The currently available control was written before Internet Explorer supported a tab browsing interface, so the control is expecting to close along with the connected iexplore.exe process. If any tabs remain open when the tab with PDF document closes, the Adobe Acrobat process 'AcroRd32.exe' remains running. Since the process is not properly terminated, it will not launch again for the next PDF document load request. There are two options for workarounds until Adobe provides an updated client solution:

- 1) When users encounter issues with loading PDF documents and only see a blank page, they can manually terminate the 'AcroRd32.exe' process. Once the process is terminated they need to re-access the desired page/report and Adobe Acrobat should load properly.
- 2) Open any PDF documents in a new window.
 - a. This can be done programmatically by modifying your web application to force a new window for each PDF document. Using the JavaScript 'window.open' command is one solution.
 - b. Users can manually force this type of action by holding down the SHIFT key when clicking on a PDF document link (or report generation link). An alternative technique would be to right click, then select 'Open in New Window'.

Cascading Style Sheets

One major area of improvements in Internet Explorer 7 is better adoption and adherence to web standards. As a result of those changes, some web design layouts no longer appear to render properly. Many of the required workarounds for previous versions of Internet Explorer are no longer needed, enabling web designers to spend less time on browser customization and more time on the actual design elements.

Question:

In testing with Internet Explorer 7, our pages don't render (layout) properly. Why do things look so funny (or why are things in the wrong place)?

Answer: Internet Explorer 7 introduced many standards compatibility improvements to help web designers and developers create common experiences across platforms. Prior versions of Internet Explorer did not properly interpret certain aspects of the HTML DOM and W3C specifications. As a result, web designers and developers devised certain 'hacks' specifically for Internet Explorer users to ensure proper layout. With the development work made in Internet Explorer 7 most of those modifications are no longer necessary. Further, Internet Explorer 7 attempts to render (layout) the CSS source material as written, causing unexpected visual outcomes.

Solution: The work performed in development of Internet Explorer 7 focused heavily on fixing most of the well known problems – including:

- Expanding Box Problem
- Three Pixel Jog
- Double Float-Margin
- Unscrollable Content

More information on Internet Explorer 7 CSS changes and workarounds can be found at <http://msdn2.microsoft.com/en-us/library/bb250496.aspx>. The Application Compatibility Toolkit (available at <http://technet.microsoft.com/en-us/windowstvsta/aa905078.aspx>) enables you to browse sites and view compatibility issues in real time. Compatibility issues are logged and contain information about the type of error and information about behavior changes with Internet Explorer 7, as well as suggestions for necessary changes to avoid compatibility issues. This document also contains more information about the [Application Compatibility Toolkit](#) and Internet Explorer 7.

Question:

Many of our critical internal websites don't work/look right, or the hyperlink areas aren't where we expect them to be. What is causing this?

Answer: This could be in part due to CSS and User-Agent string changes made in Internet Explorer 7. Application servers or HTML code could be improperly rendered in Internet Explorer 7 if the application logic is not properly handling the User-Agent string setting. More information on that issue can be found [here](#). If the User-Agent string issue is not the problem, the issue may be related to standards compatibility improvements made in Internet Explorer 7. Many of the CSS related enhancements made in Internet Explorer 7 no longer require the use of CSS 'hacks' that were needed to properly render content in previous versions.

Solution: Microsoft believes strongly in the need for application compatibility for our users and enterprise customers. It is essential to help ensure legacy applications continue to function as routinely as possible, while continuing to advance browser capabilities. Microsoft implemented two solutions, to help address this issue:

Conditional Comments

[Document Declaration Type \(DOCTYPE\) modes](#) [see next section]

Conditional Comments allow web designers and developers to add code in their HTML content that will only be read (and rendered) by Internet Explorer browsers. For example:

```
<![if IE]>  
SOME HTML CODE  
<![endif]>
```

In addition, Conditional Comments support browser version numbering, so you are able to define precise formatting controls for a specific browser target. For example:

Internet Explorer 6 only comment:

```
<![if IE 6]>  
SOME HTML CODE  
<![endif]>
```

Internet Explorer 7 only comment:

```
<![if IE 7]>  
SOME HTML CODE  
<![endif]>
```

To resolve some layout issues, you should examine HTML code for the use of Conditional Comments, looking for anything which does not identify a specific Internet Explorer version. If you find the use of untargeted comments, you can modify the Conditional Comment to exclude Internet Explorer 7 and re-test the page. To modify a Conditional Comment to exclude Internet Explorer 7 (and future versions), the comment would need to appear as:

```
<![if lte IE 7]>  
SOME HTML CODE  
<![endif]>
```

More information and details on Conditional Comments can be found at

<http://msdn2.microsoft.com/en-us/library/ms537512.aspx>.

