Browsium Catalyst
Administration Guide
Version 4.6

browsium

Browser Management for Enterprise

www.browsium.com

Administration Guide

This guide has been created for IT administrators to assist in installing, configuring, and deploying Browsium Catalyst. This version of the guide is designed for use with Browsium Catalyst 4.6.

For more information about Browsium Catalyst, other Browsium products, or to contact Browsium Support, please visit www.browsium.com.

Table of Contents

Section One

Introduction

In this section, you will learn:

- ✓ What is Browsium Catalyst
- ✓ The components which make up Browsium Catalyst
- ✓ What to expect from Browsium Catalyst

1. Introduction

As a rule, the IT department is responsible for determining the standard desktop configuration and web browser for the organization. In the past, most IT groups opted to use Internet Explorer for its flexibility in management as well as IE having been included with Windows. As the web has evolved, pressures have been put on the IT group to offer more browser choice to their users. The challenge of offering choice was complicated by the need for IT to properly manage the alternate browser offerings.

Browsium Catalyst is designed to help IT organizations to address these challenges and offer a selection of browser choice to end users – all without losing management functionality. Catalyst provides the ability to deliver browser change and still ensure business process remains uninterrupted.

Catalyst is more than just about controlling browser behavior. Catalyst is designed to solve IT challenges around legacy web application compatibility issues, challenges around embracing emerging technologies and reducing support costs. Lastly, Catalyst provides the bridge needed for IT to address consumerization in the workplace.

1.1. Browsium Catalyst Explained

Browsium Catalyst is the first multi-browser redirection and management tool of its kind. Unlike other browser redirection engine solutions which have been designed and implemented as part of a vendor specific solution, Catalyst is platform and browser agnostic. By removing any reliance on a specific vendor technology, Catalyst enables an IT organization to be in complete control regardless of how they want to implement the solution.

Catalyst provides the ability to safely deploy multiple browsers to end users and still control which browser can be used for a given website or web application. In delivering this level of control, IT organizations finally have a toolset to enable browser choice. Now IT can deliver a multi-browser solution in a manner that makes sense for the organization. Business needs and end user choice are decoupled from web application contingencies, giving IT the flexibility to meet multi-browser requests with confidence.

Catalyst makes sense to the user since they have to do nothing special or different; the add-ons do all the switching automatically. The solution is seamless. The end user can focus on doing their work and avoiding problems or downtime.

Catalyst is controlled by a hierarchical system of rules, defined using the Catalyst Configuration Manager. Understanding this system is the key to understanding Catalyst. The Configuration Manager provides tools to define criteria by which web applications are loaded in a desired browser. In addition to simply specifying a website to open in a given browser, Catalyst offers the ability to control the user experience when being redirected.

For example, some web applications not only need to be opened in a specific browser, but the application requirements are to open each link in a new session. Catalyst can do that with ease. If the requirement is to open content in a new tab, no problem. Catalyst can even block requests entirely, helping provide an extra layer of immediate protection when a security advisory is issued for an exploit on a given browser.

Catalyst is designed to keep users where IT wants them to go – while not getting in the way when IT hasn't set a policy for a given location. Catalyst enables IT administrators to ensure the right browser is used for the right application, but undefined applications can be accessed with the browser of choice for the user.

Catalyst is built on an opt-in basis. In other words, Catalyst intervenes when – and only when – it is instructed to act.

1.2. Browsium Catalyst Configuration Manager and Client

The Catalyst Configuration Manager is the main interaction point for IT administrators using Catalyst. Catalyst has been designed to work in a traditional IT setting and deploy using existing technology systems in use at your organization.

The basic design of the Catalyst Configuration Manager is such that it easily matches the architecture and needs of your organization. Using a distributed solutions approach, web application owners, business units or the IT organization use the Catalyst Configuration Manager to create rules and configurations for their specific needs; alternatively, a single administrator can manage all the rules and settings.

Browsium Client, a lightweight, unified client agent that works with all modules in the Browsium browser management suite, includes extensions for four browsers – Microsoft Internet Explorer, Microsoft Edge, Google Chrome, and Mozilla Firefox. Catalyst also supports the ability to define and configure additional custom browsers – either additional versions of Chrome or Firefox, or an arbitrary browser to set as a target for redirection. Once Browsium Client has been installed on a system, the Browsium Controller process will load at user logon and read the Catalyst configuration from the system.

Catalyst supports both local- and Group Policy-managed settings to provide the most flexibility and truly deliver an enterprise-ready solution. Once the configuration is loaded, the browser extensions monitor the navigation process for each browser and communicate with the Controller to ensure the correct rule is followed and the appropriate browser is invoked.

Section Two

Installation

In this section, you will learn:

- ✔   About the Browsium Catalyst components
- ✔   Software requirements for Browsium Catalyst
- ✔   How to install Browsium Catalyst

2. Installation

Browsium Catalyst is simple to install – administrators need both Browsium Catalyst Configuration Manager and Browsium Client, while end users only need Browsium Client installed. This section provides details on the specific components of Catalyst, as well as information on how to manage Browsium Client installation in an enterprise environment.

Browsium Client is a lightweight, unified client agent that works with all modules in the Browsium browser management suite. Install Browsium Client once and enable Ion, Proton, and Catalyst services as required for your organization.

2.1. Catalyst Components

Browsium Catalyst is comprised of two main parts: a configuration manager tool for defining rules and settings, and a client for interpreting the configurations for end users. The client includes a Controller process and browser extensions for Microsoft Internet Explorer, Microsoft Edge, Google Chrome, and Mozilla Firefox.

- **Catalyst Configuration Manager (Catalyst-Setup.exe)**
  Catalyst Configuration Manager (`BrowsiumCatalystManager.exe`) is used to create and manage Catalyst configurations (browsers, rules, and settings), which will ultimately be deployed to end user PCs. This application is not intended to be run by end users, so Catalyst-Setup.exe should not be installed broadly – installation of this package should be limited to system administrators, web application owners, or business unit managers.

- **Browsium Client (Browsium-ClientSetup.exe)**
  Browsium Client is responsible for loading Catalyst configuration data and redirecting browser traffic based on that configuration. The client package should be installed on all Catalyst-licensed PCs in your organization. The package consists of two components:

  - **Browsium Controller**
    The Browsium Controller (`BrowsiumController.exe`) is the main component of the client infrastructure used by Catalyst to handle rules implementation and redirection. The Client consists of a background process/listener service that must be running for Catalyst to operate. Without this component, the browser extensions cannot communicate properly and redirection will fail to function properly. Browsium Management Service (`BrowsiumService.exe`) runs as a Windows service with System privileges to handle content loading for projects that require elevated permissions. Browsium Monitor (`BrowsiumMonitor.exe`) is responsible for ensuring that Controller is running reliably.

  - **Browsium Client Extensions for Microsoft Internet Explorer, Microsoft Edge, Google Chrome, and Mozilla Firefox and Edge**
    Browsium Client installs extensions to each browser to enable communication between the browsers and the Browsium Controller.

2.2. System Requirements

The following system specifications are required to run Catalyst.

- Microsoft Windows
  - Windows 7 or later
  - Windows Server 2008 R2 or later (for multi-user terminal servers)
- Microsoft Internet Explorer 8, 9, 10, or 11
- Google Chrome 39 or later
- Mozilla Firefox 48 or later (41 or later using Browsium legacy extension)
- Microsoft Edge 40 or later (machine must be domain joined)
- Microsoft .NET Framework Version 3.5 SP1 or later (full version only)
- 1 GHz or faster 32-bit (x86) or 64-bit (x64) processor
- 1MB or more system memory; 2GB when used on multi-user terminal servers

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

2.3. Installing Browsium Catalyst Configuration Manager

This section covers the installation process for the Browsium Catalyst Configuration Manager. The Browsium Catalyst Configuration Manager is only required, and should only be installed, on systems for Administrator use and building deployment configurations. End users SHOULD NOT have the Configuration Manager installed. End users should only have the Browsium Client installed. Administrators will need the Browsium Catalyst Configuration Manager and the Browsium Client be installed on their systems for project development and testing.

The steps for installing Catalyst Configuration Manager are as follows:

1. Locate the Catalyst Configuration Manager Installation file (***Catalyst-Setup.exe***) and double click to run the program.
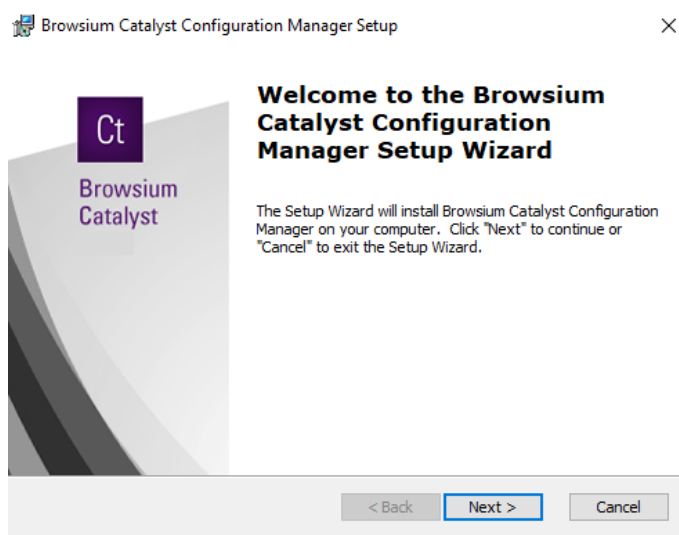
2. Confirm you have read and agreed to the End-User License Agreement (EULA) by clicking '**I agree to the terms in the License Agreement**' and **Next** to continue with installation.
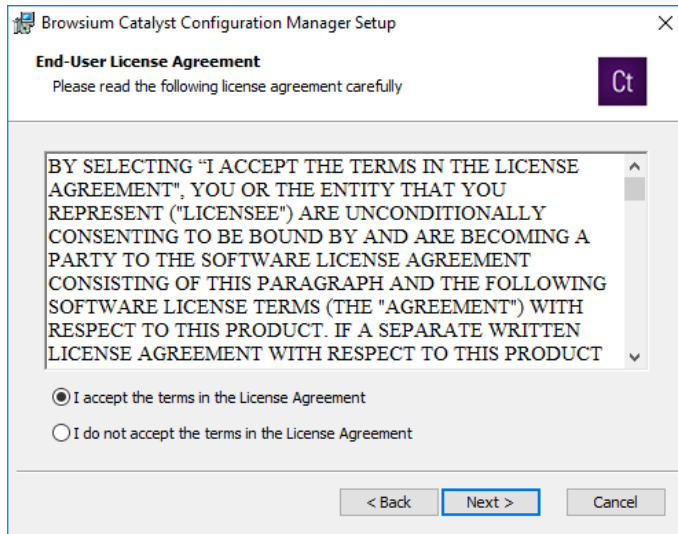
3. By default, the installer places the required files in C:\Program Files\Browsium\Catalyst Manager (32-bit systems) or C:\Program Files (x86)\Browsium\Catalyst Manager (64-bit systems) on the system drive.
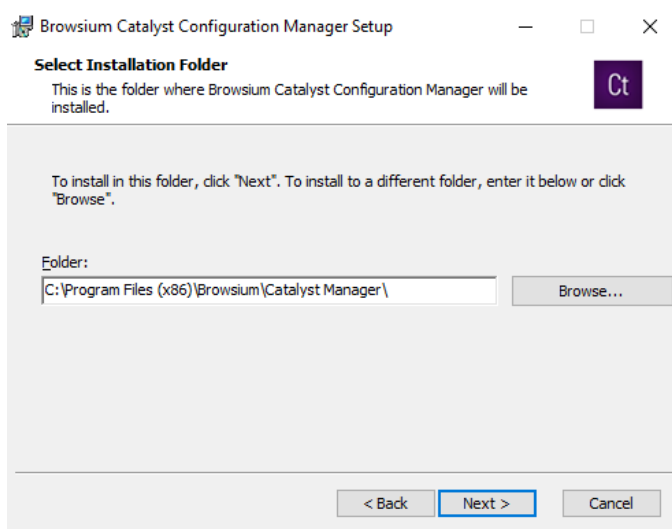
Select an installation location and click **Next.**

4. Now you're ready to install Catalyst Configuration Manager. Click **Install**.
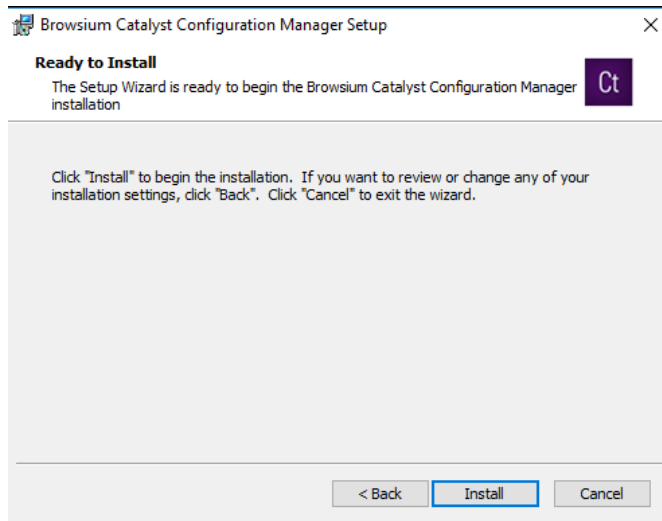
Catalyst Configuration Manager installation requires Administrator rights so the installer may generate a UAC prompt before installing.

5. 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 Catalyst.
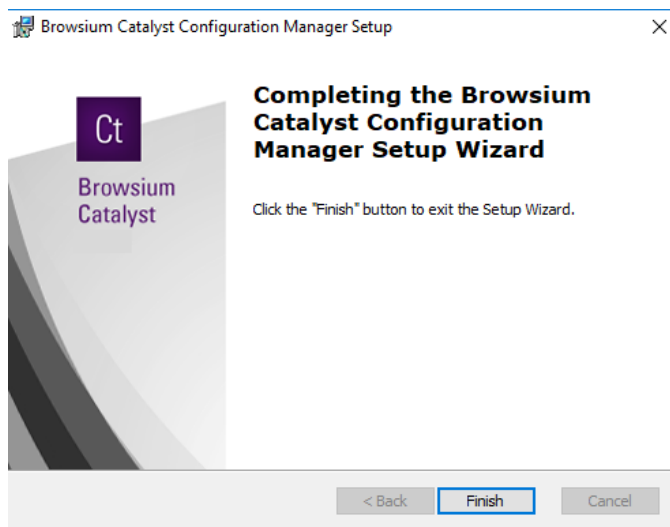
2.4. Installing Browsium Client

This section covers manual installation of Browsium Client. Network and command line installation options can be found in Command Line Switches for the Client Installer. Administrator rights are required to run the Client Installer. Once installed, Browsium Client runs under any user account and does not require special user permissions or elevation.

You'll find Browsium Client installation program in the zip file containing the Browsium Catalyst software that you downloaded from Browsium.com.

1. To start Browsium Client installer process, double-click on the `Browsium-ClientSetup.exe` file. Administrative rights are required to properly complete the installation process. The first screen provides a basic introduction. Click **Next** to get started.
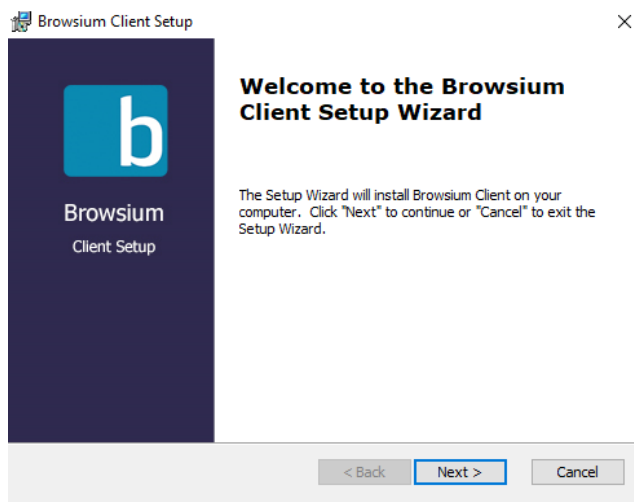
2. The next screen contains the End User License Agreement (EULA) for Browsium software. Read and accept the terms of the EULA to proceed, then click **Next**.
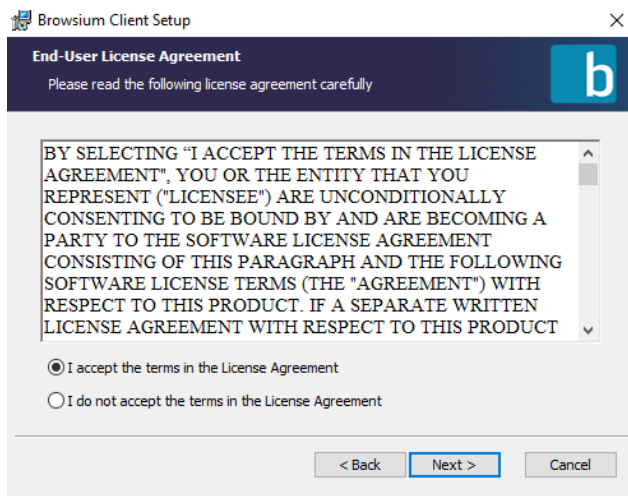
3. To function properly, the Browsium Client requires browser extensions must be installed and enabled. All supported Web browser extensions are installed during Setup and enabled during first run of the Browsium Client. To ensure proper operation in deployment, the Browsium Client will periodically monitor extensions state and enable them if they become disabled. Installing and enabling extensions often requires open browser windows to be closed.

