

Browsium Ion 3.3 Administration Guide



Bonding Browsers to Business
www.browsium.com



Administration Guide

This guide has been created for IT administrators to assist in installing, configuring, and deploying Browsium Ion. This is version of the guide is designed for use with Browsium Ion 3.3.

For more information about Browsium, or to contact customer support, please visit <http://www.browsium.com>.



Table of Contents

1. Introduction.....	6
1.1. Browsium Ion Overview	7
1.2. Browsium Ion Explained.....	8
1.2.1. Document Modes.....	8
1.2.2. Profiles.....	9
1.2.3. Rules.....	9
1.3. Ion Configuration Manager.....	10
1.4. Browsium Ion Integration with Internet Explorer.....	11
2. Installation.....	13
2.1. Ion Components.....	14
2.2. Software Requirements.....	15
2.3. Installing the Browsium Ion Client.....	16
2.4. Installing Browsium Ion Configuration Manager	20
2.5. Available Command Line Switches for the Installer	23
2.5.1. Installation Options	23
2.5.2. Display Options.....	23
2.5.3. Restart Options	23
2.5.4. Logging Options.....	24
2.5.5. Repair Options.....	24
2.5.6. Command Line Installation Examples.....	24
2.6. Upgrading from Evaluation to Licensed Version.....	25
3. Ion Configuration Manager Overview.....	28
3.1. Menu Bar	29

3.2. The Project Node.....	30
3.3. The Profile Node.....	32
3.3.1. Rendering Mode Selection	32
3.3.2. Profile Editor.....	36
3.4. The Rules Manager Node.....	50
3.4.1. Rule Editor.....	52
3.5. The Project Settings Node.....	53
3.5.1. Listener Service Port.....	53
3.5.2. Show Splash Screen on Start	53
3.5.3. Clear Cache on Start.....	53
3.5.4. Invalid Certificate Handling	54
3.5.5. Custom Ion Preferences.....	54
3.6. The Tools Node.....	55
3.6.1. Regex Builder	55
4. Configuration Tutorial	58
4.1. How to Create a Profile.....	59
4.2. Working with Rules.....	62
4.2.1. Creating a Rule.....	62
4.2.2. Copying a Rule	66
4.2.3. Removing a Rule.....	66
4.3. Working with Custom Files, Registry Settings and ActiveX Controls	68
4.3.1. Java Version Manager	73
5. Rule Writing Basics	77
5.1. Recommendations for Writing Rules that Work Well	77
5.2. Working with Web Application Assessment Tools.....	78
5.3. Ion Rules Behavior	78
5.4. Why Users Can't Create Their Own Rules	79
6. Ion Deployment in the Enterprise.....	81
6.1. Understanding Ion Registry Setting Locations	81
6.1.1. Working with Test Configurations in Group Policy Environments	83

6.1.2. Multi-user Testing Configurations.....	84
6.2. Enabling the Ion Client for Enterprise Deployment	84
6.2.3. Understanding the Add-on List Policy	85
6.2.4. Determining the GUID/CLSID of the Ion Client Add-On.....	87
6.3. Readyng End User PCs to Use the Ion Client	89
6.4. Deploying Ion Configurations	90
6.4.1. Project File Deployment Option A: Network Share or Web Server Location	90
6.4.2. Project File Deployment Option B: Client File System Location	92
6.5. Verifying Deployment Settings	93
6.6. Understanding the Ion 3.3 Project File Caching Process.....	95
6.6.1. Ion Project File Caching Behavior	95
6.6.2. How Ion Handles Project File Polling.....	95
6.7. Managing Ion Logging Settings	96
A. Troubleshooting.....	99
A.1. Ion Rule Fails To Engage	99
A.2. Correct Profile Does Not Load for a Web Site.....	100
A.3. Ion Not Working Properly in IE6 or IE7	101
A.4. Ion Not Working Properly on Windows 8/8.1	101



Section One

Introduction

In this section you will learn:

- ✓ Why Browsium Ion is your one-stop Web Application Continuity framework
- ✓ What components make up the Browsium Ion system
- ✓ How you can seamlessly integrate Browsium Ion into your enterprise

1. Introduction

Many organizations would like to upgrade to a modern version of Internet Explorer and Windows 7 or Windows 8, but can't. That's because they depend upon web applications which were designed for legacy versions of Internet Explorer. The various changes in layout, rendering, scripting and security designs in modern IE versions prohibit those applications from functioning properly without extensive and expensive modifications or upgrades.

Browsium Ion ("Ion") allows enterprises to upgrade to modern technologies like Windows 7 or Windows 8 and current Internet Explorer versions while maintaining compatibility with existing legacy web applications—all without changing a single line of server code.

But Ion provides more than just legacy Internet Explorer compatibility for the web applications you are running today – it helps ensure *Web Application Continuity* as your business moves forward. Web Application Continuity provides solutions to manage, maintain, and operate your web applications regardless of the underlying browser lifecycle. Your web applications are built to serve the business for years, but the rate of browser innovation has accelerated to the point it outpaces the standard IT technology system lifecycle.

Ion decouples the browser/web application dependencies that exist today, and enables your organization to deploy technologies when you want, without breaking existing line-of-business applications. More importantly, Ion is seamless to users, integrated into the browser and is more easily managed than virtualization solutions.

1.1. Browsium Ion Overview

Ion delivers Web Application Continuity by delivering the ability to use and control multiple browser modes, fully customize the browsing environment, and rollback web plugin support to legacy versions all inside modern versions of Internet Explorer. Ion lets organizations enjoy the features and better security of the latest platforms and migrate their legacy Internet Explorer applications on their own timeframes and budgets.

More importantly, Ion is a solution that makes sense.

Ion makes sense to users because it's seamless. It lets them work in the ways they're used to, unlike virtualization approaches that require special training or hardware. In fact, many users never notice they're using Ion.

Ion certainly makes sense to IT administrators because it's been designed with them in mind. Ion is easy to deploy and manage using the same tools and techniques they use for Internet Explorer. There's no server component, so Ion works well for in-house and remote workers alike. With minimal overhead, and support for 32- and 64-bit versions of Windows, Ion runs everywhere IT administrators need it to. Finally, it's isolated approach to running legacy browser settings greatly reduces attack surface when compared to running a legacy operating system and browser.

With Ion integrated into your business, the security and functionality benefits of Windows 7, Windows 8, and a modern Internet Explorer are no longer out of reach.

1.2. Browsium Ion Explained

Browsium Ion is markedly different from any other web application compatibility offerings, so to understand what Ion is, it's easiest to start by identifying what it is not: Ion is *not* virtualization, it is *not* a repackaged standalone version of Internet Explorer, *nor* is it a solution that enables running multiple versions of the Internet Explorer browser on the same Windows installation. Installing Ion on your system does *not* modify the default system browser registry values or settings.

Ion is controlled by a hierarchical system of Profiles and Rules, defined using the Ion Configuration Manager. Understanding this system is the key to understanding Ion. The Configuration Manager provides the configuration tool that allows Ion to sidestep compatibility issues by delivering different versions of rendering modes, registry settings and ActiveX controls to the web applications that need them. And with the custom ActiveX, File and Registry settings feature, you can manage browser extensions that are otherwise incompatible with a particular version or architecture of Windows or Internet Explorer.

Ion works by managing the way your natively installed version of Internet Explorer renders content. When invoked automatically by a Rule, Ion ensures web applications load the needed version of an add-on, inject the proper JavaScript code, and carry forward legacy Internet Explorer environment and security settings – all without modifying the Windows and Internet Explorer installations or reducing the security posture of the system. Ion renders legacy web applications and content directly inside the Internet Explorer window, just like any other content in the browser. Ion is fully integrated with Internet Explorer settings and uses existing browser settings and file system setups so you only need to manage the exceptions. This approach requires fewer resources and makes administration easy.

More specifically, Ion acts as a broker between the browser and the requested web content. Ion uses software isolation to display content and settings on an IT-controlled opt-in basis. In other words, Ion intervenes when – and only when – it is told to by IT administrators.

1.2.1. Document Modes

Document modes are the foundation of how Ion is able to make legacy applications work. When a user requests a page that matches an Ion rule, Ion displays the content using an alternate document mode specified by that rule. The following document modes are available in Ion 3:

- Default (let Internet Explorer choose)
- Adaptive IE Quirks Mode (combination of IE Quirks Mode and IE7 Standards Mode)
- Emulate IE7 Mode

- Emulate IE8 Mode
- Emulate IE9 Mode
- Emulate IE10 Mode
- IE Quirks Mode
- IE Edge Mode
- IE7 Standards Mode
- IE8 Standards Mode
- IE9 Standards Mode
- IE10 Standards Mode

1.2.2. Profiles

Profiles are groups of settings and browser configurations as well as the specific environment settings, options/values and add-ons required for your web application. Profiles let you specify the configuration and settings needed to run your web application properly, even if that configuration varies from the default IE installation or settings. By providing this type of 'side by side' rendering environment, Ion ensures web applications run as expected, even when the default system settings and applications are upgraded to the latest modern IE version.

Profiles are easy to create and customize for your specific web application needs. To get started with Profiles, simply select the browser engine required for your web application and Ion will create a Profile using that browser engine and commonly required default settings. From there you can add Custom File associations (to fix issues like running multiple versions of Java inside Internet Explorer or load older CRM/ERP application controls), define HTTP Headers (to ensure web applications operate correctly), as well as a range of other options.

While Profiles are highly customizable, they also inherit system settings that are not explicitly changed. This streamlines management by ensuring you don't need to recreate every policy setting and control point; simply manage the ones you need to set for compatibility.

Information on the specifics of each browser engine available in Ion can be found in the [Profiles List](#) section.

1.2.3. Rules

Rules are actions triggered by a simple string or IP address match or more complex and granular regular expressions. Rules provide very granular control over the conditions which bring them into play, and can work broadly against entire zones or target specific elements on a page. See the [Rules Manager](#) section to learn more.

1.3. Ion Configuration Manager

The Ion Configuration Manager is the main interaction point for IT administrators using the Ion system. Designed to work efficiently in your business and deploy using your existing technology systems, the Ion Configuration Manager can be utilized in ways that best meet the needs of your organization. Using a distributed architecture approach, each web application team – or business unit – can use the Ion Configuration Manager to create Rules and configurations for their specific needs; alternatively, a single administrator can manage all the Rules, Profiles and Settings.

If you choose the distributed approach, you can merge configurations so users will have the combined set of rules and configurations needed for line-of-business application compatibility. Using the Ion Configuration Manager, define the sites that need a specific rendering mode and settings, define and deploy the necessary Rules and those sites will then be rendered by Internet Explorer using Ion to control the experience. By integrating with Internet Explorer, Ion provides a seamless and natural browsing user experience. Your users won't need to know they are viewing sites using Ion – everything will just work as expected in Internet Explorer.

This is the power of Ion: You control the behavior. Only the sites you configure to be managed by Ion are controlled by an Ion configuration. Sites that need no special browser configuration or security settings load using your default Internet Explorer settings. Once your Ion configuration is complete, simply deploy the XML formatted Project file and configure client systems to read it. Client systems can be configured to read the location of the Project file via Group Policy or simple registry edits – you decide what works for your organization.

1.4. Browsium Ion Integration with Internet Explorer

Unlike any other web application compatibility solution available today, Browsium Ion is fully integrated inside Internet Explorer. Ion is a clean and user-friendly solution that enables a single browser to access both legacy web applications and the latest standards-based web technologies.

In addition to being a better solution for end users, Ion offers many benefits for the IT administrator. As an integrated component of Internet Explorer, Ion reduces installation and patch management overhead, by using Internet Explorer's local settings and configurations.

Internet Explorer Group Policy Options (GPOs) cascade down and are respected by Ion, so you only need to manage one set of policies. In cases where a Group Policy setting controls a feature not supported by the rendering mode specified by an Ion Rule, Ion simply ignores the setting. This ensures consistency across systems running Ion, even in environments that have not standardized on a single version of Internet Explorer, and avoids limitations or issues that may arise in future versions of Internet Explorer.



Section Two

Installation

In this section you will learn:

- ✓ About the Browsium Ion System Components
- ✓ Software Requirements for Browsium Ion
- ✓ How to Install Browsium Ion
- ✓ How to Configure Command Line Switches for Network Distribution

2. Installation

Browsium Ion is simple to install – administrators need both the Browsium Ion Configuration Manager and the Ion Client, while end users only need the Ion Client installed. The Browsium Ion Client requires Administrator permission to install, but can run using standard user permissions so system access can remain tightly controlled. This section provides details on the specific pieces of the Ion system as well as information on how to manage the Ion Client installation in a corporate environment.

2.1. Ion Components

The Ion system is comprised of two main components: an administrative tool for defining configurations, and a client for interpreting the configurations for end users.

- **Ion Configuration Manager (Ion-AdminSetup.exe)**

The Ion Configuration Manager (**BrowsiumIonManager.exe**) is the single management interface for the Ion system. This application provides the central point for creating, configuring and managing Projects, Profiles and Rules. Since the Ion Configuration Manager is not meant for end users, this package should not be installed broadly – installation of this package should be limited to system administrators and web application/business unit owners.

- **Ion Client (Ion-ClientSetup.msi)**

The Ion Client is responsible for loading Ion configuration data and invoking Ion-managed instances of Internet Explorer for end-users. The client package must be installed on all PCs in your organization that require web application remediation with Ion. The Ion Client consists of two core components:

- **Ion Controller & Broker**

The Ion Controller (**BrowsiumIonController.exe**) is the main component of the client infrastructure used by Ion to handle content loading, rules implementation and redirection. The Ion Broker (**BrowsiumIonBroker.exe**) handles communication between the Ion Controller and Ion-managed instances of Internet Explorer. The Ion Monitor (**BrowsiumIonMonitor.exe**) is responsible for ensuring that Controller is running reliably.

- **Ion Client Add-On for Internet Explorer**

Ion installs an add-on to facilitate communication between Internet Explorer and the Ion Controller and Ion Broker.

2.2. Software Requirements

The following minimum system specifications are required to run Browsium Ion.

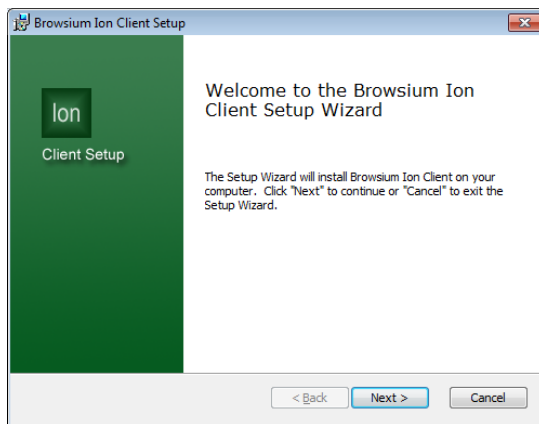
- Operating System:
 - Windows XP
 - Windows 7
 - Windows 8
 - Windows Server 2003
 - Windows Server 2008 R2
 - Windows Server 2012 R2
- Microsoft Internet Explorer 8, 9, 10, or 11
- .NET Framework Version 3.5 SP1
- 1 GHz or faster 32-bit (x86) or 64-bit (x64) processor
- 1GB system memory
 - 2GB system memory when used on multi-user Windows Servers

While Browsium Ion can be installed on systems with less than 1GB of memory, users may experience performance issues.

2.3. Installing the Browsium Ion Client

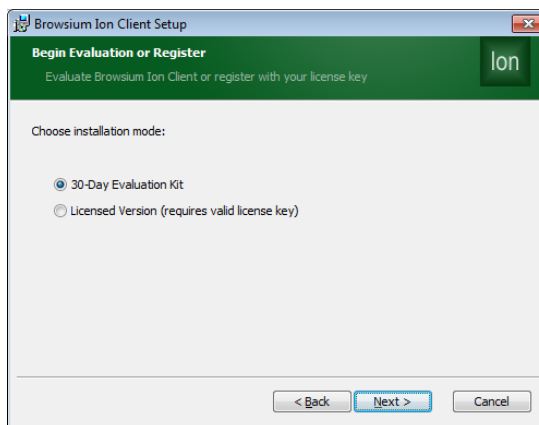
This section covers manual installation of the Browsium Ion Client. Network and command line installation options can be found in [Available Command Line Switches for the Installer](#). You will need Administrator rights to run the Client Installer. Once installed, the Ion Client can run under any user account and does not require special user permissions or elevation.

1. To start the Client Installer process, double-click on the Ion-ClientSetup.msi file. To properly complete the installation process you will need an account with Administrator rights. The first screen provides a basic introduction. Click Next to get started.

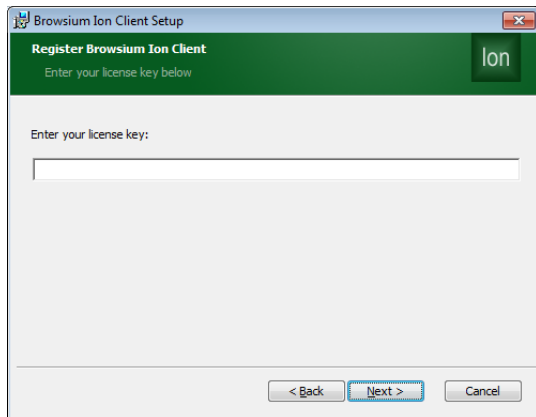


Once the installation process has begun, you will be presented with two choices for the installation mode. One installs the Ion Client software as a 30-Day Evaluation Kit. The second installs the fully licensed version, requiring a license key provided by Browsium.

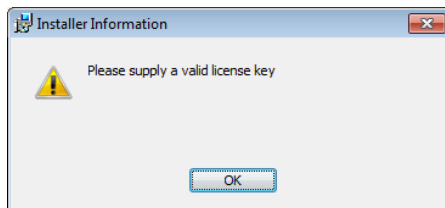
When you have completed your 30-Day evaluation of Ion and are ready to install the License, the process is defined in [section 2.6](#).



If you have chosen the Licensed Version, you may now enter the license key that has been provided by Browsium and then click Next. The license key can be copied from your Browsium Ion Download page and pasted into the empty box.

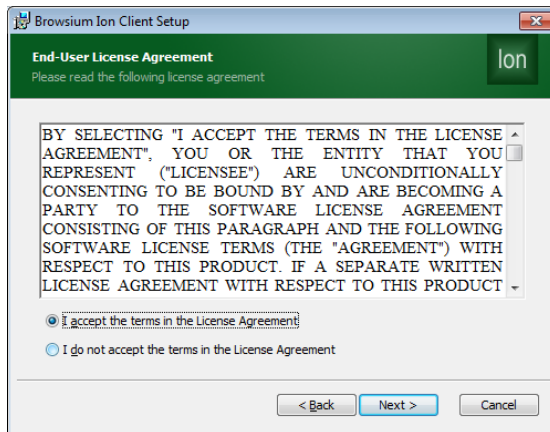


Your license key is then validated. An invalid product key will result in the following error:

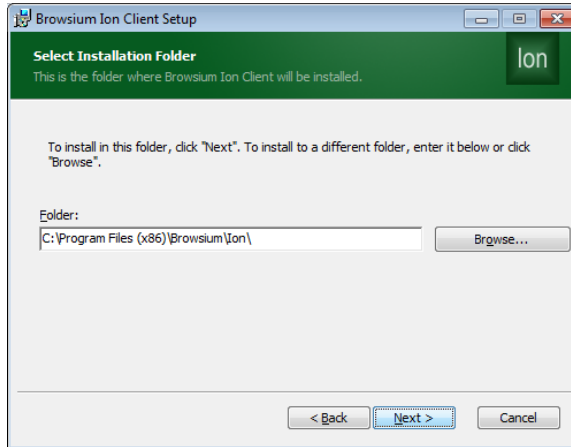


If you believe your key is valid, please contact Browsium Support. You may install the 30-day Evaluation Kit now and update the license key later. Be sure to delete the invalid key after clearing the error dialog before clicking Back to the previous screen.

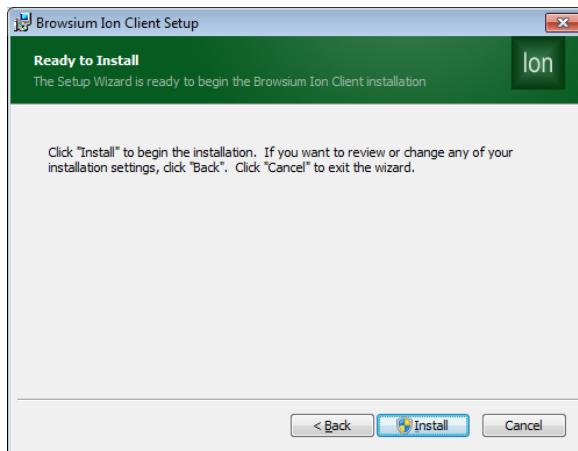
2. The next screen contains the End User License Agreement (EULA) for Browsium Ion software. You will need to read and agree to the terms of the EULA in order to proceed.



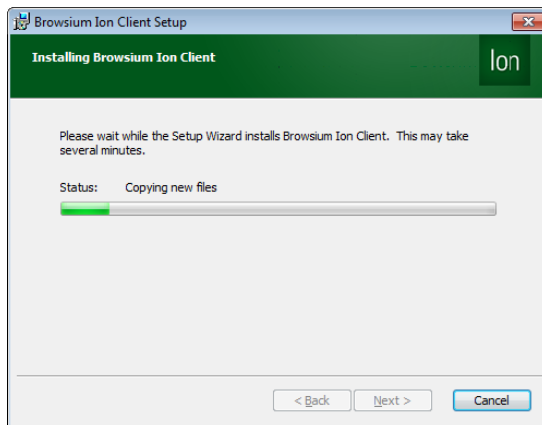
This page allows you to specify where to install the Ion Client program files. When you have specified the desired location, click Next. Browsium recommends installing into the default location.



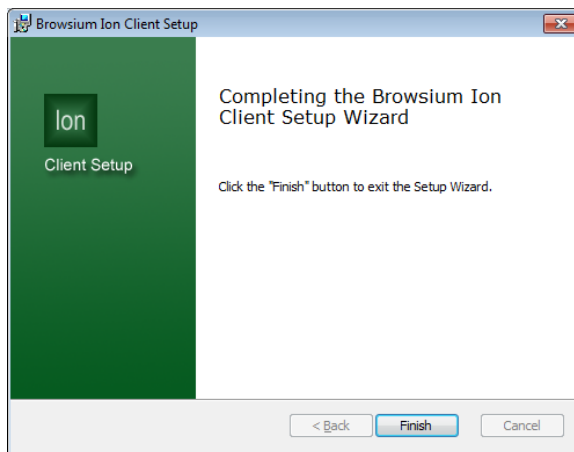
3. Now you are ready to install the Ion Client. Simply click Install to proceed.



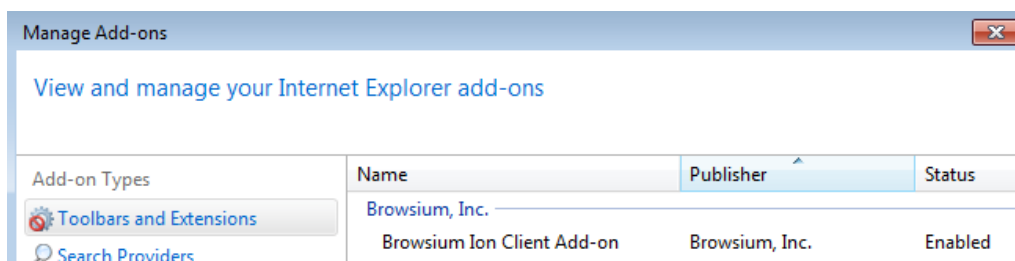
During the process you will see a progress bar:



When the Ion Client installation process has finished, you will see the following screen indicating success.