For information about DOCTYPE issues, [see this section](#).

Question:

We're having layout issues with Internet Explorer 7 and I've heard about the CSS changes and different DOCTYPE modes ('quirks' and 'standard'). What do these modes mean to me?

Answer: To help ensure application compatibility across versions, Internet Explorer utilizes the DOCTYPE mode to define a set of 'rules' by which to render a webpage. More information and details on DOCTYPE and how to set the modes for Internet Explorer can be found at <http://msdn2.microsoft.com/en-us/library/ms535242.aspx>. Internet Explorer supports both 'quirks' and 'standards' modes as defined by the DOCTYPE declared in the HTML source. In 'quirks' mode, Internet Explorer 7 has been designed to ensure compatibility so that existing content continues to render as it

did in previous versions of Internet Explorer. By contrast, in 'standards' mode Internet Explorer 7 is designed to work towards improved conformance with W3C recommendations, which may include changes that impact existing content.

Solution: Microsoft continues to make investments for better adherence to web design standards, and recommends using 'standards' mode wherever possible. By using 'standards' mode, your applications will be better designed to work properly with future versions of Internet Explorer. As a last resort for application compatibility issues, you should select the DOCTYPE options that work best for your environment.

Question:

Our layout issues are resolved if we use the 'quirks' mode DOCTYPE. Can we force Internet Explorer 7 to always use 'quirks' mode?

Answer: No. Unfortunately, the design of the DOCTYPE modes are made such that the browser receives the DOCTYPE information in the HTML payload from the server and renders based on that information. There is no mechanism to force Internet Explorer 7 (or any version) to ignore the DOCTYPE mode information.

Solution: While there is no way to modify Internet Explorer 7 settings to always use the 'quirks' mode DOCTYPE, it may be possible to modify your web applications to always specify the use of 'quirks' mode. This is accomplished by modifying web server configurations to inject the following header in all HTML content:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
```

Using this DOCTYPE will force Internet Explorer to use 'quirks' mode to render pages, maintaining the expected content display.

If your pages already contain DOCTYPE declarations and it is impractical to modify that content, there is another option. Due to a legacy issue with Internet Explorer, the browser will enter 'quirks' mode if there is any NON-WHITE space prior to the DOCTYPE declaration. Using this technique you can inject a single line (using web server header additions) and leave content pages untouched. An example of text to trigger 'quirks' mode:

```
<!--use this to trigger quirks mode for IE7-->
```

Simply injecting that immediately before the DOCTYPE declaration will trigger Internet Explorer 7 to use 'quirks' mode. As this configuration change is done at the web server level, you may have the ability to granularly control which applications and pages utilize this header feature. This technique will also work *even if* the DOCTYPE declared by the HTML page is specifically set to use 'standards' mode.

Question:

Our page content is in all the wrong places (DIV layers overlapping and in the wrong place) and I can't figure out where boxes start and end. Is there a tool to let me get more detail on how Internet Explorer 7 wants to render my page?

Answer: Yes, Microsoft created the Internet Explorer Developer Toolbar to help web developers and designers better work with the DOM and CSS content in Internet Explorer 7.

Solution: The Developer Toolbar can be downloaded from <http://www.microsoft.com/downloads/details.aspx?familyid=e59c3964-672d-4511-bb3e-2d5e1db91038&displaylang=en>. Once installed, you can enable it under the Tools menu. The Developer Toolbar is designed to be used for real time review, analysis and viewing of HTML and CSS page content. One useful option offered by this tool is the ability to outline by DIV layer, which provides a visual guide to layout issues and enables designers and developers to quickly see where Internet Explorer 7 is placing content. In addition, the Developer Toolbar enables you to drill down into specific content and parameter details so you can review the values while looking at rendered content.

Security Changes

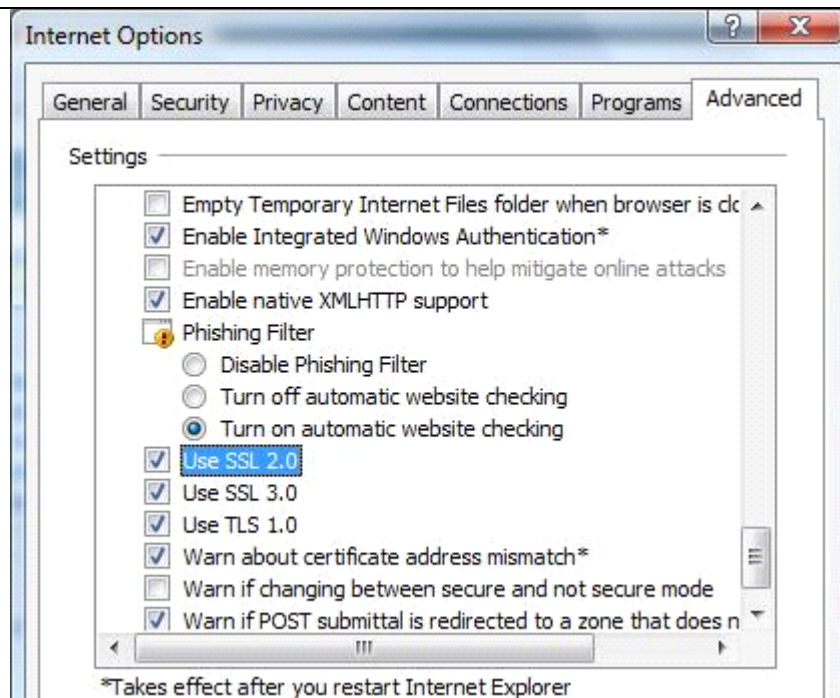
Internet Explorer 7 introduced many new security features and default configurations in order to enhance browser security increase user information safety. Some of those changes affect the way in which Internet Explorer works and may cause issues for enterprise deployments.

Question:

With Internet Explorer 7 we get authentication failure or connection failure errors when trying to access certain HTTPS sites. These sites worked fine before, does Internet Explorer 7 not work with HTTPS sites?

Answer: Internet Explorer 7 modified the default settings for HTTPS connections to remove default support for SSL v2 certificates. This change was made in order to help increase the overall security of HTTPS connections. SSL v2 certificates do not offer the same level of encryption as the current SSL v3 option, and therefore it is not recommended to use SSL v2 for an encryption mechanism.

Solution: Microsoft recognizes there are legacy applications which may use SSL v2 certificates, so support for SSL v2 was not removed from the product entirely. It was disabled by default. The recommended solution for this issue is to obtain new SSL v3 certificates for any older/legacy web servers and applications that use SSL v2. If that is not possible, you can enable support for SSL v2 in the 'Advanced' tab of the 'Internet Options' control panel:



There is no Group Policy Object setting to enable this feature. To enable it for your enterprise deployments you would need to either:

- 1) Use the Internet Explorer Administration Kit (IEAK) 7 to build a custom Internet Explorer 7 package, and then deploy that MSI package using your Software Management System (SMS) infrastructure (or similar software delivery system).
- 2) Create a custom registry file to make this change and deploy it via login scripts. The registry key is located at:

[HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Internet Settings\]

Under that key you will need to create the **SecureProtocols** REG_DWORD entry with the flag options:

"The SSLv2 flag is 8 (0x008).
 The SSLv3 flag is 32 (0x020).
 The TLSv1 flag is 128 (0x080).

Hence, if all protocols are enabled, the value is 0x0A8. The IE 7 default is 0xA0.

These changes can be made using Active Directory configuration changes or you can use login scripts to run a registry file update for installations that do not have AD infrastructures. The registry file needs to contain both of the lines above and have a .reg extension for it to properly import to the system registry.

Question:

Why do we see "Certificate Errors: Navigation Blocked" page for sites that used to work fine?

Answer: Internet Explorer 7 modified the default security settings for HTTPS connections to increase

overall security in a number of instances. As a result of the new security default settings, IE 7 will block users (by presenting a red-colored information page) in any of the following instances when visiting an HTTPS site:

- Certificate was issued to a hostname other than the current URL's hostname
- Certificate was issued by an untrusted root
- Certificate is expired
- Certificate is revoked

