



Adding External Launch Points to the GUI Client

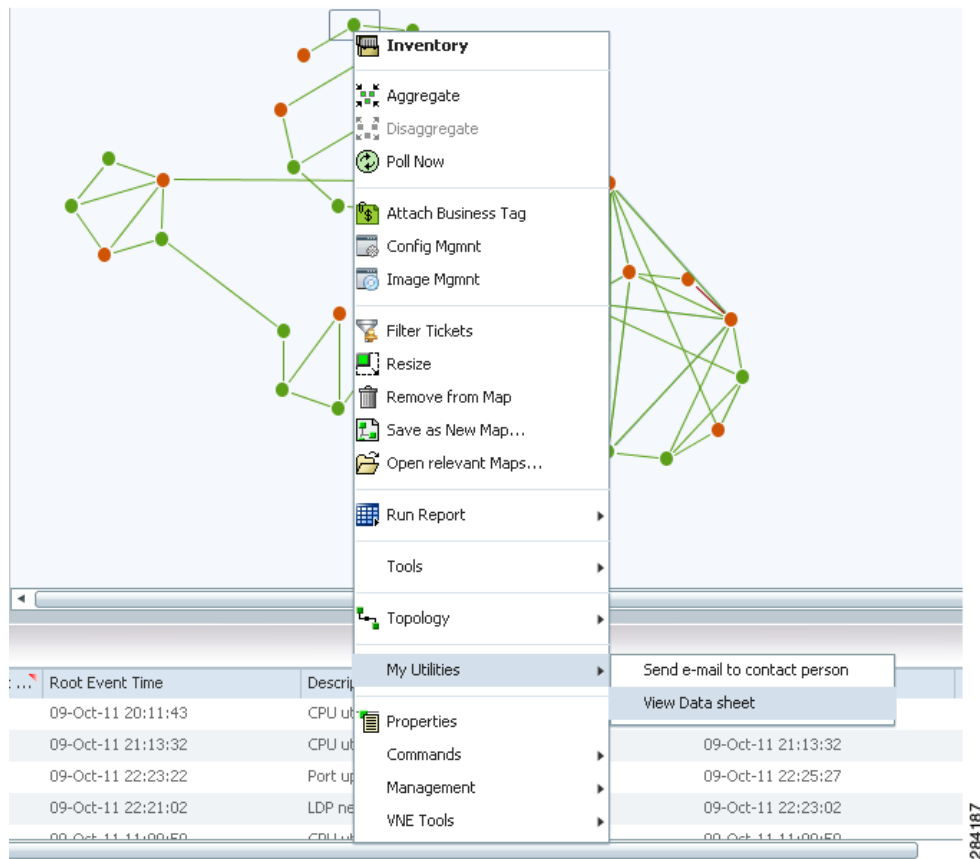
These topics describe how to add launch points from Prime Network to external applications:

- [Overview: Adding External Launch Points, page 8-1](#)
- [Adding an External Launch Point to the GUI Client, page 8-3](#)
- [Deleting an External Launch Point, page 8-10](#)
- [Sample BQL Scripts for Launching External Applications, page 8-11](#)

Overview: Adding External Launch Points

If you have configurator or higher privileges, Prime Network allows you to add right-click menu options to elements in order to launch external applications or URLs. These are called external launch points. When the external launch point is clicked, it invokes the external application (for example, a script or batch file), or opens the specified URL in the default browser. [Figure 8-1](#) illustrates a new launch point named My Utilities.

Figure 8-1 External Launch Point Example



You can add launch points to any IMO, including network elements, links, tickets, and events. You identify the IMO according to IMO type. The launch points appear as additional right-click menu options on all IMOs of the specified type.

You can narrow down the instances on which the external launch point appears by filtering by any of the properties of the IMO. You can also create a launch point on a property of a referenced IMO.

When you add a launch point, the site.xml file is updated on the Prime Network gateway. The new launch point (right-click menu option) will appear in all Prime Network clients that connect to the gateway.