To confirm the Ion installation has completed properly, launch Internet Explorer and select Tools->Manage Add-ons to ensure the Browsium Ion Client Add-on is listed and Enabled. For enterprise deployments, you will Enable the add-on via Group Policy as described in [chapter 6](#).

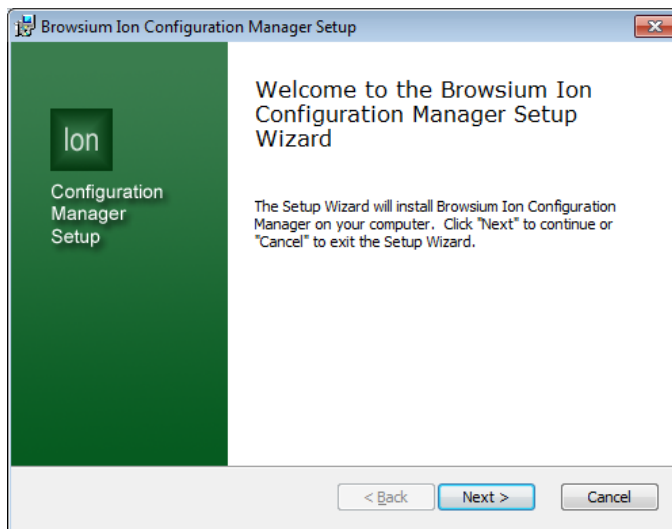


2.4. Installing Browsium Ion Configuration Manager

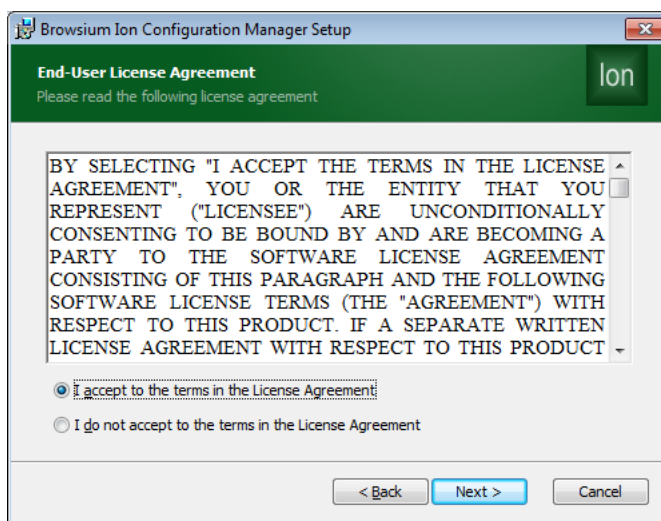
This section covers the installation process for the Browsium Ion Configuration Manager. The Browsium Ion Client must also be installed on the system in order to test Rules and Profiles.

The steps for installing the Ion Configuration Manager are as follows:

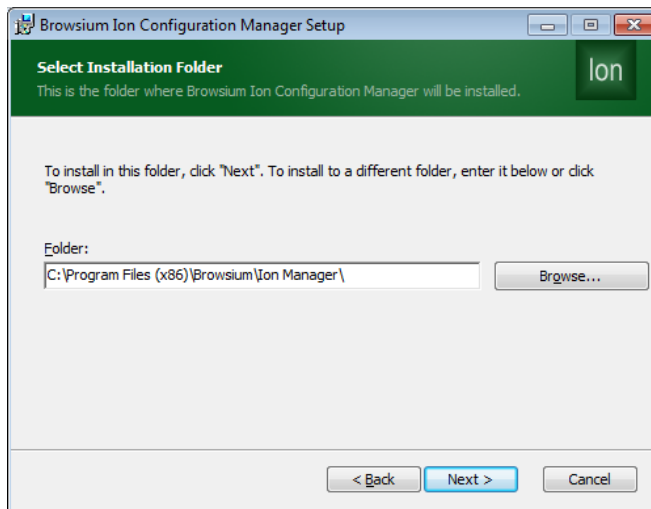
1. Locate the Ion Configuration Manager Installation file (***Ion-AdminSetup.msi***) and double click to run the program.



2. Confirm you have read and agreed to the End-User License Agreement (EULA) by clicking '**I agree to the terms in the License Agreement**' and **Next** to continue with installation.

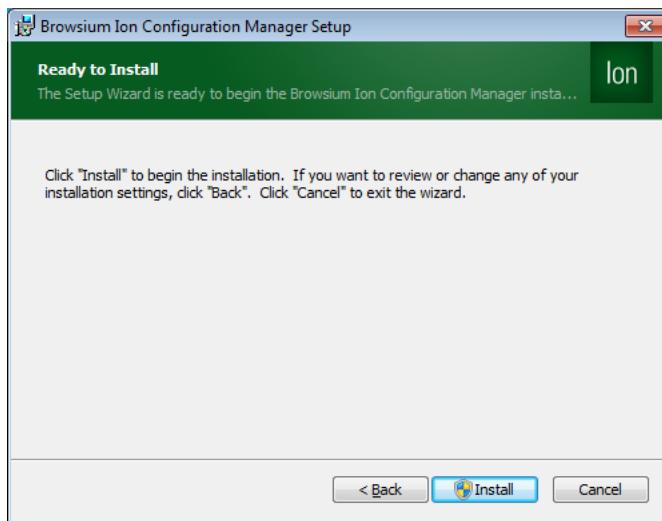


By default, the installer places the required files in "`\Program Files\Browsium\Ion Manager`" (32-bit systems) or "`\Program Files (x86)\Browsium\Ion Manager`" (64-bit systems) on the system drive.



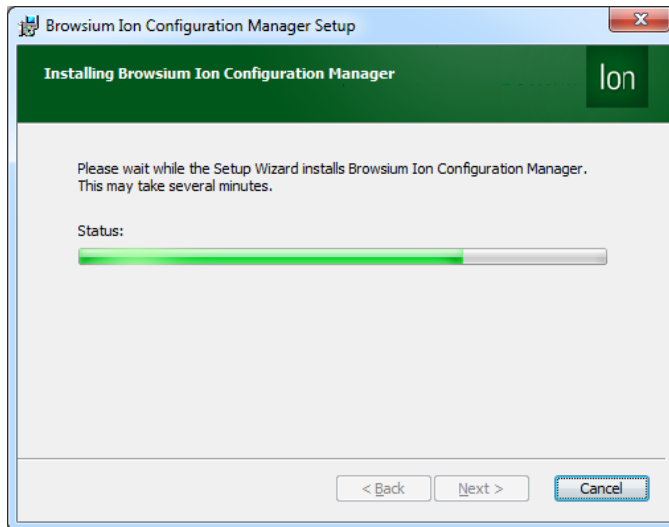
Select an installation location and click **Next**.

- Now you're ready to install the Ion Configuration Manager. Click **Install**.

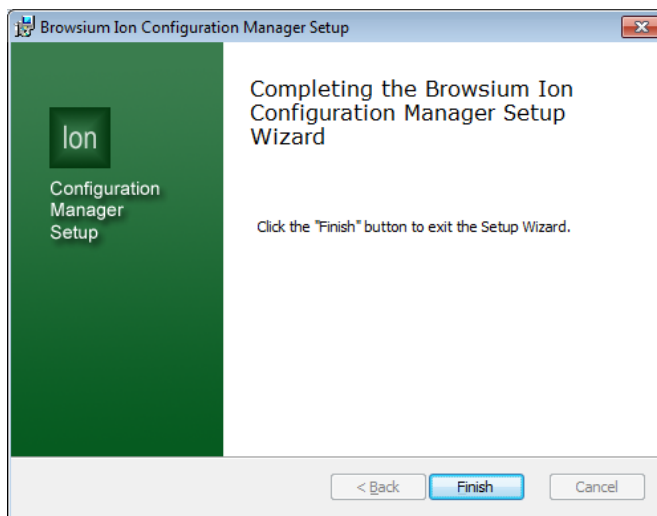


The Ion Configuration Manager requires Administrator rights so the installer may generate a UAC prompt before installing.

- During the installation process you will see a progress window



This screen will be displayed when the installation is complete and all necessary files have been configured. Click **Finish** and you are ready to begin working with Ion.



2.5. Available Command Line Switches for the Installer

Ion supports network-based installations using Windows Installer (MSIEXEC.EXE) for organizations that use software distribution systems or want to deploy via installation scripts and logon applications. Ion provides for several options that are controlled by the following switches. Note: You must run Command Prompt as Administrator on Windows 7 and later.

2.5.1. Installation Options

Switch	Description
<code>/j<u m> <Product.msi> [/t <Transform List>] [/g <Language ID>]</code>	Advertises a product – ‘m’ to advertise to all users, ‘u’ to advertise to the current user
<code></uninstall /x> <Product.msi ProductCode></code>	Uninstalls the product
<code>APPDIR=<path></code>	Installs product to a specific directory, other than the default location
<code>OPT_PID=<license key></code>	Installs with an Ion license key

2.5.2. Display Options

Switch	Description
<code>/quiet</code>	Quiet mode, no user interaction
<code>/passive</code>	Unattended mode - progress bar only
<code>/q[n b r f]</code>	Sets user interface level, where: n - No User Interface b - Basic User Interface r - Reduced User Interface f - Full User Interface (Default)
<code>/help</code>	Shows help information

2.5.3. Restart Options

Switch	Description
<code>/norestart</code>	Do not restart after the installation is complete
<code>/promptrestart</code>	Prompts the user for restart if necessary
<code>/forcerestart</code>	Always restart the computer after installation

2.5.4. Logging Options

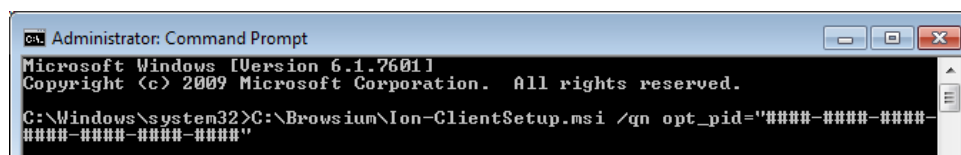
Switch	Description
<code>/l[i w e a r u c m o p v x + ! *] <LogFile></code>	Install keeping a log file, where: i - Status messages w - Nonfatal warnings e - All error messages a - Start up actions r - Action-specific records u - User requests c - Initial UI parameters m - Out-of-memory or fatal exit information o - Out-of-disk-space messages p - Terminal properties v - Verbose output x - Extra debugging information + - Append to existing log file ! - Flush each line to the log * - Log all information, except for v and x options
<code>/log <LogFile></code>	Equivalent of <code>/!*</code> <code><LogFile></code>

2.5.5. Repair Options

Switch	Description
<code>/f[p e c m s o d a u v] <Product.msi ProductCode></code>	Repairs a product

2.5.6. Command Line Installation Examples

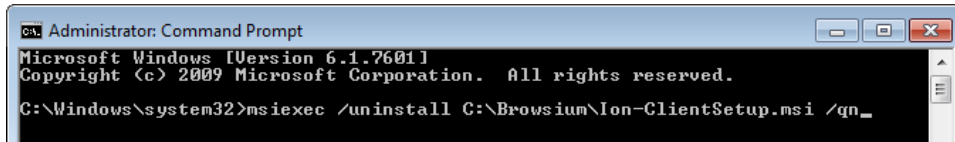
The following example will install Ion-ClientSetup.msi with an Ion license key in Quiet Mode with No User Interface. Launch the Command Prompt as Administrator, enter the path to Ion-ClientSetup.msi (located in C:\Browsium for this example), add the /qn switch, and substitute the hash marks (#) with your Ion license key provided by Browsium.



```
Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>C:\Browsium\Ion-ClientSetup.msi /qn opt_pid="####-####-####-####-####-####-####-####"
```

The following example will uninstall Ion-ClientSetup.msi in Quiet Mode with No User Interface. Launch the Command Prompt as Administrator, enter "msiexec /uninstall" followed by the path to Ion-ClientSetup.msi and add the /qn switch.



More information on deploying the Ion Client to ensure the browser extension is enabled by default can be found in [section 6.4](#).

2.6. Upgrading from Evaluation to Licensed Version

You have two options when your 30-day evaluation is complete and you wish to upgrade your Ion Client software to the licensed version. Choose the options that best fits your organization's deployment and systems management processes.

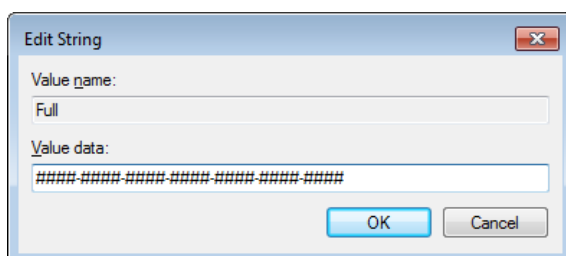
The first option is to enter the license key in the system registry utilizing a registry editing tool or command line script. You must have administrator rights to edit the registry.

Find or create the string value Full at the following registry path:

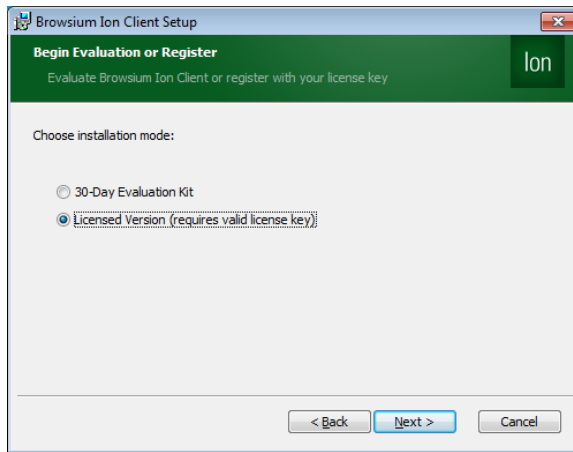
For 32-bit systems: HKEY_LOCAL_MACHINE\SOFTWARE\Browsium\Ion

For 64-bit systems: HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Browsium\Ion

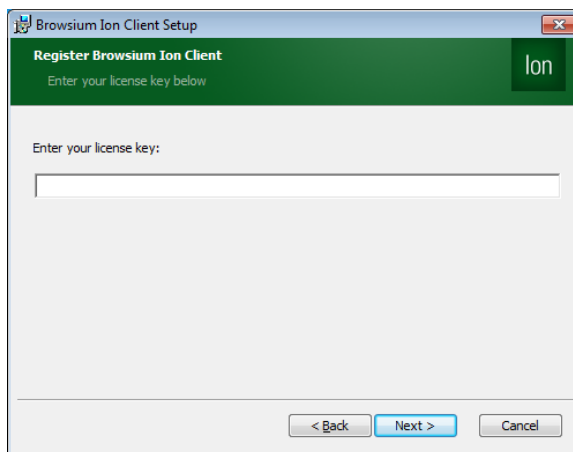
Then modify the value data of Full and enter your complete 28 character license key.



The second option is to uninstall the client software with the Windows Program and Features uninstall utility. Then reinstall Ion-ClientSetup.msi, select Licensed Version, click next.



Then enter the license file provided by Browsium and continue with the install process as before.



You may also install the Ion Client from a command line using the OPT_PID= <license key> switch documented in [section 2.5](#).



Section Three

Introduction to Ion Configuration Manager

In this section you will learn:

- ✓ More about the Browsium Ion Configuration Manager
- ✓ Where to Find Settings in the Browsium Ion Configuration Manager
- ✓ How to Navigate the Screens, Settings and Options in the Browsium Ion Configuration Manager

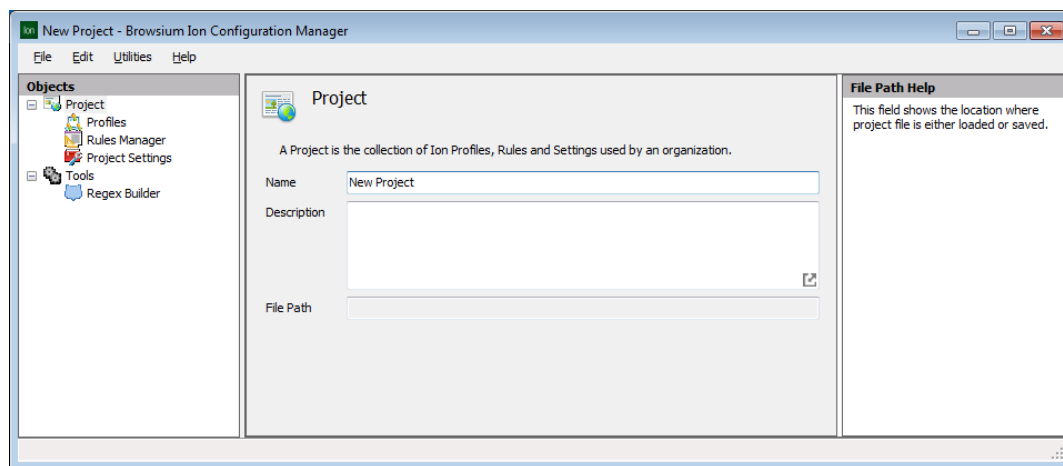
3. Ion Configuration Manager Overview


The Ion Configuration Manager enables you to create Profiles and manage Rules that define the websites you need to manage using the Ion rendering process. This section looks at the various elements of the Ion Configuration Manager. The Configuration Manager is designed with the look and feel of an MMC snap-in, with three main functional areas:

Objects Pane (Left) – Tree view containing Project and Tools nodes

Content Pane (Center) – Main data and content window

Actions Pane (Right) – Contextual links for common tasks and steps



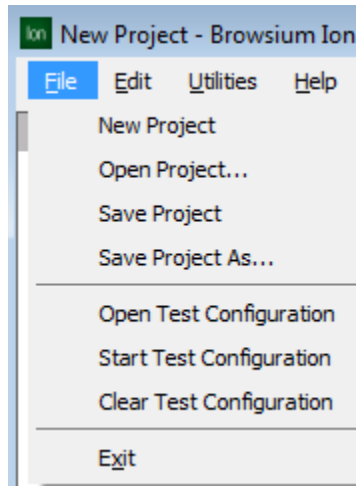
Some data entry fields will display an  icon in the lower right hand corner of the text entry field. Clicking this link on any screen will open the extended and powerful text editor. The text editor is provided for easily working with complex text fields, as well as offering syntactical highlighting for easy and readable visual layout. When done editing using the text editor, simply click the OK or Cancel buttons to save (or discard) changes made.

The Actions Pane contains regularly used and context related common functions. The Add, Edit, Delete, Copy, Cut, Paste items appear in the Actions Pane based on the specific task being completed. The behavior of these items is the same in all scenarios, for example the copy and cut items always copy or cut the selected object. Some interfaces may not support items such as Copy and Paste, and will not be displayed.

See the [Project Node](#), [Profile Node](#), [Rules Manager Node](#), [Project Settings Node](#) and [Tools Node](#) sections for a detailed description of each Objects Pane Node.

3.1. Menu Bar

The Ion Configuration Manager Menu Bar provides access to key functions related to creating, editing, and testing Ion projects.

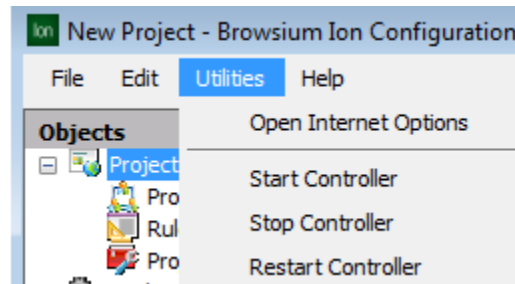


To aid in configuration testing and tuning, Ion Configuration Manager allows administrators to automatically apply project file pointers directly in the system registry and start or restart the Ion Controller to process the new configuration. This is performed via Start Test Configuration from the File menu, which writes the file pointer to the LoadFromFileName registry value in HKEY_CURRENT_USER\SOFTWARE\Browsium\Ion before starting or restarting the Ion Controller.

This option reduces delays and overhead of manually editing the registry or deploying registry changes via Group Policy. Open Test Configuration will open the Project file found in the LoadFromFileName pointer (if Start Test Configuration had been executed prior). Clear Test Configuration will clear the LoadFromFileName pointer in the registry and stop the Ion Controller.

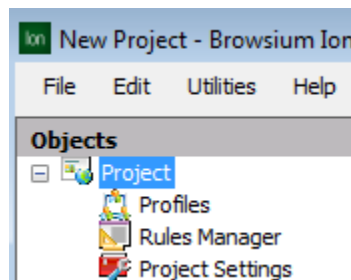
Start Test Configuration will automatically save the current project before writing the LoadFromFileName registry value and restarting the Controller. If the current project has not been saved prior, you will be prompted to save the file to a new location before continuing.

You can use the Utilities menu to launch Internet Explorer and manage the BrowsiumIonController.exe ("Controller") process. You may need to Start/Stop/Restart the Controller in order to load new configurations or reproduce troubleshooting steps.

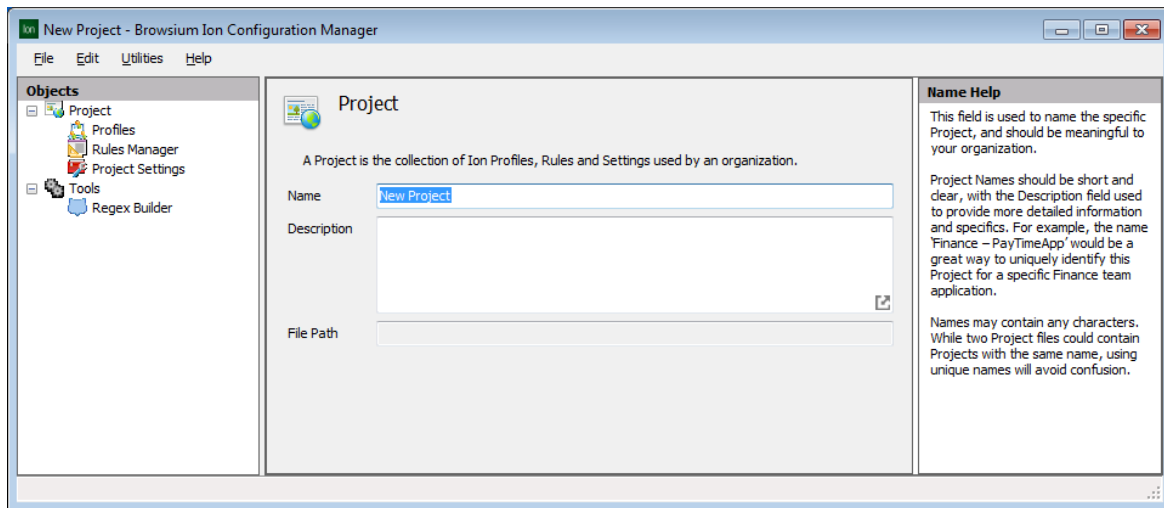


3.2. The Project Node

The Project Node displays the configuration file currently loaded in the Ion Configuration Manager. An individual Project file (.BCX file extension) contains all of the configuration information needed for a given set of Rules and Profiles. The Ion Configuration Manager is designed to load one Project per instance. Loading multiple Projects can be done by opening each Project in another Ion Configuration Manager instance. Rules and Profiles (and individual configuration items that make up a Profile) can be copied and pasted between Projects.