Click **Yes** to allow the Browsium Client to enable the extensions and retain control over the setting once deployed. Selecting **No** only installs the browser extensions, the Browsium Client extensions will not be automatically enabled or setup for ongoing management. Clicking either the **Yes** or **No** options will require all open browser windows be closed, and the installer will close them automatically.

If you don't want to close open browser windows select **Cancel** at this time.

Closing browser windows automatically may cause unsaved work to be lost. Verify browser window activity prior to allowing the installation process to close any browser window.
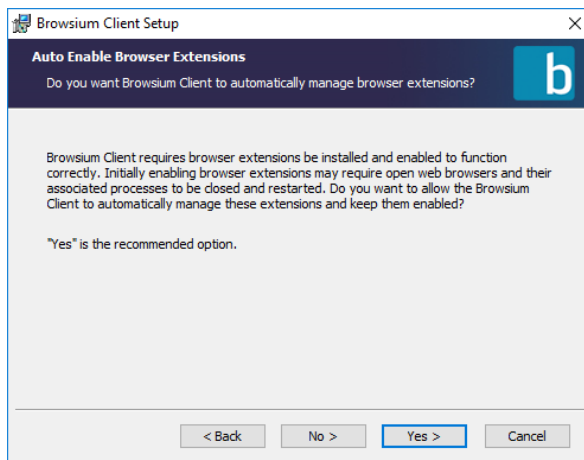
4. Selecting Typical installs Browsium Client Extensions for Internet Explorer, Chrome, and Firefox, and installs the software in the default Program Files directory for 32-bit applications – 'Program Files' on 32-bit systems and 'Program Files (x86)' on 64-bit systems. If you don't need to change these defaults, select **Typical**, then **Next** to proceed with Browsium Client installation.
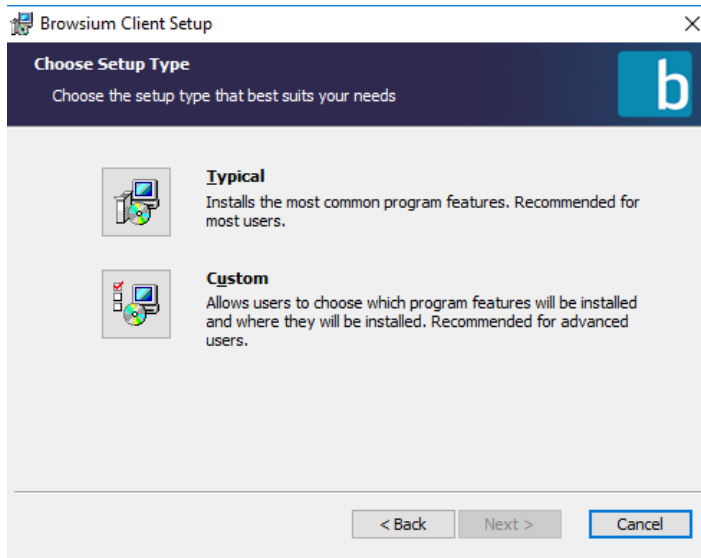
Selecting Custom enables manual selection of browser extensions to be installed, as well as the installation directory.
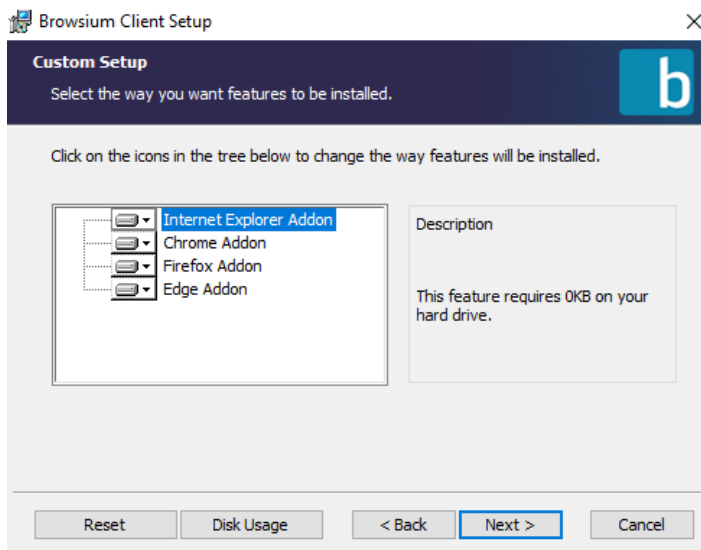
5. Optionally, specify the Client Configuration, or click Next to keep the default, blank values for the Browsium Client Management Server (BCMS) and the Browsium Client Registration Token. These values can be set using the command line switches as detailed in section 2.6. They can also be set via Group Policy or other remote registry configuration tools in an enterprise deployment as detailed in section 5 – Deploying Browsium Client.
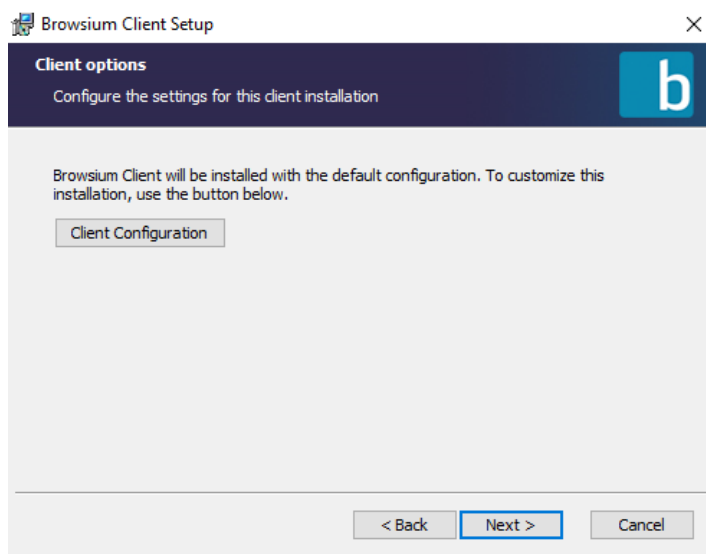
6. Selecting the Client Configuration button presents a dialog that requests the URL of your BCMS and your Browsium Client Registration Token. The BCMS URL is required to communicate with the server for client management and authorized usage. If the BCMS value is not provided here, the Browsium Client will operate for 30 days before it stops working. You do not need to setup a BCMS prior to deploying the Browsium Client, but it is advised so as to avoid unexpected client shutdowns at the end of the grace period.

   A Client Registration Token is optional and only required for Proton deployments. Click **OK** and then **Next** to continue.
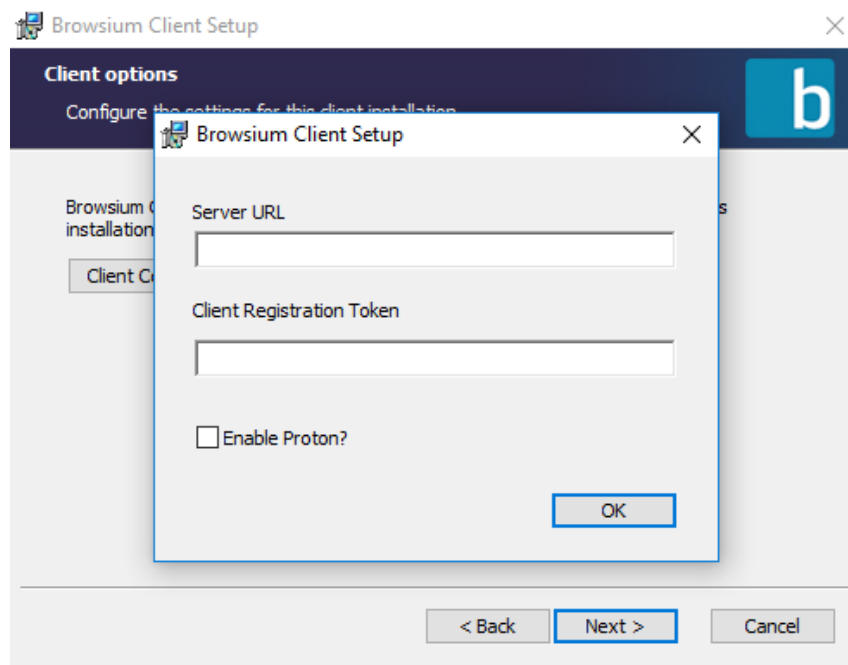
If you are entering a Server URL, it must be specified using HTTPS, the Fully Qualified Domain Name (or Machine Name) and port, followed by '/server' to ensure Browsium Client can communicate with your Browsium Client Management Server.

7. Browsium Client is now ready to install. Click **Install** to proceed.
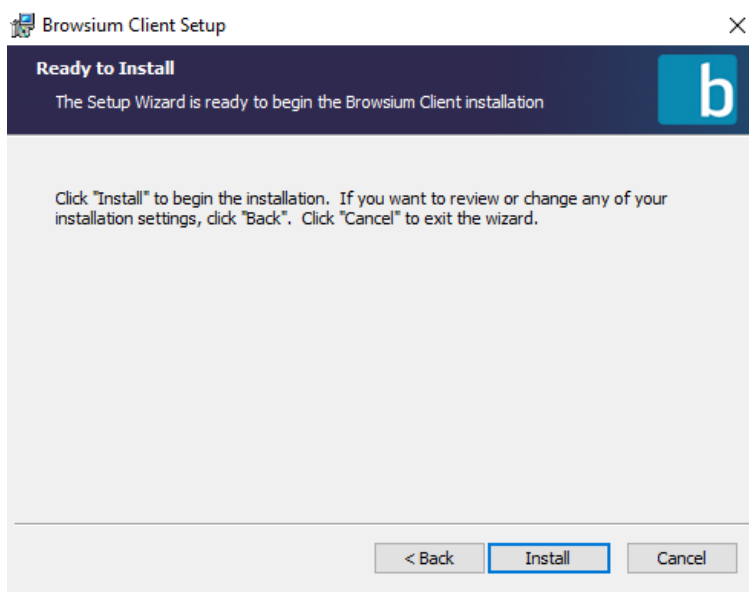
8. The following screen will display to indicate success when Browsium Client installation process has finished.
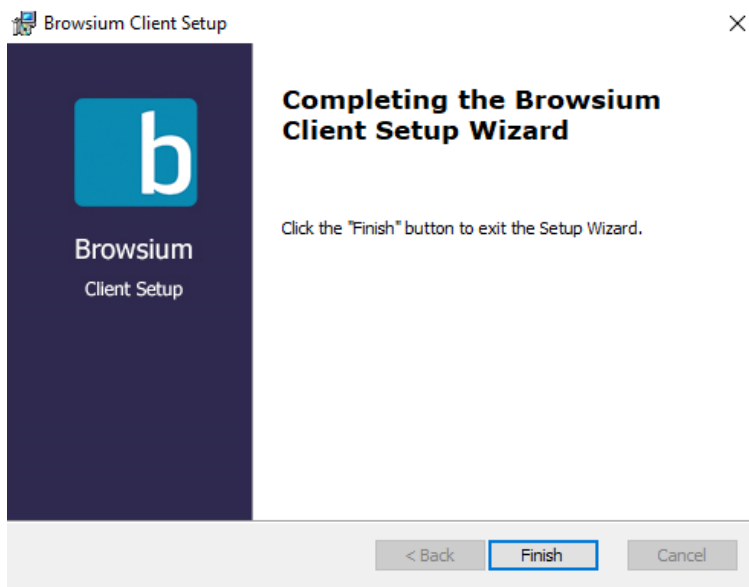
2.4.1. Confirming Browser Extension Installation

By default the Browsium Client will enable and manage the extension activation process. This section is only required if choosing to manually control enabling the extension process.

To confirm Browsium Client installation has completed properly, launch Internet Explorer and select Tools->Manage Add-ons and ensure Browsium Internet Explorer Extension is listed and Status is set to Enabled.
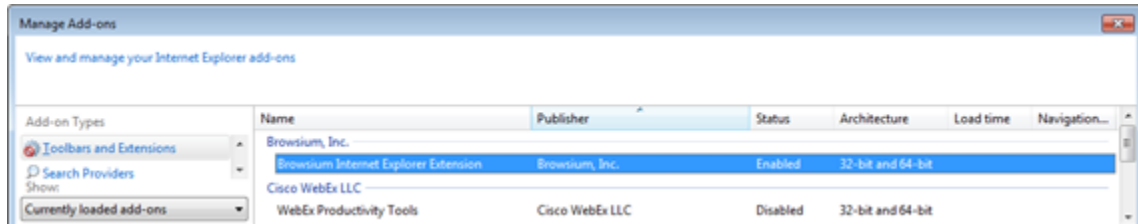
To confirm Browsium Client installation has completed properly for Google Chrome, launch Chrome and open the Settings menu. Select the More Tools / Extensions menu item and verify Browsium Client Extension is installed and the Enabled checkbox is set. You may also enter chrome://extensions in the address bar to navigate directly to the Extensions page.
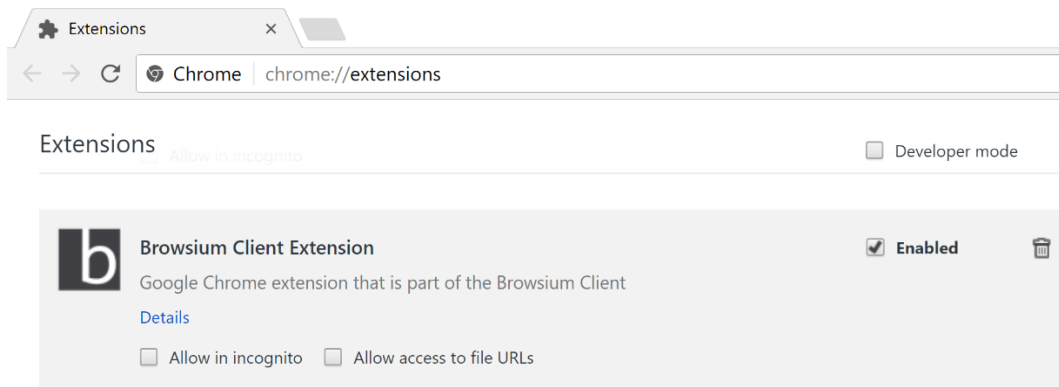
To confirm Browsium Client installation has completed properly for Mozilla Firefox, launch the browser and open the Settings menu and select Add-ons (or navigate directly to about:addons in the address bar). Select the Extensions item from the left column and verify the Browsium Proton Client Extension is installed and Enabled (which means the Disable button is visible; when Disabled, the Enable button is visible).
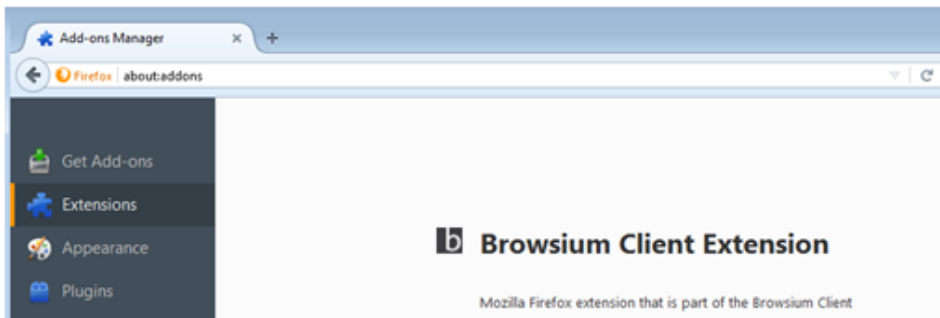
Browsium Client can be deployed using enterprise software distribution systems, including scripting of the BCMS URL, and Client Authentication Token, along with automatically enabling the extensions for Internet Explorer and Chrome.