The scripts or batch files used in the external launch point definition can be copied to one of the following locations:

- The Prime Network client machine—In this case, the launch point will only be functional on that client.
- The Prime Network gateway, under the Main/webstart/scripts directory—In this case, the scripts will be downloaded automatically to all clients associated with the gateway, upon next login. The launch points will be functional from all these clients. Please follow the instructions in [Enabling Automatic Download of BQL Scripts to Clients, page 8-10](#), to enable the automatic download of the scripts to the clients.
- A shared location.

Adding an External Launch Point to the GUI Client

Adding an external launch point involves the following steps:

1. [Identifying the IMO Context and Properties, page 8-3](#)
2. [Creating BQL Scripts, page 8-4](#)
3. [Running BQL Scripts, page 8-10](#)
4. [Enabling Automatic Download of BQL Scripts to Clients, page 8-10](#)



Note

You should maintain a list of all the IMO contexts where launch points have been added, as there is no way to get such a list from the system. This will enable you to update or delete external launch points at a later stage.

Identifying the IMO Context and Properties

To identify the IMO context and properties:

-
- Step 1** Select a network element from the Tree pane.
 - Step 2** Press **F2**. The All Properties viewer appears in a separate window.
 - Step 3** Make a note of the IMO type. If you are in the Map pane, you should make a note of the IMO type that is part of ContainedIMO and not part of IHierarchyNode; for example, IPortConnector or IManagedElement.

Example 1: Identifying IManagedElement in the Tree pane:

```
com.sheer.imo.IManagedElement
{ [ManagedElement (Key=ana7609-1) ] }
```

Example 2: Identifying IPortConnector in the Tree pane:

```
com.sheer.imo.IPortConnector
{ [ManagedElement (Key=ana7609-1) ] [PhysicalRoot] [Chassis] [Slot (SlotNum=5) ] [Module] [Port (Port
Number=GigabitEthernet5/1 - Missing Pluggable Port) ] }
```

Example 3: Identifying IManagedElement in the Map pane:

```
{ [HierarchyNode (Id=1001) ] [ContainedImo] }
  key=Imo, value=com.sheer.imo.IManagedElement
    { [ManagedElement (Key=ana7609-1) ] }
```

- Step 4** Make a note of the property key name; for example, in IManagedElement write down IP; in IPortConnector write down PortAlias.

Example 1: IManagedElement

```
key=DeviceName, value=ana7609-1
key=IP, value=172.20.68.72
```

Example 2: IPortConnector

```
key=Location, value=5.GigabitEthernet5/1 - Missing Pluggable Port
key=PortAlias, value=GigabitEthernet5/1 - Missing Pluggable Port
```

Creating BQL Scripts

Read the *Cisco Prime Network Integration Developer Guide* for an understanding of Broadband Query Language (BQL), and as a prerequisite to understanding this section.

Table 8-1 describes the format and syntax that you must follow when writing the BQL `set` command to launch an external application:

Table 8-1 BQL Command Format and Syntax

Property / Type	Value	Description
management.IExternalLaunch		
ID / OID	<p>Examples:</p> <p>For network elements:</p> <pre>{ [ExternalLaunch (ContextImoType=com.sheer.imo.ManagedElement) (Name=example1)] }</pre> <p>For tickets:</p> <pre>{ [ExternalLaunch (ContextImoType=com.sheer.imo.newalarm.ITicket) (Name=example36)] }</pre>	<p>The OID contains the context and unique name of the external launch point.</p> <p>It defines the IMO type on which the external launch point is defined, for example:</p> <ul style="list-style-type: none"> • IManagedElement—Network elements. • IEvent (or any other subtype, like ISyslogAlarm)—Events and their subtypes. • ITicket—Tickets. • ITopologicalLink—Links.