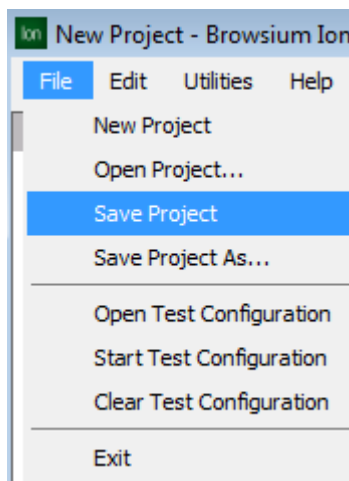


Loading a Project (or creating a New Project) brings up the Project details in the Content Pane.



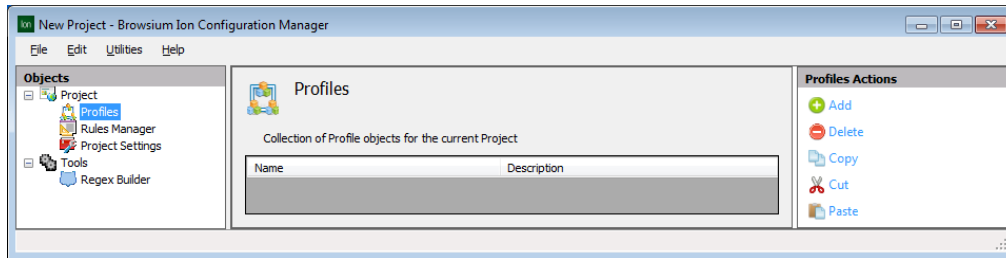
Project details (Name and Description) can be edited on this screen. Projects are assigned a default name and a blank description and should be edited to reflect information and labeling relevant to your organization. The project's File Path is blank until you save the project file. This field cannot be edited but will be updated automatically if the project is saved to a new location.

Use the File menu to Save a Project for future editing or deployment.



3.3. The Profile Node

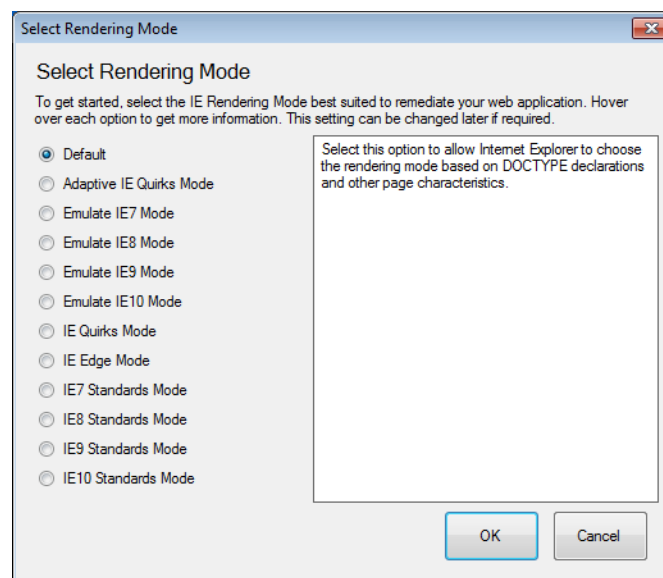
Profiles contain the rendering mode, environment settings, and specific browser plug-ins to be used in rendering content. To create a new Profile use the Actions Pane 'Add' item.



The Actions Pane includes the Delete, Copy, Cut, Paste items which are helpful when working on a Project populated with several Profiles. The Copy, Cut and Paste items integrate with the system clipboard and are especially useful when working with multiple Ion Configuration Manager instances loaded with unique Project files.

3.3.1. Rendering Mode Selection

Profiles are based on rendering modes. Rendering modes are a collection of settings, anchored by Internet Explorer Document Modes, designed to provide maximum backward compatibility for business applications that require legacy versions of Internet Explorer. This screen shows the available rendering mode templates:



This chart provides a list of the Internet Explorer rendering modes accessible by Ion:

Rendering Mode	Description
Default	This creates an empty Profile with no Document Mode or customized values. All settings are default settings or blank values.
Adaptive IE Quirks Mode	<p>Renders content by dynamically selecting either the IE7 Standards or Quirks Mode rendering behavior based on the presence and location of a DOCTYPE on the page. Pages will be rendered using the IE7 Standards behavior if the page contains a DOCTYPE in the 1st position (no text preceding the declaration) and that DOCTYPE declaration is anything other than 'QUIRKS'. Pages will be rendered using the Quirks Mode behavior if the page has no DOCTYPE declaration, expressly includes a QUIRKS DOCTYPE declaration or if the DOCTYPE declaration is NOT in the 1st position.</p> <p>In either case the IE6 UA string is declared regardless of which rendering Mode is selected.</p>
Emulate IE7 Mode	Standards Mode directives are displayed in Internet Explorer 7 Standards Mode and Quirks Mode directives are displayed in IE5 Mode.
Emulate IE8 Mode	Standards Mode directives are displayed in Internet Explorer 8 Standards Mode and Quirks Mode directives are displayed in IE5 Mode.
Emulate IE9 Mode	<p>Standards Mode directives are displayed in Internet Explorer 9 Standards Mode and Quirks Mode directives are displayed in IE5 Mode.</p> <p><i>Note: This Profile will default to the highest available version of IE Standards Mode engine on systems running versions lower than IE9. E.g. IE8 systems will use IE8 Standards Mode, etc.</i></p>
Emulate IE10 Mode	<p>Standards Mode directives are displayed in Internet Explorer 10 Standards Mode and Quirks Mode directives are displayed in IE5 Mode.</p> <p><i>Note: This Profile will default to the highest available version of IE Standards Mode engine on systems running versions lower than IE10. E.g. IE8 systems will use IE8 Standards Mode, IE9 systems will use IE9 Standards Mode, etc.</i></p>

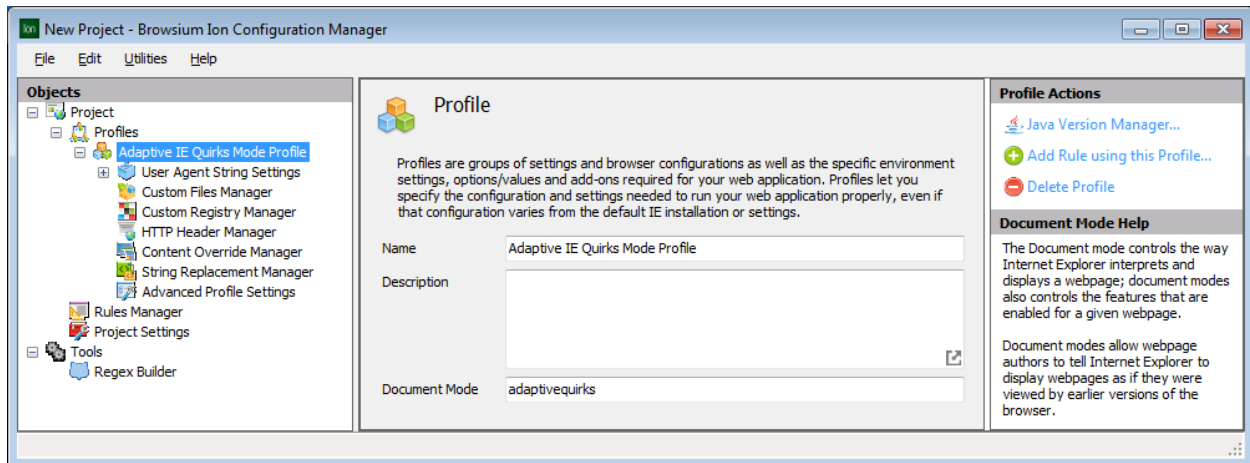
IE Quirks Mode	Renders content as if it were displayed in Quirks Mode by Internet Explorer. IE Quirks is similar to the rendering behavior in IE 5.5. This Profile can be used to force a web application to render using 'Quirks' Mode directives.
IE Edge Mode	<p>To plan for future browser version continuity, Ion includes the 'Default IE Edge Mode Profile' to enable rules to always use the default (latest) browser engine installed on the system.</p> <p><i>Note: Ion supports multiple versions of IE (8, 9, 10). Customers are advised to use caution in selecting this value as it will have important behavioral, rendering and scripting differences when used on different Internet Explorer versions.</i></p>
IE7 Standards Mode	Renders content as if it were displayed in Standards Mode by Internet Explorer 7. This Profile can enable scenarios where a web application renders properly using the IE8's 'IE7 Mode' but the given line-of-business application may require granular setting configurations that are not possible or appropriate changes to make globally to the IE8 browser configuration.
IE8 Standards Mode	Renders content as if it were displayed in Standards Mode by Internet Explorer 8.
IE9 Standards Mode	<p>Renders content as if it were displayed in Standards Mode by Internet Explorer 9.</p> <p><i>Note: This Profile will default to the highest available version of IE Standards Mode engine on systems running versions lower than IE9. E.g., IE8 systems will use IE8 Standards Mode.</i></p>
IE10 Standards Mode	<p>Renders content as if it were displayed in Standards Mode by Internet Explorer 10.</p> <p><i>Note: This Profile will default to the highest available version of IE Standards Mode engine on systems running versions lower than IE10. E.g. IE8 systems will use IE8 Standards Mode, IE9 systems will use IE9 Standards Mode, etc.</i></p>

Any Profile can be modified to suit specific web application compatibility issues. For example, Ion makes it possible to specify different versions of the Java Runtime Environment (JRE) on a per-Profile basis, whereas changing the JRE version for Internet Explorer globally would only allow for a single version to be loaded at a time.

Care should be used when selecting IE Edge, Emulate IE9, Emulate IE10, IE9 Standards and IE10 Standards since the rendering engines, and therefore rendering behavior, will be very different when the Profile is invoked on systems running Internet Explorer versions lower than the mode specified.

3.3.2. Profile Editor

The Profile Editor is used to modify settings, values and configurations for a given Profile. To get started, highlight a Profile to view the details in the Content Pane.



Profile details (Profile Name, Profile Description, and Document Mode) can be edited in this screen. Profiles are assigned default values based on the Template selected when creating a new Profile. Most values can be left at defaults, but Profile Name and Description should be edited to reflect information and labeling relevant to your organization.

3.3.2.1. Profile Editor

Profile

Profiles are groups of settings and browser configurations as well as the specific environment settings, options/values and add-ons required for your web application. Profiles let you specify the configuration and settings needed to run your web application properly, even if that configuration varies from the default IE installation or settings.

Name: Adaptive IE Quirks Mode Profile

Description:

Document Mode: adaptivequirks

This list provides an explanation of the items on the main Profile Editor screen. Additional information about each setting is available from within the Ion Configuration Manager using the 'rollover' information panel display inside the Content Pane.

Profile Name – The Profile name is a friendly name for your reference, and can be modified to suit your organization.

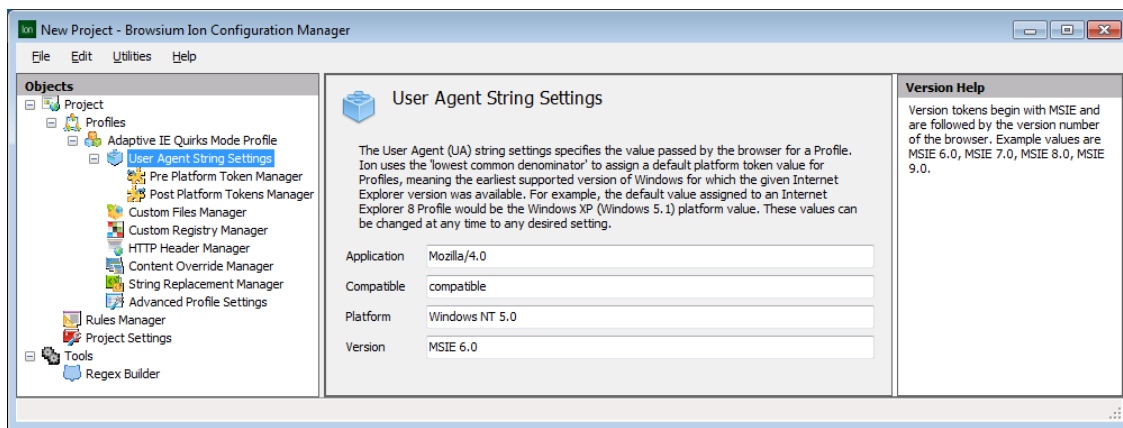
Profile Description – The description field provides a place to include a more complete explanation of the Profile.

Document Mode – Ion allows you to define which Document Mode is included for pages loaded using the specified Profile. Internet Explorer Document Modes are used to trigger different rendering and scripting engine behaviors. The default Document Mode included with each Profile is set based on the rendering mode selected when the Profile is created. The Document Mode can be changed at any time and is exclusive of the rendering mode choice. More information about Document Modes can be found at [http://msdn.microsoft.com/en-us/library/ie/cc288325\(v=vs.85\).aspx#DCModes](http://msdn.microsoft.com/en-us/library/ie/cc288325(v=vs.85).aspx#DCModes).

3.3.2.2. User Agent String Settings

User Agent (UA) Settings let you specify the value passed by the browser for a given Profile. For example, the default UA string used by IE6 is that of a Windows XP Service Pack 2 system. Some common web application compatibility issues can be resolved by simply adjusting the UA string values. Using this feature, Profiles can be configured to send the desired UA string without impacting the standard Internet Explorer UA string values. UA strings are defined in a Profile by the originally selected rendering mode template, but the values can be changed at any time.

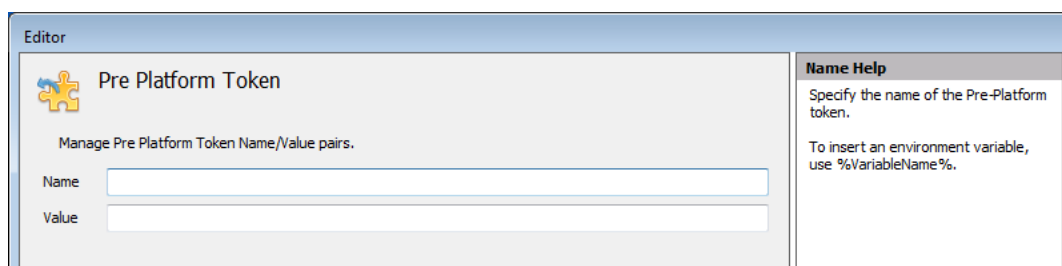
More information on the User Agent string elements and potential values can be found at <http://msdn.microsoft.com/en-us/library/ms537503%28v=vs.85%29.aspx>.

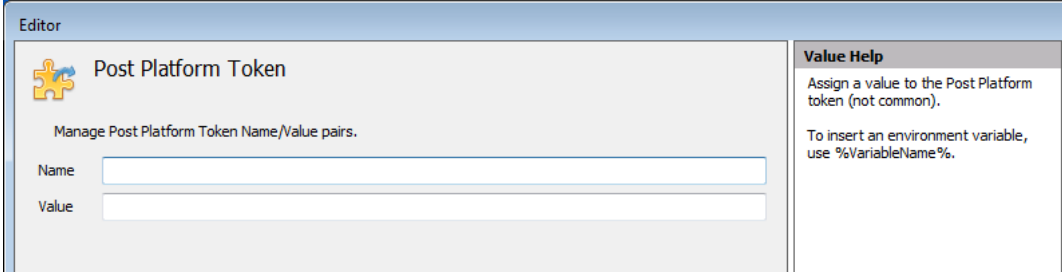


Ion provides the ability to uniquely and individually define each portion of the UA String value. The most commonly adjusted values are Platform and Version.


As a general rule, Profiles created using any of the rendering mode templates will set the UA string values to the minimum supported platform for the Internet Explorer release on which that rendering mode is designed. For Example, the UA string values for an Emulate IE8 Profile will be set as Windows NT 5.1 (Windows XP version number) as IE8 minimum support starts with Windows XP. Platform Token Editor.

In addition to the base UA String values, Ion provides a mechanism to specify Pre- and Post-Platform Token values that are included in the full UA String value.





Editor

 **Post Platform Token**

Manage Post Platform Token Name/Value pairs.

Name

Value

Value Help

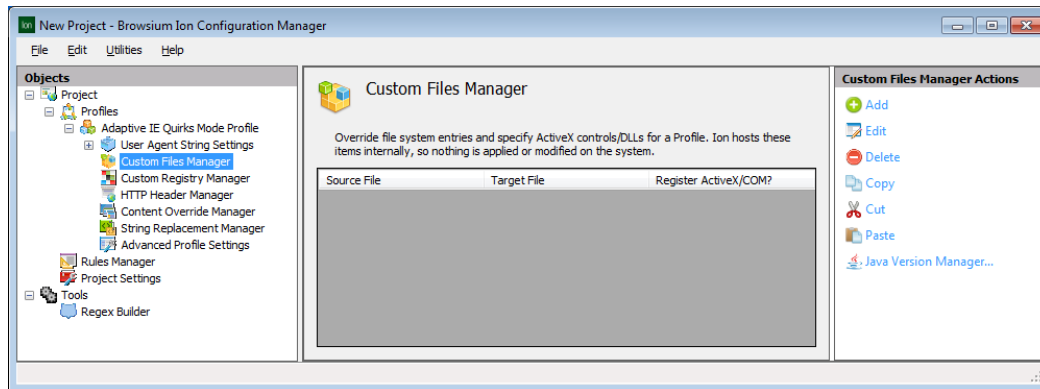
Assign a value to the Post Platform token (not common).

To insert an environment variable, use %VariableName%.

Common uses of the Pre- and Post-Platform Tokens define the installed add-ons and versions. Some web applications examine these values for proper functionality so this feature enables including the required values. Other web applications are unable to process long UA String values and this feature can be used to reduce the set of Pre- and Post-Platform Tokens to a manageable level.

3.3.2.3. Custom Files Manager

Ion allows you to create specific file system entries as well as define custom ActiveX controls for a Profile. Ion hosts these items internally, so nothing is applied or modified on the system. Any required file system entries or ActiveX controls should be defined here.

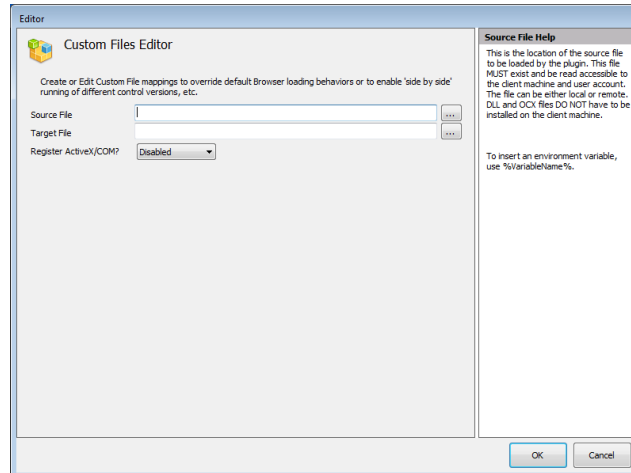


This screen displays all configured custom file and custom ActiveX mappings. To add a value, use the link in the Actions Pane on the right-hand side. To edit a value, highlight the item in the Content Pane and use the 'Edit' link in the Actions Pane. The Custom Files Manager supports the Delete, Copy, Cut and Paste items as well.

The last item listed in the Actions Pane is the Java Version Manager. This wizard is designed to automate the creation of the Custom Files entries required to run a specific version of Java for your web application.

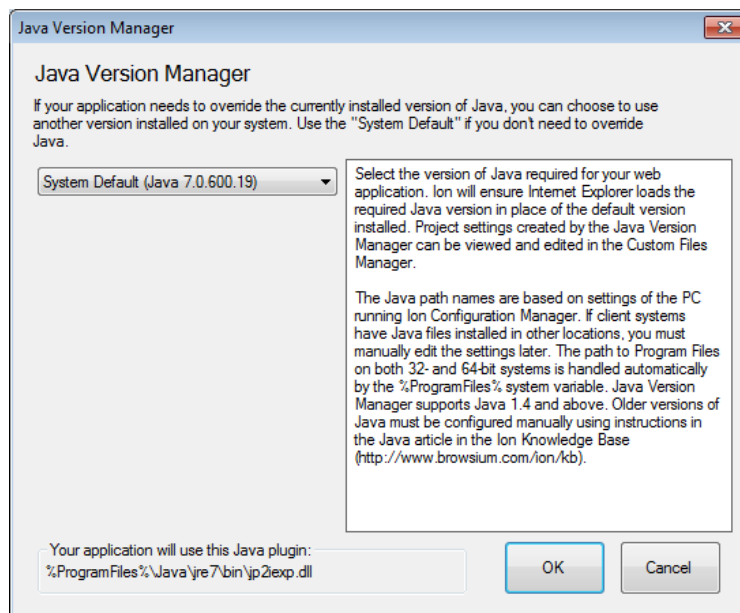
3.3.2.3.1. Custom Files Editor

Use this screen to add or edit Custom File mappings. More information on this feature and an example of creating a Custom File mapping can be found in the [Working with Custom Files, Registry Settings and ActiveX Controls](#) section.



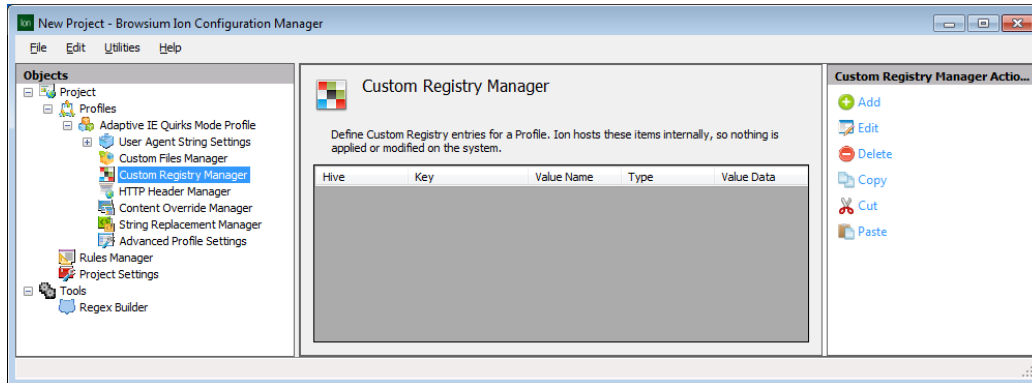
3.3.2.3.2. Java Version Manager

Using the Java Version Manager is the easiest and fastest way to define specific versions of Java for use by your web application. By default, Internet Explorer and the JRE are designed to use the most recent version of Java installed on the system. The Java Version Manager provides a dropdown list of all Java versions installed on the system running Ion Configuration Manager, enabling you to simply select that version and the wizard creates the Custom Files entries. More information on this feature and an example of creating a Custom Files mapping can be found in the [Java Version Manager](#) section.