2.5. Command Line Switches for Browsium Client Installer

Browsium Client setup (`Browsium-ClientSetup.exe`) supports network-based installations using Windows Installer for organizations that use software distribution systems or want to deploy via installation scripts and logon applications. Browsium Client setup passes through command line parameters to Windows Installer (msiexec.exe), and as such, is able to provide all of the standard Windows Installer command line functionality as documented by Microsoft. The constant '//' may be used any place the command line requires the package name. Documented below are some examples of common options:

2.5.1. Installation Options

Switch	Description
`</uninstall // | /x //>`	Uninstalls the product
`APPDIR=<path>`	Installs product to a specific directory, other than the default location
`SERVER_URL=<Proton Server URL>`	Installs the Proton Server URL in ConfigurationServerURL value in the registry
`ION_PRODUCT_KEY`	Installs an Ion license key in the registry
`CATALYST_PRODUCT_KEY`	Installs a Catalyst license key in the registry
`KEEP_EXTENSIONS_ENABLED`	1 or 0. Enables the automatic management of the Browsium Client extension

2.5.2. Display Options

Switch	Description
`/quiet`	Quiet mode, no user interaction
`/passive`	Unattended mode - progress bar only

`/q[n|b|r|f]`	Sets user interface level, where:
	    n - No User Interface
	    b - Basic User Interface
	    r - Reduced User Interface
	    f - Full User Interface (Default)

`/help`	Shows help information

2.5.3. Restart Options

Switch	Description
/norestart	Do not restart after the installation is complete
/promptrestart	Prompts the user for restart if necessary
/forcerestart	Always restart the computer after installation (the default if no other option is selected)

2.5.4. Logging Options

Switch	Description
/l[i|w|e|a|r|u|c|m|o|p|v|x|+|!|*] **<LogFile>**	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
/log <LogFile>	Equivalent of /l* <LogFile>

2.5.5. Repair Options

Switch	Description
/f [p][o][e][d][c][a][u][m][s][v] **{Package | ProductCode}**	Install keeping a log file, where: p - Reinstalls only if file is missing o - Reinstalls if file missing or older version installed e - Reinstalls if file missing or equal or older version is installed d - Reinstalls if file missing or different version installed c - Reinstalls if file is missing or the stored checksum does not match the calculated value a - Forces all files to be reinstalled u - Rewrite all user-specific registry entries m - Rewrites all computer-specific registry entries s - Overwrites all existing shortcuts v - Runs from source and re-caches the local package

2.6. Upgrading from Evaluation to Licensed Version

If you have chosen to trial Catalyst in a proof of concept using the 30-day evaluation period, or simply started your deployment without a Browsium Client Management Server (BCMS) in place and now want to ensure all Catalyst clients function properly, you will need to have a BCMS in place.

Clients must contact the BCMS server to obtain authorization and continue functioning as licensed software. Failure to regularly connect to the BCMS server (at least once every 30 days) will result in the client machine becoming 'unauthorized' and the software will no longer function.

Upgrading clients to licensed status is simply a matter of configuring them to contact the BCMS server. As long as the BCMS server includes a valid license file, provided by Browsium, client communication with the BCMS will ensure registration and authorization happen automatically. Clients DO NOT need the license file deployed to any systems.

For more information about BCMS please see the Browsium Client Management System Administration Guide.

Section Three

# Introduction to the
Catalyst Configuration Manager

In this section, you will learn:

- ✔ More about Catalyst Configuration Manager
- ✔ How to manage browsers, rules, and settings in Catalyst Configuration Manager

3. Catalyst Configuration Manager Overview

Catalyst Configuration Manager enables you to create and manage rules that define the websites you want to open using a browser designated by Catalyst. This section looks at the various elements of Catalyst Configuration Manager. Configuration Manager is designed with the look and feel of an MMC snap-in, with three main functional areas:

Objects Pane (Left) – Tree view enabling access to the Browser Manager, Rules Manager, and Project Settings for a given project
Content Pane (Center) – Main data and content window
Actions Pane (Right) – Contextual links and descriptions for common tasks and steps
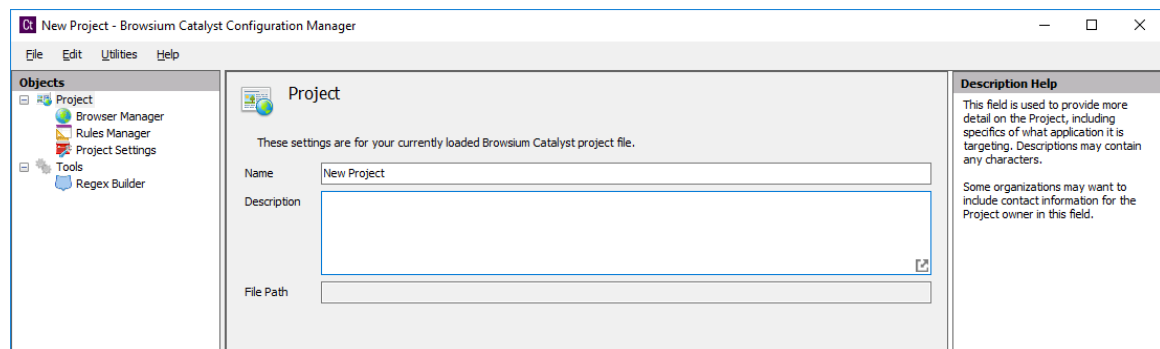
Some data entry fields will display an icon in the lower right hand corner of the text entry field. Clicking this link on any screen will open the extended and powerful text editor. The text editor is provided for easily working with complex text fields, as well as offering syntactical highlighting for 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. Some interfaces may not support items such as Copy and Paste, and will not be displayed.

See the Project Node, Browser Manager Node, Rules Manager Node, and Project Settings Node sections for a detailed description of each Objects Pane Node.

3.1. Menu Bar

Catalyst Configuration Manager Menu Bar provides access to key functions related to creating, editing, and testing Catalyst projects.
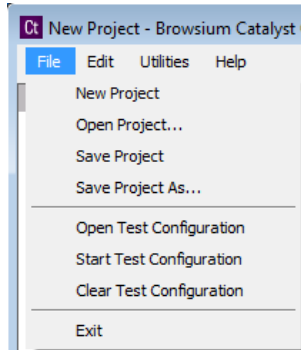
To aid in configuration testing and tuning, Catalyst Configuration Manager allows project developers to automatically apply project file pointers directly in the system registry and start or restart Browsium Controller to process the new configuration. This is performed via Start Test Configuration from the File menu, which writes the file pointer to the LoadFromFileName registry value in HKEY_LOCAL_MACHINE\SOFTWARE\(Wow6432Node)\Browsium\Catalyst before starting or restarting Browsium Controller.

This option reduces delays and overhead of manually editing the registry or deploying registry changes via Group Policy. Open Test Configuration opens the project file found in LoadFromFileName (if Start Test Configuration had been run prior). Clear Test Configuration clears LoadFromFileName in the registry and stops Browsium Controller.

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

The Edit menu provides Cut, Copy, and Paste functions to assist in project editing and moving rules between projects. These functions are also available in the Actions pane for the Rules Manager for easy access. The Edit menu also has a Find function (available from the menu or via Control-F) which searches for specific strings inside the project.
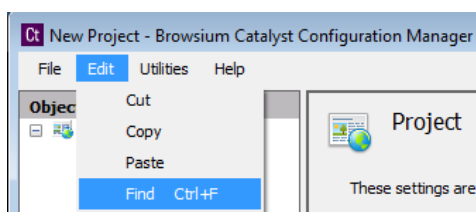
The Find function will search all text in the Project Node, excluding the Project Node itself, starting with the current location of the cursor, and continue looping through the project continuously as you select Find Next.
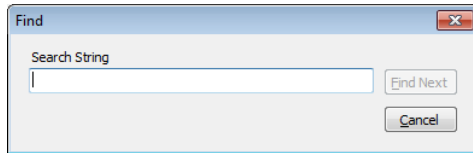
If the string is not found anywhere in the project, you will receive a 'string not found' error.

The Find function does not highlight the matched strings in the project file. It simply brings the appropriate manager node or page into focus.

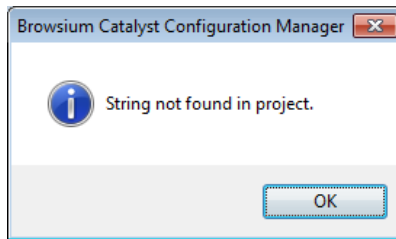You can use the Utilities menu to manage Browsium Controller (`BrowsiumController.exe`) process. You may need to Start/Stop/Restart Browsium Controller to load new configurations or reproduce troubleshooting steps.
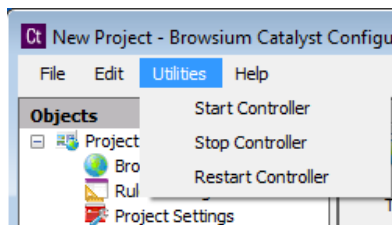
3.2. Project Node

The Project node displays the configuration file currently loaded in Catalyst Configuration Manager. An individual project file (`.CAX` file extension) contains all of the configuration information needed for a given set of browsers and rules. Catalyst Configuration Manager is designed to load one project per instance. Loading multiple projects can be done by opening each project in another Catalyst Configuration Manager instance. Browsers and rules can be copied and pasted between projects.
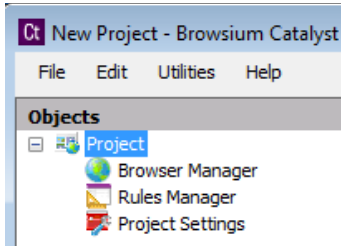
Loading a project (or creating a new project) brings up the Project details in the Content Pane.
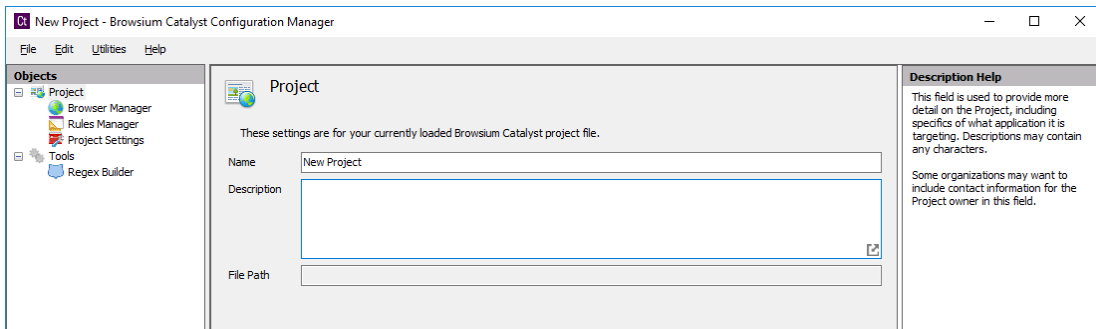
Project details (Name and Description) can be edited on this screen. Projects are assigned a default name and a blank description and should be edited to reflect information and labeling relevant to your organization. The project's File Path is blank until you save the project file. This field cannot be edited, but will be updated automatically if the project is saved to a new location.

3.3. Browser Manager Node

The Browser Manager contains the list of defined browsers for a given system. Catalyst supports four preset browsers – Google Chrome, Microsoft Edge, Microsoft Internet Explorer, and Mozilla Firefox– by default. Catalyst Configuration Manager will attempt to automatically determine the installation path for each installed browser. If a required browser is not present or not installed on the machine where the Browsium Client is running and a rule requires the given browser, that rule is skipped and may result in a misconfiguration or unexpected behavior.
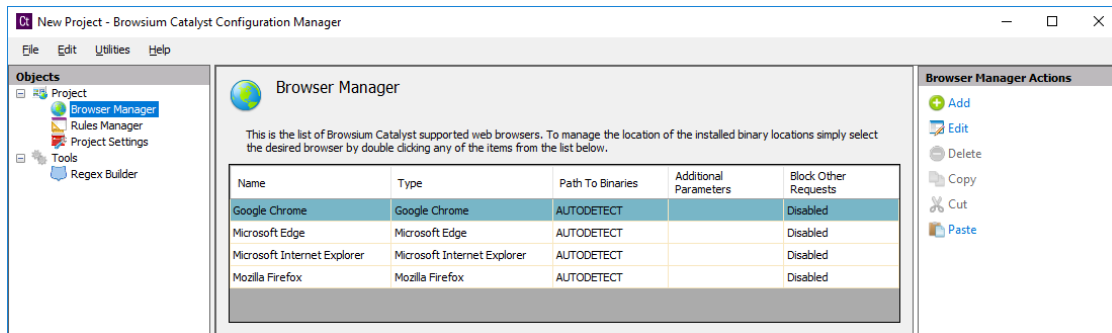
Selecting a browser and then selecting Edit from the Browser Manager Actions pane will bring up the Browser Editor. For the preset browsers (Chrome, Edge, Internet Explorer, and Firefox), you will see four read-only fields and one field (Block Other Requests) which can be toggled between Disabled and Enabled. Custom browsers allow editing of all fields.
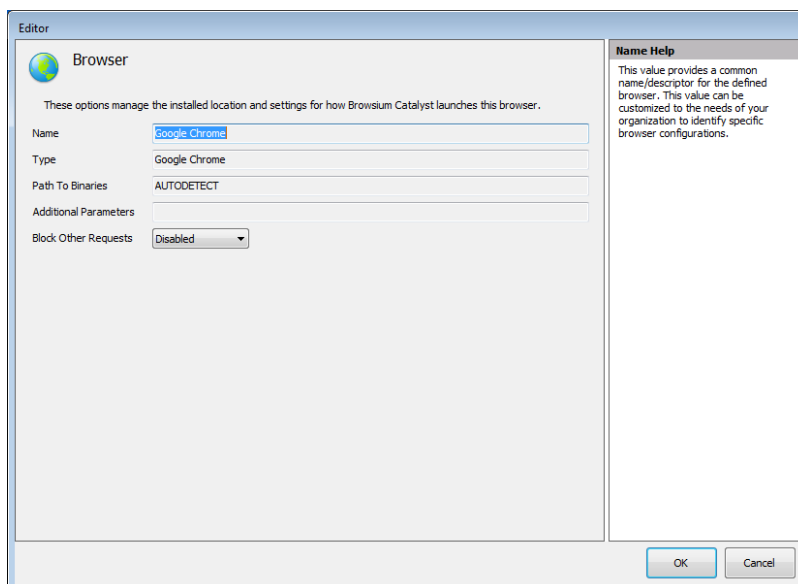
Browser Name – This is the name of the browser. Browser names are ready-only for the preset browsers, but can be modified to reflect naming relevant to your organization for custom browsers.

Browser Type – The Browser Type value is used by Catalyst to identify which type of browser is defined by the setup. This value is required to support multiple browser instances and variations. Setting the Browser Type incorrectly may cause unexpected behaviors.

Path to Binaries – This is the path location containing the application binaries. Catalyst needs to have the accurate location of the binaries to properly load the defined browser when rule conditions are met. Errors in the path location will cause Catalyst to fail to properly load a browser or web content. Paths to preset browsers are always automatically detected. Use of the system variable %ProgramFiles% will ensure that the path to a custom browser works properly on both 32-bit and 64-bit Windows systems. See the example in section 3.3.1.

The Path to Binaries field must not contain quotes, even if the path includes spaces. Catalyst will properly handle the spaces to locate to browser executable.

Additional Parameters – In addition to launching a desired browser, Catalyst can open the browser using additional parameters specified here. Ensure any additional parameters are correct for the specific browser as incorrect items may cause the browser to stop loading.

Additional Parameters are often used when creating a custom browser for Google Chrome to launch Chrome in Application Mode (with no address bar or menus). Use the syntax --app=${url} to launch Chrome in Application Mode.

Block Other Requests – By default, Catalyst is designed to only intervene in content loading and redirection when explicitly defined by a rule. Setting this option will prevent the user from loading any content in the specified browser **unless** the content matches a rule.

Use caution when setting Block Other Requests to Enable, as users may become confused when the browser redirects any address entered that does match rules targeted for that browser.

3.3.1. Add a Custom Browser

Selecting Add from the Browser Manager Actions pane enables the addition of a custom browser. Custom browsers are typically Target Browsers only, meaning that they can be the destination browser for a rule, but cannot process rules themselves because they don't have a Catalyst browser extension. Examples include Opera and Safari, which would be set as Type = Custom.

However, a custom browser can be created of type Google Chrome and be capable of processing rules. This is typically done to launch Chrome with special parameters, such as Application Mode or Kiosk Mode.

The following example configures a custom browser named Chrome App Mode, launching Chrome in Application Mode. Although the Type is still Custom, Catalyst recognizes Google Chrome as the binary and allows rules to be processed.