Table 8-1 BQL Command Format and Syntax (continued)

Property / Type	Value	Description
FilterParameters / String	<pre> Example: <Filter type="management.IExternalLaunchFilter" > <FilterParameters type="IMObjects_Array"> <management.IExternalLaunchFilterParameter type="management.IExternalLaunchFilterParameter" > <Value type="String">reg-exp</Value> <Property type="String">SysLocation</Property> </management.IExternalLaunchFilterParameter> <management.IExternalLaunchFilterParameter type="management.IExternalLaunchFilterParameter"> <Value type="String">reg-exp</Value> <Property type="String">ObjectId.Key</Property> </management.IExternalLaunchFilterParameter> <management.IExternalLaunchFilterParameter type="management.IExternalLaunchFilterParameter"> <Value type="String">reg-exp</Value> <Property type="String">@IBusinessObject.Notes</Property> </management.IExternalLaunchFilterParameter> <management.IExternalLaunchFilterParameter type="management.IExternalLaunchFilterParameter"> <Value type="String">reg-exp</Value> <Property type="String">~time~</Property> </management.IExternalLaunchFilterParameter> <management.IExternalLaunchFilterParameter type="management.IExternalLaunchFilterParameter"> <Value type="String">reg-exp</Value> <Property type="String">ScriptMetadataOids</Property> <Match type="String">all</Match> </management.IExternalLaunchFilterParameter> <management.IExternalLaunchFilterParameter type="management.IExternalLaunchFilterParameter"> <Value type="String">1-100</Value> <Property type="String">CpuUsage</Property> </management.IExternalLaunchFilterParameter> </FilterParameters> <Match type="String">any</Match> </Filter> </pre>	<p>Enables you to apply a filter to narrow down the instances on which the external launch point will be available. For example, you can use this filter if you want only IMOs with specific properties to have the external launch point, rather than all IMOs of the specified type.</p> <p>You can use any of the IMO's properties in the filter.</p> <p>The filter property value can be:</p> <ul style="list-style-type: none"> Numbers—a pipe-delimited list of numbers and ranges, such as: "1 4-6 8" Strings and other scalar properties—a Java regular expression (<code>reg-exp</code>) that is checked for containment, i.e., a full match is not required. Scalar properties are converted to string values, and null values are converted to empty strings (""). Enum properties—either the enum number or string value. Arrays—value is one of the above, with the option to specify whether any (at least one) or all of the items must match the filter value. <p>Multiple filter criteria are also supported. The <code>match any</code> or <code>match all</code> option should include the logical condition between the criteria elements.</p>

Table 8-1 BQL Command Format and Syntax (continued)

Property / Type	Value	Description
ExternalBatchToExecute / String	<p>Examples:</p> <ul style="list-style-type: none"> Specifying a batch file located on the client machine: <pre><ExternalBatchToExecute type="String">C:\runthis.bat</ExternalBatchToExecute></pre> Specifying a batch file located on the gateway machine, this batch file will be automatically downloaded from the gateway machine: <pre><ExternalBatchToExecute type="String">foldername/runthis.bat</ExternalBatchToExecute></pre> Specifying a URL: <pre><ExternalBatchToExecute type="String">http://www.cisco.com</ExternalBatchToExecute></pre> Passing an environment variable: <pre><ExternalBatchToExecute type="String">%java_home%\bin\javadoc</ExternalBatchToExecute></pre> 	<p>Defines the details of the executable steps or batch file to be invoked. You:</p> <ul style="list-style-type: none"> Must specify the parameters as an array when using a script. Cannot pass any parameters in the URL. You must use batch files or LineToExecute to pass the parameters. Can include more commands as part of a batch file. Can include an environment variable when invoking the batch file. <p>To define a string that represents the URL or script which is executed along with its parameters, use the LineToExecute property.</p>

Table 8-1 BQL Command Format and Syntax (continued)