3.3.2.4. Custom Registry Manager

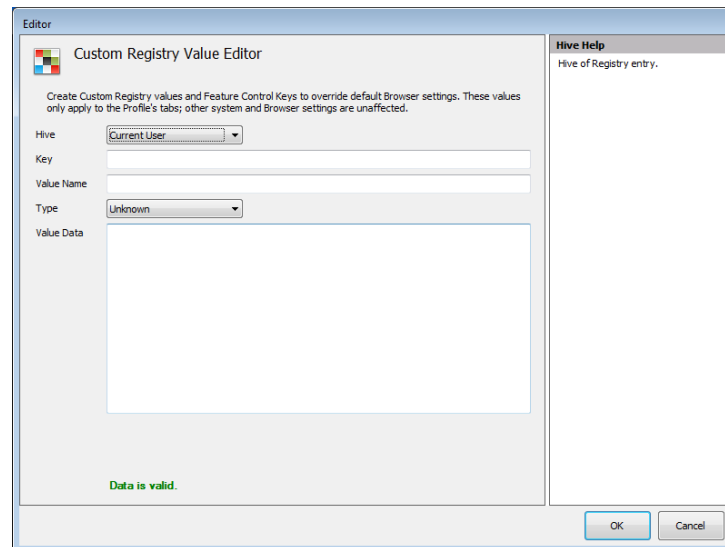
This screen allows you to create specific registry entries for a Profile. Ion provides an environment to host these items internally, so nothing is applied or modified on the system. Any required registry entries should be defined here.



To add a value, use the link in the Actions Pane on the right hand side and select the 'Add' item. To edit a value, highlight the item in the Content Pane and use the 'Edit' item in the Actions Pane. The Custom Registry Manager supports the Delete, Copy, Cut and Paste items as well.

3.3.2.4.1. Custom Registry Value Editor

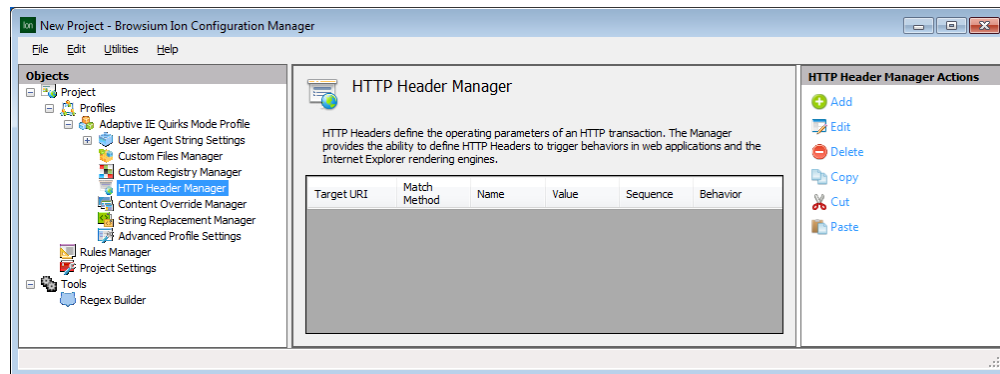
Any type of Registry entry can be created using this interface. Simply choose the Hive location and entry values, and then click OK.



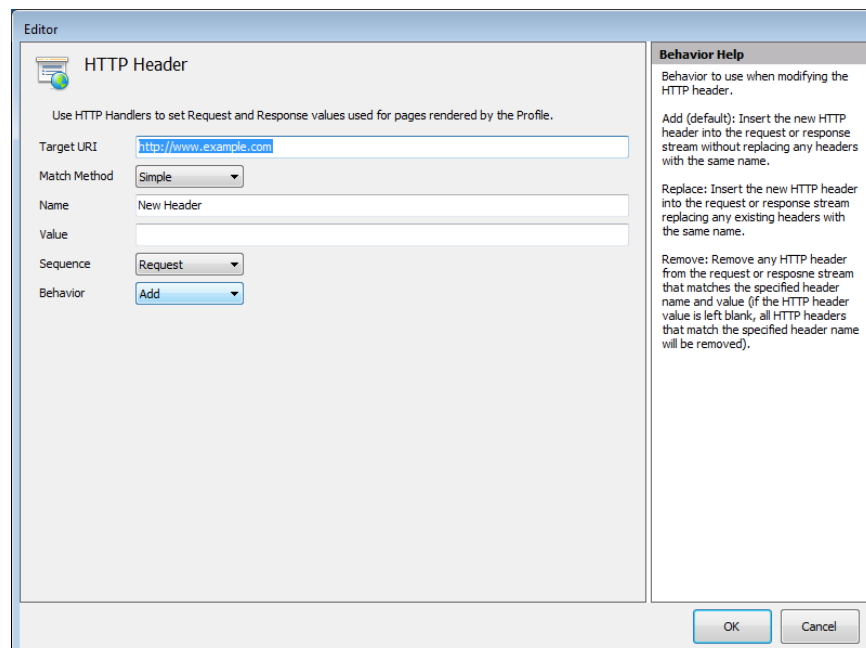
For more information on working with Custom Registry settings, see the [Working with Custom Files, Registry Settings and ActiveX Controls](#) section.

3.3.2.5. HTTP Header Manager

This screen displays the list of defined Custom HTTP headers. HTTP header fields are components of the message header of requests and responses and some web applications rely on these headers to trigger certain behaviors and functionality. Ion enables easy creation and management of HTTP Headers.



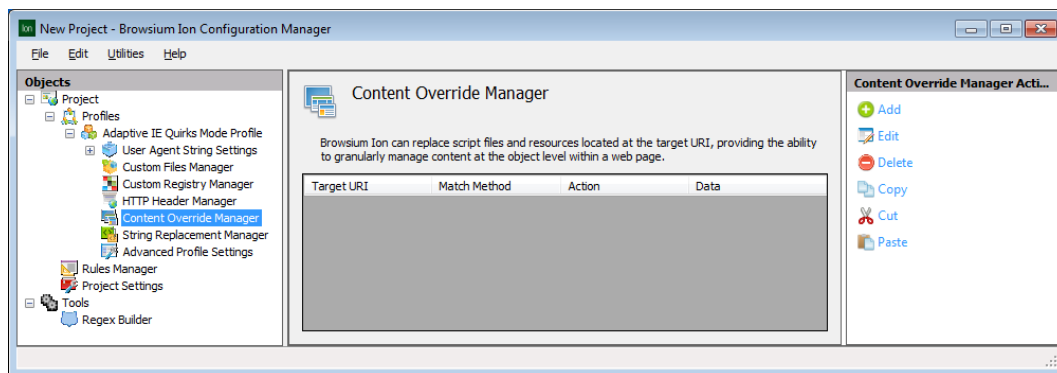
A list of available HTTP Header fields can be found at http://en.wikipedia.org/wiki/List_of_HTTP_header_fields. To add an HTTP Header, use the link in the Actions Pane on the right-hand side and select the 'Add' item. To edit a value, highlight the item in the Content Pane and use the 'Edit' item in the Actions Pane. The HTTP Header Manager supports the Delete, Copy, Cut and Paste items as well.



3.3.2.6. Content Override Manager

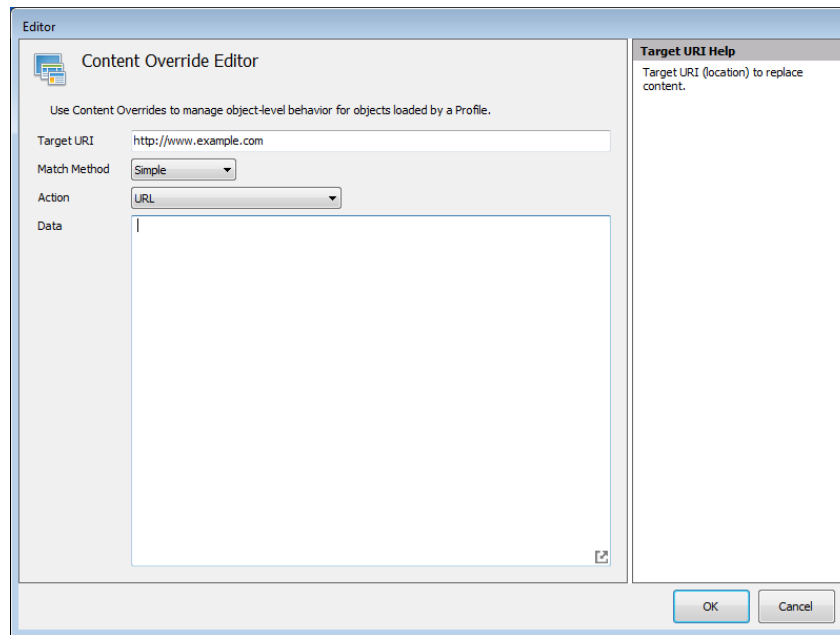
Some older web applications may include content (JavaScript, HTML, CSS, images, etc.) resulting in compatibility issues with newer browser rendering engines that cause the application to work incorrectly. These issues can be as simple as layout issues, more complex coding related issues that prevent the site from loading or even serious issues that cause the web browser to crash when loading specific content.

Ion provides the ability to replace or block inline or linked content for a given web page to help solve these compatibility issues.



3.3.2.6.1. Content Override Editor

Start by defining the Target URI – even though these settings are part of a Profile which would be triggered by a Rule, Content Overrides have an additional layer of granularity and work with content regardless of the location so the Target URI must be specified for the content being replaced.



The Content Override feature provides the ability to control the loading behavior taken by the browser for the specified content. The available Actions are:

- **Block:** Stops the content from being downloaded.
- **URL (Default):** Redirects the request for the content at the specified URI to the URL specified in the Data field.
- **File – Body Only:** Loads content from the file specified in the Data field.
- **File – Header and Body:** Loads content from headers and the file specified in the Data field.
- **Content:** Replaces content from the specified URI with the content in the Data field.

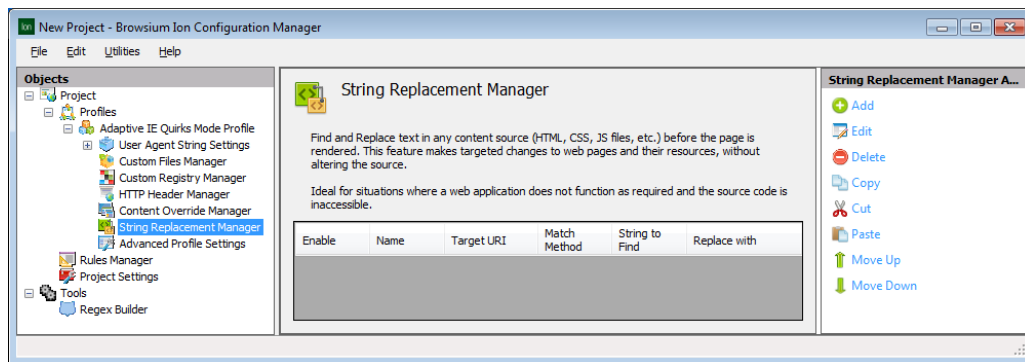
The Data field will contain content based on the selected Action.

In addition to application compatibility needs, the Content Override feature can be used to augment security and prevent downloading known malicious or restricted content. For example, using the Regex match method an organization can build a complex matching string to prevent downloading any exe, dll or other data types.

3.3.2.7. String Replacement Manager

Some older web applications need simple, targeted fixes to make an application work exactly right. Often the issues can be as simple as resizing an image or changing a word in some JavaScript. Changing the web application source code is a great way to go – but many organizations are using ‘off the shelf’ applications for which they don’t have source code access to change what is needed. Some commercial applications have license agreements that prohibit any modification to their source code or binaries. In addition, changing the source code is impractical for ‘roll outs’ where some users will remain on older browsers while some users get the latest versions.

Ion offers the ability to edit text ‘inline’ before the browser rendering begins, enabling an organization to fix virtually any issue without touching the source code. These changes are highly targeted and can be used to make pinpoint changes to the affected area of the web application.



Making a string replacement is easy. It all starts with the Target URI – even though these settings are part of a Profile which would be triggered by a Rule, String Replacements are designed to work with content regardless of the location so the Target URI must be specified for the strings being replaced.

3.3.2.7.1. String Replacement Editor

Browsium has worked with dozens of customers to resolve highly complex web application compatibility issues. Many of the more complex issues require a series of String Replacements. Since any given replacement can be targeted at a highly specific area, Ion includes data fields to define a Name and Description of each string replacement.

This metadata information is not used by Ion but provides administrators or developers an easy and convenient place to read notes about each item that could be used to convey history on that issue and avoid others from having to troubleshoot from scratch.

String Replacements can be easily enabled or disabled via the Enable checkbox, easing configuration development and troubleshooting.

The core of a String Replacement is the string to find – the pattern match can be Simple or RegEx depending on the Match Method property selected.

Enter the value to be used as the replacement text and click OK to finish.

String Replacement

Find and Replace text in any content source (HTML, CSS, JS files, etc.) before the page is rendered. This feature makes targeted changes to values within web pages and their resources, without needing to alter the source.

Name:

Description:

Enable: ☒

Target URI:

Match Method:

Find/Replace Method:

String to Find:

Replace with:

Name Help

This field is used to name the item and should be meaningful to your organization. Names should be short and clear, with the Description field used to provide more detailed information.

While two items can have the same name, care should be taken to use unique names to avoid confusion.

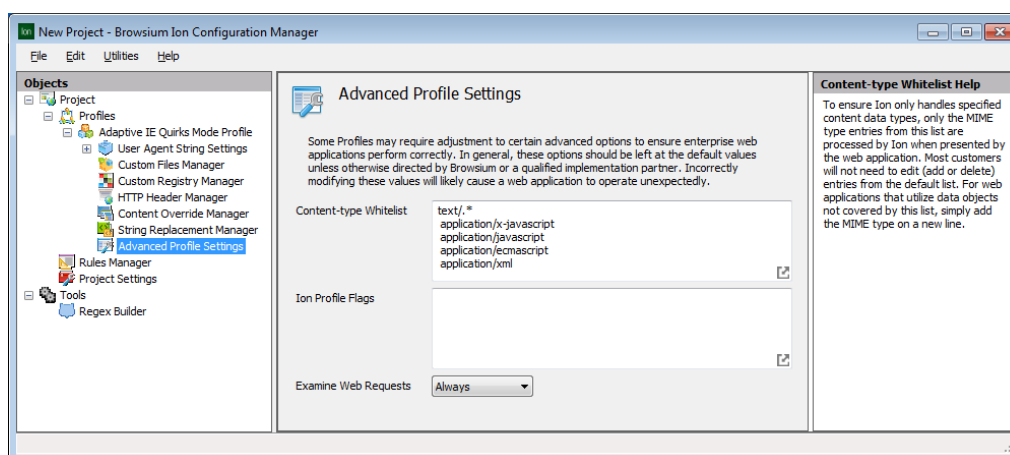
OK Cancel

3.3.2.8. Advanced Profile Settings

Since web applications can be a combination of different development technologies, some Profiles may need some additional advanced configurations. For example, a web application can integrate JavaScript (interpreted) code while also calling an ActiveX control that runs outside the IE process model. Offering an array of options to developers with which to make their applications enables web development to be extremely powerful. Those same benefits can mean a specific application has unique and specific requirements.

To help accommodate the unique scenarios that may be required, Ion offers additional behavioral and environmental controls using these options. As a rule, most web applications will work appropriately without modifying any of the entries or values in this area.

Customers are encouraged to avoid adding, deleting or modifying the settings below unless directed by Browsium or an integration partner. Incorrect modification or deletion of the items here can result in unexpected and unwanted application behaviors.



3.3.2.8.1. Content-type Whitelist

The Content-type Whitelist feature is provided to address complex and unique web application scenarios that require uncommon data types. By default, Ion is designed to handle only those data types listed here. Any Content-type not included on this list will be ignored by Ion and passed along to Internet Explorer for handling by the IE MIME type handler function.

In the case where some environmental, string manipulation or other modification is required of a given Content-type, the value should be added here. Adding additional Content-types is easy. Simply find the 'Click to add a new item...' line, select that entry and begin adding the required value. The values in the Content-type Whitelist are freeform text, not selected from a list, so care should be taken to ensure spelling and syntax are correct or Ion will not function as expected.

Customers are encouraged to avoid changing or deleting any of the pre-defined Content-type Whitelist entries. Incorrect modifications or deletions may result in Ion behaving unexpectedly.

3.3.2.8.2. Ion Profile Flags

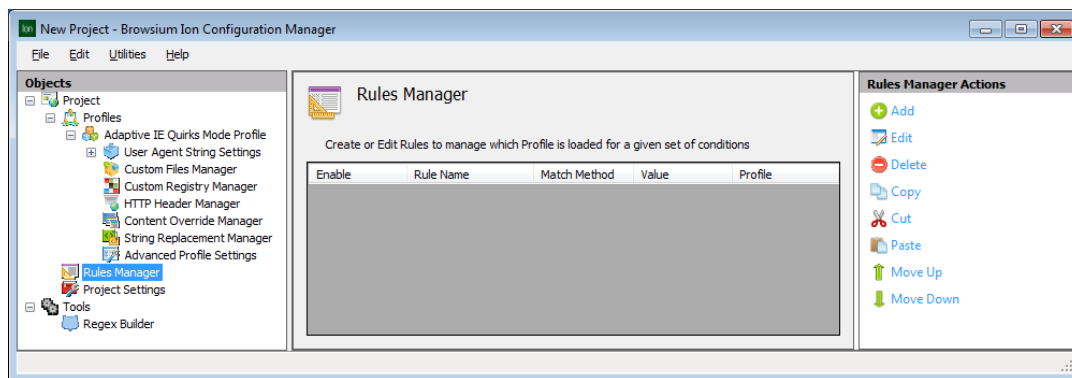
Ion Profile Flags provide a mechanism to add additional functionality, logging and management capabilities without the need to update the Ion binaries. The feature is designed mainly for use in complex debugging scenarios. Certain Ion Profile Flag values can also be used to extend Ion functionality to meet an array of unique and specific edge case scenarios. Unless directed by Browsium or an integration partner, the Ion Profile Flags value should remain blank.

3.3.2.8.3. Examine Web Requests

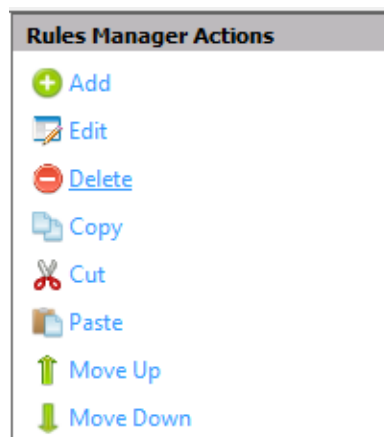
Examine Web Requests determines when to send web requests and responses through the internal proxy for examination and content modification. 'Always' bypasses the routing logic and forces all traffic through the internal proxy. Auto allows Ion to decide and reduces proxy routing where possible. By default, Profiles are set to use 'Always Examine Web Requests' to ensure consistent application handling.

3.4. The Rules Manager Node

The Rules Manager is the main interface for creating, editing and managing evaluation criteria to determine which websites are rendered using an Ion-managed instance instead of the default instance of Internet Explorer. This section provides details on the various elements of this interface.



The Content Pane shows the hierarchical rules list that Ion uses to determine how to handle web application rendering. The heading for each column in this window refers to the specific rule element (e.g. Rule Name, Match Method, Value, Profile) for a given Rule. Use the Actions Pane to work with Rules using the 'Add', 'Edit', 'Delete', 'Copy', 'Cut' and 'Paste' items. The Actions Pane also contains the 'Move Up' and 'Move Down' items for re-ordering Rules.



Add Rule - To create a new rule, click the Add Rule link in the Actions Pane to bring up the Rule Editor window. The next part of this section provides details on the options and values in the Rule Editor window. See the [How to Create a Rule Section](#) for details on creating rules.

Edit Rule - URLs and rule definitions can be complex, and complexity may lead to typographical errors. In the event you make a mistake or need to revise a Rule, simply double click the line to edit (or highlight the line and click the Edit Rule link in the Actions Pane) to make the necessary changes.

Delete Rule - To delete a Rule, select it from the Rules Manager Window, then click the Delete Rule link in the Actions Pane.

To disable a Rule rather than remove it, uncheck the checkbox in the Enable column.

Copy – To copy a Rule, select the item from the Rules Manager Window and click the Copy item in the Actions Pane.

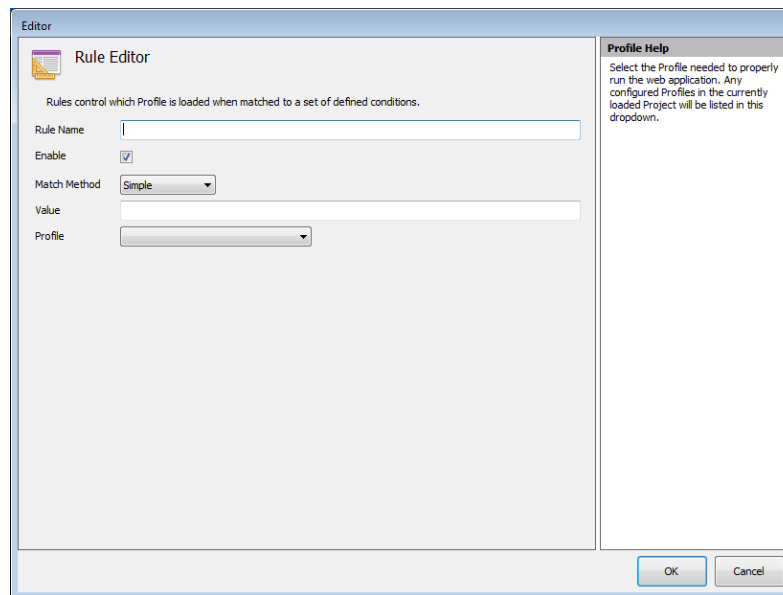
Cut - To cut a Rule (which copies the item to the clipboard rather than delete it entirely), select the item from the Rules Manager Window and click the Cut item in the Actions Pane.

Paste – The Paste item in the Actions Pane can be used to paste items from the clipboard to the currently loaded, active Project. For example, you can Copy/Paste items within a single Project or Copy from one Project and Paste into another.

Move Rule Up/Move Rule Down - By default, rules are ordered in the sequence they are added. Since rules are evaluated in the order they are stored, the sequence of rules can be critical to the proper functionality of your web application in Ion. To manually adjust the order of a Rule, simply highlight the Rule and use the Up and Down buttons to move it to the proper placement.

3.4.1. Rule Editor

The Rule Editor window is used for creating new Rules or editing existing Rules.



Rule Name - You can name a Rule using any characters without restriction or character count limit. Rule names are for your use and identification only, and have no impact on functionality. Browsium recommends using Rule names that clearly describe what the Rule is used for, so it can be easily identified in the Rules Manager display.

Enable (checkbox) - All Rules are enabled by default when created. Some Rules may contain complex parameters or complex paths and it may be more practical to simply disable a Rule rather than remove it if the Rule is not needed or to test ordering behaviors, etc. In addition, some users may choose to keep only a single Rule set and want to disable a specific Rule for some given period of time. To disable a Rule, click the checkbox to remove the check or click the checkbox next to the Rule name in the Rules Manager.

Match Method – Most Rules require a simple pattern match of the URL in order to trigger Ion to manage the web application experience and load the required Profile. Some complex URL configurations require advanced pattern matching techniques, so Ion provides the ability to match based on regular expressions (Regex) if needed.