Solution: Microsoft recognizes there are legacy applications which may utilize older or internally issued certificates that now trigger the new warning page. The intent of this change was to increase security and protect users from online threats. Microsoft recommends upgrading HTTPS sites to meet the newer level of security and validation checking. To accommodate for those scenarios where that is not practical, you are able to modify the default settings in your deployment. In addition, certain situations, such as allowing the use of expired or revoked certificate should never be over-ridden. Doing so is likely to create a reduction in user and system security. Microsoft also strongly advises against modifying the default settings regarding 'certificate address mismatch'. Disabling this change may expose your users to improperly configured systems on the Internet, and may result in compromised security of the machine.

A frequent scenario is users being blocked by default by the use of internally issued certificates for intranet sites (untrusted root issue). To resolve that error you should use Group Policy to deploy the internal root certificate used by your organization. You can also use the Certificate Manager Tool (information at [http://msdn2.microsoft.com/en-us/library/e78byta0\(VS.80\).aspx](http://msdn2.microsoft.com/en-us/library/e78byta0(VS.80).aspx), but the tool is included in Visual Studio) to build a login script to manually install the custom root certificate. That certificate will be stored in the local machine certificate store, not as a part of Internet Explorer. More information on working with the certificate store, and how to add your own certificates, can be found at <http://technet2.microsoft.com/windowsserver/en/library/2c03582f-00b2-43e5-ae1d-493894ad0fd71033.msp?mfr=true>.

Question:

We have users locked-down as standard users. Before Internet Explorer 7 our support staff could use the 'Run As' option to access Windows Explorer as an admin and perform necessary tasks (e.g. installing software). With Internet Explorer 7 we are no longer able to use that option. Is something wrong with our configuration?

Answer: No, this change was made to help increase the overall security of Internet Explorer and protect users against online threats. The issue is related to the process separation of Windows Explorer and Internet Explorer. Prior to the release of Internet Explorer 7, those two processes were linked and support staff/system administrators were able to use that link in order to open a Windows Explorer window (by entering a local file system location in the address bar) with administrative credentials. Now that we have separated those two processes, this option is no longer possible. Attempting to use the 'Run As' option to launch Internet Explorer 7 will perform as expected and launch the process with administrative privileges. When the administrator enters a local file system location in the Internet Explorer 7 address bar, the separate Windows Explorer process is called – and it launches with the logged in user credentials.

Solution: Since this change was made by design to increase security, there is no way to disable this setting or simple workaround. It was intended to create a barrier between the two processes and isolate Windows Explorer from potential external compromise. The only way to access the local file system with administrator level credentials would be to logon to the local machine directly to perform the necessary tasks. In the case of remote support, Microsoft recommends using Remote Desktop Connections in order to login with the proper credentials required to perform the required task.

Question:

We're using Internet Explorer 7 and having issues on Windows Vista with some legacy applications that read and write to the local file system. Why can't it read and write the files properly? Is there something different about Internet Explorer 7 on Windows XP?

Answer: The main difference between Internet Explorer 7 on Windows XP and Windows Vista is that Windows Vista offers an additional security protection to help prevent unintentional system access by malicious software or other parties. This feature is known as Protected Mode, and helps protect users and systems by restricting the ability of the Internet Explorer process from reading and writing to critical areas of the file system. Protected Mode allows reading and writing from the 'user hive' area of the file system, but prohibits activity to areas outside that perimeter. In making this change, some applications fail to work properly as they are attempting to read and write from restricted areas. Protected Mode does make an effort to assist improperly written applications by 'virtualizing' data reads and writes from the file system, however this should not be used as a remediation option for Protected Mode issues.

Solution: To resolve issues with Protected Mode application compatibility problems, customers are encouraged to rewrite the application (or extensions) to be Protected Mode aware. Resource information about working with Protected Mode is available at <http://msdn2.microsoft.com/en-us/library/bb250462.aspx>. In addition, the document at [http://msdn2.microsoft.com/en-us/library/bb250493\(VS.85\).aspx](http://msdn2.microsoft.com/en-us/library/bb250493(VS.85).aspx) on security and compatibility issues in Internet Explorer 7 will provide information about how to use the [Application Compatibility Toolkit](#) to diagnose specific compatibility issues with Protected Mode problems.