3.4. Rules Manager Node

The Rules Manager is the main interface for creating, editing and managing evaluation criteria which Catalyst uses to determine which browser opens each web application. This section contains details on the various elements of this interface.
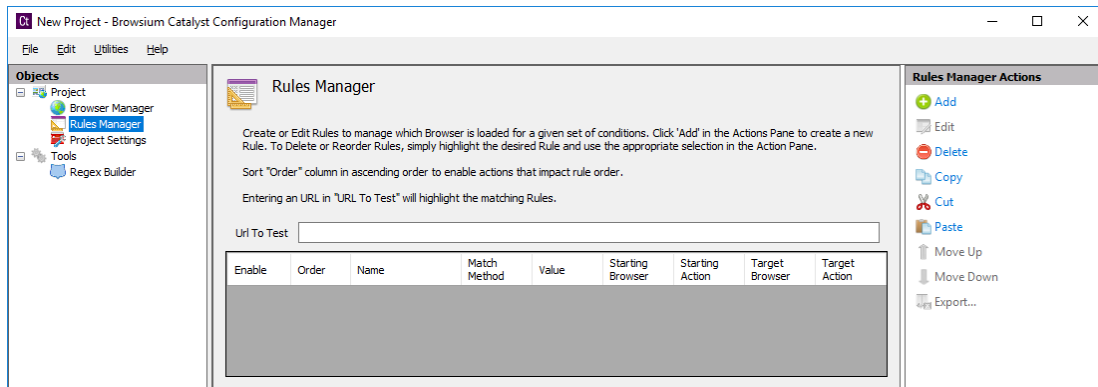
The Rules Manager shows the hierarchical rules list that Catalyst uses to determine what (if any) action to take when a web address is entered. The **Url To Test** box allows administrators an easy way to enter a given Url and see which Rule will be triggered by the value. The heading for each column in this window refers to the specific rule element for a given rule. The Rules Manager Actions pane will display the available options for managing rules.
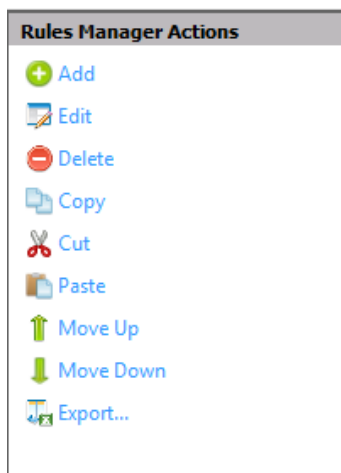
Add – To create a new rule, click the Add link in the Actions Pane to bring up the Rule Editor window. By default, the rule will be added at the bottom of the list in Rules Manager. If you first select a rule in Rules Manager, the new rule will be added immediately below the selected rule. See the description of the Move Rule Up/Move Rule Down feature for more information on rule order. The Rule Editor 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. If 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 Enable checkbox.

Copy – To copy a rule, select the item from the Rules Manager Window and click Copy 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 Cut 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 – Rules are evaluated in the order they are stored, so the sequence of rules can be critical to the proper functionality of your web applications in Catalyst. By default, rules are ordered in the sequence they are added. You can override this default by selecting a rule in Rules Manager prior to adding a new one. The new rule will be added immediately below the selected rule. To manually adjust the order of a rule, simply highlight the rule and use the Move Up and Move Down actions to move it to the proper placement.

Export – Saves the complete list of rules to a CSV file, with the following fields: Enable, Name, Match Method, Value, Starting Browser, Starting Action, Target Browser, Target Action. This file can be used to share information about the Catalyst rules with other members of the IT team. Note that Catalyst has no rules import function, so modifications to this CSV file cannot be programmatically applied to your Catalyst project file.

3.4.1. Rule Editor

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

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. To disable a rule, click the checkbox to remove the check or click the checkbox next to the rule name in the Rules Manager.

Description - Most organizations will have multiple staff members working with rule creation and Catalyst configurations. To reduce confusion when working with multiple team members, the description field can be used to provide notes or even detailed explanations of the purpose for the given rule. Some organizations also require the owner of the rule to be identified in the description to streamline internal communications when a problem or question arises with a given rule.

Match Method – Most rules require a simple string match of the URL to trigger Catalyst to direct the navigation to the chosen browser. Some complex URL configurations require advanced pattern matching techniques, so Catalyst provides the ability to match based on

regular expressions (RegEx) if needed. You may also use Zone as the match method so that rules are triggered based on the Internet Explorer Zone of the target web application.

Rules can be set to trigger based on following types of pattern matching.

Simple – Host Name Only (default): Performs case-insensitive, substring matches on the Fully Qualified Domain Name (FQDN). E.g., will match "example" in http://www.example.com.

Simple – Full Url: Performs case-insensitive, substring matches on the entire Url including subpath AND query string.

Starts With: Performs case-insensitive, 'starts with' string matches. If provided, the protocols must exactly match. (e.g., http://www.example.com will not match https://www.example.com)

RegEx: Performs complex matching using regular expressions. Be careful to escape special characters such as periods.

Zone: Matches URLs that are in specified zones.

Value – This field contains the string or integer to conditionally match for a Catalyst rule to be triggered. If Match Method is set to Zone, the Value field will change to a dropdown menu containing Computer, Internet, Local Intranet, Restricted Sites, and Trusted Sites.
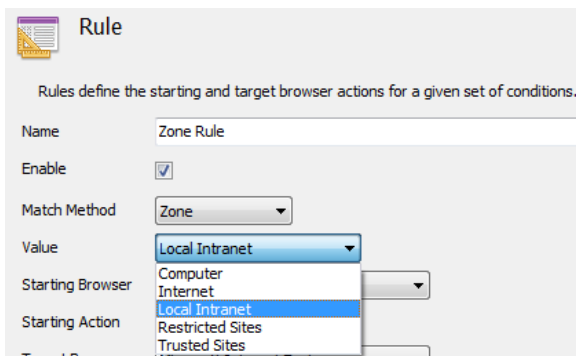
Starting Browser – This field determines which browser will evaluate the rule. The default is Any, meaning all browsers running the Browsium Client extension will evaluate the rule. A specific Starting Browser can be chosen so that Catalyst will evaluate the rule in that specific browser.
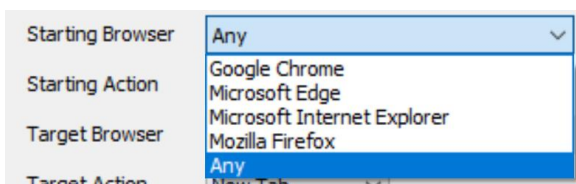
Starting Action – This field defines the Starting Browser navigation behavior. Same Page instructs the Starting Browser to cancel the desired navigation and remain on the current page, while opening the new URL in the Target Browser. Redirect navigates the Starting Browser to a redirection page (which can be customized by Catalyst admins), informing users as to why their navigation action opened in a different browser. Close Tab closes the Starting Browser tab (or entire browser if only one tab is open) as it navigates the Target Browser to the desired URL.

Target Browser – This field defines desired browser that will load the web page when the rule is triggered by the navigation action performed by the end user. When Target Browser is set to 'None', the navigation action will be aborted, effectively blocking end user access to that URL. When Target Browser is set to 'Any', the matching URL will be allowed in every browser, even those that have 'Block Other Requests' enabled.

Target Action – This field defines the Target Browser navigation behavior. New Tab opens the page in a new tab. New Window opens the page in a new window. New Session opens the page in the specified browser using a new user session.

3.5. Project Settings Node

The Project Settings Node gives you the ability to edit global settings for Catalyst configurations. These settings encompass features such as the Splash Screen behavior, Default Browser behavior and choice, and the location of custom redirection messages.
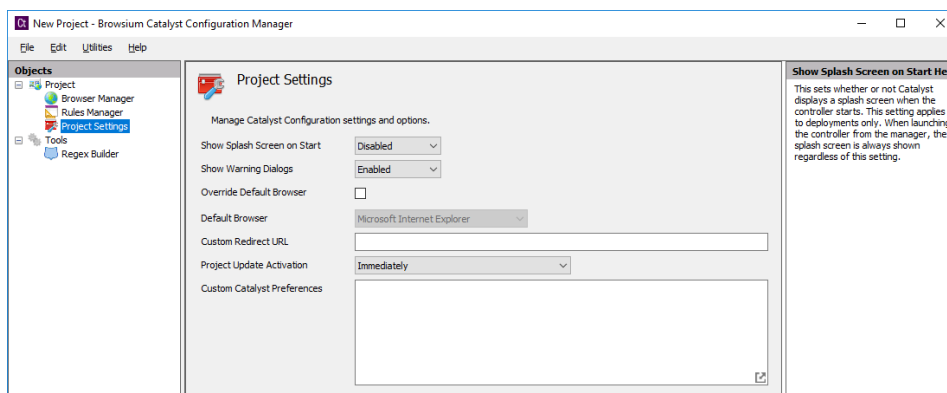
Show Splash Screen on Start – By default, Catalyst disables showing the splash screen (Catalyst graphic with the product name and version number) on client systems when the Controller is started. This option can be enabled during project configuration. The splash screen is only displayed at startup (usually set to user login) so users should not regularly see this display.

Show Warning Dialogs - This sets whether or not Catalyst displays certain warning dialogs when a test configuration is started. This setting only displays a dialog on starting a test configuration when using the manager – deployed clients will never show warning dialogs regardless of the value for this setting.

Override Default – Defines whether Catalyst should override the default browser setting on the system, making Catalyst the default handler of HTTP and HTTPS traffic. It is checked by default, and must be checked for Catalyst to seamlessly handle navigation from desktop shortcuts and links in emails. However, if it is unchecked, Catalyst rules will still be honored as long as the Browsium Client extension is running in the default browser.

Default Browser – Allows an administrator to define the default browser (as defined by Catalyst) to be used for desktop shortcuts and links in emails when no rule exists. This setting also defines the default Target Browser for new rules.

Custom Redirect URL – Enables administrators to specify the URL for a custom page displayed when a rule includes Starting Action = Redirect.

Project Update Activation – Project Update Activation enables new Catalyst configurations to be loaded on the fly without restarting Browsium Controller (which typically requires a reboot or

logoff/logon of the client computer). The field can be set to load the new configuration immediately (the default), at a scheduled time, or upon next Browsium Controller restart.

Custom Catalyst Preferences – Provides the ability to extend Catalyst functionality to support additional features and settings that may be needed for certain environments. Most customers will not need to use these optional values.

Custom Catalyst Preference	Description
Catalyst.IEAddonDetectRedirects = true	When this flag is set to "true", Internet Explorer will detect 30x redirection (typically a 302 redirect) and evaluates rules on the redirected URL. By default, 30x redirected URLs are ignored.
Catalyst.IEAddonNoZoneHopRecovery = true	There are two different "Same page" starting action behaviors implemented in the Browsium Client Internet Explorer extension when navigating between zones. Default behavior is to detect a zone hop and navigate back to the last page in the history of the starting browser. However, this functionality is unreliable in some environments where Internet Explorer's travel log API is incorrectly providing incomplete travel log information. For such cases an alternate zone hop behavior was implemented that leaves Internet Explorer at "about:blank" page after a zone hop. This alternate behavior can be invoked by setting this flag to "true".

3.6. Regex Node

The Regex (Regular Expression) Builder is for administrators to create and validate complex Regular Expressions that can be used in the rules evaluation criteria.
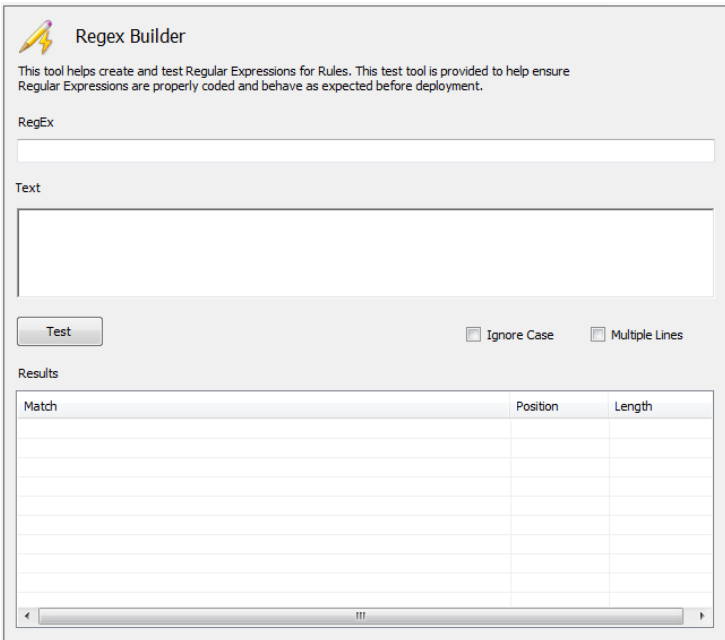
The Regex Builder provides a simple integrated tool for writing and testing Regular Expressions before using them as rules evaluation criteria to help avoid errors in pattern matching.

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 and test a Catalyst project

4. Configuration Tutorial

Once the Browsium Catalyst Configuration Manager and Browsium Client installation is complete, you can begin configuring which sites to load in the desired browser. The Browsium Catalyst Configuration Manager is provided as a simple tool to create and manage the rules and settings that govern Catalyst behavior. Its output is a Catalyst project (or configuration), which instructs the Catalyst service within Browsium Client to take action.

It's critical to enable Browsium Client extensions within Internet Explorer, Edge, Chrome, and Firefox before proceeding with this tutorial. If the Browsium Client was installed using default settings then all extensions will be enabled. If the default option was not selected ensure the manual process is followed. Read section 5.2 - Managing Browsium Client Extensions to learn how to manually deploy Browsium Client throughout your enterprise and centrally enable the extensions.

Client PCs must have Browsium Client installed, with all extensions enabled, to test or use the configurations created in the Catalyst Configuration Manager.

Windows 8 and later do not allow any program to set the default browser. If you're testing or running this tutorial on Windows 8 and later, be sure to read section 5.3 - Readying Windows 8 and later for Catalyst. This section will also provide guidance for deploying Catalyst throughout an enterprise running Windows 8 and later, as steps must be taken to configure Catalyst as the default browser on every PC.

4.1. Create and Test a Project

Catalyst offers a few ways to deliver powerful options for rule matching in order to meet the specific needs of your environment. In this tutorial, we will create a new project, add a rule, and then test the configuration by simulating a client deployment. We'll do all of this on a single system running both Catalyst Configuration Manager and Browsium Client.

For this example, we have identified a website, www.yourang.us, which must be opened using Internet Explorer – it will not work properly in any other browser.

To create the rules needed for the YouRang site, use the following steps:

1. Open Catalyst Configuration Manager. A new project is created by default. There are a variety of ways to open existing project files, but this section assumes this is your first time creating a Catalyst project. Therefore, we'll have you start building your configuration from an empty, new project.
2. Begin by naming your project "Catalyst test project" with description "My first project".
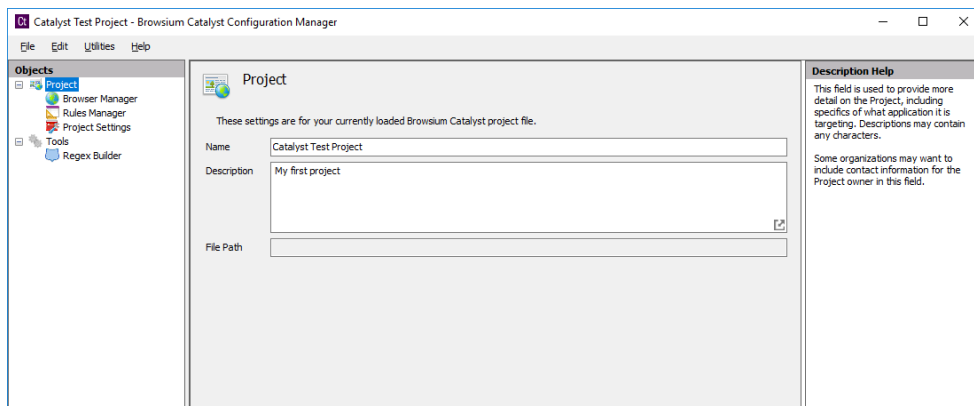
Click the Rules Manager, click the '**Add Rule**' link in the Actions pane to bring up the Rule Editor screen. Start by entering a name for the rule. Rule names are friendly names for organizational and identification purposes only and have no effect on the behavior of a rule. For this example, we will choose "YouRang Portal".
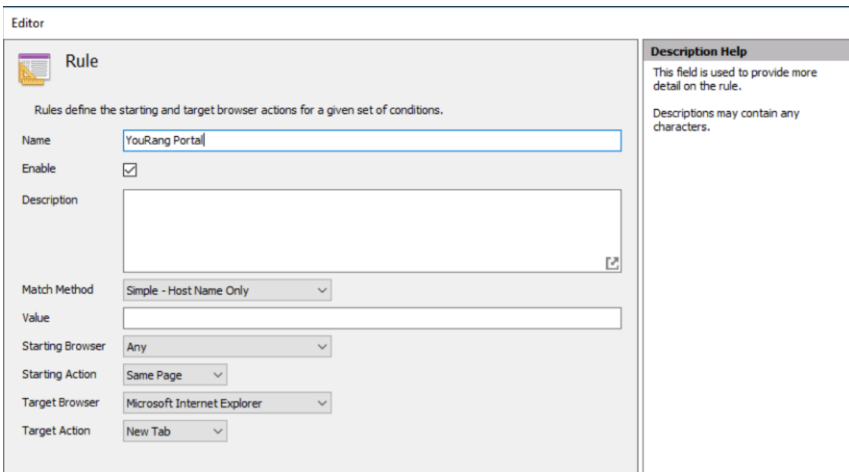
3. Keep the Enable checkbox selected to ensure the rule is active and Catalyst will trigger when the proper conditions are met.