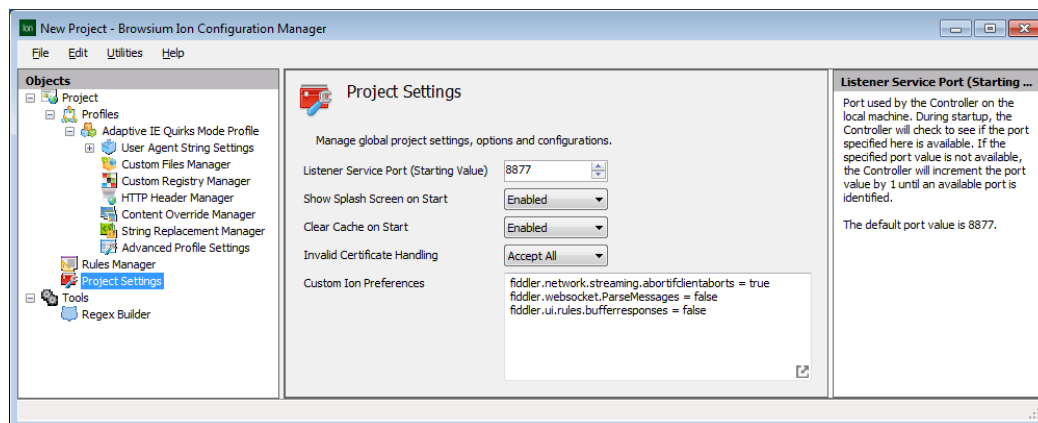
Value – This field should contain the string or integer to conditionally match in order to Ion to manage content rendering.

Profile - The Profile selection determines which browser Profile Ion uses to load the matching website. Any configured Profiles will be listed in this dropdown. In a default configuration there

are no Profiles available. To change the Profile used for a given Rule, simply change the value in this field and save the configuration.

3.5. The Project Settings Node

The Project Settings node gives you the ability to edit global settings for Ion configurations that will be applied to all Profiles. These settings encompass features such as the Listener Service Port, Load from File Name parameters, and generic security and loading behaviors.



3.5.1. Listener Service Port

Port used by the Controller on the local machine. During startup, the Controller will check to see if the specified port is available. If that port is not available, the Controller will automatically increment the port value by 1 until an available port is identified. The default port value is 8877.

3.5.2. Show Splash Screen on Start

Ion defaults to showing the Ion splash screen (Ion graphic with the product name and version number) on client systems when the Controller is started. The splash screen is only displayed at startup (usually set to user login) so users should not regularly see this display. This option can be disabled during project configuration.

3.5.3. Clear Cache on Start

By default, Ion will clear the Internet Explorer cache when the Controller starts and detects a configuration change since the last time it started. This is done to ensure proper rendering and eliminate any potential of older/stale files from being loaded via the system cache versus being

obtained from the server. When files are loaded from the local system cache, Ion may be unable to properly manage the web application and cannot ensure configurations will work as expected.

3.5.4. Invalid Certificate Handling

When presented with invalid HTTPS Certificates, Internet Explorer 6 did not generally warn users or prevent navigations. HTTPS Certificate handling behavior has become more secure in recent versions of Internet Explorer, such that invalid HTTPS Certificates will trigger blocked navigation, user prompts or related behaviors that prevent web applications from functioning as they did previously. Ion provides the ability to manage how invalid HTTPS Certificates are handled. Available options are:

- **Reject All:** Rejects all invalid HTTPS Certificates and navigation is cancelled.
- **Accept All (Default):** Accept all invalid HTTPS Certificates and continue with navigation.

3.5.5. Custom Ion Preferences

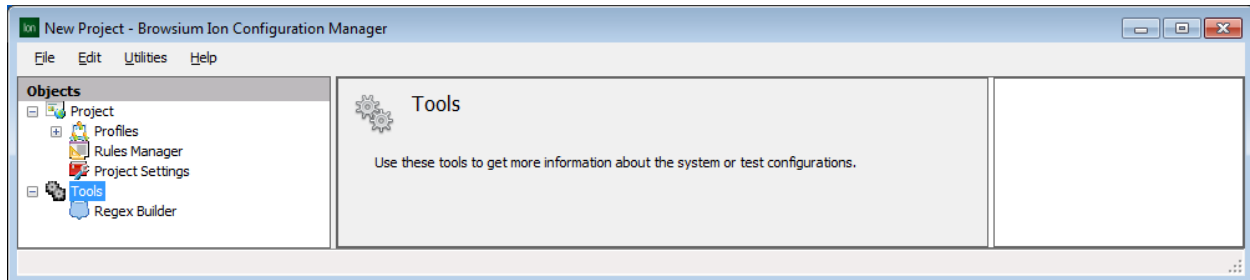
This section is used to add features or product behaviors that may be needed by customers for unique and specific browser or web application settings. Ion provides the ability to control sub-component behaviors through these preference options. Customers running older or custom developed web applications may need the ability to control Ion to respond in non-standard ways for specific and defined scenarios.

For example, an organization may have an application that is designed to force users to present login credentials rather than use their current login user information. Ion is designed to work only using the integrated Windows user credentials, as controlled by Internet Explorer settings. In this case a custom Ion preference could be used to change the default Ion behavior rather than modifying global IE settings that may impact other applications where the logged in user credentials are preferred.

To avoid unexpected behavior, the default options should not be modified.

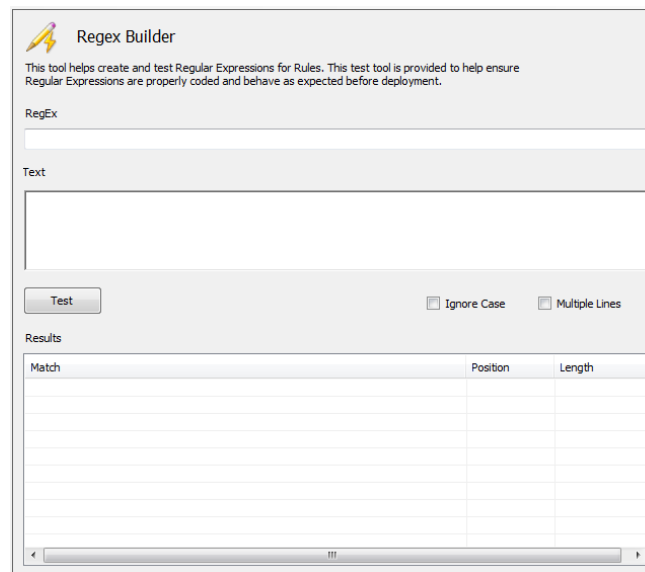
3.6. The Tools Node

The Tools Node was designed to provide a set of 'helper' applications for creating, testing and designing Profiles and Rules. Additional Tools may be added later, and can be installed by simply placing the specific DLL file in the directory with the Browsium Ion Configuration Manager.



3.6.1. Regex Builder

The Regex (Regular Expression) Builder is for administrators to create and validate complex Regular Expressions that can be used in the rules evaluation criteria.



The Regex Builder provides a simple integrated tool for writing and testing Regular Expressions before using them as rules evaluation criteria to help avoid errors in pattern matching.

For more information on the Regex Builder, including a step by step example, please read the 'Using regular expressions in Ion rules' document in the Knowledge Base section of <http://support.browsium.com>.

More information on building and testing Regular Expressions can be found online, including regexhero.net or regexpal.com.



Section Four

Configuration Tutorial

In this section you will learn:

- ✓ How to Create a Profile
- ✓ How to Create Rules in Ion
- ✓ How to Remove Rules in Ion
- ✓ How to Work With Custom Files and Registry Editor Options

4. Configuration Tutorial

Once the Browsium Ion Configuration Manager installation is complete, you can begin configuring which sites to remediate with Ion using the Ion Configuration Manager. The Ion Configuration Manager is provided as a simple interface to create, edit, and test Ion configurations before deployment. As a security design, Browsium Ion only uses Ion Profiles to manage the settings for sites explicitly identified in Ion Rules. This tutorial will take you through the process of creating and testing Ion Profiles and Rules.

Systems must have the Browsium Ion Client installed to test or use the configuration created in the Ion Configuration Manager.

4.1. How to Create a Profile

Ion Profiles provide the ability to bind specific custom file system, registry and file settings to a specific rendering mode. Ion includes eleven Profiles templates that can be used to create Profiles: 'Default', 'Adaptive IE Quirks', 'Emulate IE7 Mode', 'Emulate IE8 Mode', 'Emulate IE9 Mode', 'Emulate IE10 Mode', 'IE Quirks Mode', 'IE Edge Mode', 'IE7 Standards Mode', 'IE8 Standards Mode', 'IE9 Standards Mode' and the 'IE10 Standards Mode'.

Any Profile can be modified to suit your needs after it is created, but the templates will remain unchanged. Some organizations may not need to make any customizations to the settings defined in the template, while others will require several changes to a Profile in order to make their web applications function properly.

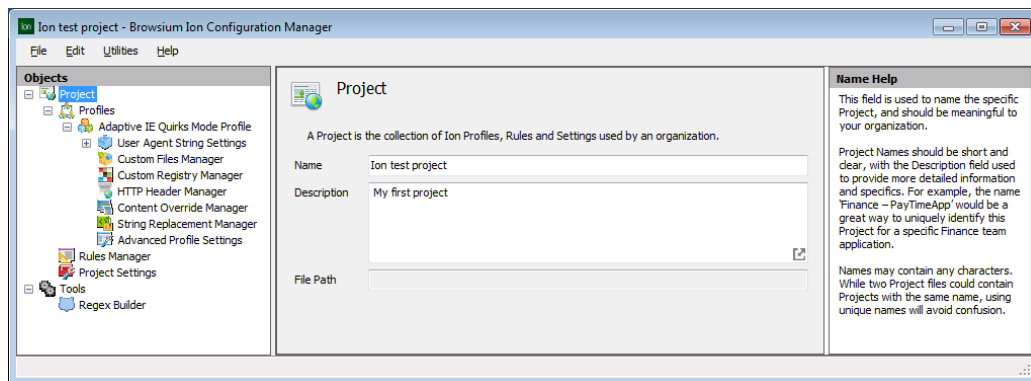
This table shows the specific browser engine version that will be loaded in each scenario:

Document Mode	If IE8 is installed	If IE9 is installed	If IE10 is installed
Adaptive IE Quirks	IE8 Quirks Mode or IE7 Standards Mode (as evaluated by DOCTYPE)	IE9 Quirks Mode or IE7 Standards Mode (as evaluated by DOCTYPE)	IE10 Quirks Mode or IE7 Standards Mode (as evaluated by DOCTYPE)
Emulate IE7 Mode	Emulate IE7 Mode for IE8	Emulate IE7 Mode for IE9	Emulate IE7 Mode for IE10
Emulate IE8 Mode	IE8 Standards Mode	Emulate IE8 Mode for IE9	Emulate IE8 Mode for IE10
Emulate IE9 Mode	IE8 Standards Mode	IE9 Standards Mode	Emulate IE9 Mode for IE10
Emulate IE10 Mode	IE8 Standards Mode	IE9 Standards Mode	IE10 Standards Mode
IE Quirks Mode	IE8 Quirks Mode	IE9 Quirks Mode	IE10 Quirks Mode
IE Edge Mode	IE8 Standards Mode	IE9 Standards Mode	IE10 Standards Mode
IE7 Standards Mode	IE7 Standards Mode for IE8	IE7 Standards Mode for IE9	IE7 Standards Mode for IE10
IE8 Standards Mode	IE8 Standards Mode	IE8 Standards Mode for IE9	IE8 Standards Mode for IE10
IE9 Standards Mode	IE8 Standards Mode	IE9 Standards Mode	IE9 Standards Mode for IE10
IE10 Standards Mode	IE8 Standards Mode	IE9 Standards Mode	IE10 Standards Mode

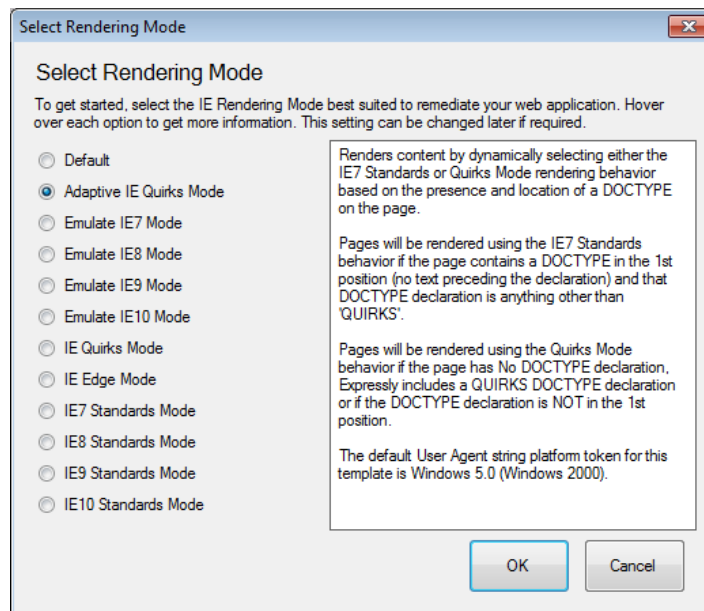
The following steps demonstrate how to create a Profile based on the 'Adaptive IE Quirks' rendering mode to render sites and load an older version of Java. By design, Internet Explorer can only load the latest version of Java installed on the PC. This design is ideal for security and developer purposes, but it impedes the ability to ensure web application compatibility. This example will illustrate how to create a Profile to load Java 1.4.2 update 19 (1.4.2_19) instead of the newer Java 1.7 update 60 (1.7.0_60) version installed on the system.

Additional information on this topic, including information on running other combinations of Java or how to run the MSJVM can be found online at <http://support.browsium.com> in the “Managing Java with Ion” document.

- 1) Open the Ion Configuration Manager. A new Project is created by default. There are a variety of ways to open existing Project files, but this section of the guide assumes this is your first time creating an Ion Project. Therefore we'll have you start building your configuration from an empty new Project.
- 2) Begin by naming your project “Ion test project” and give it the description “My first project”.

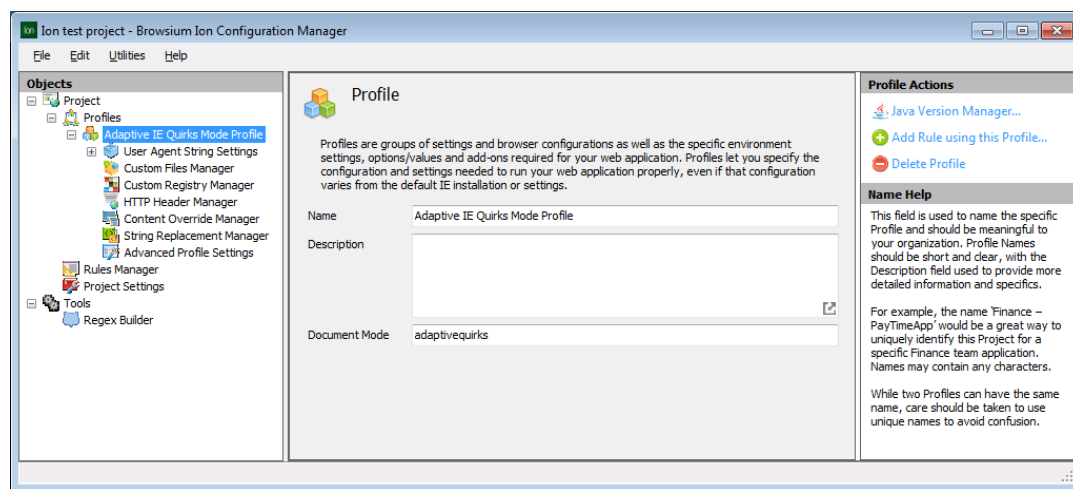


- 3) Click on the Profiles Node, then click the 'Add Profile' link in the Actions pane (or right-click on the Profiles Node and select 'Add Profile') to open the Select Rendering Mode screen:



Select the Rendering Mode needed for your web application and click 'OK'. If you are unsure which Rendering Mode to use, Browsium recommends starting with the Adaptive IE Quirks Mode for legacy web applications. With newer web applications you can start using the latest available Rendering Mode and then working backwards as needed.

- 4) Enter a name and description for the new Profile. The name and description fields are for reference purposes only and can be anything useful or meaningful to your organization and environment. Profile Names and Descriptions can be changed at any time.



The Profile will be pre-populated with the most commonly needed settings for the specified Profile. Changes to these values can be made at any time.

4.2. Working with Rules

Rules are the basis for how Ion is invoked to load content in managed IE instances. The Rules system is simple and powerful, but working with it requires some care to ensure the behaviors are as expected. This section provides the basics on creating and editing Rules using the Ion Configuration Manager.

4.2.1. Creating a Rule

Ion provides several options for rule matching in order to meet the specific needs of your environment. In this example we have identified a website, www.aggrid.com, which is not compatible with IE8 and above, so we need to make a Rule to enable Ion when users visit that web application.

The easiest way to create a Rule using the Profile you just created is to use the 'Add Rule using this Profile' link in the Actions pane. Rules can also be created using the following steps:

1. Click the Rules Manager Node, and then click the '**Add Rule**' link in the Actions pane to bring up the Rule Editor window. Enter a name for the Rule. Rule names are friendly names for organizational and identification purposes only and have no effect on the behavior of a rule. For this example, we will choose "Ion Demo Site".

Keep the Enable checkbox selected in order to make this Rule active for use.

Editor

Rule Editor

Rules control which Profile is loaded when matched to a set of defined conditions.

Rule Name:

Enable: ☒

Match Method:

Value:

Profile:

Enable Help

By default, this item is set to Enable. If set to Disable, it will not be loaded.

An organization may choose to set the item to Disable rather than delete it from a Project file in case it is needed at a later date.

OK Cancel

2. There are two Match Method options for ensuring the Rule is triggered when desired conditions are met. The Simple method does a simple pattern match within the requested resource, and the RegEx method is provided for scenarios that require a set of complex matching criteria. Most Rules will only need the Simple method. For this example we will leave everything set to the default.
3. Enter a Value to check for Rule matching conditions. For this example we will use the domain name for our application - aggrid.com.

The screenshot shows the 'Rule Editor' dialog box. The title bar says 'Editor'. Inside, the main area is titled 'Rule Editor' and contains the text 'Rules control which Profile is loaded when matched to a set of defined conditions.' Below this, there are several fields: 'Rule Name' with the value 'Ion Demo Site', 'Enable' with a checked checkbox, 'Match Method' with a dropdown menu set to 'Simple', 'Value' with the text 'aggrid.com', and 'Profile' with an empty dropdown menu. On the right side, there is a 'Rule Name Help' section with two paragraphs of text. At the bottom right, there are 'OK' and 'Cancel' buttons.

Rule Editor

Rules control which Profile is loaded when matched to a set of defined conditions.

Rule Name: Ion Demo Site

Enable: ☒

Match Method: Simple

Value: aggrid.com

Profile:

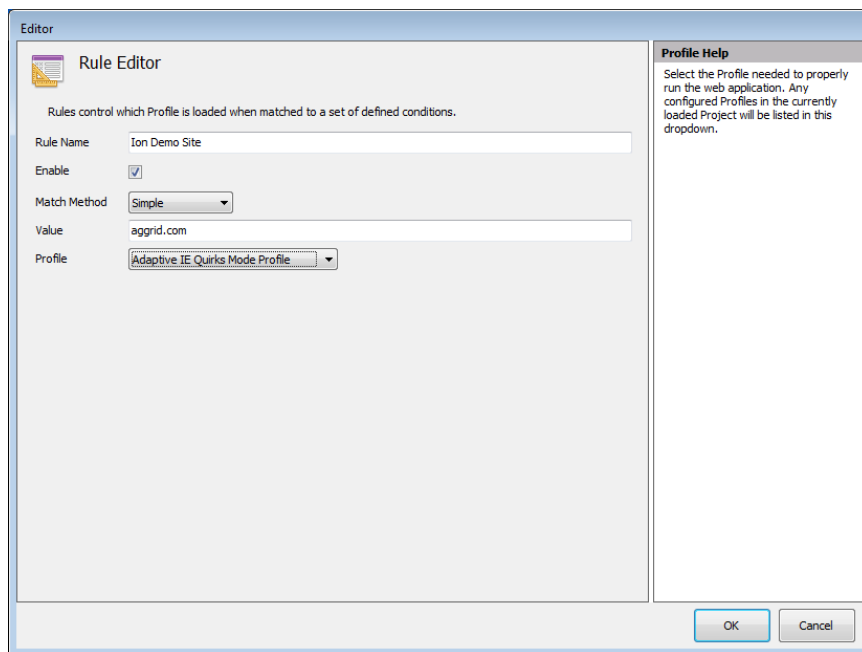
Rule Name Help

This field is used to name the specific Rule, and should be meaningful to your organization. Rule Names should be short and clear and may contain any characters. For example, the name 'HR - Benefits System' would be a great way to uniquely identify this Rule for a specific HR team application.

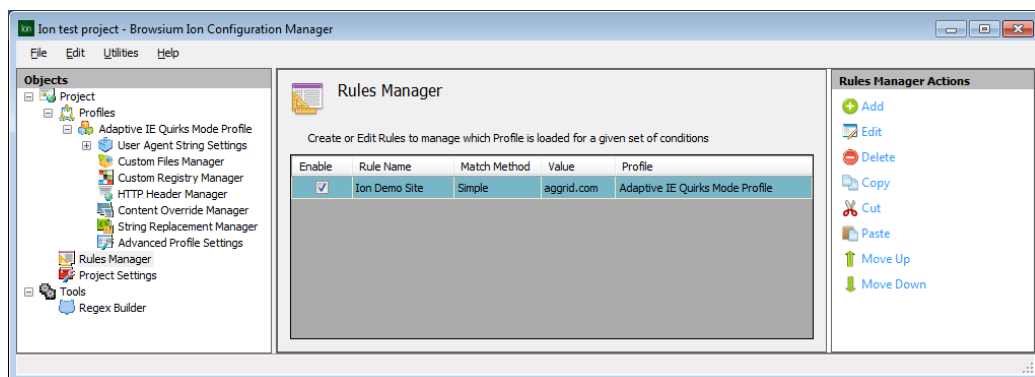
While two Rules could contain the same name, care should be taken to use unique names to avoid confusion.

OK Cancel

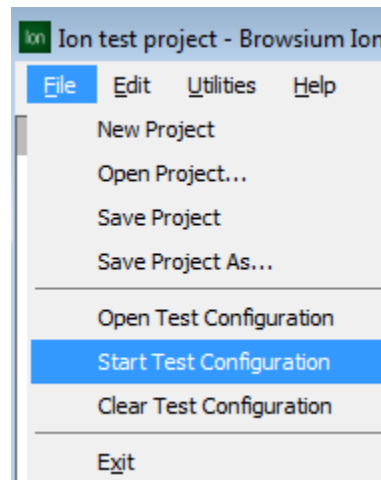
4. If you used the Action pane link to create this Rule, the Profile is already defined. Otherwise simply choose "Adaptive IE Quirks Mode Profile" as the Profile for this rule.



5. When you are done creating the Rule (or changing a setting), remember to click the '**OK**' button to save the Rule to the Rules Manager.

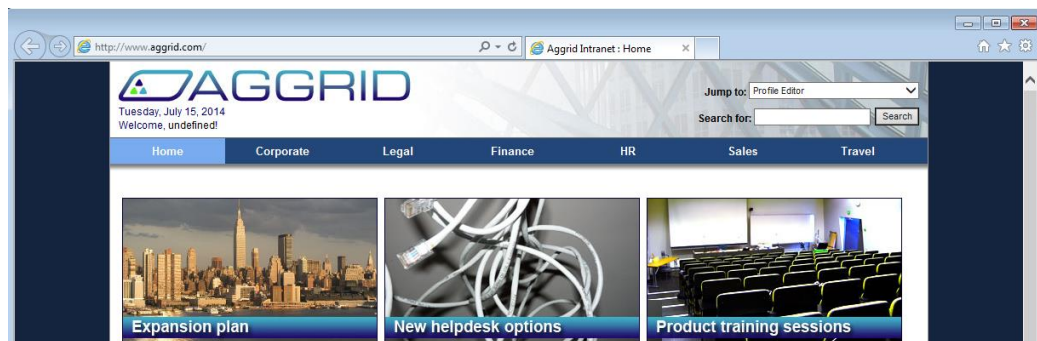


6. Using the 'Start Test Configuration' menu option will prompt you to save the project file and then immediately and automatically apply project file pointers directly in the system registry and start the Ion Controller to process the new configuration.