Property / Type	Value	Description
LineToExecute / String	<p>Example:</p> <p>Launching a URL with the Gateway IP, network element IP, and logged-in username parameters, using / as delimiter:</p> <pre><LineToExecute type="String">http://url.example.com/\$GW\$/com.sheer.imo.IManagedElement.IP\$/USERNAME\$/LineToExecute></pre> <p>Example:</p> <p>Launching a URL from the soft property from the interface of the supported IMO:</p> <pre><LineToExecute type="String">http://url.example.com/\$GW\$/com.sheer.imo.IEthernetService.NetworkElement#softproperty\$/USERNAME\$/LineToExecute></pre>	<p>Defines a string that represents the URL or script which is executed along with its parameters.</p> <p>The syntax includes:</p> <ul style="list-style-type: none"> • \$—Used when you specify an IMO property definition. • ~—Used when you specify a Soft Property definition. • \$number\$—The parameter whose index is a number. • \$USERNAME\$—The logged-in username. • \$GW\$—The Prime Network gateway IP address. <p>Note The & character should be written as &amp; when used inside a BQL command.</p> <p>You can define the launch point of the external application from a soft property of the supported IMO by using the OID of the supported IMO.</p> <p>Note The soft property need to be already defined in the IMO of the supported network element.</p>
MenuCaption / String	<p>Example:</p> <pre><MenuCaption type="String">ping</MenuCaption></pre>	<p>The wording of the right-click menu option to be used for the external launch.</p> <p> Note Unless you want to create a mnemonic, do not use an underscore (_) character at the beginning of the string. For example, if <code>_scripts</code> is specified as the menu caption, the result will be having the menu caption as <code>scripts</code>, where <code>s</code> is the mnemonic.</p>
MenuPath / String	<p>Example:</p> <pre><MenuPath type="String">external launch demo</MenuPath></pre>	<p>The menu path to be followed to get to the external launch point. You can have a sub-menu separated by a forward slash (/).</p>
Parameters / IMObjects_Array	<pre><Parameters type="IMObjects_Array"></pre>	<p>All external launch parameters must be passed in an array. See management.IExternalLaunchParameter, page 8-8.</p>

Table 8-1 BQL Command Format and Syntax (continued)

Property / Type	Value	Description
Role / String	Example: <code><Role type="string">CONFIGURATOR</Role></code>	Defines the minimum user role required to use the launch point—administrator, configurator, and so on. If this is not defined, all users will have access to the launch point.
Selection Limit / Int	Example: <code><SelectionLimit type="int">2</SelectionLimit></code>	Defines the number of items that must be selected for the launch point to be displayed. For example, if the selection limit is 2, the right-click menu option will not appear if only one item is selected, or if three items are selected. It will only appear if two items are selected.
management.IExternalLaunchParameter		
ID / OID	<code>{[ExternalLaunchParameter(Index=1)]}</code>	Index value defines the order in which the parameters will be passed to the script.
IMOType / String	Example 1: <code><IMOType type="String">com.sheer.imo.IPortConnector</IMOType></code> Example 2: <code><IMOType type="String">com.sheer.imo.newalarm.ITicket</IMOType></code>	Defines the IMO context type. You can also leave this field empty. You can pass the parameter not only from the IMO and its parent on which you set the menu but also from the IMO tree up to IManagedElement. It can be the parent or the higher level IMO depending on the IMO that you have selected for your menu.
PropertyName / String	Example 1: <code><PropertyName type="String">PortAlias</PropertyName></code> Example 2: <code><PropertyName type="String">LatestState</PropertyName></code> Example 3 (retrieving properties of a contained IMO): <code><PropertyName type="String">ObjectId.Key</PropertyName></code> Example 4 (retrieving properties not available at time of execution): <code><PropertyName type="String">Port#PortDescription</PropertyName></code>	Defines the property or soft property name within the IMO context type. This value is used as an argument when invoking the application. You can also have any constant value. If the property value is another IMO, you can retrieve the properties of this IMO using a dot notation (see Example 3). If the property value is an OID, you can retrieve the properties of the referenced IMO using the # notation (see Example 4).

Table 8-1 BQL Command Format and Syntax (continued)

Property / Type	Value	Description
RunSingleCommand / Boolean	<p>Example:</p> <pre><RunSingleCommand type="boolean">true</RunSingleCommand></pre> <p>The script will be called once, with the required properties of all selected IMOs passed at the same time.</p>	If the value is set to true, a single script will be run for all the selected objects. If set to false, a script will be run for each of the selected objects.
ReplaceNullWith / String	<p>Example:</p> <pre><ReplaceNullWith type="String">N/A</ReplaceNullWith></pre>	<p>If you want a null value not to be ignored or replaced with an empty string, you can use the ReplaceNullWith string. This is optional.</p> <p>In the example, the null value will be replaced by N/A. See Defining Soft Property Parameters for a BQL Script for more information on the usage of this property.</p>