4. Select a Match Method from the dropdown menu. There are five Match Method options for ensuring the rule is triggered when desired conditions are met. The **Simple – Host Name Only** method does a simple string match of the FQDN against the text in the Value field. The **Simple – Full Url** method does a simple string match of the Url against the text in the Value field. The **RegEx** method is provided for scenarios that require a set of complex matching criteria. **Zone** will match the Internet Explorer Zone of the target website. Most rules will only need either the **Simple – Host Name Only** or **Simple – Full Url** methods. For this example, we will leave everything set to the default (Simple – Host Name Only).
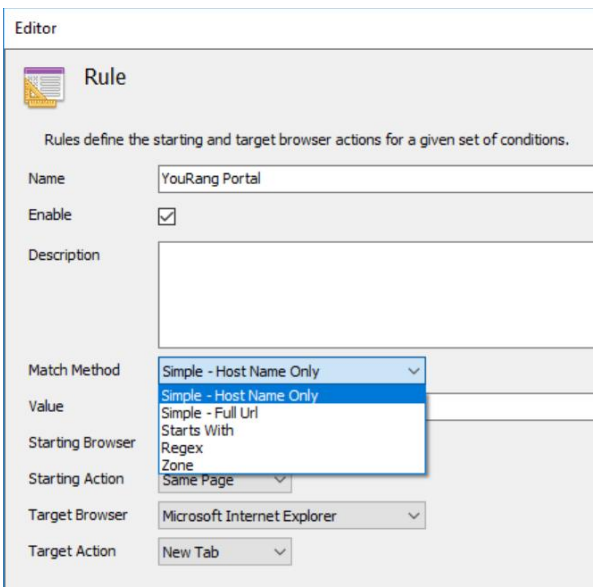
5. Enter a Value to check for rule matching conditions. For this example, we will use "yourang.us" to match our portal site.

6. The 'Starting Browser' option allows administrators to define if the user must initiate a rule action from a specific browser, or if the rule should be triggered regardless of which browser is active at the time. The default value for this setting is 'ANY' to ensure the broadest rule coverage.

7. Catalyst provides the ability for administrators to control the user experience behavior when a rule is triggered. Administrators can set Catalyst to leave the user on the same page, redirect them (and display a redirection notice page) or close the active tab. By default the 'Starting Action' option is set to 'Same Page' to avoid interrupting the user activity and simply leaving the user at their last successful navigation.
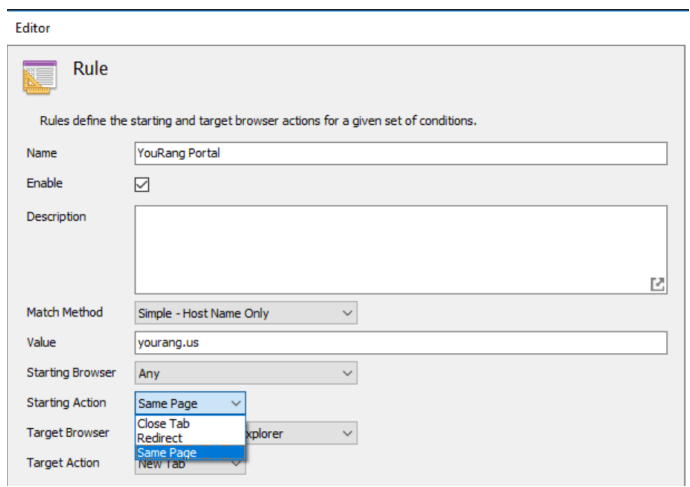
8. The 'Target Browser' setting defines which browser is loaded when the rule conditions are met. By default, this value is set to the Default Browser value listed in the Project Settings node – in this case 'Microsoft Internet Explorer'. Administrators should set this value to the desired browser. If the purpose of the rule is to block navigation (e.g. for security purposes), simply set the value to 'NONE'. For this example, we will load the YouRang portal in Microsoft Internet Explorer.
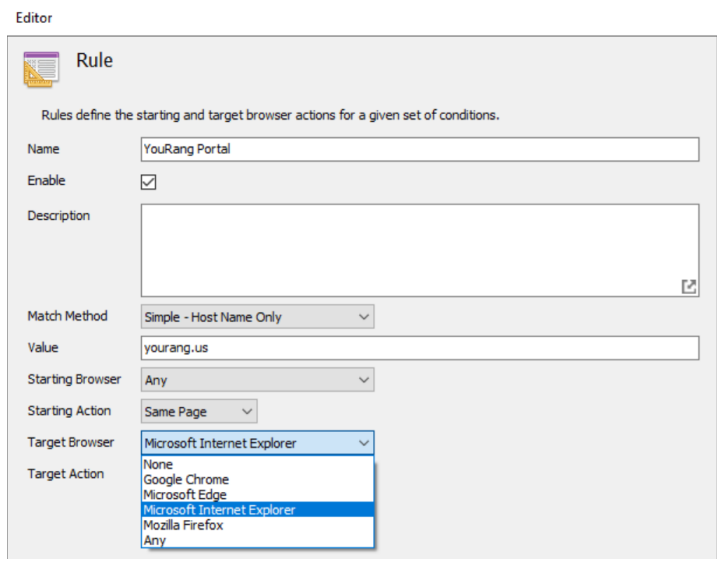
9. The final rule option is Target Action. Catalyst offers the ability to granularly control browser behaviors when loading content, offering the ability to load sites in a new tab,

new window or new session. By default, the 'Target Action' value is set to 'New tab'. For this example, we'll stay with the default 'New tab'.
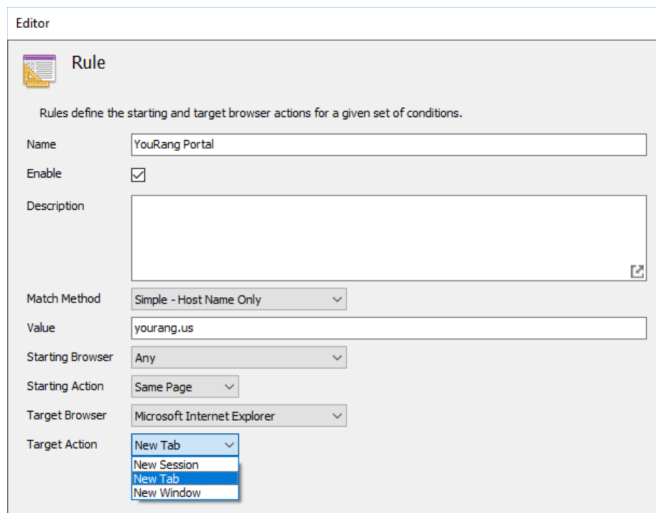
Depending on the type of Browser, some of the actions not applicable to those browsers won't be visible. Microsoft Edge only supports only New Tab action. Microsoft Internet Explorer is the only browser supporting the New Session action.

10. When you are done creating the rule, click OK to save the rule to the Rules Manager.
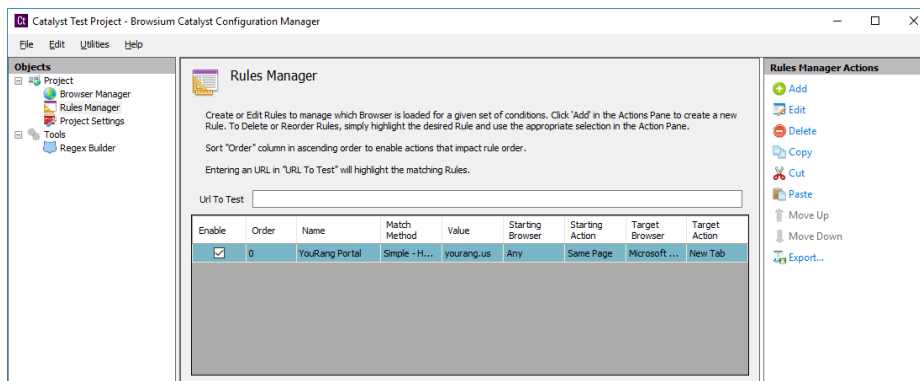
11. For our Catalyst test project, we don't need to modify any browsers in the Browser Manager or the Project Settings. We will leave everything at its default value.

12. Now we're ready to test our configuration. 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 Browsium Controller to process the new configuration.
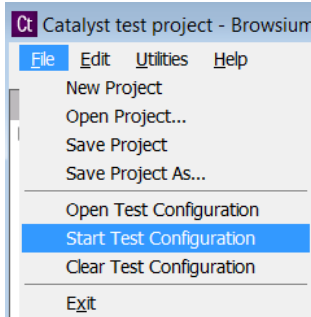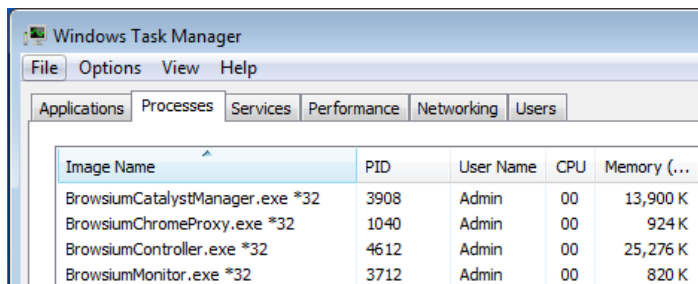
Projects should be saved regularly to ensure work is not accidently lost. Catalyst Configuration Manager does not auto-save work in progress, other than when Start Test Configuration is executed.

Once Start Test Configuration has been executed, Browsium Controller will display the Browsium Splash Screen to inform you that the Controller has started. Note that this occurs even though our configuration has the Show Splash Screen on Start setting disabled (as it is by default). This is because Catalyst Configuration Manager will always override this setting when it starts the Controller during project development. If you have configured Ion or Proton, you'll see these entries here as well. Both are 'Not Active' in our tutorial.
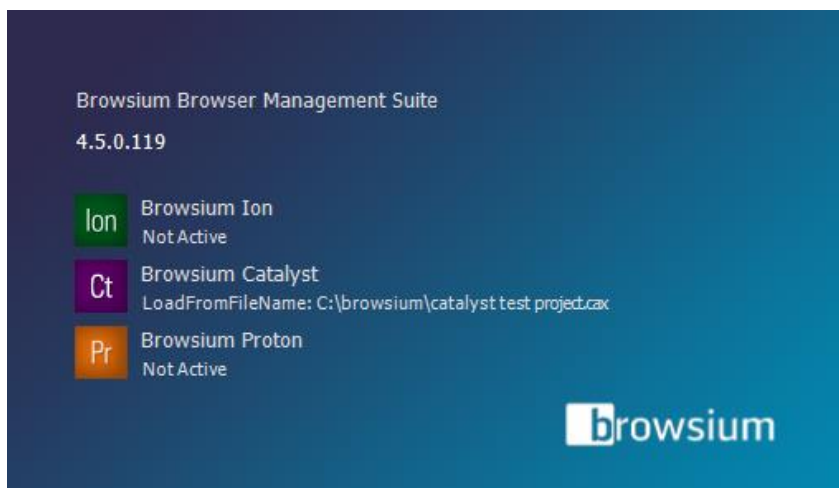
13. After the test configuration has been started, you may want to confirm that the Browsium Client processes are all running by viewing them in the Task Manager. The key processes are BrowsiumController.exe and BrowsiumMonitor.exe. Those will only run when the Controller is running with a valid Catalyst configuration (or a valid Proton or Ion configuration, as all three modules share a single Browsium Controller).

BrowsiumChromeProxy.exe is part of Browsium Client Extension for Chrome and always runs when Browsium Client is installed. BrowsiumCatalystManager.exe is the Catalyst Configuration Manager and only runs when the Manager application is running during project development and testing.

14. Now that we've confirmed all process are running correctly, it's time to test the configuration in our browser. Simply open Chrome (or Firefox) and enter the URL www.yourang.us in the address bar. Catalyst will automatically stop the navigation in Chrome and open Internet Explorer to the YouRang site.
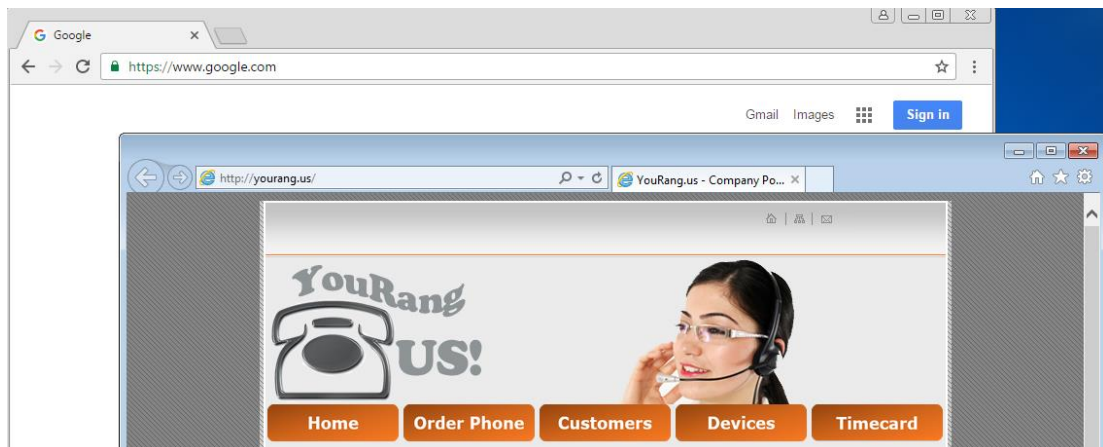
Section Five

Catalyst Deployment in the Enterprise

In this section you will learn:

- ✔ How to use Group Policy to manage Browsium Client extension settings for each browser
- ✔ To automatically enable and lock down Browsium Client extensions on remote systems
- ✔ To configure other settings to improve the Browsium Client experience for end users

5. Catalyst Deployment in the Enterprise

Like any enterprise software deployment, Catalyst deployment in the enterprise requires careful planning. This begins with Browsium Client, which is designed to be centrally deployed and managed by IT, and be completely invisible to end users.

The Catalyst configuration required by Browsium Client is delivered by an independently-deployed project file. Once Browsium Client is installed on all PCs in the enterprise, and configured with a pointer to the project file, no additional direct management is necessary.

To provide maximum flexibility for managing enterprise deployments, Catalyst supports a variety of methods for deploying Browsium Client and hosting project files (containing Catalyst rules and settings). In this section, we'll examine the available deployment options, provide recommendations, explain the loading & caching behaviors, and offer specific deployment guidance for typical enterprise scenarios.

The Catalyst configuration is read from a Catalyst project file – a single, XML-formatted data file containing the browsers, rules, and settings that were created by Catalyst Configuration Manager. The file can be pushed out to a known location on end user PCs, stored on a shared network location, or hosted at an easily accessible URL. The project file can be updated easily and separately from Browsium Client 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 Catalyst 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. As many enterprise organizations are standardizing on a single Windows image, standardizing on a single Catalyst configuration is a modern best-practice.

A client system loads its Catalyst configuration by reading the project file referenced in the LoadFromFileName registry value. As a Catalyst administrator, you determine the best way to deploy this registry value 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. The registry preference extension for Group Policy is often the most efficient way to streamline the LoadFromFileName registry value deployment. 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 Catalyst Knowledge Base. See section 5.4 for details on deploying project files and LoadFromFileName registry pointers.

5.1. Deploying Browsium Client

To get the most out of Catalyst, you'll want to install Browsium Client on every licensed PC in your enterprise. This is easy to do using Browsium-ClientSetup.exe and some careful planning.

Catalyst seat licenses are required for each PC so check your Browsium license agreement before deploying.

It is assumed that the reader of this guide is familiar with enterprise deployment of client software and has available tools to perform a scaled deployment. Section 2.6 of this guide details the command line switches available when installing Browsium Client in an enterprise environment. These options range from silent installation with no installation user interface visible to end users to pre-configuring the BCMS location in the client registry.
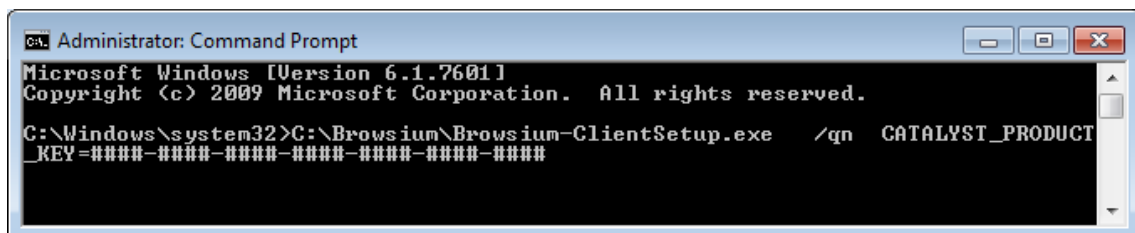
5.1.1. Installing Browsium Client

To deploy Browsium-ClientSetup.exe simply with a Catalyst license key, there are two simple steps:

1) Retrieve a copy of Browsium-ClientSetup.exe from the Browsium Catalyst zip file you received from Browsium.
2) Ensure you have the location of your Browsium Client Management Server (BCMS) to ensure Catalyst clients properly authorize for use. If Catalyst clients are unable to connect to the BCMS, they will fail to operate after 30 days.