Projects should be saved regularly to ensure work is not accidentally lost. Ion Configuration Manager does not auto-save work in progress.

7. Once the test configuration has been started, simply open Internet Explorer and browse to www.aggrid.com. The Aggrid website will open in an Ion-managed instance of Internet Explorer.



4.2.2. Copying a Rule

Creating Rules is fast and easy, but some customers have demonstrated the need to create nearly identical Rules for several related sites. For example, your organization may have a development, test and production version of the same system. Each of those systems have unique hostnames, but the URL string is identical beyond that. Ion 3.3 includes the ability to copy/paste a Rule to quickly duplicate it rather than having to create it from scratch again and again.

The process is simple. Just select the Rule you need, click the 'Copy' option in the Actions pane and then hit 'Paste' to add a duplicate entry into the Rules Manager. Pasted Rules will be appended to the end of the existing Rules list so you may need to re-order Rules when finished copying all of the needed Rules. Rules can be re-ordered by simply selecting the Rule and using the Move Up or Move Down items in the Actions Pane.

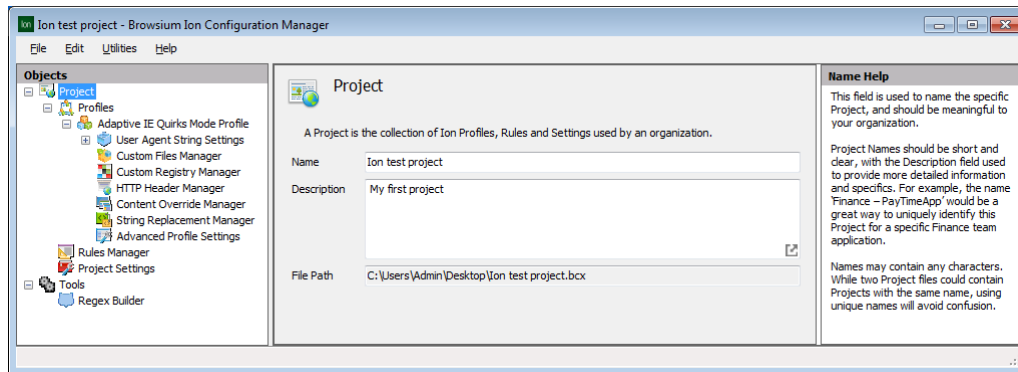
The Ion Configuration Manager uses both the Copy and Paste steps, rather than a single 'Duplicate' option since the process is designed to be the same for copying within a Project as well as across multiple Projects. A single instance of the Ion Configuration Manager can only load a single Project file at a time, so working with two Project files requires loading each in a unique Ion Configuration Manager instance. For example, if your organization has separate Projects for different business units and deploys configurations based on OU membership, copying Profile and Rule settings between configurations is easy. Just open each Project file in its own Ion Configuration Manager, select the item you want, hit Copy and switch to the target Project file and hit Paste.

4.2.3. Removing a Rule

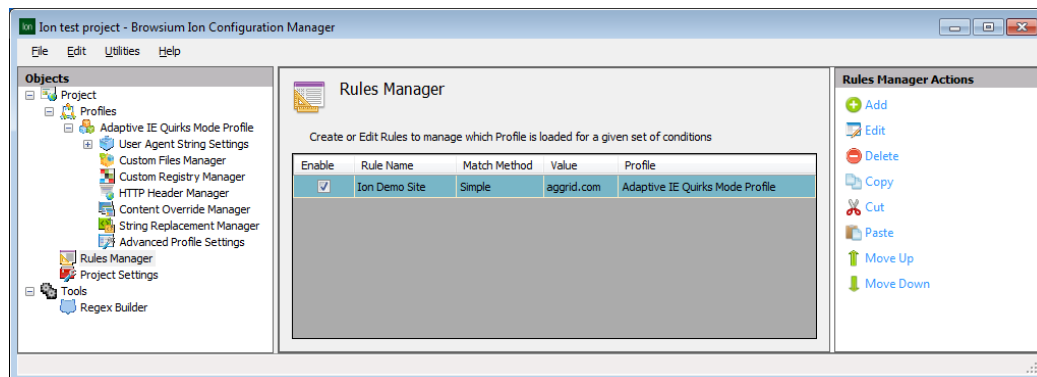
Rules are easily removed using the Browsium Ion Configuration Manager when it is no longer needed. In this example, assume we have taken the needed steps to mitigate the compatibility issues with the web application running on www.aggrid.com, so we no longer need to render the site using Ion.

You can remove a Rule by following these steps:

1. Open the Ion Configuration Manager and open the Project containing the Rule you want to remove, or simply double-click on the Project file (*.bcx) in Windows Explorer.



2. With the Project open, click the Rules Manager node to bring up the ordered list of Rules. Select the Rule you wish to remove from the list of Rules.



3. Click the Delete Rule in the Actions pane.

Remember to save the configuration using the File menu (either using Save, Save As, or Start Test Configuration) before closing the Ion Configuration Manager to ensure the deleted Rule is actually removed from your configuration.

4.3. Working with Custom Files, Registry Settings and ActiveX Controls

One of the most powerful features of Ion is that it lets you use files and settings other than those traditionally installed on the PC. This is accomplished, in part, by creating additional Ion profiles.

To illustrate how this feature works, the example in this section uses the Browsium demo website, "aggrid.com", which worked properly in IE6 and is also incompatible with the latest version of Java. To accommodate this site, create a Profile and include a custom ActiveX control setting to load the older, compatible version of the Java Runtime Environment.

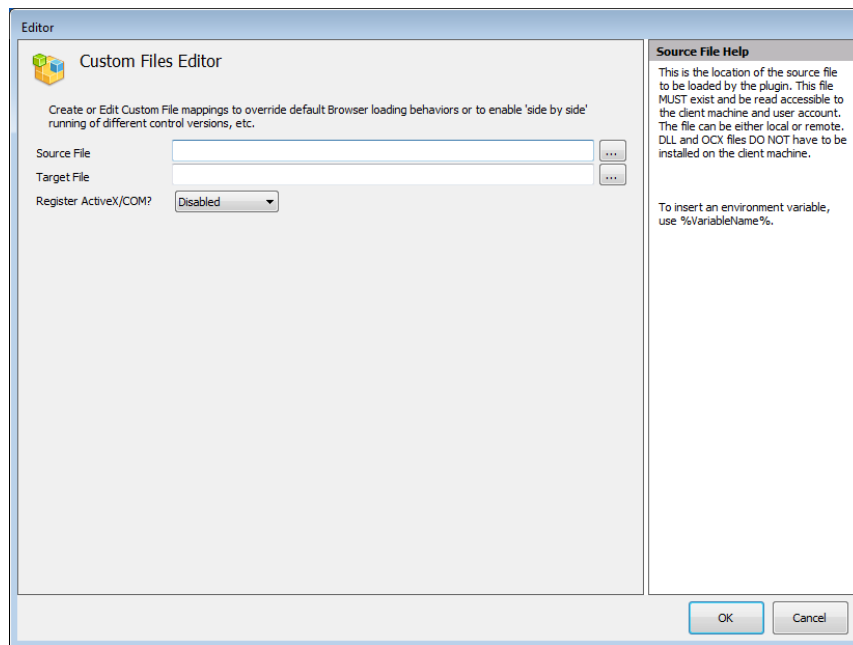
The example used here (requiring Java 1.4.2_19) shows an ActiveX control setting that requires only one file, but it is important to understand Custom ActiveX controls may be comprised of several files (either [.OCXs](#) and/or [.DLLs](#)) as well as a range of registry entries.

Visit the Browsium support site (<http://support.browsium.com>) for more information on settings and guidance for commonly requested controls.

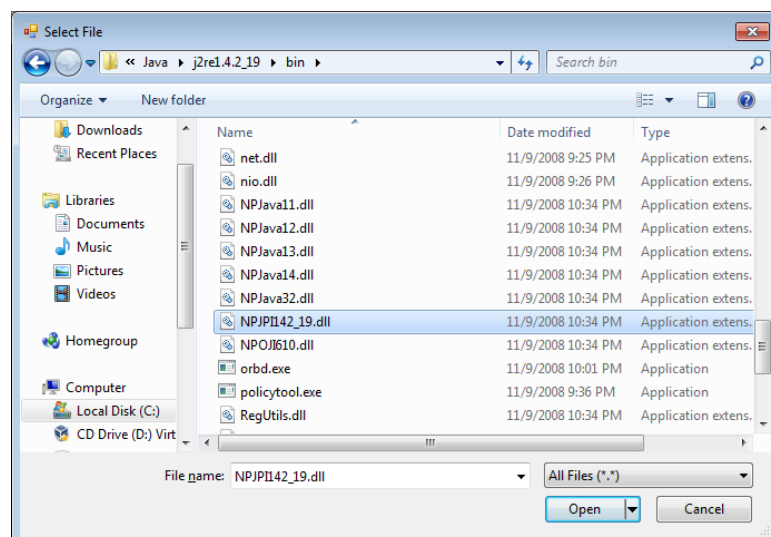
The following steps demonstrate how to create the configuration needed to solve the Java compatibility issues on the "aggrid.com" site. This example requires your test system to have a current version of Java 7 installed, along with Java 1.4.2_19.

This example is being shown using the long-form method of setting up a custom file redirection for Java as a proxy for learning how to do this for any ActiveX Control. However, it is recommended that you use the Java Version Manager when solving Java compatibility issues. See [section 4.4.1](#) for guidance on using the Java Version Manager.

- 1) Create a Profile with Adaptive IE Quirks Mode and name it "Adaptive IE Quirks + Java 1.4" using the process outlined in the [How to Create a Profile](#) section. Then select the Custom Files Manager from under the Profile node. Click the '**Add**' link in the Actions pane to will bring up the "Custom File Editor" window where you need to identify the source and target files for Ion to manage.

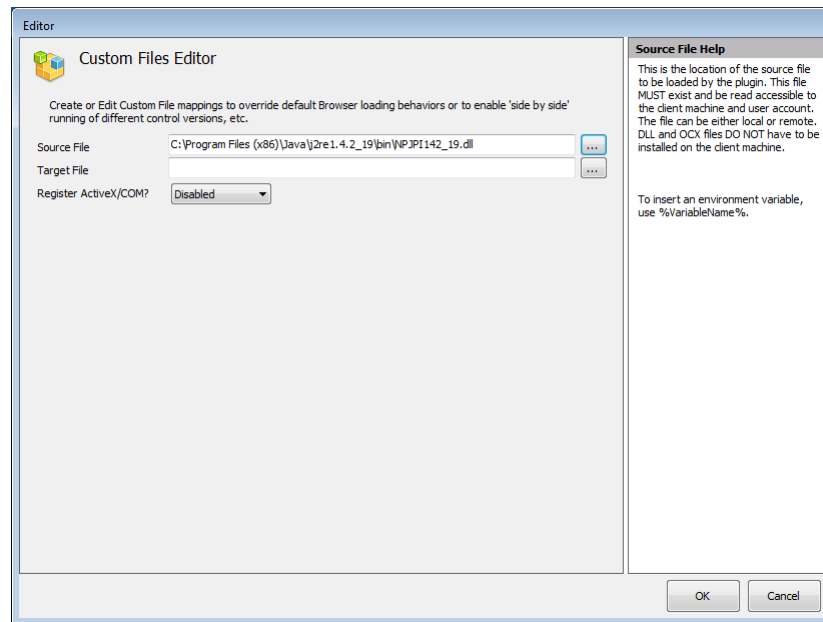


- 2) The Source File is the item needed for your application – in this case the Java 1.4 file (NPJPI142_19.dll). Either manually enter the path of the file or select it by clicking the '...' button to open the file dialog. The required .dll can be found under the "bin" directory where Java was installed.

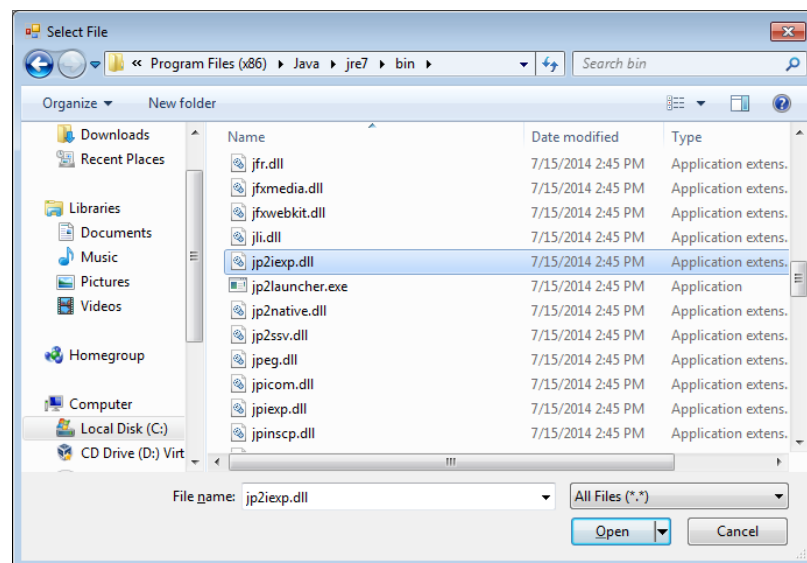


Highlight the desired file and click **Open** to select it.

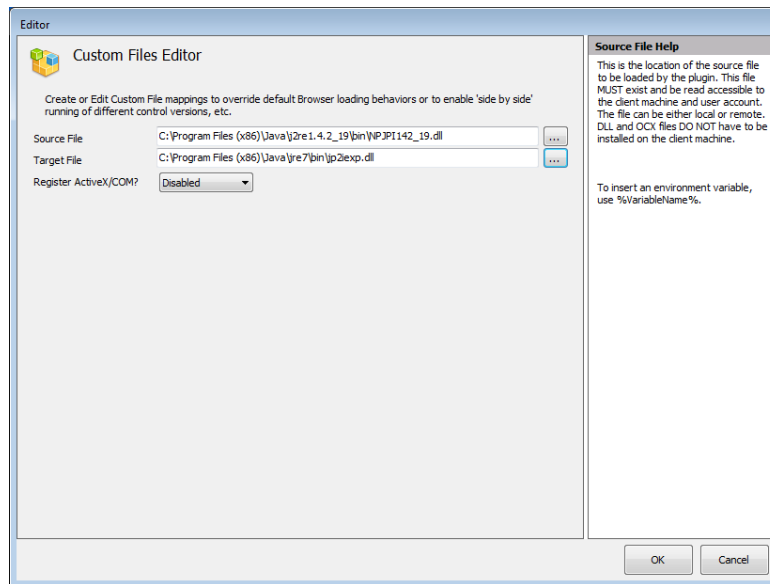
The location and filename will be entered in the Source File text box.



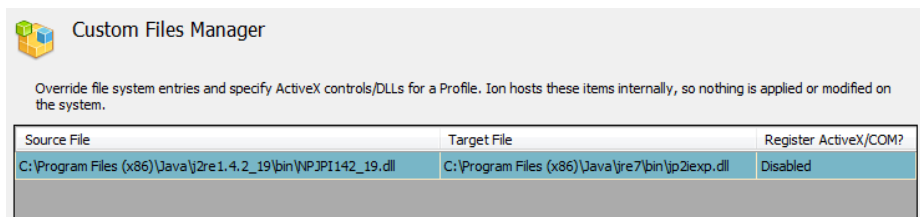
- 3) The Target File is the file to be overridden – in this case Java 7 file (JP2IEXP.DLL). Either manually enter the path and location of the file or select it by clicking the '...' button to open the file dialog. The Target File can be found under the "bin" directory where Java 7 was installed.



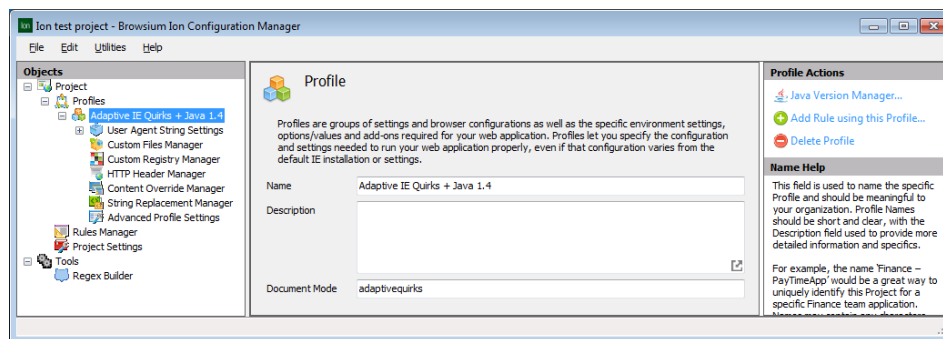
- 4) Leave the '**Register ActiveX Control/COM?**' set to 'Disabled' and select '**OK**'.



- 5) Your Custom Files Manager window should now look like this:



Now you need to create a Rule to instruct Internet Explorer to automatically render the "aggrid.com" site using this Ion Profile. To get started, click the Profile name "Adaptive IE Quirks + Java 1.4" in the Objects pane.



- 1) Click the 'Add Rule using this Profile' link in the Actions pane and you'll see the following:

Editor

Rule Editor

Rules control which Profile is loaded when matched to a set of defined conditions.

Rule Name:

Enable: ☒

Match Method:

Value:

Profile:

Rule Name Help

This field is used to name the specific Rule, and should be meaningful to your organization. Rule Names should be short and clear and may contain any characters. For example, the name 'HR - Benefits System' would be a great way to uniquely identify this Rule for a specific HR team application.

While two Rules could contain the same name, care should be taken to use unique names to avoid confusion.

OK Cancel

- 2) Give the rule a friendly name (e.g. 'Aggrid.com with Java 1.4'), leave the Match Method set to Simple and enter 'aggrid.com' in the '**Value**' field. Finally, leave the Profile set to "Adaptive IE Quirks + Java 1.4".

Editor

Rule Editor

Rules control which Profile is loaded when matched to a set of defined conditions.

Rule Name:

Enable: ☒

Match Method:

Value:

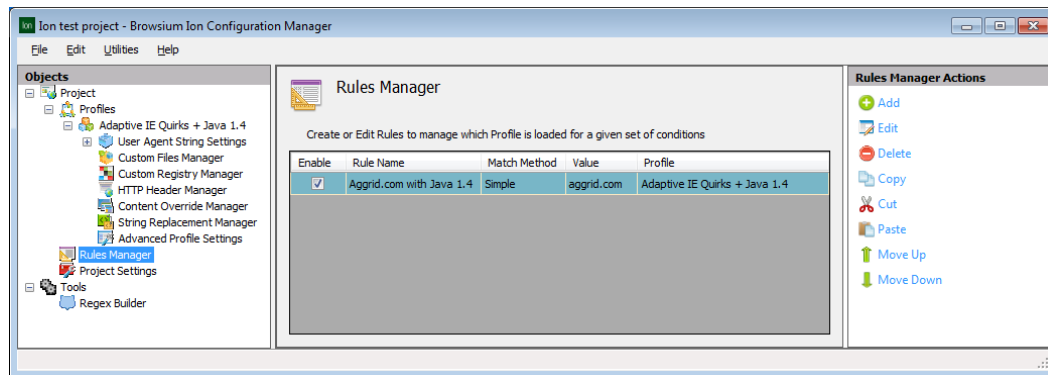
Profile:

Profile Help

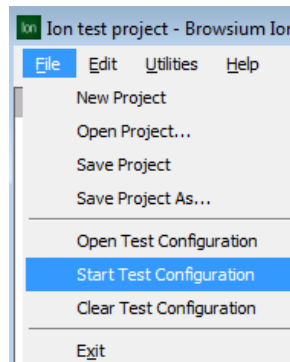
Select the Profile needed to properly run the web application. Any configured Profiles in the currently loaded Project will be listed in this dropdown.

OK Cancel

- 3) Click '**OK**' to add this Rule to the list:



- 4) Your Rule and Profile are now ready to use. Select '**File**' from the menu bar, then '**Start Test Configuration**' and visit www.aggrid.com/timesheets.html to test your configuration.

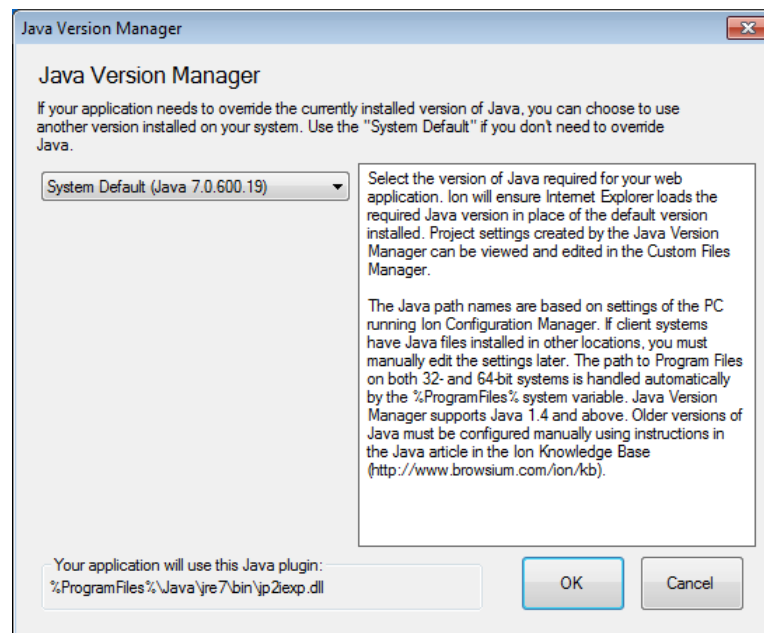


4.3.1. Java Version Manager

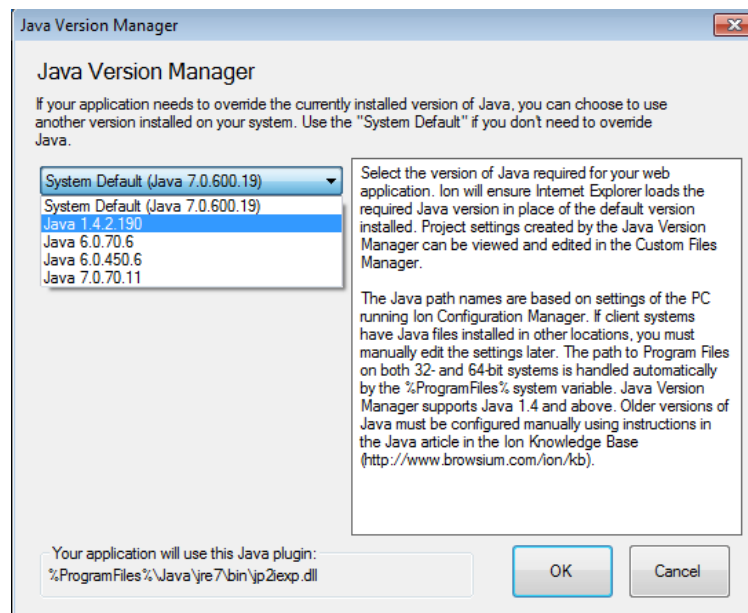
Ion includes the Java Version Manager wizard to make it easier for organizations to manage Java versions with Ion. The Java Version Manager can be used instead of manually creating the required Custom Files entries. This example uses the Java Version Manager to automatically create the same Custom Files entries as were done in the previous example. Again, you must have a current version of Java 7 installed, along with Java 1.4.2_19 for this example.