Defining Soft Property Parameters for a BQL Script

You can use a soft property as a parameter in an external launch BQL command. You can specify the soft property that you want to define the external launch to be performed. To specify this, enter the following in the line to execute property of the BQL Command:

```
<LineToExecute
type="String">http://url.example.com/$GW$/$com.sheer.imo.IEthernetService.NetworkElemnt#~soft
property~$/$USERNAME$</LineToExecute>
```

To define a soft property parameter you need to write the soft property name (not the label) in the PropertyName entry, and add the following entry to the parameter definition:

```
<SoftProperty type="boolean">true</SoftProperty>
```

If you do not add this entry, by default the value is assumed to be false.

Parameters that have the value true for the soft property entry are not validated in the definition of the command.

The launch point will be visible even if the IMO does not have the soft property defined on it. When a soft property is not defined, an appropriate message is displayed after the command is run.



Note

If you create an ExternalBatchtoExecute-based external launch command that includes a soft property and the specified soft property does not exist on the IMO, the output will have an empty string (" "). This is to ensure that any logic built in to the script which depends on the parameter order, functions as desired. You may use the ReplaceNullWith external launch command parameter if you want to get a different string for the null value.

Running BQL Scripts

See the *Cisco Prime Network Integration Developer Guide* to understand how to run a BQL script.

Enabling Automatic Download of BQL Scripts to Clients

You have the option to have the scripts downloaded automatically to all clients associated with the gateway, upon next login.

To enable automatic download of scripts to clients:

-
- Step 1** Create a directory for your scripts under the Main/webstart/scripts directory on the Prime Network gateway.
- Step 2** Run [~/Main/scripts]% updateXLaunchScripts.pl on the gateway to update the auto-deployable scripts jar.
- The next time a client connects to the gateway, the scripts will be downloaded and extracted by the client application.
-

Deleting an External Launch Point

You can delete an external application launch point using the BQL **Delete** command.



Note

See the *Cisco Prime Network Integration Developer Guide* for an understanding of BQL, and as a prerequisite to understanding this section.

To delete the launch point:

-
- Step 1** Create a BQL script with the **Delete** command as shown below:
- ```
<?xml version="1.0" encoding="UTF-8"?>
<command name="Delete">
 <param name="oid">
 <value>
 <management.IExternalLaunch>
 <ID type="Oid">{[ExternalLaunch(ContextImoType=com.sheer.imo.IPortConnector)
(Name=example4)]}</ID>
 </management.IExternalLaunch>
 </value>
 </param>
</command>
```
- Step 2** Run the BQL script.
- Step 3** Launch Prime Network Vision to verify that the launch point has been removed.
-

# Sample BQL Scripts for Launching External Applications

This chapter provides example scripts for launching external applications using BQL.



**Note**

See [Table 8-1 on page 8-4](#) for details on the syntax used in the commands.

For a BQL Example That:	See
Runs a single script on selected tickets using their latest state and last modification time as parameters	<a href="#">Example 1, page 8-11</a>
Runs a single script on selected tickets that include Layer 2 in their description, using their latest state and last modification time as parameters	<a href="#">Example 2, page 8-12</a>
Includes a soft property parameter in a BQL command to create an external launch point on a network element	<a href="#">Example 3, page 8-13</a>
Uses a batch file to invoke an application	<a href="#">Example 4, page 8-13</a>
Executes a script named C:/script.batch using PortAlias and the logged-in username as parameters	<a href="#">Example 5, page 8-14</a>
Launches a URL that uses the parameters values gateway IP, NE IP and the logged-in username; and uses / as a delimiter	<a href="#">Example 6, page 8-14</a>
Launches a URL that uses the parameters a, b, and c whose values are, respectively, the soft property showClock, a constant string "constant", and the logged-in username	<a href="#">Example 7, page 8-15</a>
Launches a URL that uses the parameters a, b, c and d whose values are, respectively, the IMO property IP, the property PortAlias, a constant string "any_const_parameter", and the logged-in username; and uses the existing Parameters element	<a href="#">Example 8, page 8-15</a>
Uses an environment variable to invoke an application	<a href="#">Example 9, page 8-16</a>
Uses multiple IMO contexts to launch an application	<a href="#">Example 10, page 8-16</a>

For details about the format and the syntax of BQL scripts, see [Creating BQL Scripts, page 8-4](#).