To learn more about the Browsium Client Management Server and how Browsium Clients connect to it, please review the BCMS Administration Guide

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

More information on deploying Browsium Client to ensure the browser extensions are enabled by default can be found in section 5.2.

5.2. Managing Browsium Client Browser Extensions

If the Browsium Client was installed using the suggested default option to allow for enabling and management of the Browsium extension, this section can be ignored and no additional action is needed. This section is included for customers that opted to control settings manually.

It is important to develop a strategy to properly deploy and manage Browsium Client software on end user PCs. As part of your strategy, two important system configuration options should be considered when using Catalyst.

The first is to ensure Browsium Client browser extensions are enabled for all supported browsers on each client PC. It is recommended that you also block end users from disabling Browsium Client browser extensions once they've been enabled.

The second is to ensure that neither Internet Explorer, Edge, Chrome, nor Firefox are selected as the 'default browser' or set to prompt to become the default – Catalyst itself (actually Browsium Client Launcher) must be the default so it can route all navigation to the appropriate browser. Catalyst will take over as the default browser automatically, every time the Controller starts.

These important configuration options can be managed by Group Policy in both Internet Explorer and Google Chrome. Mozilla Firefox does not natively support Group Policy today.

Alternatively, you can manage the enforcement of the browser settings for Internet Explorer and Chrome by adding or changing registry settings manually. To modify settings manually in the local PC registry, administrators will need to use a registry editor. The default Windows registry editor which must be launched from the Run command is regedit.exe. For a large organization, registry edits can be scripted and applied using a variety of enterprise management tools.

The remainder of this section covers management of Browsium Client browser extensions.

5.2.1. Enable the Browsium Internet Explorer Extension via Group Policy

A critical element of any Catalyst deployment is ensuring the Browsium Internet Explorer Extension 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 Browsium Internet Explorer Extension during deployment.

The most common way to enable the Browsium Internet Explorer Extension 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 Internet Explorer Extension, which must be located in the registry once it is installed in your environment.

Group Policy - 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 Browsium Client add-on to the Add-on List and set the value to 1.
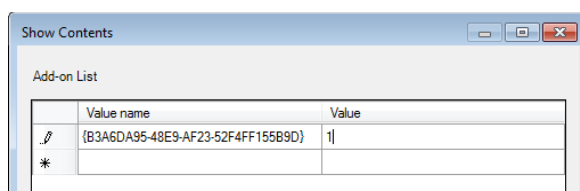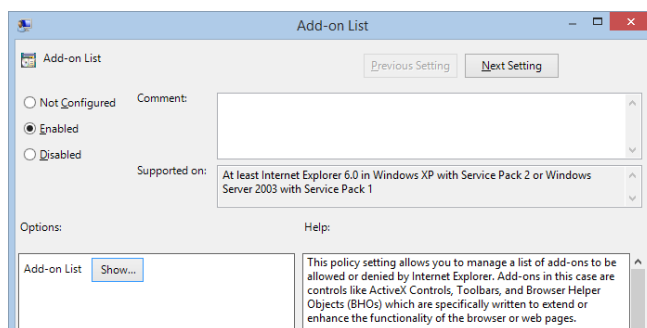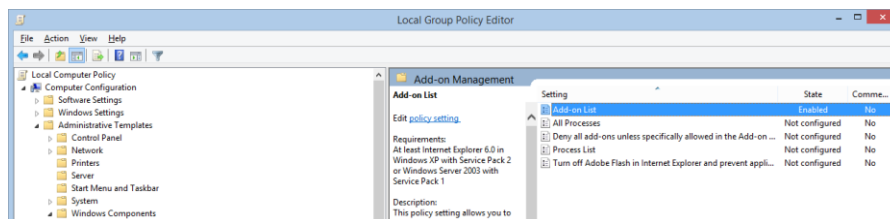
Determining the GUID/CLSID of the Browsium Internet Explorer Extension

After installing Browsium Client, use Internet Explorer Tools menu to choose Manage add-ons.
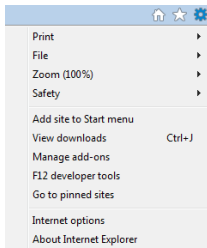
You'll then be presented with the Manage Add-ons interface where you will see Browsium Internet Explorer Extension in the list.
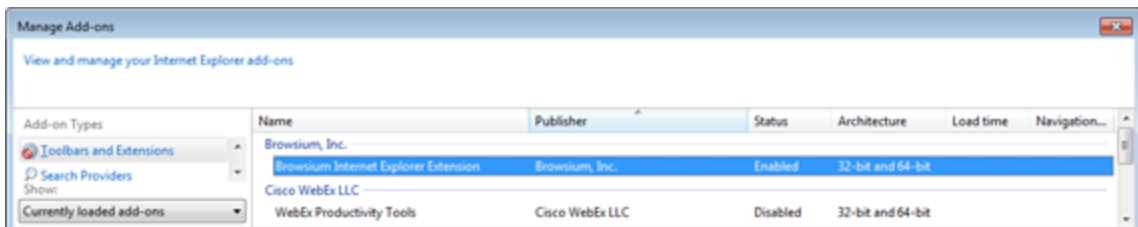
Right Click on the Browsium Internet Explorer Extension and choose "More Information" from the dropdown menu.
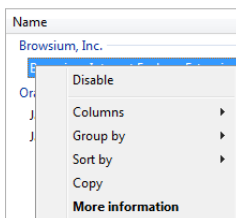
The CLSID, (Class ID) will appear in the dialog box.

More Information   ✕

Name:	Browsium Internet Explorer Extension
Publisher:	Browsium, Inc.
Type:	Browser Helper Object
Status:	Enabled
Architecture:	32-bit and 64-bit
Version:	4.5.0.119

File date:	Wednesday, August 1, 2018, 7:32 PM
Date last accessed:	Today, August 3, 2018, 2:44 PM
Class ID:	{B3A6DA95-9243-48E9-AF2E-52F4FF155B9D}
Use count:	3
Block count:	0
File:	BrowsiumIEBHOx64.dll
Folder:	C:\Program Files (x86)\Browsium\Client

Copy

Remove   Close

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 extension in the policy.

To set this policy with a manual or automated registry entry, an administrator can create a registry value based on the GUID/CLSID of the extension in either of the following keys and then set the desired value. When you enter the GUID/CLSID be sure to include the open brace at the beginning and the close brace at the end.

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

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

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

Name: *GUID* of add on which is {B3A6DA95-9243-48E9-AF2E-52F4FF155B9D}

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.

The Add-on (CLSID) lists are empty by default.
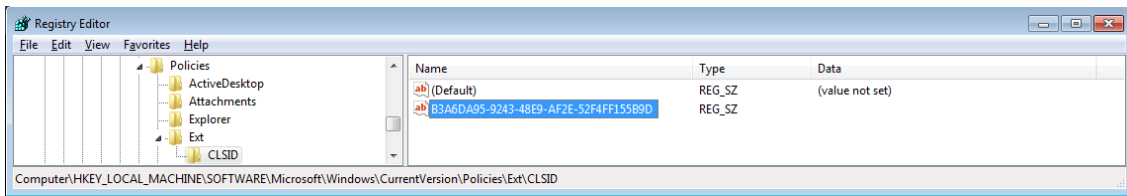
5.2.2. Disable Internet Explorer's Default Browser Check

By default, some versions of Internet Explorer will prompt the user to select it as the default browser. Since Catalyst becomes the default browser, you will want to prevent this behavior.

With Group Policy settings and local registry settings, you can remove the ability for end users to change the default browser to Internet Explorer. Depending on which Group Policy template is on the system, this policy will vary. This policy allows you to prevent Internet Explorer from checking to see whether it is the default browser and prevents the user from changing it.

Before you enable this policy, you will want to uncheck the box "Tell me if Internet Explorer is not the default browser". The check box is on the Program tab in the Internet Options dialog box. You can uncheck the box with a registry setting in the HKCU hive. The path to the registry key is below. The value for "Check Associations" should be "no".

[HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main]"Check_Associations"="no"

The Group Policy settings are listed below. All of them can be found at the following path:

Path: User Configuration\Administrative Templates\Windows Components\Internet Explorer

For IE8 & IE9:

> **Policy Name:** Prevent changing default browser check

For IE10:

> **Policy Name:** Disable changing default browser check

For IE11:

> **Policy Name:** Notify users if Internet Explorer is not the default web browser

If you enable this policy, the **Internet Explorer Should Check to See Whether It Is the Default Browser.** Also, the check box on the **Programs** tab in the **Internet Options** dialog box appears dimmed and the user cannot change the default browser to IE. If you disable this policy or do not configure it, users can determine whether Internet Explorer will check to see if it is the

default browser. When Internet Explorer performs this check, it prompts the user to specify which browser to use as the default.

5.2.3. Enable Browsium Client Extension for Microsoft Edge

Microsoft Edge requires systems to be domain joined and logged in with a domain user account to use the Browsium Client extension. In addition, application sideloading needs to be enabled.

To do this, you will need to configure the following Group Policy setting:

> Local Computer Policy\Computer Configuration\Administrative Templates\Windows Components\App Package Deployment\Allow all trusted apps to install

This setting must be set to 'Enabled' or the Browsium Client extension will not be installed and enabled for Microsoft Edge.

5.2.4. Enable Browsium Client Extension for Google Chrome

To ease your Group Policy setup, several templates can guide you through the configurable options. Group Policy templates, and associated guidance, are provided by Google and can be found on Google's support site. You may find additional settings (beyond those documented here) that you may wish to enforce or enable based upon your organization's preferences.

By default, Chrome automatically disables all extensions that are side-loaded (installed by a 3rd party program, like Browsium Client installation package), requiring users to enable them manually. The only way to centrally enable Browsium Client Extension for Chrome for enterprise deployment is via Group Policy for domain-joined systems.

The policy **Configure the list of force-installed extensions** (a.k.a. ExtensionInstallForcelist) allows you to specify a list of extensions that will be installed silently and enabled by default, without user interaction. This policy also works for side-loaded extensions, effectively overriding the default behavior in Chrome.

Each item of the list is a string that contains an extension ID and an update URL, separated by a semicolon (;). The extension ID is the 32-letter string found e.g. on chrome://extensions when in 'Developer mode'. The update URL must point to an Update Manifest XML document as described at http://code.google.com/chrome/extensions/autoupdate.html. Note that the update URL set in this policy is only used for the initial installation; subsequent updates of the extension will use the update URL indicated in the extension's manifest.

For each item, Google Chrome will retrieve the extension specified by the extension ID from the update service at the specified update URL and silently install it. Users will be unable to uninstall extensions that are specified by this policy. If you remove an extension from this list, it will be automatically uninstalled by Google Chrome. Extensions specified in this list are also automatically whitelisted for installation; the **Configure extension installation blacklist** (a.k.a. ExtensionInstallBlackList) does not affect them.

A by-product of the ExtensionInstallForceList policy is that managed extensions are silently installed in Chrome, enabled by default, and block users from disabling or removing them. This is desired for enterprise deployment of Browsium Client. If this policy is 'Not Configured', users can delete any extension in Chrome, including Browsium Client Extension, from the Extensions page. This is undesirable, as side-loaded extensions that are deleted are automatically blacklisted and re-enabling them is tricky (but achievable). Contact _Browsium Support_ if this happens.

To force-enable Browsium Client Extension for Chrome, and lock it down so users can't disable or delete it, you will use the **Configure the list of force-installed extensions** policy. This process requires an XML Manifest, which references the Proton extension's .crx file. Both must be available on a server or in the Chrome web store. Browsium is hosting these files for all customers on browsium.com.
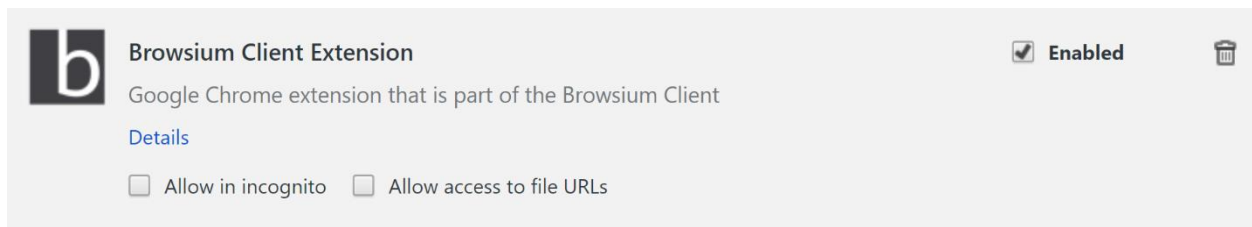
Follow these steps to ensure that this method is properly configured using Group Policy for your domain-joined systems. These instructions assume you're using the ADM template. The Group Policy location will change if using ADMX.

As of Chrome 33, the ExtensionInstallForceList policy is only enforced for domain-joined systems. All client PCs in your environment must be joined to a Windows domain or you will not be able to centrally manage Browsium Client Extension. Attempting to configure ExtensionInstallForceList via the Local Policy Editor will result in unpredictable behavior of the extension.

1. Install Browsium Client software.
2. Download the Group Policy templates for Chrome from the Google support site.
3. Import the Google Chrome Group Policy template into your Group Policy editor.
4. Enable the policy **Configure the list of force-installed extensions**.
5. Enter the following value by selecting the 'Show...' button in the Options window and apply the setting.

   (This is the Browsium Client Extension ID followed by the URL for the manifest XML document, with no spaces in the string.)

   pdphfkkbpakmngokamdgaomjnffoipca;http://www.browsium.com/crx/browsium-4.6/browsium-chrome-4.6.xml

Browsium Client Extension for Chrome may have a different version number from the other Browsium Client and Catalyst components as maintenance versions are released. See the version number section of the *Catalyst Version History KB article* for details on the release date and version number for the Browsium Client Extension for Chrome.

5.2.5. Disable Chrome's Default Browser Check

Group policy can be used to configure the default browser checks in Google Chrome and prevent users from changing them. If you 'Enable' this setting, Chrome will always check on startup whether it is the default browser and automatically register itself if possible. If this setting is 'Disabled', Chrome will never check if it is the default browser and will disable user controls for setting this option (the desired state when using Catalyst). If this setting is 'Not Configured', Chrome will allow the user to control whether it is the default browser and whether user notifications should be shown when it isn't.

For all users running Catalyst, the **Set Chrome as Default Browser** setting (a.k.a. DefaultBrowserSettingEnabled) should be "Disabled" in your Group Policy editor. The path for this setting in the Local Group Policy Editor is Local Computer Policy\Administrative Templates\Classic Administrative Templates (ADM)\Google\Google Chrome.
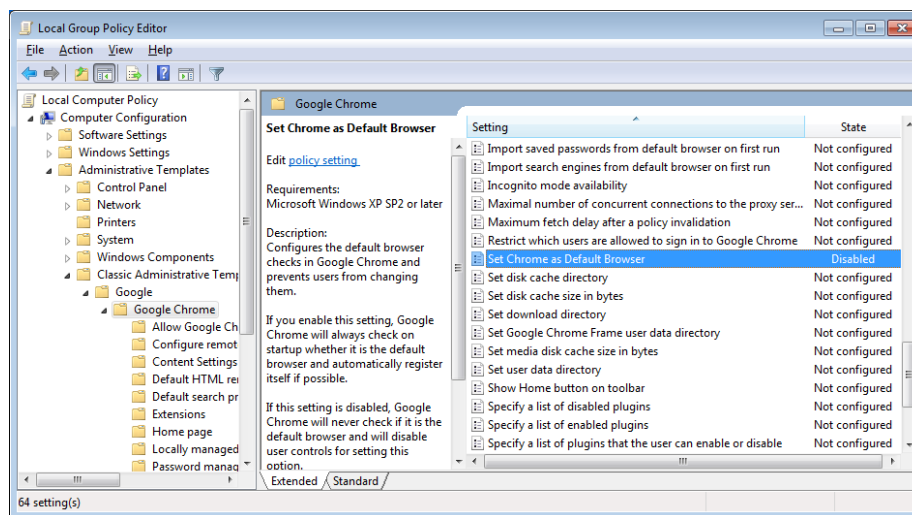
5.2.6. Enable the Browsium Client Extension for Mozilla Firefox

Mozilla Firefox does not allow client software to install an extension and enable it by default. Nor does it contain a centralized management facility to keep users from tampering with extensions like Internet Explorer and Chrome. However, Browsium Client includes a facility to automatically enable the extension every time the user logs onto the system, so administrators can ensure that the Firefox extension is always enabled.

To enable Browsium Client Extension for Firefox automatically, create the registry value:

... for **32-bit** Windows systems:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run

Enable Browsium Extension (REG_SZ) = "C:\Program files\Browsium\Client\BrowsiumController.exe" /ef

... for **64-bit** Windows systems:

HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Windows\CurrentVersion\Run

Enable Browsium Extension (REG_SZ) = "C:\Program Files (x86)\Browsium\Client\BrowsiumController.exe" /ef

5.2.7. Disable Firefox's Default Browser Check

This policy configures the default browser checks in Mozilla Firefox and prevents users from changing them. If you enable this setting, Firefox will not check on startup whether it is the default browser and also will not allow the user to change this setting.