The Java Version Manager loads options available on the system with the Ion Configuration Manager installed. The file locations used by the Java Version Manager will be based on the locations of Java on that machine. If the location of Java files on the target/client system does match the system running the Ion Configuration Manager, the entries will need to be manually modified to the proper paths and filenames in order to ensure the Java versioning works properly.

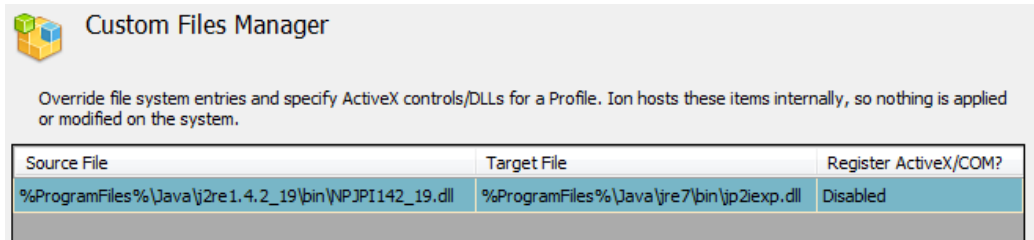
- 1) Launch the Java Version Manager by clicking the 'Java Version Manager' link in the Actions pane.



- 2) Since the aggrid.com web application requires Java 1.4.2 update 19, simply select that from the dropdown list and click OK.



- 3) After completing these steps you should have the following entry in the Custom Files manager screen:



The Java Version Manager will use the %ProgramFiles% environment variable rather than a fully qualified path to help avoid issues where end user systems may have different configurations – particularly 32-bit vs. 64-bit Windows.



Section Five

Rule Writing Basics

In this section you will learn:

- ✓ Recommendations for Rule Writing
- ✓ Using Test Mode for Instant Testing
- ✓ How Rules and Settings are Applied with Ion
- ✓ How Ion Keeps Users Secure Through Admin-only Configurations

5. Rule Writing Basics

You completely control the Ion experience. No rules are included 'out of the box'. There is nothing hidden from you and Ion does not make any decisions for you. We can, however, provide some guidance on writing rules that might work effectively for your organization.

For security and performance purposes, we have designed Ion not to interfere with content that fails to match a declared Rule. This opt-in model prevents unwanted content from being displayed using anything other than the standard Internet Explorer experience. While the Ion design increases security and protects both systems and users from inadvertently loading malicious content, unmatched rule content can still be displayed.

5.1. Recommendations for Writing Rules that Work Well

Ion was designed so users browsing the web are unaware anything 'special' about the content they are viewing and to remove from users the burden of ensuring line-of-business applications 'just work'. This is the key to a successful experience. Users should not be concerned with the browser rendering engine that they are using with line-of-business applications and websites. Anything else can disrupt and interfere with workflow and efficiency. When writing rules with the Ion Configuration Manager, there are some things to take into consideration that can help your users have a better experience.

Any browser upgrade requires an organizations begin the process by assessing which applications and websites are incompatible with the newer (target) version of Internet Explorer. Without a detailed review of which applications need remediation, organizations may look at building broad sets of Rules. Most internal web applications are accessed using only a few defined hostnames, at first, you may want to write a Rule that covers all of these hostnames. This will help ensure that all your internal web applications render as they have in previous versions of Internet Explorer. Over time, you can build a library of rules that cover the specific web applications which either do not work with later versions of the browser or still need to be tested. Then, you can remove the overarching hostname rule so that the specific rules apply.

5.2. Working with Web Application Assessment Tools

While Ion provides a solution for Web Application Continuity issues, you must create a rule set to make your line-of-business applications function properly. For organizations that have been actively assessing application compatibility issues in their environment, making the rules list can be relatively straightforward. For organizations just starting their assessment efforts, making the rules could prove complex – use broad rules sparingly and they should be short-term solutions used while more in depth assessment is performed.

5.3. Ion Rules Behavior

Since Ion uses a Rule-based opt-in model, the ordering of Rules is critical to ensure web applications function properly. The Ion design allows an organization to manage configurations in either centralized or distributed models. If a centralized team manages the configuration, there is less likelihood of overlapping configurations.

Organizations which choose to distribute configuration settings to various business units or web application teams may encounter conflicting configurations. In these instances Ion will always use the 'Last In, First Out' (LIFO) method, meaning that newer configuration values will overwrite previously read values for the same name. These conflicts can be avoided by using your organizations defined change control process or appointing a release manager to verify configurations prior to deploying in production. The Ion Configuration Manager does not offer a mechanism to validate or rationalize multiple configuration files to look for overlapping conditions or errors.

In addition to the process outlined here, specific examples and step by step directions for using Group Policy to control which Ion configuration is loaded by the client system can be found online at support.browsium.com.

5.4. Why Users Can't Create Their Own Rules

Only Administrative installs of Ion are capable of creating rules. The justification for this design boils down to a simple reason: security.

Older web applications and ActiveX controls were not designed with modern exploits in mind and may not be updated as frequently as their newer counterparts (or at all, in the case of end-of-life software). That's why it's important to run modern, up-to-date software for normal, everyday web browsing. Ion helps organizations do just that, allowing them to migrate to the latest platforms while continuing to use the legacy software on an as needed basis.

A user faced with an incompatible web application might be tempted to use Ion excessively or even exclusively. But running Ion to render like IE6 and exposing outdated ActiveX Controls to the Internet just isn't a good idea.

Compatibility problems are frustrating and incur a cost to users' productivity. Since users are not given a way to enable Ion manually to 'fix' something that appears to be broken, we encourage companies to use existing support feedback mechanisms, such as helpdesk or support escalation systems, to give users ways to provide feedback and get new sites added to the Ion rules lists.

Ion does not install any legacy Internet Explorer components that may be exploited by a malicious site or application. Installing Ion will not add any potentially exploitable legacy Internet Explorer software. The Ion design simply manages how Internet Explorer renders content and the environment in which it runs. If a web application requires removing or lowering security settings present in the newer version of Internet Explorer you are running, those changes will be restricted to the pages loaded by sites that are defined by Rule.



Section Six

Managing Ion in the Enterprise

In this section you will learn:

- ✓ Enabling the Ion Client in the enterprise
- ✓ Options for deploying Ion Project files
- ✓ Deploying registry pointers to Ion Project files
- ✓ Using Ion with Group Policy

6. Ion Deployment in the Enterprise

To provide maximum flexibility for managing enterprise deployments, Ion supports a variety of methods for hosting configuration (Ion Profiles, Rules, and Settings) locations. In this section, we'll examine the available options, provide recommendations, explain the loading/caching behaviors and offer specific deployment guidance for typical enterprise scenarios.

The Ion Client configuration is read from an Ion Project file – a single, XML-formatted data file containing the entire Ion configuration. The file can be pushed out to a known location on end user PCs, stored at a shared network location, or hosted at an easily accessible URL. The Project file can be updated easily and separately from the Ion executable code, making ongoing support and maintenance easy to integrate into existing organizational processes.

Each client system must be made aware of the location of the Ion Project file containing the configuration that is relevant to that user's web applications. Your organization may use one configuration for all end users, or have separate configurations for various departments or geographic locations.

A client system loads its Ion configuration by reading a file path from the LoadFromFileName registry value. As an Ion administrator, you must determine the best way to deploy this registry value and data to your organization. You can do this using a variety of methods, ranging from scripted registry editors to Group Policy.

Many organizations prefer to use Group Policy to deploy this registry value, particularly if different organization units require different configurations. The [registry.preference.extension for Group Policy](#) is often the most efficient way to streamline the deployment of the LoadFromFileName registry value and data. Another option is to use Browsium's ADM and ADMX templates, which can be easily customized for the location of your Project file. These templates, with usage instructions, can be found in the [Browsium Ion Knowledge Base](#). See [section 6.4](#) for details on deploying Ion Project files and LoadFromFileName registry pointers.

6.1. Understanding Ion Registry Setting Locations

Before we look at deployment specifics for enterprise client systems, it is important to understand methodologies for deploying configurations on test systems during configuration development. Ion makes it easy to test a configuration without requiring a centralized deployment methodology. This is done via the Start Test Configuration option from the File menu of Ion Configuration Manager.

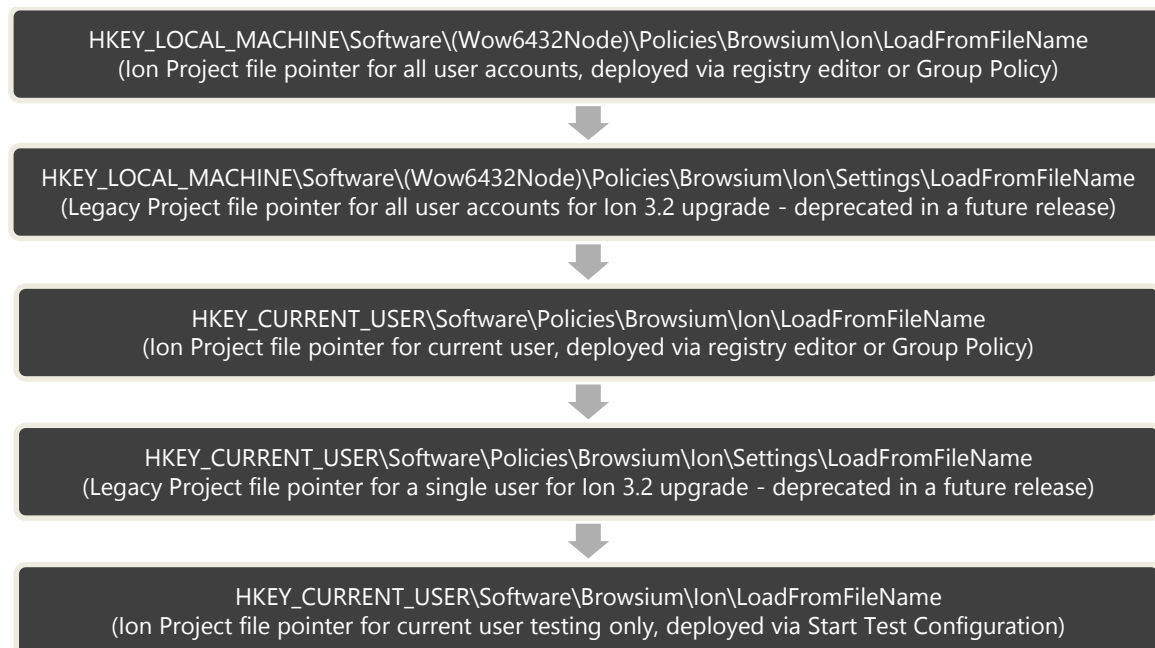
This option will save the Project file on disk, place a pointer to that file directly in the system registry, and start the Ion Controller to process the new configuration. The pointer is written to the LoadFromFileName registry value in HKEY_CURRENT_USER\SOFTWARE\Browsium\Ion.

Start Test Configuration should only be used for project development and testing as it requires manual operation of Ion Configuration Manager which should never be made available to end users. In addition, this registry location should only be used for testing as it is easily overwritten by the Ion Configuration Manager and may not persist during an Ion Client upgrade.

The last concept that must be understood before embarking on an Ion deployment is the precedence hierarchy for the evaluation of configurations when multiple LoadFromFileName values are found on a system. Ion follows this hierarchy to load the configuration that will be used on a given end user system (and on test systems). Once a valid configuration is found, Ion will stop searching and that configuration will be used.

Deploying different Ion configurations using multiple methodologies on a single PC may cause unpredictable results as only the configuration highest in the hierarchy will be used.

The following table provides the hierarchy of Ion configuration precedence. The string "(Wow6432Node)" in the registry path denotes the Wow6432Node registry key that will be included in the path on 64-bit Windows systems. 32-bit Windows systems do not contain this key, hence the use of parentheses in the example.



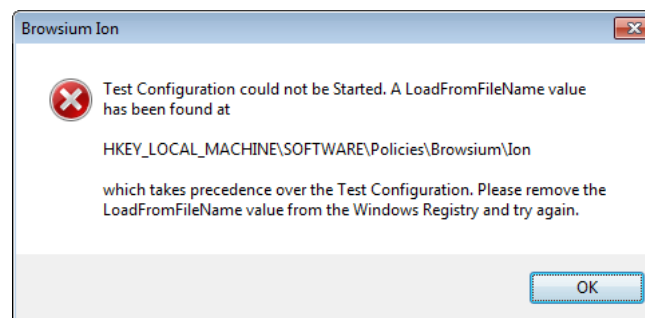
Ion 3.2 and earlier looked in \Browsium\Ion\Settings for the LoadFromFileName registry value. Ion 3.3 supports this location for backward compatibility to ease enterprise upgrades. This location will be deprecated in a future release.

6.1.1. Working with Test Configurations in Group Policy Environments

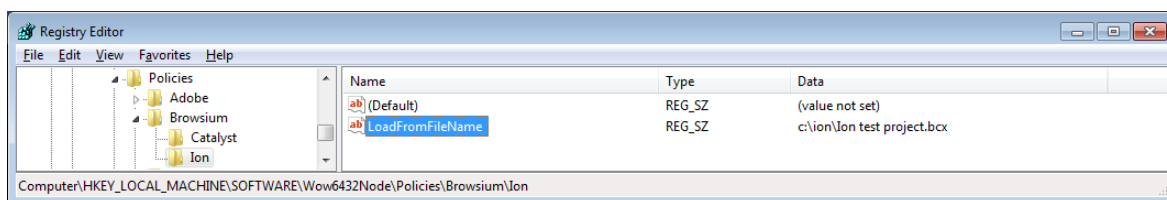
Understanding the configuration loading hierarchy is an important part of ensuring test configurations will load as expected during project development. Testing on systems that are also configured with a LoadFromFileName value via Group Policy may cause issues when Start Test Configuration is applied. As identified in the section above, Ion reads LoadFromFileName from all other \Policies\Browsium\Ion locations prior to the Test Configuration location.

Browsium recommends testing Ion on systems that are not controlled by Ion Group Policy settings to avoid configuration hierarchy conflicts.

For example, if Group Policy is deploying a LoadFromFileName value in the location at the top of the configuration hierarchy, Start Test Configuration will generate an error as follows:



In order for Ion to read the LoadFromFileName value created by Start Test Configuration, an additional step is required to disable the value created by Group Policy. To disable the Group Policy loading behavior, open the Registry Editor and navigate to the key listed in the error message above, and remove or rename the LoadFromFileName value.



Customers may find that changing the value of the registry key to LoadFromFileName-OFF is a convenient way to disable the registry value, rather than deleting it, while making that registry value's original intent clear. Making any small change in the key name will prevent Ion from reading the key value and therefore Ion will load only the Test Configuration as desired. When

testing is complete, remember to revert the changes made using the Registry Editor or manually force a Group Policy update.

Group Policy refresh rates may nullify the Registry changes when policies are refreshed and the managed keys are recreated. If this occurs when the Ion Controller is running, no changes will happen. If the Controller is restarted after the policies are refreshed, the Controller will read the higher level policy values which will result in the Test Configuration values failing to be read.

6.1.2. Multi-user Testing Configurations

As mentioned previously, the Ion Configuration Manager should be installed only for Administrators and Project Developers – not for end users. The Start Test Configuration option can only be used for single user testing. The Start Test Configuration option only saves and loads the configuration for the current user.

The Start Test Configuration does not support multi-user testing.

This design cannot meet the requirement to perform extended testing with multiple users or business groups before enterprise wide deployment. Administrators looking to test a configuration with a group of users should follow the standard configuration deployment approach for their organization. Once the Ion Project file is deployed, the Controller process must be restarted on each client system to load the new configuration.

More information on deployment options, settings and configuration options can be found in the [Deploying Ion Configurations](#) section.

6.2. Enabling the Ion Client for Enterprise Deployment

A critical element of any Browsium Ion deployment is ensuring the Ion Client Add-on is set to 'enabled' in Internet Explorer on all client PCs. To prevent malicious or unwanted add-ons from impacting the user experience, recent versions of Internet Explorer require user confirmation before any new add-on is enabled, unless that add-on is set to 'enabled' during the deployment process. In this section, you will learn the procedures for centrally enabling the Ion Client Add-on during deployment.

The most common way to enable the Browsium Ion Client Add-on during deployment is by utilizing Group Policy to make the necessary registry changes on client PCs. Alternative methods to modify the registry on client PCs, such as a Visual Basic Script, can also be employed. The following guidance is adapted from articles on Microsoft's TechNet website, and includes the

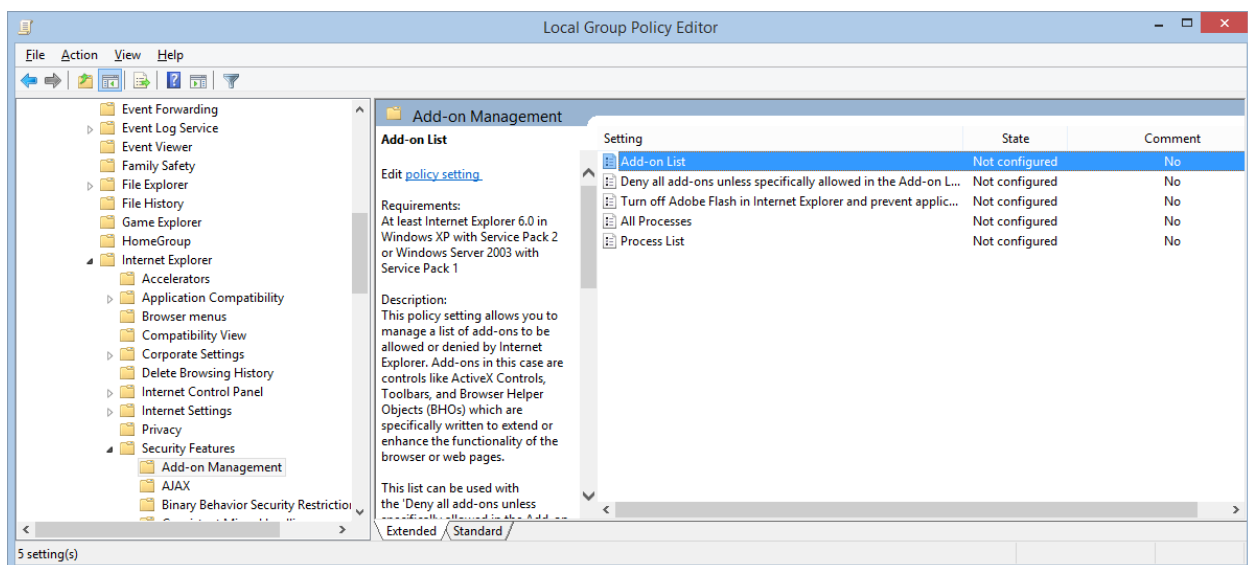
process to identify the GUID/CLSID of the Browsium Ion Client Add-on, which must be located in the Registry once it is installed in your environment.

6.2.3. Understanding the Add-on List Policy

Administrators can control the use of specific add-ons in Internet Explorer through the **Add-on List Policy**. Administrators can choose to enable or disable an add-on as well as allow a specific add-on to be managed by the user.

Policy Name: add-on list

Path: User Configuration or Computer Configuration node; Administrative Templates\Windows Components\Internet Explorer\Security Features\Add-on Management. To set this policy, an administrator can enable the policy and enter the GUID/CLSID of the Ion add-on to the Add-on List and set the value to 1.



To manually set this policy, an administrator can create a registry value based on the GUID/CLSID of the add-on in either of the following keys and then set the desired value:

HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Ext\CLSID

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\Ext\CLSID

Each add-on is a value in this Registry key with the following properties.

Name: *GUID* of add on

Type: **REG_SZ**

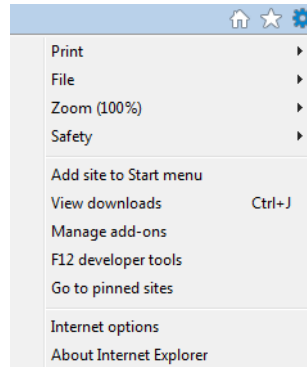
Value:

- 0 - Add-on is disabled and cannot be managed by the end user.
- 1 - Add-on is allowed and cannot be managed by the end user.
- 2 - Add-on is allowed and can be managed by the end user.

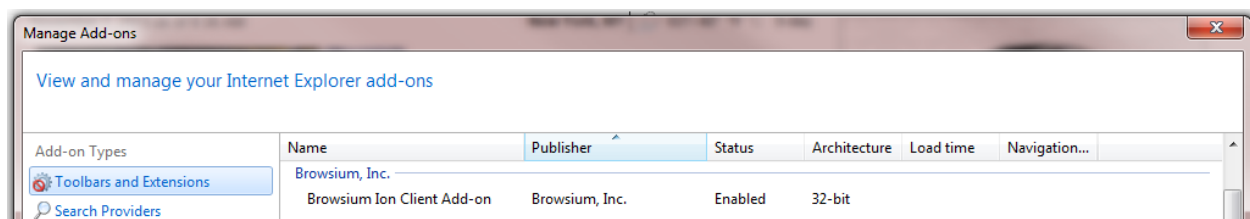
The Add-on (CLSID) lists are empty by default.

6.2.4. Determining the GUID/CLSID of the Ion Client Add-On

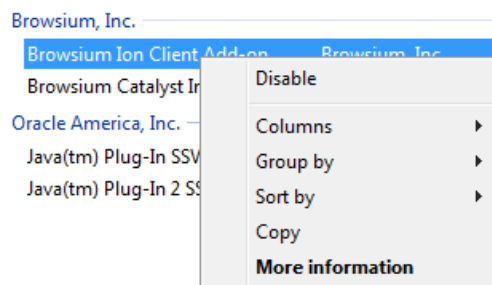
After installing the Browsium Ion Client, go to the Tools menu in Internet Explorer and choose Manage add-ons.



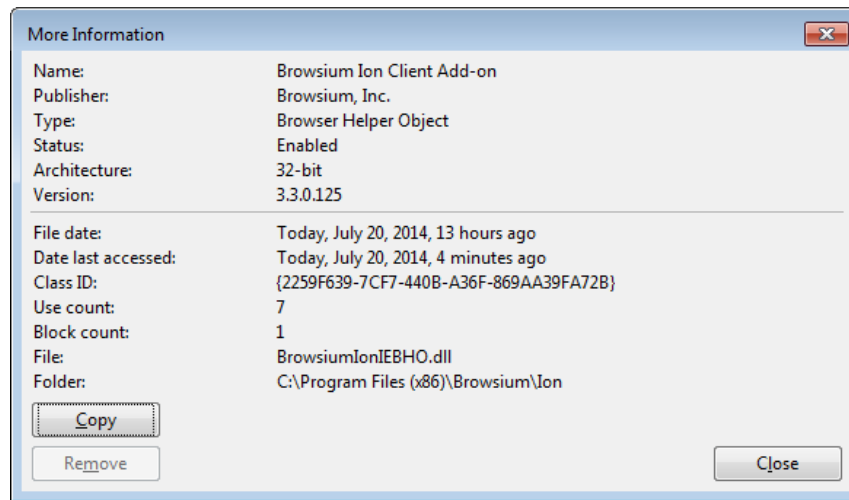
You'll then be presented with the Manage Add-ons interface where you should see Browsium Ion Client Add-on in the list among the Toolbars and Extensions that are currently loaded in Internet Explorer.