**Note**

<param name = "replace"> is always set to **true**. This value replaces the old value with the new value based on the script that is invoked.

## Example 1

This example runs a single script on the selected tickets with their latest state and last modification time as parameters.

```
<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>

 <ID type="Oid">{[ExternalLaunch(ContextImoType=com.sheer.imo.newalarm.ITicket
) (Name=print state)]}</ID>
 <ExternalBatchToExecute type="String">C:\script.bat</ExternalBatchToExecute>
```

```

 <MenuCaption type="String">Print state</MenuCaption>
 <MenuPath type="String">My apps</MenuPath>
 <Parameters type="IMObjects_Array">
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=1)] }</ID>
 <IMObjectType type="String">com.sheer.imo.newalarm.ITicket</IMObjectType>
 <PropertyName type="String">LatestState</PropertyName>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=2)] }</ID>
 <IMObjectType type="String">com.sheer.imo.newalarm.ITicket</IMObjectType>
 <PropertyName type="String">LastModificationTime</PropertyName>
 </management.IExternalLaunchParameter>
 </Parameters>
 <RunSingleCommand type="boolean">true</RunSingleCommand>
 </management.IExternalLaunch>
</value>
</param>
<param name="replace">
 <value>true</value>
</param>
</command>

```

## Example 2

This example runs a single script on the selected tickets with their latest state and last modification time as parameters, for tickets that include Layer 2 in their description.

```

<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>
 <ID type="Oid">{ [ExternalLaunch (ContextImoType=com.sheer.imo.newalarm.ITicket)
 (Name=print state)] }</ID>
 <ExternalBatchToExecute type="String">C:\script.bat</ExternalBatchToExecute>
 <LatestState type="String">Layer 2</LatestState>
 <MenuCaption type="String">Print state</MenuCaption>
 <MenuPath type="String">My apps</MenuPath>
 <Parameters type="IMObjects_Array">
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=1)] }</ID>
 <IMObjectType type="String">com.sheer.imo.newalarm.ITicket</IMObjectType>
 <PropertyName type="String">LatestState</PropertyName>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=2)] }</ID>
 <IMObjectType type="String">com.sheer.imo.newalarm.ITicket</IMObjectType>
 <PropertyName type="String">LastModificationTime</PropertyName>
 </management.IExternalLaunchParameter>
 </Parameters>
 <RunSingleCommand type="boolean">true</RunSingleCommand>
 </management.IExternalLaunch>
 </value>
 </param>
 <param name="replace">
 <value>true</value>
 </param>
</command>

```

**Example 3**

This example includes a soft property parameter in a BQL command for an external launch point on a network element.

```
<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>

 <ID type="Oid">{ [ExternalLaunch(ContextImoType=com.sheer.imo.IManagedElement) (Name=InfoVista-Launch-test-2)] }</ID>
 <ElementType type="String">Cisco 7206VXR</ElementType>
 <ExternalBatchToExecute
 type="String">C:/IV/launch-report.bat</ExternalBatchToExecute>
 <MenuCaption type="String">test4</MenuCaption>
 <MenuPath type="String">Performance/Test4</MenuPath>
 <Parameters type="IMObjects_Array">
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=1)] }</ID>
 <IMObjectType type="String">com.sheer.imo.IManagedElement</IMObjectType>
 <PropertyName type="String">showClock</PropertyName>
 <SoftProperty type="boolean">true</SoftProperty>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=2)] }</ID>
 <IMObjectType type="String">com.sheer.imo.IManagedElement</IMObjectType>
 <PropertyName type="String">IP</PropertyName>
 <SoftProperty type="boolean">false</SoftProperty>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=3)] }</ID>
 <IMObjectType type="String">com.sheer.imo.IManagedElement</IMObjectType>
 <PropertyName type="String">ElementTypeKey</PropertyName>
 </management.IExternalLaunchParameter>
 </Parameters>
 <Vendor type="String">Cisco</Vendor>
 </management.IExternalLaunch>
 </value>
 </param>
 <param name="replace">
 <value>true</value>
 </param>
</command>
```