Microsoft does not recommend disabling Protected Mode as a solution to application compatibility. Removing this security mechanism may expose users and computers to unnecessary system compromise, loss of resources or unwanted installation of software.

Application Compatibility Toolkit (Internet Explorer Test Tool)

With the introduction of new products and platforms, there is always a need for tools to help plan and prepare for migrations. Internet Explorer 7 was released about the same time as the Windows Vista operating system, and is the browser shipped with Windows Vista. This section addresses the tools released in order to help enterprise customers test their environments for application compatibility as they evaluate Windows Vista, specifically Internet Explorer 7.

Question:

I want to start testing my internal sites, applications as well as our major partner applications. Are there

any tools to help me find any problems while I'm testing?

Answer: Along with the introduction of Internet Explorer 7 for Windows XP, Microsoft released the Windows Vista operating system which includes Internet Explorer 7. Microsoft created the Application Compatibility Toolkit to help enterprise planning and testing for migrations to Windows Vista. Included in the Application Compatibility Toolkit is the Internet Explorer Test Tool, which focuses on specific issues relating to Internet Explorer 7. While these tools are designed for Windows Vista migration planning and testing, the Internet Explorer Test Tool provides useful insight and information about compatibility issues which impact Internet Explorer 7 on Windows XP as well.

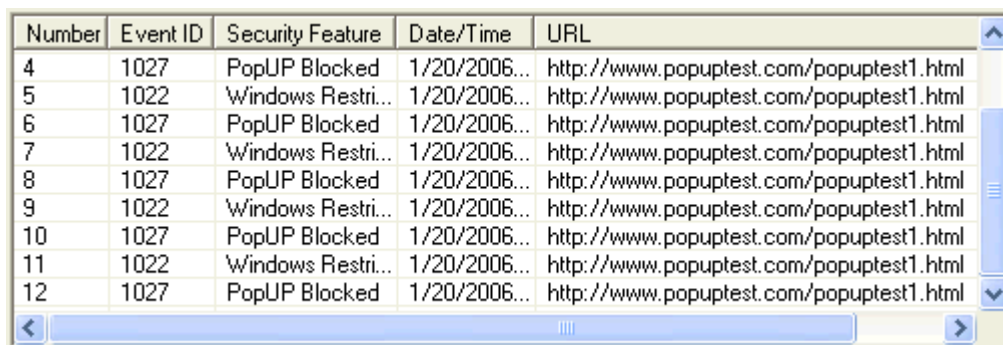
Solution: The Internet Explorer Test Tool is one component of the Application Compatibility Toolkit (available at <http://www.microsoft.com/downloads/details.aspx?FamilyID=24da89e9-b581-47b0-b45e-492dd6da2971&displaylang=en>). More information and details about the Internet Explorer Test Tool can be found at <http://technet2.microsoft.com/WindowsVista/en/library/8d2a6345-0d45-4909-bc0c-f9e0f86ad8b11033.mspx?mfr=true>. The step by step guide to configuring and working with the Application Compatibility Toolkit is available at <http://www.microsoft.com/downloads/details.aspx?FamilyId=BC931BFD-D279-4060-B370-08FA52F97A8D&displaylang=en>.

Question:

Using the Internet Explorer Test Tool is helpful at identifying areas that may not work properly in my environment, but the logs are pretty long. How do I figure out which of this information will really cause problems with my deployment?

Answer: The purpose of the Internet Explorer Test Tool is to provide details on all issues that may impact compatibility before you begin widespread deployments. While the tool may gather a large amount of data, the information in it is important and should be reviewed to ensure there are no critical problems that will impact your environment.

Solution: To help sort through the potentially large amount of data collected by the Internet Explorer Test Tool, the tool was designed to quickly and easily provide detailed information on each entry. An example of data collected by the Internet Explorer Test Tool is:



Number	Event ID	Security Feature	Date/Time	URL
4	1027	PopUP Blocked	1/20/2006...	http://www.popuptest.com/popuptest1.html
5	1022	Windows Restri...	1/20/2006...	http://www.popuptest.com/popuptest1.html
6	1027	PopUP Blocked	1/20/2006...	http://www.popuptest.com/popuptest1.html
7	1022	Windows Restri...	1/20/2006...	http://www.popuptest.com/popuptest1.html
8	1027	PopUP Blocked	1/20/2006...	http://www.popuptest.com/popuptest1.html
9	1022	Windows Restri...	1/20/2006...	http://www.popuptest.com/popuptest1.html
10	1027	PopUP Blocked	1/20/2006...	http://www.popuptest.com/popuptest1.html
11	1022	Windows Restri...	1/20/2006...	http://www.popuptest.com/popuptest1.html
12	1027	PopUP Blocked	1/20/2006...	http://www.popuptest.com/popuptest1.html

The 'More Details' section of the logging output provides more information about why something was blocked (or logged), for example:

Log Reason	PopUP Blocked
Date/Time	1/20/2006 12:01:44 PM
URL	http://www.popuptest.com/popuptest1.html
More Details	UrZone 3 PopupURL http://www.popuptest.com/popup2.html PopupFlags 262144
<input type="checkbox"/> Raw Data	

The tool provides useful and actionable information for issues that break web applications or prevent expected behaviors. By reading the detailed information from the logs, you should be able to specifically identify and easily resolve critical issues. Many issues identified with the Internet Explorer Test Tool are related to the security related changes made in Internet Explorer 7. See the document at [http://msdn2.microsoft.com/en-us/library/bb250493\(VS.85\).aspx](http://msdn2.microsoft.com/en-us/library/bb250493(VS.85).aspx) for more detailed information on security related compatibility issues identified in your logs.