Right Click on the Browsium Ion Client Add-on and choose 'More information' from the dropdown menu.



The Class ID (CLSID) will appear in the dialog box.



Click the "Copy" button and then paste the contents of this dialog box (including the Class ID) to Notepad for later reference and save the text file. When you make the registry changes documented above, you will need to use the Class ID to identify the add-on in the Registry.

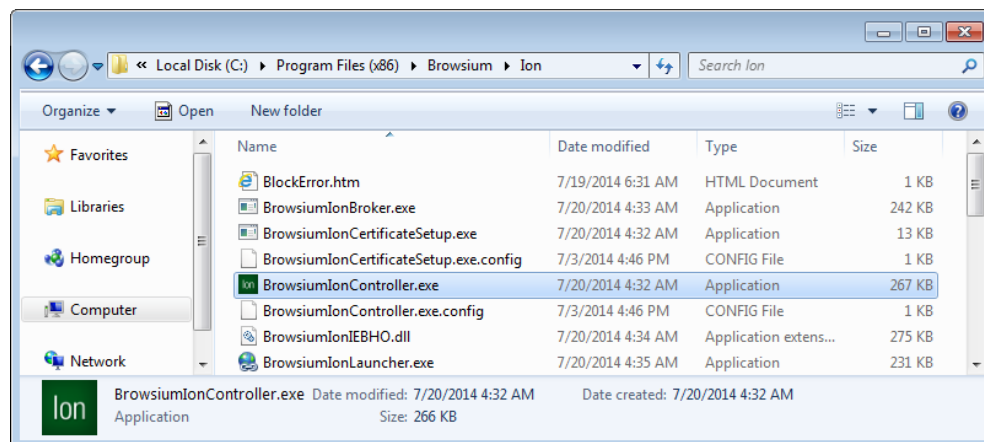
More information, including troubleshooting techniques, can be found on the Microsoft TechNet website in the article [Internet Explorer Add-on Management and Crash Detection](#).

6.3. Readyng End User PCs to Use the Ion Client

After you have deployed the Ion Client (**Ion-ClientSetup.msi**) to your end user PCs, it is very important to follow a few steps to ensure the Ion software is running and ready to process the Profiles and Rules contained in your Ion configuration.

In addition to ensuring that the Ion Client Add-on is enabled, per the guidance in [section 6.2](#), the Ion Controller process must also be running. The Ion Controller does not start automatically after installation of the Ion Client, even if an Ion configuration is present. The Ion Controller can be started by restarting the client computer (and will start automatically upon every logon) or by manually starting **BrowsiumIonController.exe** found in the Browsium\Ion directory in Program Files (Program Files (x86) on 64-bit Windows systems). This operation can be scripted by executing "BrowsiumIonController.exe /Start".

If this latter method is used, all Internet Explorer windows must be closed before the Ion configuration will take effect.



If the Ion Configuration Manager has been installed, it can be used to start, stop, and restart the Controller as required – also requiring Internet Explorer windows to be closed. This method should only be used on systems run by administrators trained in creating and modifying Ion configurations.

6.4. Deploying Ion Configurations

Since Ion is a client solution with no server component, getting the configuration settings to the client is critical for proper operation. Configuration management is extremely flexible and can be tailored to meet the design and requirements of your environment.

Ion's Project files are standard XML documents, allowing you to take full advantage of this versatile and very compact format. Configurations are easy to update by simply replacing the Project file on end user systems, network share or web server.

By design, Ion will not look in any specific location for a Project file – you must configure where each client system will look for the configuration. This only needs to be done once, no matter how often the Project file is updated – provided the file name and file location do not change. The following steps provide guidance to specify where Ion should look to find the Project file.

There are two options for deploying the Ion Project file – on each client system or hosted on a file share or web server. Browsium recommends using the hosted method for deploying configurations to provide the most flexibility. The Ion Client makes the network calls needed to 'pull down' the configuration from the hosted Project file, so no additional packaging or user login configuration steps are needed when using the hosted option.

When using a Project file loaded from a client system, administrators must use some other software distribution solution (SMS, SCCM, login script, file copy, etc.) to deploy the Project file to the client file system.

6.4.1. Project File Deployment Option A: Network Share or Web Server Location

In this section you will learn how to instruct Ion to load the configuration from a network share or web server location. To do this, you must edit the system registry manually (for local testing) or via a script or Group Policy (for remote deployment) to create the LoadFromFileName registry value and data at the appropriate location. Browsium recommends using the registry preference extension for Group Policy or the Browsium ADM and ADMX templates as the most efficient way to streamline deployment of this registry value.

The Ion project file (.bcx) must be stored in a user-readable location on the network share or web server. For network share locations, Browsium recommends users have only read access permissions to ensure the file is not accidentally removed or modified. For web server locations, organizations must ensure users have both security credential and network access to the

resource. For example, remote/mobile users will need to have VPN access if the web server hosting the Ion configuration file is not publically accessible and the user's system has not cached the configuration from a prior connection.

If the network share or web server is unavailable when the Ion Controller starts, and no configuration is found in Ion's cache from a previous connection, the Ion Controller will not start. As long as the user successfully connects to the server once during Ion Controller startup, the Project file (configuration) will be cached indefinitely.

The following registry keys and associated values must be created, depending on the system and user accounts being targeted:

For **per-machine** settings on **32-bit** Windows systems, find or create:

HKEY_LOCAL_MACHINE\Software\Policies\Browsium\Ion

For **per-machine** settings on **64-bit** Windows systems, find or create:

HKEY_LOCAL_MACHINE\Software\Wow6432Node\Policies\Browsium\Ion

Then create or populate the following String Value in the Ion key:

LoadFromFileName (REG_SZ) = \\Server\Share\Ion\... [the path to your Ion Project file]

This setting will direct the Ion software to the Project file the next time the Ion Controller is restarted.

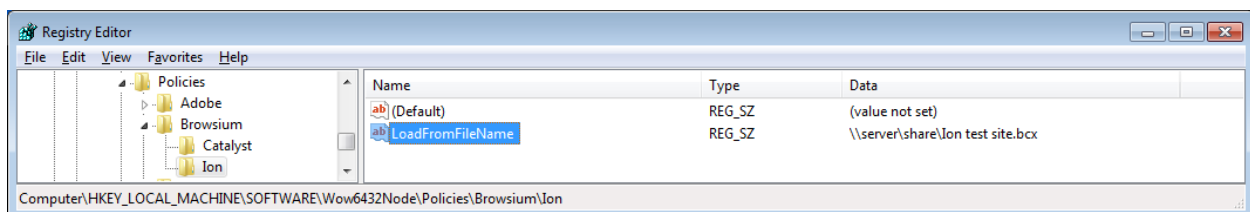
Slashes in the file path must be escaped with a slash when invoking Regedit.exe via a .reg file. So \\Server\Share becomes \\\Server\Share. http://server/file remains http://server/file.

In this example, LoadFromFileName has been configured to use the Ion Project file "Ion test site.bcx" in the \\server\share directory on a 64-bit Windows system. These entries can be scripted and delivered to the Registry on remote clients via the following text in a .reg file.

Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Policies\Browsium\Ion]

"LoadFromFileName"="\\server\share\Ion test site.bcx"



6.4.2. Project File Deployment Option B: Client File System Location

In this section you will learn how to instruct Ion to load the configuration file from a local file system location. To do this, you must edit the system registry manually (for local testing) or via a script or Group Policy (for remote deployment) to create the LoadFromFileName registry value and data at the appropriate location. Browsium recommends using the [registry.preference extension for Group Policy](#) or the [Browsium ADM and ADMX templates](#) as the most efficient way to streamline deployment of this registry value.

The Ion Project file (.bcx) must be stored in a user-readable location on the client PC. If not already on the client PC, administrators must have a distribution plan and process for ensuring the Project file is copied to the client PC prior to starting the Ion Controller. If Ion is unable to find the Project file in the defined location, and none exists in the cache from a previous configuration, the Ion Controller will not start. For more information on Ion file caching behaviors see the [Ion 3.3 Project File Caching Behavior](#) section.

The following registry keys and associated values must be created, depending on the system and user accounts being targeted:

For **per-machine** settings on **32-bit** Windows systems, find or create:

HKEY_LOCAL_MACHINE\Software\Policies\Browsium\Ion\

For **per-machine** settings on **64-bit** Windows systems, find or create:

HKEY_LOCAL_MACHINE\Software\Wow6432Node\Policies\Browsium\Ion\

Then create or populate the following String Value in the key:

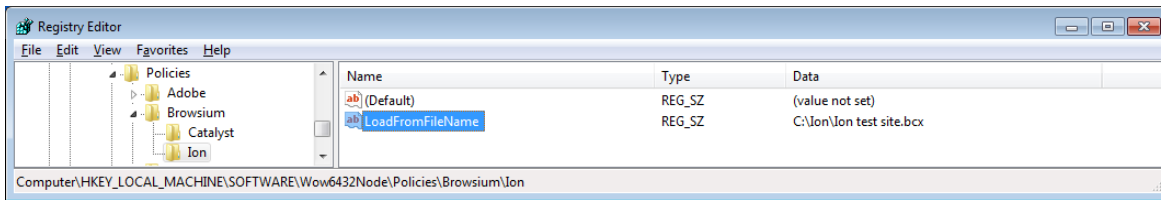
LoadFromFileName (REG_SZ) = C:\directory\... [the path to your Ion Project file]

This setting will direct the Ion software to the configuration file the next time the Ion Controller is restarted.

Slashes in the file path must be escaped with a slash when invoking Regedit.exe via a .reg file. So C:\directory becomes C:\\directory in the registry value.

In this example, LoadFromFileName has been configured to use the file "Ion demo site.bcx" in the C:\Ion directory on a 64-bit Windows system. These entries can be scripted and delivered to the Registry on remote clients via the following text in a .reg file.

```
Windows Registry Editor Version 5.00
[HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Policies\Browsium\Ion]
"LoadFromFileName"="C:\\Ion\\Ion demo site.bcx"
```



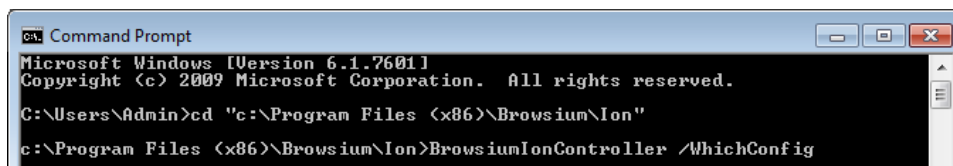
6.5. Verifying Deployment Settings

Browsium Ion provides a command-line function to query the local system and identify the location of the configuration defined on the local system. The command is **"BrowsiumIonController /WhichConfig"** and can be run from a standard Command Prompt.

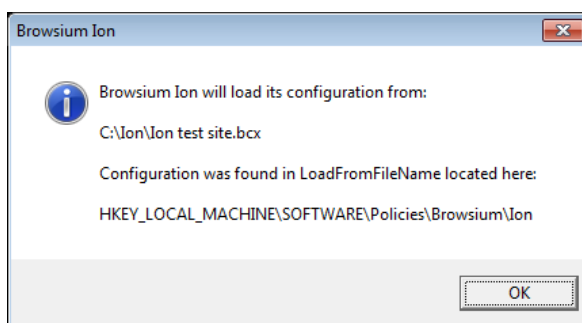
In the following example, a system is setup with a configuration deployed to HKEY_LOCAL_MACHINE\Policies\Wow6432Node\Browsium\Ion and a value "C:\Ion\Ion demo site.bcx" stored in LocalFromFileName.

The query uses registry reflection and therefore the result ignores the Wow6432Node key on 64-bit Windows systems although it is in the path of the registry key value.

Executing this command ...

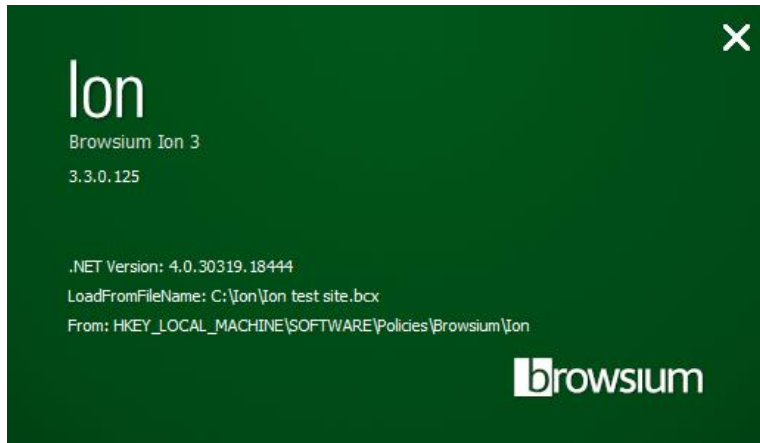


... results in the configuration acknowledged as:



The design of the 'WhichConfig' tool is to show the configuration TO BE LOADED at the next Controller restart. The value shown may be different from the currently loaded configuration, if the value in the registry key has been changed since the Controller was last started.

In addition to the command line option, administrators running the Ion Configuration Manager can quickly check which configuration will load when the Controller is restarted by viewing About Box. Select 'About Browsium Ion Configuration Manager' from the Help menu. When the Ion splash screen loads, hit the Ctrl button to reveal the configuration setting information. In addition to the configuration loading data you will see the installed .NET version – another way to ensure the system is ready and working properly for testing.



6.6. Understanding the Ion 3.3 Project File Caching Process

Browsium Ion reads the Project file (configuration) when the Controller is started, then continues operating using that configuration until the Controller restarts and looks for a new Project file. Ion intentionally does not support 'dynamic' updates to the working configuration. If a new configuration were to be loaded dynamically, end users could be negatively impacted. For example, users entering data to a reporting tool would see their sessions closed before they may have completed their entries.

To avoid end user complaints and helpdesk calls, Ion will not force close the browser for any reason. Organizations wanting to load a new configuration must instruct users to close Internet Explorer and restart the Browsium Ion Controller. Alternatively, organizations could use existing systems management tools to remotely perform the tasks, or wait for the users to logoff and logon again which will trigger the Ion Controller process to load the new Project file.

6.6.1. Ion Project File Caching Behavior

Ion attempts to load the configuration from the Project file defined in the LoadFromFileName registry value whenever the Controller is started. The location of the file can be local on the client system, a network share, or HTTP location. The Controller must be able to reach the location where the file is stored, and the user must have at least read permissions on the Project file. When the configuration is loaded, a copy of the Project file is cached locally in the user's AppData directory (or equivalent based on Windows version and user directory structure).

If the original Project file is not available (network location is temporarily unreachable, the file has been moved, renamed, or deleted, etc.), Ion will load the previously cached copy of the Project file and operate using that configuration until the Controller is restarted and a new Project file is available.

This capability makes Ion extremely robust and able to remain fully functional regardless of what happens to the Project file after it has been loaded and the Ion processes are running.

6.6.2. How Ion Handles Project File Polling

Some customers deploy Browsium Ion in a staged rollout to their enterprise user base, which requires updating Project files as new web application or Java version management settings are required. In order to ensure client systems obtain the updated Project file, Ion includes a configuration polling mechanism to check for modified versions of the Project file. When Ion identifies a different file in the target location defined in the LoadFromFileName registry value,

by comparing the MD5 hash of the cached file against the target file, the target file is retrieved and cached locally. If the hash matches the current cached file, the target Project file is not retrieved. To ensure a robust user experience, and avoid corrupting data entry or other business processes, newly cached Project files will not be loaded until the Controller is restarted, most commonly at the next user logoff/logon.

The client polling mechanism checks the target Project file location at a regular time sequence interval, using a randomly generated seed time point. The randomly generated seed value helps ensure large scale deployments do not simultaneously try to retrieve the Project file, avoiding a denial of service or network traffic storm issue.

Ion simply compares the computed MD5 hash of two files and uses the target file if the two signatures are different. No effort is made to determine which file is newer (either by time/date stamp or content analysis) so administrators must ensure updates to the target file location are done carefully to avoid clients regressing to an older Project file configuration by mistake.

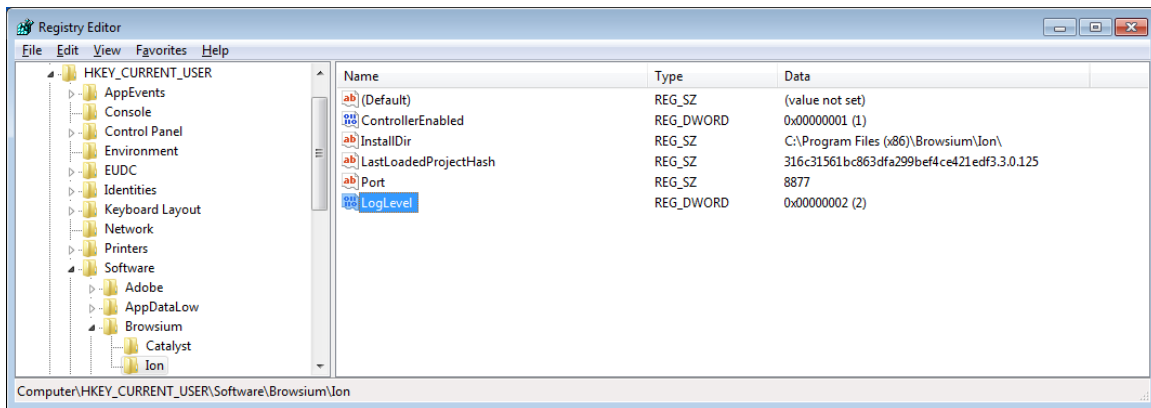
6.7. Managing Ion Logging Settings

In a default installation, Ion logs Warning level information which will provide basic information about the configuration and any important errors that may occur on client machines. Ion can be configured to record logging information in order to troubleshoot problems or validate configuration settings on the local machine. Ion logging is written to the standard Windows event log under an application-level source named 'Ion'. The Logging Level setting determines the amount and type of data collected in the Windows event log. This table summarizes the various levels and data collected:

Value	Level	Description
1	Error	Writes Error entries
2	Warning	Writes Warn and Error entries (Default)
4	Info	Writes Info, Warn and Error entries

To adjust the level of logging a registry key and value must be configured on the target machine. Once the logging value key is created, the controller must be restarted to recognize the specified logging value.

Ion looks for the presence of the logging key at HKEY_CURRENT_USER\Software\Browsium\Ion. The key is a REG_DWORD type, with value name "LogLevel". The value of the key can be 1, 2 or 4 (see chart above).



If no LogLevel registry value is present in the defined registry location (the default Ion Client state), or any invalid values are found at that location, Ion will revert to using the default level 2 – Warning.



Appendix A

Appendix A: Troubleshooting

In this section you will learn:

- ✓ How to Recognize Issues with an Ion Configuration
- ✓ What to do When Ion is Not Working as Expected

A. Troubleshooting

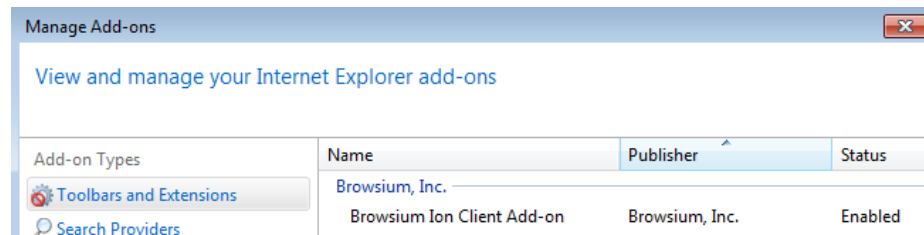
A.1. Ion Rule Fails To Engage

You may encounter a scenario in which Ion fails to engage on one or websites based on rules created.

The following points may guide you to a resolution:

- **Review System Prerequisites**
 - Check to see that the target computer meets the performance and storage requirements to run Ion.
 - Confirm that .NET 3.5 SP1 or later is installed on the computer running the Ion Client.
- **Verify the Ion Installation Files**
 - If you are using a computer where both the Ion Configuration Manager and Ion Client are present, ensure that they are installed and running.
 - If you are using a computer where only the Ion Client is installed, verify that it is correctly installed and running.
- **Ensure Matching Versions of the Ion Configuration Manager and Ion Client are Installed**
 - If you installed previous versions of Ion, make sure that you are using the Ion Configuration Manager for which the Major, Minor, and Update numbers match that of the Ion Client currently installed on a target machine.
- **Confirm the Ion Executable Files are Running**
 - Check to see that the Ion Controller (**BrowsiumIonController.exe**) and Ion Monitor (**BrowsiumIonMonitor.exe**) are running on the client PC.
 - Verify the Ion Broker process (**BrowsiumIonBroker.exe**) is running when a Rule has been invoked.
- **Ensure the Ion Plugin is Enabled and Running**
 - Confirm the Browsium Ion Client Add-on is seen and loaded by Internet Explorer
 - Open Internet Explorer and open the Manage Add-Ons dialog. Do you see the Browsium Ion Client Add-on installed? Is it enabled?

- An Internet Explorer instance that correctly loads the Ion Client will display the following information in the Manage Add-Ons dialog:



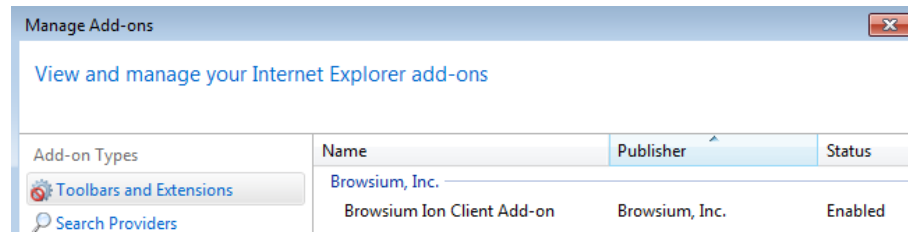
- **Visit the Forums or Contact Support**

- If all of these steps fail, consider posting a question on the Browsium Support Forums.
- If you have a support evaluation or contract, contact your service integrator, or the Browsium Support team for one-on-one guidance

A.2. Correct Profile Does Not Load for a Web Site

If you are experiencing an issue where a specific Ion Profile does not load for one or more web sites, the following steps may help resolve your issue.

1. Check that you have Rule covering the website or web application in question.
 - a. If testing locally, try removing any older settings by using the Clear Test Configuration file menu option, then Start Test Configuration. Also ensure the system is not being prevented from loading the Test Configuration as a result of Group Policy settings being applied. See the [Understanding Ion Registry Setting Locations](#) section for more information.
2. Open the Task Manager and check that the **BrowsiumIonController.exe** (Controller), the **BrowsiumIonBroker.exe** (Ion Broker) processes are running.
3. In Internet Explorer, open the Manage Add-Ons window, and check that the Browsium Ion Client Add-on is enabled. Your configuration should match the configuration in the screenshot below:



A.3. Ion Not Working Properly in IE6 or IE7

Browsium Ion is only supported on Internet Explorer 8, 9, 10, and 11, so you may see unexpected behavior when trying to install and run Ion with other versions of Internet Explorer.

A.4. Ion Not Working Properly on Windows 8/8.1

Browsium Ion is only supported for the desktop version of Internet Explorer included with Windows 8/8.1. By design, Ion is unable to integrate with the Modern (aka 'Metro') version of Internet Explorer.