**Example 4**

This example uses a batch file to invoke an application.

```
<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>

 <ID type="Oid">{ [ExternalLaunch(ContextImoType=com.sheer.imo.IManagedElement) (Name=example1)] }</ID>
 <ElementType type="String">Cisco</ElementType>
 <Vendor type="String">Cisco</Vendor>
 <ExternalBatchToExecute
 type="String">C:\runthis.bat</ExternalBatchToExecute>
 <MenuCaption type="String">ping</MenuCaption>
 <MenuPath type="String">external launch/file example</MenuPath>
 </management.IExternalLaunch>
 </value>
 </param>
</command>
```

```

 <Parameters type="IMObjects_Array">
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=1)] }</ID>
 <IMObjectType type="String">com.sheer.imo.IManagedElement</IMObjectType>
 <PropertyName type="String">IP</PropertyName>
 </management.IExternalLaunchParameter>
 </Parameters>
 </management.IExternalLaunch>
</value>
</param>
<param name="replace">
 <value>>true</value>
</param>
</command>

```

### Example 5

This example executes a script `C:/script.batch` with parameters `PortAlias` and the logged-in username.

```

<?xml version="1.0" encoding="UTF-8"?
<command name="Set"
 <param name="imo">
 <value>
 <management.IExternalLaunch>
 <ID type="Oid">{ [ExternalLaunch (ContextImoType=com.sheer.imo.IPortConnector)
 (Name=example)] }</ID>
 <MenuCaption type="String">My Example</MenuCaption>
 <MenuPath type="String">My Menu/My Sub Menu</MenuPath>
 <LineToExecute type="String">c:/script.batch
 $com.sheer.imo.IPortConnector.PortAlias $USERNAME$</LineToExecute>
 </management.IExternalLaunch>
 </value>
 </param>
 <param name="replace">
 <value>>true</value>
 </param>
</command>

```

### Example 6

This example launches a URL with parameter values Gateway IP, the NE IP and the logged-in username, using / (forward slash) as delimiter.

```

<?xml version="1.0" encoding="UTF-8"?
<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>
 <ID type="Oid">{ [ExternalLaunch (ContextImoType=com.sheer.imo.IPortConnector)
 (Name=example)] }</ID>
 <ElementType type="String">IOU</ElementType>
 <Vendor type="String">Cisco</Vendor>
 <MenuCaption type="String">My Example</MenuCaption>
 <MenuPath type="String">My Menu/My Sub Menu</MenuPath>
 <LineToExecute type="String">http://url.example.com/GW/com.sheer.imo.IManagedElement.IP/$USERNAME$</LineToExecute>
 </management.IExternalLaunch>
 </value>
 </param>
 <param name="replace">
 <value>>true</value>

```

```

 </param>
</command>

```

### Example 7

This example launches a URL with parameters a, b, and c, whose values are the soft property `showClock`, a constant string `constant`, and the logged-in username.

```

<?xml version="1.0" encoding="UTF-8"?>
<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>
 <ID type="Oid">{ [ExternalLaunch (ContextImoType=com.sheer.imo.IPortConnector)
 (Name=example)] }</ID>
 <ElementType type="String">IOU</ElementType>
 <MenuCaption type="String">My Example</MenuCaption>
 <MenuPath type="String">My Menu/My Sub Menu</MenuPath>
 <LineToExecute type="String">http://url.example.com/my.action?a=~com.sheer.i
 mo.IManagedElement.showClock~&b=constant&c=$USERNAME$</LineToExecute>
 </management.IExternalLaunch>
 </value>
 </param>
 <param name="replace">
 <value>true</value>
 </param>
 </command>