For all users on a PC, the **Disable Firefox Default Browser Check** setting should be "enabled" in your Group Policy editor. The path for this setting is Local Computer Policy\Administrative Templates\Classic Administrative Templates (ADM)\Mozilla Firefox.

This setting will make the following changes to the PC's registry once the policy is propagated:

Data type: REG_DWORD

Windows registry location:
HKEY_LOCAL_MACHINE\Software\Policies\Firefox\FirefoxCheckDefault

Example value: 0x00000001

The value in this case should set be "0"

Data type: REG_SZ

Windows registry location:
HKEY_LOCAL_MACHINE\Software\Policies\Firefox\FirefoxCheckDefaultType

The value in this case should be "Locked"

5.3. Understanding Catalyst Project Registry 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. Catalyst 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 Catalyst Configuration Manager.

This option saves the project file to disk, places a pointer to the file in the registry, and starts Browsium Controller to load the new configuration. The pointer is written to the LoadFromFileName registry value in HKEY_LOCAL_MACHINE\SOFTWARE\(Wow6432Node)\Browsium\Catalyst.

Start Test Configuration should only be used for project development and testing as it requires the Catalyst Configuration Manager which should never be made available to end users. In addition, the HKEY_LOCAL_MACHINE\SOFTWARE\(Wow6432Node)\Browsium\Catalyst registry location should only be used for testing as it is easily overwritten by Catalyst Configuration Manager and may not persist during a Browsium Client upgrade.

The last concept that must be understood before embarking on a Catalyst deployment is the precedence hierarchy for the evaluation of configurations when multiple LoadFromFileName values are found on a system. Catalyst 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, Catalyst will stop searching and that configuration will be used.

Deploying different Catalyst 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 Catalyst 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 chart.

HKEY_LOCAL_MACHINE\SOFTWARE\(Wow6432Node)\Browsium\Catalyst\LoadFromFileName
Catalyst project file pointer for configuration testing only, deployed via Start Test Configuration.
Do not use for enterprise deployment.

HKEY_LOCAL_MACHINE\SOFTWARE\(Wow6432Node)\Policies\Browsium\Catalyst\LoadFromFileName
Catalyst project file pointer for all user accounts, deployed via registry editor or Group Policy.

HKEY_CURRENT_USER\SOFTWARE\Policies\Browsium\Catalyst\LoadFromFileName
Catalyst project file pointer for a specific user account, deployed via registry editor or Group Policy.
This registry location may be deprecated in a future release.

Catalyst 2.0 and earlier used a registry key called RulesFilePath. That key has been deprecated in Catalyst 3.0 and later. Only LoadFromFileName values will be used to determine the active Catalyst configuration. Contact Browsium Support if you need assistance upgrading from an older version of Catalyst.

5.3.1. Working with Test Configurations in Group Policy Environments

Understanding configuration loading hierarchy is critical to ensuring configurations load as expected during project development. Testing on systems that receive a LoadFromFileName value via Group Policy may not behave as expected.

Catalyst reads LoadFromFileName from the Test Configuration location prior to Group Policy locations to ensure that the project you are testing is used by Browsium Controller. However, as soon as you Clear Test Configuration, the configuration deployed via Group Policy will become the active project. This will make it difficult to compare your new Test Configuration with a native, unmanaged environment. To achieve a clean Catalyst test system, you must remove your system from the domain, exclude it from Catalyst project file Group Policy settings, or temporarily delete the LoadFromFileName value delivered by Group Policy. (It will return on the next Group Policy refresh.)

Browsium recommends testing Catalyst on systems that are not controlled by Catalyst project file Group Policy settings to avoid configuration conflicts.

5.3.2. Multi-user Testing Configurations

As mentioned previously, Catalyst Configuration Manager should be installed only for administrators and project developers – not for end users. However, this creates a challenge when working on a multi-user server system, such as a Citrix XenDesktop or Microsoft Terminal Server environment.

Start Test Configuration writes to the HKEY_LOCAL_MACHINE registry hive. This means the Catalyst configuration will be deployed for all users on a multi-user system if any user with administrator privileges uses Start Test Configuration from Catalyst Configuration Manager. Therefore, Browsium recommends that project development and testing be performed on a single-user system only.

Start Test Configuration in Catalyst Configuration Manager always writes to the shared HKEY_LOCAL_MACHINE registry hive on a multi-user system.

5.4. Deploying Catalyst Configurations

Since Catalyst 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.

Catalyst'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, Catalyst 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 location do not change. The following steps provide guidance to specify where Catalyst should look for the project file.

There are two options for deploying the Catalyst 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. Browsium 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.

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

In this section you will learn how to instruct Catalyst 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 Catalyst project file (.cax) must be stored in a user-readable location on the network share or web server. For network share locations, Browsium recommends client systems have only read access permissions to ensure the file is not accidently removed or modified. For both network share and web server locations, organizations must ensure client systems have both security credentials for, and network access to, the resource.

Security credentials must be established for the **end user**, not the **client system** as required for Ion, as the Catalyst project file is accessed directly by the Browsium Controller, which runs with user privileges.

Remote/mobile users will need to have VPN access if the file or web server hosting the Ion configuration file is not publicly 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 Browsium Controller starts, and no configuration is found in Catalyst's cache from a previous connection, the Controller will not start (provided neither Proton nor Ion are configured on the client system). As long as the user successfully connects to the server once during Controller startup, the 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-user** settings on **32-bit** or **64-bit** Windows systems, find or create:

> HKEY_CURRENT_USER\Software\Policies\Browsium\Catalyst

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

> HKEY_LOCAL_MACHINE\Software\Policies\Browsium\Catalyst

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

> HKEY_LOCAL_MACHINE\Software\Wow6432Node\Policies\Browsium\Catalyst

Then create or populate the following String Value in the key:
LoadFromFileName (REG_SZ) = \\Server\Share\Catalyst\... [the path to your Catalyst project file]

This setting will direct the Catalyst software to the project file the next time Browsium 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 Catalyst project file "Catalyst test.cax" 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\Catalyst]
> "LoadFromFileName"="\\\\server\\share\\Catalyst test.cax"
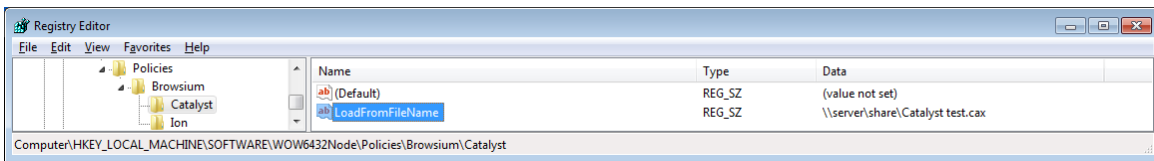
5.4.2. Project File Deployment Option B: Client File System Location

In this section you will learn how to instruct Catalyst 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 Catalyst project file (.cax) 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 Browsium Controller. If Catalyst is unable to find the project file in the defined location, and none exists in the cache from a previous configuration, Browsium Controller will not start the Catalyst service. For more information on Catalyst file caching behaviors see the Catalyst 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-user** settings on **32-bit** or **64-bit** Windows systems, find or create:

    HKEY_CURRENT_USER\Software\Policies\Browsium\Catalyst\

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

    HKEY_LOCAL_MACHINE\Software\Policies\Browsium\Catalyst\

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

    HKEY_LOCAL_MACHINE\Software\Wow6432Node\Policies\Browsium\Catalyst\

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

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

This setting will direct the Catalyst service to the configuration file the next time Browsium 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 "Catalyst test.cax" in the C:\Browsium 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\Catalyst]
"LoadFromFileName"="C:\\Browsium\\Catalyst test.cax"
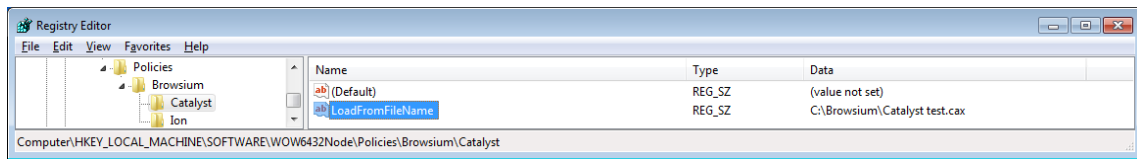
5.5. Readying Windows 8 and Windows 10 for Catalyst

For Catalyst to work properly in all scenarios, it must be the default browser in Windows. In earlier versions of Windows, Catalyst was able to set itself the default browser. Microsoft has made significant changes with Windows 8 and later in this area which impacts all browsers, forcing users to take specific action or IT to manage default browser settings centrally.

Prior to Windows 8, Catalyst automatically configured itself as the default browser, or more specifically, the default program for the HTTP and HTTPS protocols. This enabled Catalyst to direct traffic to the appropriate browser for desktop shortcuts and links to websites in applications, such as email programs.

With Windows 8 and Windows 10, Microsoft has blocked applications from programmatically configuring the default browser. Only end users (or IT administrators via Group Policy) can set a default browser. This is even true of Internet Explorer, which can only invoke the Set Default Programs control panel to when selecting "Make Internet Explorer the default browser" from Internet Options.
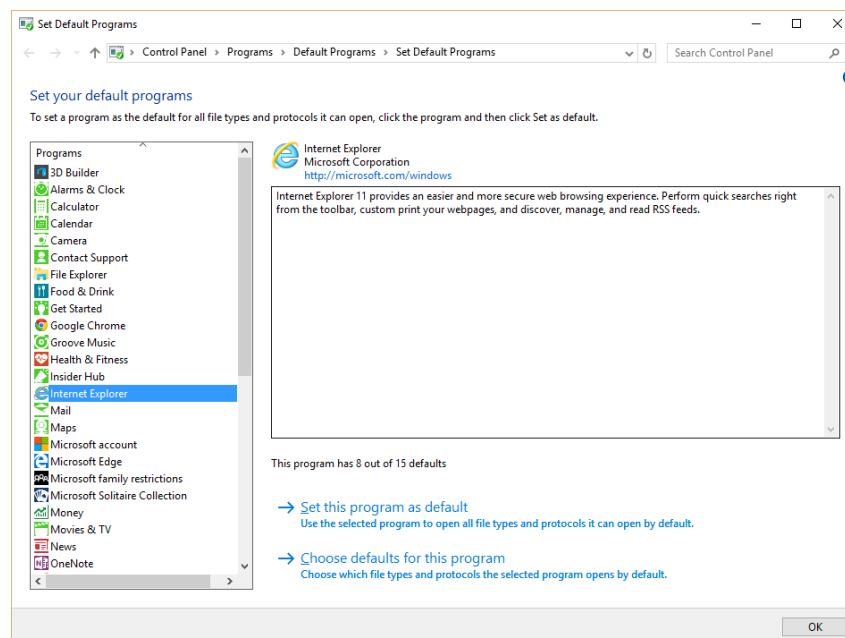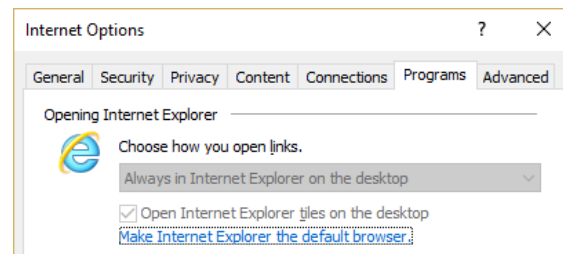
Because of these new restrictions in Windows 8 and later, default browser settings must be manually configured for Catalyst. The following sections will guide you through this process for two scenarios:

- **Catalyst Project Development**, which includes evaluation and testing of Catalyst
- **Catalyst Enterprise Deployment**, to set the default browser in a managed enterprise

5.5.1. Set Catalyst as Default Browser for Project Development

When developing Catalyst projects (or configurations), or simply evaluating Catalyst for future purchase, it's not practical to configure Group Policy to set Catalyst as the default browser on Windows 8 and later. Instead, it's much easier to set the default browser using the Default Programs control panel in Windows.

After creating a project in Catalyst Configuration Manager and executing Start Test Configuration for testing, Catalyst will alert you that it wants to become the default browser in Windows – assuming the Project Setting "Override Default Browser" is checked (which is the default setting in all projects). This works in much the same way as Internet Explorer, described in the prior section.
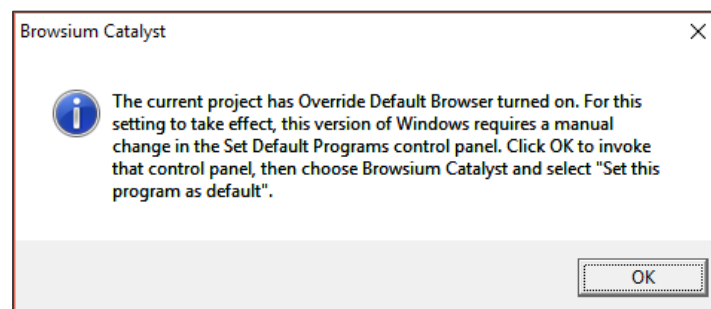
Clicking OK in this dialog will automatically launch the Set Default Programs control panel. It will also start Browsium Controller so you will see the Catalyst Splash Screen while the control panel is loading.
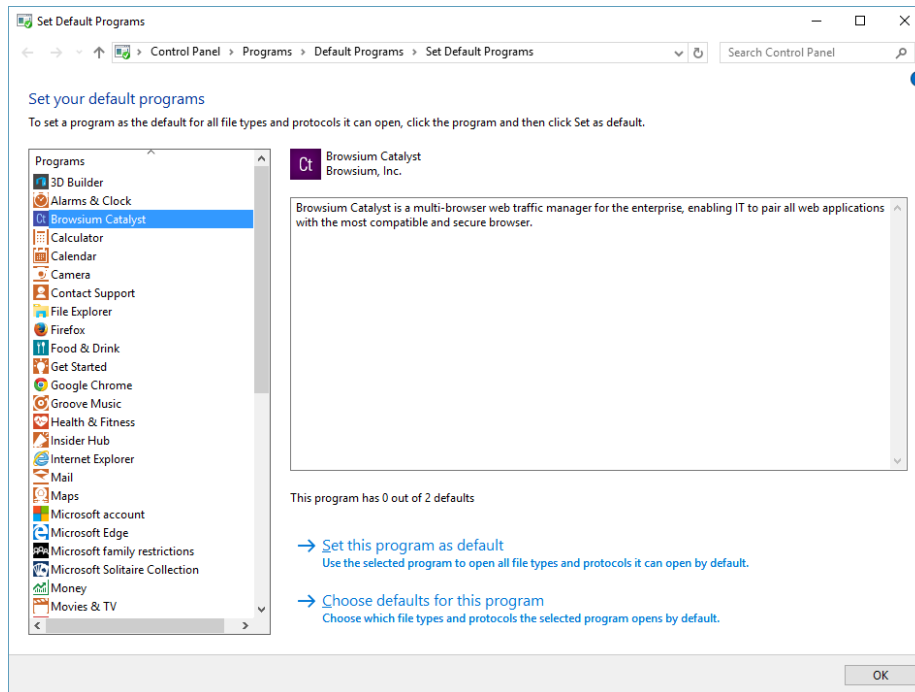
You'll notice that Catalyst has "0 out of 2 defaults". You want it to have both defaults – which correspond to HTTP and HTTPS. Select the option 'Set this program as default'. Alternatively, to see the individual selections, select 'Choose defaults for this program' and then choose Select All to set Catalyst as the default for both HTTP and HTPPS and then click 'Save'.
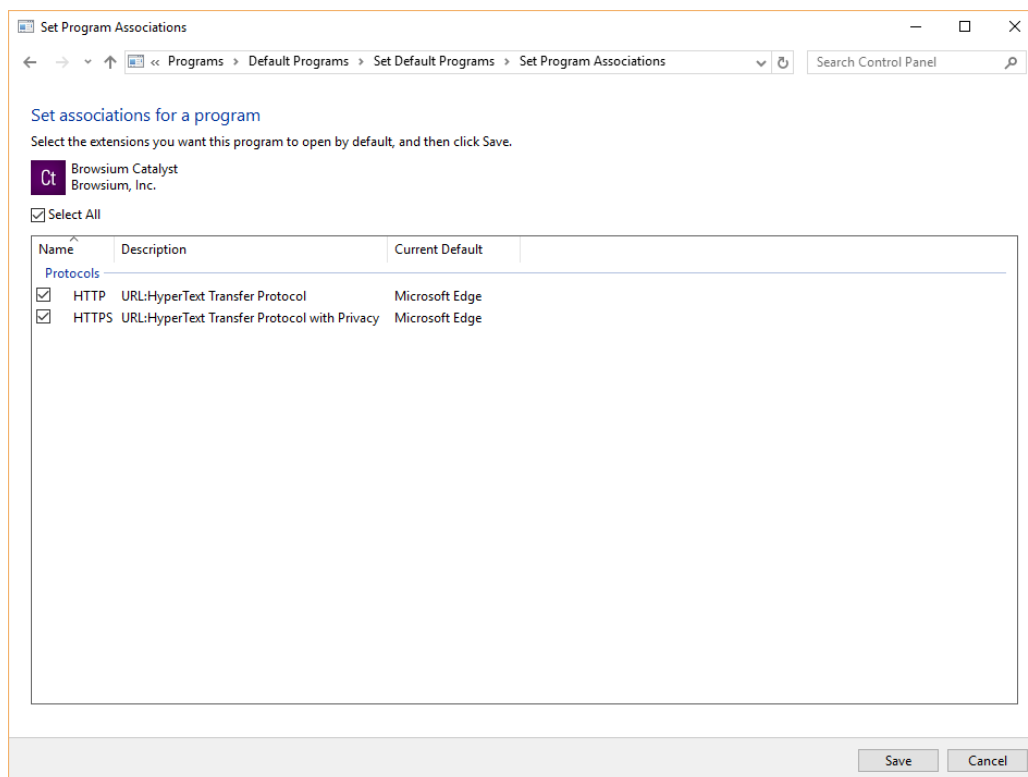
Catalyst will hold these defaults until you change the settings by selecting another browser as the default if prompted. If you plan to continue to use Catalyst, you should not change the default browser, per the guidance earlier in section 5.2. Catalyst has a built-in fail-safe if the Controller is stopped, either manually or by the Clear Test Configuration option in Catalyst Configuration Manager, where it will restore Internet Explorer as the default program for HTTP, and HTTPS, even though the Default Programs control panel still shows Catalyst owning the setting for these protocols.