Question:

Many of the errors I see in the reporting logs are about CSS errors, but the sites look fine to me, can I just ignore those errors? What will happen if we don't fix them before we deploy?

Answer: The errors you see in your logs are important to note, but they may not require action to successfully deploy Internet Explorer 7 in your environment.

Solution: Many of the standards support design changes made in Internet Explorer 7 can be mitigated by the use of certain DOCTYPE modes. While it is not recommended to build new content using the 'quirks' mode settings, legacy content may be left alone and the DOCTYPE may remain unchanged. If you do not see any layout or other rendering issues in your testing, you can safely ignore the errors.

However, if you do identify certain pages and layouts that do not appear to be correct, the log information gathered by the Internet Explorer Test Tool will help you understand the specific changes you need to make before you begin deployment. To help developers discover these sites and their workarounds, IE7 will create a log whenever it discovers a workaround that has been rendered unnecessary by the improvements in IE7. The following list shows some of the CSS fixes in IE7:

- 0 (Star Filter) - Selector String
- 1 (Strict Comment Filter) - Property Name
- 2 (Underscore Filter) - Property Name

Installation issues

Internet Explorer 7 was tested thoroughly to ensure the highest possible product quality and user experience. Much of that time was spent on the setup and installation process, to help prevent

installation problems and try to ensure a malware free system prior to users launching Internet Explorer 7 for the first time. Despite the extensive testing performed, along with the broad user community feedback during the Beta testing phases, end user machines can have a virtually unlimited combination of configurations so it is impossible to ensure all installations will go flawlessly.

Question:

I've tried installing Internet Explorer 7 and the installation is not working. I can't get a functioning browser at all now, do I need to reinstall the system?

Answer: No, there are a few options you can try to remove the Internet Explorer 7 installation and try to install again.

Solution: Internet Explorer 7 installs as a separate program and registers in the Add/Remove control panel. The easiest mechanism for removing inoperable Internet Explorer 7 installations is to use that interface to remove it. The steps needed are:

Click 'Start', then 'Run'

In the Open box, type `appwiz.cpl`, and then click OK.

In the list of programs, click 'Windows Internet Explorer 7', and then click the 'Remove' button.

Follow the wizard instructions to uninstall Internet Explorer 7.

If that method does not work properly, you can manually run the uninstall program by the following (you will need to ensure hidden folders and files are visible):

Click 'Start', then 'Run'

In the Open box, type `%windir%\ie7\spuninst\spuninst.exe`, and then click OK.

Follow the wizard instructions to uninstall Internet Explorer 7.

Question:

In our testing Internet Explorer 7 requires two reboots to install and update, can I get this down to one?

Answer: No, unfortunately the shared system components used by Internet Explorer and the rest of the Windows operating system make this impossible. During install many of the core OS components are being loaded, unloaded and modified. The system must be restarted in order to properly complete the registration activity, as well as ensure all shared components have been unloaded by other programs. Failing to properly reboot may cause the system to end up in an unknown and unsupported state.

Solution: The system must reboot twice in order to properly complete the installation and updates. Certain updates can be completed in the first reboot, while others are performed after the first set are complete. The order integrity must be maintained to avoid having a machine in an unsupported state.

Question:

We'd like to minimize our application compatibility issues, can we deploy Internet Explorer 7 along with

Internet Explorer 6?

Answer: No, that installation option is not possible. Internet Explorer is a Windows system component and contains several core operating system pieces that are quite different between Internet Explorer versions 6 and 7. One of those core elements is MSHTML.DLL, which is the main rendering engine used by Internet Explorer. Many of the standards support and layout improvements made in Internet Explorer 7 are only possible with the newer MSHTML.DLL. The Windows operating system cannot have two of these files loaded and available, or the system would not operate properly.