```

### Example 8

This example launches a URL with parameters a, b, c, and d, whose values are the IMO property `IP`, property `PortAlias`, a constant string `any_const_parameter`, and the logged-in username, using the existing `Parameters` element.

```

<?xml version="1.0" encoding="UTF-8"?>
<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>
 <ID type="Oid">{ [ExternalLaunch (ContextImoType=com.sheer.imo.IPortConnector)
 (Name=example)] }</ID>
 <ElementType type="String">IOU</ElementType>
 <MenuCaption type="String">My Example</MenuCaption>
 <MenuPath type="String">My Menu/My Sub Menu</MenuPath>
 <LineToExecute type="String">http://url.example.com/my.action?a=1&b=$2
 $&c=$3$&d=$USERNAME$</LineToExecute>
 <Parameters type="IMObjects_Array">
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=1)] }</ID>
 <IMOType type="String">com.sheer.imo.IManagedElement</IMOType>
 <PropertyName type="String">IP</PropertyName>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=2)] }</ID>
 <IMOType type="String">com.sheer.imo.IPortConnector</IMOType>
 <PropertyName type="String">PortAlias</PropertyName>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=3)] }</ID>

```

```

 <IMOType type="String"></IMOType>
 <PropertyName type="String">any_const_parameter</PropertyName>
 </management.IExternalLaunchParameter>
</Parameters>
</management.IExternalLaunch>
</value>
</param>
<param name="replace">
 <value>>true</value>
</param>
</command>

```

### Example 9

This examples uses an environment variable to invoke an application.

```

<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>

 <ID type="Oid">{ [ExternalLaunch (ContextImoType=com.sheer.imo.IManagedElement) (Name=example3)] }</ID>
 <ElementType type="String">Cisco</ElementType>
 <Vendor type="String">Cisco</Vendor>
 <ExternalBatchToExecute type="String">%EMS_A%runthis.bat</ExternalBatchToExecute>
 <MenuCaption type="String">environment variable</MenuCaption>
 <MenuPath type="String">External Launch</MenuPath>
 <Parameters type="IMObjects_Array">
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=1)] }</ID>
 <IMOType type="String"></IMOType>
 <PropertyName type="String">www.cisco.com</PropertyName>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=2)] }</ID>
 <IMOType type="String">com.sheer.imo.IManagedElement</IMOType>
 <PropertyName type="String">IP</PropertyName>
 </management.IExternalLaunchParameter>
 </Parameters>
 </management.IExternalLaunch>
 </value>
 </param>
 <param name="replace">
 <value>true</value>
 </param>
</command>

```

### Example 10

This example uses multiple IMO contexts to launch an application.

```

<command name="Set">
 <param name="imo">
 <value>
 <management.IExternalLaunch>

 <ID type="Oid">{ [ExternalLaunch (ContextImoType=com.sheer.imo.IPortConnector) (Name=example4)] }</ID>
 <ElementType type="String">IOU</ElementType>

```



```

<Vendor type="String">Cisco</Vendor>
<ExternalBatchToExecute
 type="String">c:runthis.bat</ExternalBatchToExecute>
<MenuCaption type="String">ping</MenuCaption>
<MenuPath type="String">External Launch</MenuPath>
<Parameters type="IMObjects_Array">
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=1)] }</ID>
 <IMObjectType type="String">com.sheer.imo.IManagedElement</IMObjectType>
 <PropertyName type="String">IP</PropertyName>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=2)] }</ID>
 <IMObjectType type="String">com.sheer.imo.IPortConnector</IMObjectType>
 <PropertyName type="String">PortAlias</PropertyName>
 </management.IExternalLaunchParameter>
 <management.IExternalLaunchParameter>
 <ID type="Oid">{ [ExternalLaunchParameter (Index=3)] }</ID>
 <IMObjectType type="String"></IMObjectType>
 <PropertyName type="String">any_const_parameter</PropertyName>
 </management.IExternalLaunchParameter>
</Parameters>
</management.IExternalLaunch>
</value>
</param>
<param name="replace">
 <value>>true</value>
</param>
</command>

```