5.5.2. Set Catalyst as Default Browser for Enterprise Deployment

When deploying Catalyst across an enterprise running Windows 8 and later, you will want to use enterprise-class management tools to set Catalyst as the default browser.

Prior to Windows 8, applications could set the default handler for a file type/protocol by manipulating the registry. This means IT could easily have a script or a Group Policy manipulating the registry. For example, for the mailto: protocol, you just needed to change the "default" value under HKEY_CLASSES_ROOT\mailto\shell\open\command

With Windows 8 and later, this method is no longer available. But Microsoft has introduced a new Group Policy mechanism for declaring these defaults in Windows 8 and later to accommodate this type of scenario. The basic idea is to have an XML file that maps programs to the file type/protocol that they should be the default for. The following steps provide guidance for configuring and deploying Catalyst as the default browser across an enterprise of Windows 8 or Windows 10 systems.

This guidance is not required for Windows 7 or Windows XP as Catalyst is able to programmatically take over the default browser setting on those systems.

1. Create your XML file or export it from a system which has Catalyst as the default browser using DISM, per Microsoft's guidance. Among the many entries in your XML file, the following association identifiers must point to Catalyst.

    ```xml
    <?xml version="1.0" encoding="UTF-8"?>
    <DefaultAssociations>
    <Association Identifier="http" ProgId="CatalystHTML" ApplicationName="Catalyst" />
    <Association Identifier="https" ProgId="CatalystHTML" ApplicationName="Catalyst" />
    </DefaultAssociations>
    ```

2. Use the new Windows 8 and later Group Policy that enables you to set the association for file types and protocols. Enable the policy "Set a default associations configuration file" found at "Computer configuration\administrative templates\Windows Components\File Explorer". This will set the following registry entry:

    `<HKLM\Software\Policies\Microsoft\Windows\System!DefaultAssociationsConfiguration>`

    This policy specifies the path for the XML file that can be either stored locally or on a network location.
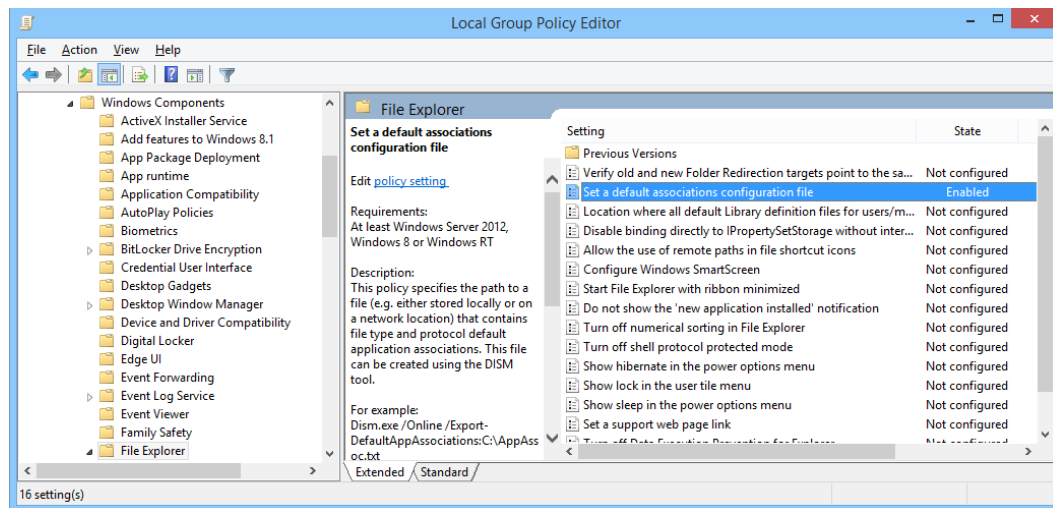
Using DISM to import the XML is not enough; you must still link it to the Group Policy Object.

In addition, the system needs to be domain-joined and the associations are applied at logon time.
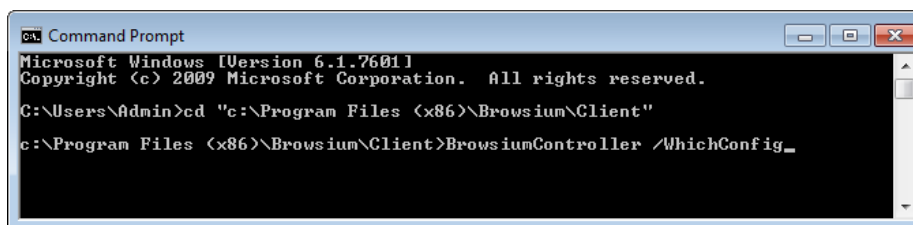
5.6. Verifying Deployment Settings

Browsium Controller provides a command-line function to query the system and identify the location of the Proton, Ion, and Catalyst configurations to be used the next time the Controller starts. The command `BrowsiumController /WhichConfig` is run from a Command Prompt.

In the following example, a system is setup with a Catalyst project file deployed to HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Browsium\Catalyst with the value "C:\Browsium\Catalyst project.cax" stored in LocalFromFileName. (This example also contains an Ion project file and a Proton Server URL.)

The query uses registry reflection so the result ignores the Wow6432Node key on 64-bit systems, although it's in the path of the registry value.

Executing this command ...

... results in the configuration acknowledged as:
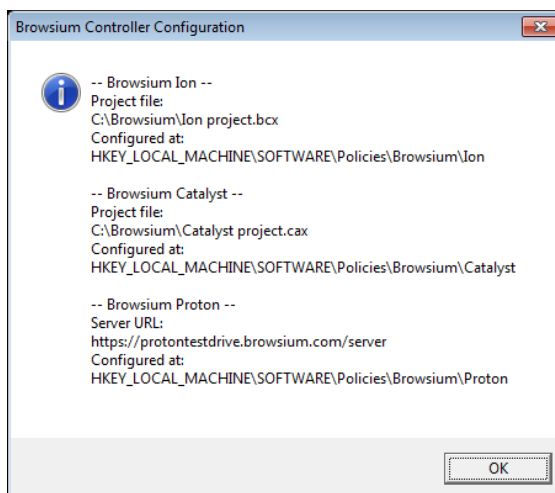
'WhichConfig' shows the Ion, Catalyst, and Proton configurations TO BE LOADED at the next Controller restart. The values shown may be different from the currently loaded configurations, if the values in the registry keys have been changed since the Controller was last started.

In addition to the command line option, administrators running Catalyst Configuration Manager can quickly check which configuration will load when the Controller is restarted by viewing About Box. Select 'About Browsium Catalyst Configuration Manager' from the Help menu. When the Catalyst About Box loads, press the Ctrl key to reveal the project file information. In addition to the LoadFromFileName registry location and project file path and name, you will see the installed .NET CLR version – another way to ensure the system is ready and working properly for testing.

In the following example, the project is read from the registry location written by Start Test Configuration, which takes precedence over all other registry locations as detailed in section 5.3.

5.7. Understanding Catalyst Project File Caching Process

Browsium Controller reads the Catalyst project file when the Controller is started, then continues operating using that configuration until the Controller restarts and looks for a new project file, or until the configuration changes if non-default Project Update Activation settings are used. See section 5.6.2 for more details on Project Update Activation.

5.7.1. Catalyst Project File Caching Behavior

Browsium Controller attempts to load the configuration from the Catalyst 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 a web server. 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 (HKCU deployment) or ProgramData (HKLM deployment) directory.

If multiple users share the same PC with unique user accounts, or when using a multi-user terminal server, each user will cache a copy of the project file in his/her AppData directory. Those files will be identical if the LoadFromFileName registry value is in HKEY_LOCAL_MACHINE (one entry for all users), or may be different if LoadFromFileName is in HKEY_CURRENT_USER (unique entry per user).

If the original Catalyst project file is not available (the network location is temporarily unreachable or the file has been moved, renamed, or deleted), Browsium Controller 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 Catalyst extremely robust and able to remain fully functional regardless of what happens to the project file after it has been loaded and Browsium Client processes are running.

5.7.2. How Catalyst Handles Project File Polling

Some customers deploy Catalyst in a staged rollout to their enterprise user base, which requires updating project files as redirection rules are required. To ensure client systems obtain the updated project file, Browsium Controller includes a Catalyst configuration polling mechanism to check for modified versions of the project file. When Browsium Controller identifies a different Catalyst project 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 ease deployment in large enterprises, Catalyst enables new configurations to be loaded on the fly without restarting the Browsium Controller (which typically requires a reboot or logoff/logon of the client computer). This functionality is governed by Project Update Activation in Project Settings. The default is set to load the new configuration immediately.
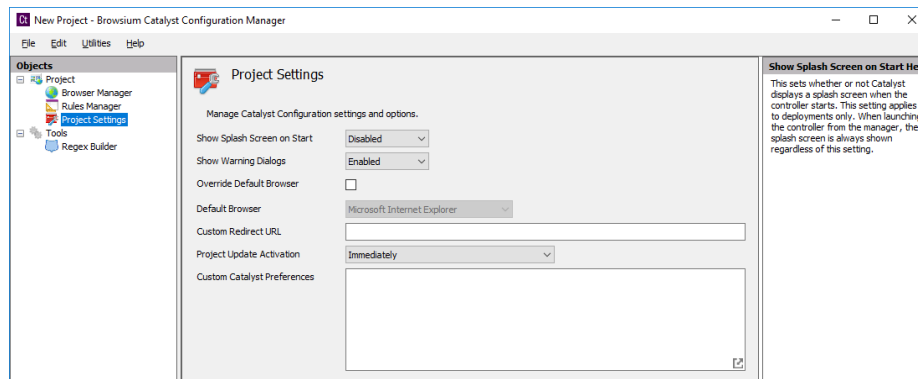
To achieve enterprise-level scale, the polling mechanism uses a random interval to discover configuration updates, as noted at the end of this section. It can therefore take up to an hour for Project Update Activation to change the in-memory configuration that Catalyst is actively using even when set to 'Immediately'.

This setting can be changed to activate the new project at a scheduled time or upon next Browsium Controller restart.
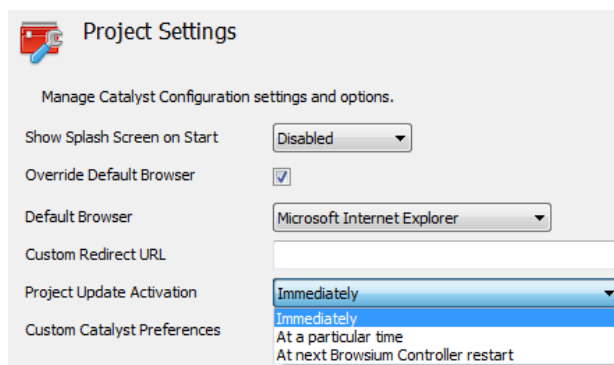
When "At a particular time" is selected, a Project Update Time field is added. This field defaults to 2am but can be changed to any time value. It uses the client's system clock so that 2am is 2am for every user, regardless of time zone.

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 issues.

Browsium Controller 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.

## 5.8. Managing Browsium Client Logging Settings

In a default installation, Browsium Client logs Warning level information which will provide basic information about the Catalyst configuration and any important errors that may occur on client systems, including errors from Proton and Catalyst if configured on the client.

Browsium Client can be configured to record more detailed logging information to troubleshoot problems or validate Catalyst configuration settings on the local system. Browsium Client logging is written to the standard Windows event log under an application-level source named Browsium. 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 value must be configured on the target system. Once the logging value is created, the Browsium Controller must be restarted to recognize the specified logging value.

Browsium Controller looks in the Windows registry for the presence of the logging setting at HKEY_CURRENT_USER\Software\Browsium\Client. The value is a REG_DWORD type, with value name "LogLevel". The data in that value can be 1, 2 or 4 (see chart above).
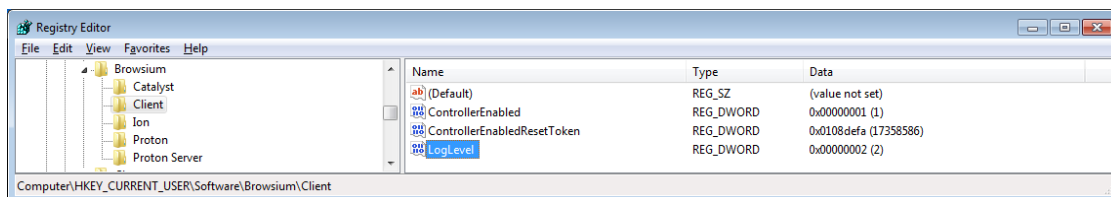
If no LogLevel registry value is present in the defined registry location (the default Browsium Client state), or any invalid values are found at that location, Browsium Client will revert to using the default Level 2 – Warning.

A. Troubleshooting

A.1. Catalyst rule fails to engage

You may encounter a scenario in which Catalyst fails to engage on one or more websites based on rules you've 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 Catalyst.

- **Confirm the Browsium Client Executable Files are Running**
    - Check to see that the Browsium Controller (`BrowsiumController.exe`) is running on the target machine.

- **Ensure the Browsium Client Extensions are Enabled and Running**
    - Confirm the Browsium Catalyst Client extensions are seen and loaded by Internet Explorer, Chrome and Firefox
        - Open each browser and open the Manage Add-Ons/Extensions dialog. Do you see the Browsium Internet Explorer Extension installed? Is it enabled?
        - For example, An Internet Explorer instance that correctly loads the Browsium Client extension will display the following information in the Manage Add-Ons dialog:
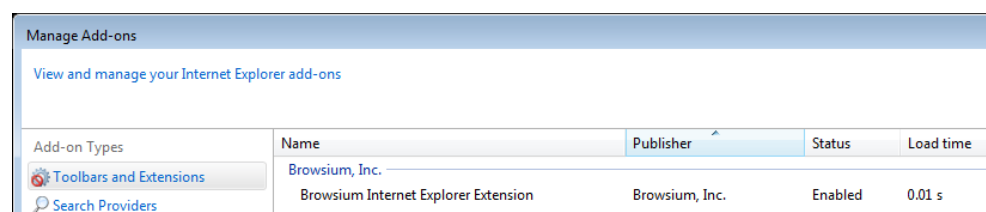
- **Visit the Knowledge Base or Contact Support**
    - If all these steps fail, consider searching the Catalyst Knowledge Base.
    - If you have a support contract, contact your systems integrator, or Browsium Support for one-on-one guidance.

A.2. Browser window doesn't get focus automatically

Some users may experience 'focus' issues where a web page or web application loads and the user is unable to interact with the browser window automatically. This issue is related to how Windows provides focus control (the ability to receive input). Users will need to click inside the browser tab window to ensure proper focus.

A.3. Catalyst did not automatically become the default browser

Microsoft made a change in Windows 8 and later, and no longer allows 3rd party software to programmatically take over as the default browser. As Catalyst must be the default browser to manage navigation through all entry points for HTTP and HTTPS, it must be set as the default by end users or via Group Policy. See section 5.5 of this guide for details on how to control browser defaults on Windows 8 and later.

A.4. Browsium Controller becomes unresponsive

Find the Controller process (`BrowsiumController.exe`) using Task Manager and kill the process. It will automatically restart. If running the Catalyst Configuration Manager, the Controller can be restarted using the Utilities menu.