Solution: Microsoft recommends that enterprise customers properly evaluate, test and then deploy Internet Explorer 7. If certain deployment blocking issues are encountered during testing, this document should help provide solutions or workarounds to any major issue. In the event a user requires having both versions of Internet Explorer installed on one machine, the solution is to use a Virtual PC image to run another instance of Windows on that machine. Microsoft regularly makes a six month development and testing image available for users. The image is refreshed shortly before it expires to ensure an operational version is always available. You can build your own VPC image to avoid regularly replacing VPC images on any machines in your environment. You can also setup a 'regular' desktop machine in your environment and remote desktop into it if you do not want to run VPC images.

Question:

During the setup and installation process, Internet Explorer wants to run the Malicious Spyware Removal Tool. What is that and do I need to let it run?

Answer: The Malicious Spyware Removal Tool (MSRT) is a simple anti-spyware tool provided by Microsoft to help identify and remove many common malicious and unwanted software applications. The tool is updated on a regular basis with the most common software signatures, and is designed to isolate and remove those applications so they do not harm the users system. By design, Internet Explorer 7 wants to run the MSRT prior to installation to help ensure the system is free of malicious or unwanted software.

Solution: Running the MSRT is an optional activity. Microsoft recommends using the MSRT to help increase the security of an Internet Explorer 7 installation. The default configuration of the standard Internet Explorer 7 package as delivered by Microsoft will prompt users to run the MSRT. If you would prefer not to run the MSRT at install (or would prefer to make other changes to the default installation package), you will need to use the Internet Explorer Administration Kit (IEAK) 7 to build a custom Internet Explorer 7 package with your preferred settings, and then deploy that MSI package using your Software Management System (SMS) infrastructure (or similar software delivery system).

Question:

We've used the Internet Explorer Administration Kit to build a custom Internet Explorer 7 build for our organization. Is there an MSI package option for the IEAK?

Answer: Microsoft re-released builds of the IEAK and Internet Explorer 7 in October 2007. IEAK versions prior to that version did not allow for the creation of MSI packages.

Solution: In order to create an MSI package and build updated custom packages of Internet Explorer 7, you will need to ensure you are using the most recent version of the IEAK 7 as well as the most recent version of Internet Explorer 7. The most recent version of the IEAK 7 can be downloaded from <http://technet.microsoft.com/en-us/ie/bb219543.aspx>. When running the IEAK 7 you will be prompted to verify you have the most recent version of Internet Explorer 7, and the IEAK 7 will download it if necessary so you do not need to download it separately.

Question:

We've used the Internet Explorer Administration Kit to build a custom Internet Explorer 7 build for our organization. Why are we seeing that some installations appear to complete properly but the customizations are not there, and other times we see some issues where SMS installations are not working?

Answer: There is an issue with the way SMS settings must be configured to properly install an IEAK 7 built package using SMS.

Solution: In order to successfully deploy Internet Explorer 7 using SMS, you need to ensure the 'run when user is logged on' flag is enabled in the SMS configuration window.

Question:

After upgrading to Internet Explorer 7, some toolbars aren't working properly (or the system is running very slowly). What is causing the problem?

Answer: While most add-ons are developed and function properly regardless of Internet Explorer version, some add-ons may experience problems based on the changes in designs made to Internet Explorer 7. Most often those issues are related to changes made on Windows Vista and Protected Mode, but other problems have come up with system performance or resource issues.

Solution: The best way to troubleshoot add-on and Browser Helper Object (BHO) issues is to run Internet Explorer in 'no add-ons' mode. In this mode, Internet Explorer loads the minimum components needed to operate, so you can verify the base installation is functioning properly. Once you have verified things are working normally with no add-ons, you should try to add back add-ons one at a time to identify which one is causing the issue. You can do this by disabling all add-ons via Manage Add-ons and disabling each control manually, then enabling one control at a time until you find the item causing the issue.

To start in No Add-ons Mode you can choose either path:

Click 'Start', then 'Run'
In the Open box, type `iexplore -extoff`, and then click OK

Or

Click 'Start' -> 'All Programs' -> 'Accessories' -> 'System Tools' -> 'Internet Explorer (No Add-ons)'

