Custom Infobox Documentation

Developed by: Ricky Brundritt, [InKnowledge](http://inknowledge.co.uk/) Date: May 21st, 2011

Table of Contents

[Description 2](#_Toc293702913)

[CustomInfobox Class 2](#_Toc293702914)

[Constructor 2](#_Toc293702915)

[Infobox Options 2](#_Toc293702916)

[Public Methods 3](#_Toc293702917)

[Example 1 3](#_Toc293702918)

[Example 2 4](#_Toc293702919)

# Description

This document is meant to accompany the V7CustomInfobox.js JavaScript file. This JavaScript file is meant to create a highly customizable and flexible infobox framework that is reusable, and easy to extend. This method takes in a map reference and a set of options. This modular plugin was designed to work with the new Bing Maps v7 Modular framework: <http://msdn.microsoft.com/en-us/library/hh125836.aspx>

This modular plugin was originally developed [InKnowledge](http://inknowledge.co.uk/) for their [Bing Maps Locator Wizard](http://bingmapslocators.cloudapp.net/) and has been extracted to be shared with the Bing Maps development community.

# CustomInfobox Class

The **CustomInfobox** class is a modular plugin for Bing Maps v7. It is designed to highly customizable yet light weight and useable. Key features include:

* Ability to dynamically change the dimensions and color of the infobox arrow
* Positions infobox towards the center of the map. This is so as to keep the infobox within the map view.
* Ability to turn tethering on or off.

The expected use case is for when users click or hover an entity on the map. However this can also be used independently of an entity simply by specifying an anchor point and content to be displayed.

## Constructor

|  |
| --- |
| var customInfobox = new CustomInfobox(map, options); |

|  |  |
| --- | --- |
| Parameter | Description |
| map | A Microsoft.Maps.Map object that the infobox is going to be used with. |
| options | An infobox option object that specifies how the infobox should render. |

## Infobox Options

When creating an instance of the **CustomInfobox** it is possible to specify options. These options allow you to customize how the infobox is rendered.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| arrowColor | string | The color of the infobox arrow. |
| arrowLength | int | The length of the arrow in pixels |
| arrowWidth | int | The width of the arrow in pixels. |
| borderColor | string | The color of the border around the content area. |
| color | string | The background color of the infobox content area. |
| minHeight | int | The minimium height of the infobox content area. |
| minWidth | int | The minimium width of the infobox content area. |
| offset | Point | The distance to offset the infobox from the anchor point. |
| orientation | byte | The orientation to display the infobox. A value of 0 will place the infobox so that the arrow points left/right. A value of 1 will place the infobox so that the arrow points up/down. |
| showArrow | bool | A flag indicating if the infobox should have an arrow or not. |
| showCloseButton | bool | A flag indicating if a close button should be displayed inside the infobox. |
| tether | bool | A flag indicating if the infobox should be tethered to its anchor. If true the infobox will remain open and will move with the map until the hide method is called. If false the infobox will close when the map moves. |

## Public Methods

|  |  |
| --- | --- |
| Name | Description |
| hide | Hides the infobox |
| show | Renders an infobox in the specified location with the specified content which can be text or custom HTML. |
| setOptions | Sets the custom infobox options. |
| getOptions | Gets the current inbox options. |

## Example 1

This is a simple example of how to display an infobox on the map using this modular plugin.

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html>  <head>  <title>Custom Infoboxes</title>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <!-- Add a reference to the Bing Maps Control -->  <script charset="UTF-8" type="text/javascript" src="http://ecn.dev.virtualearth.net/mapcontrol/mapcontrol.ashx?v=7.0"></script>    <!-- jQuery reference. Used to ensure cross browser support. -->  <script type="text/javascript" src="http://code.jquery.com/jquery-1.5.2.min.js"></script>  <script type="text/javascript">  var map, customInfobox;  function GetMap() {  map = new Microsoft.Maps.Map(document.getElementById("myMap"), { credentials: "Your Bing Maps Key" });  //Register and load the Custom Infobox Module  Microsoft.Maps.registerModule("CustomInfoboxModule", "scripts/V7CustomInfobox.min.js");  Microsoft.Maps.loadModule("CustomInfoboxModule", { callback: function () {  //Create an instance of the custom infobox control  customInfobox = new CustomInfobox(map);  }  });  }  function displayInfobox() {  customInfobox.show(map.getCenter(), "Hello World");  }  </script>  </head>  <body onload="GetMap();">  <div id='myMap' style="position:relative; width:600px; height:500px;"></div><br />  <input type="button" value="Show Infobox" onclick="displayInfobox()" />  </body>  </html> |

## Example 2

This example shows how to tie a custom infobox to a pushpin

|  |
| --- |
| <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  <html>  <head>  <title>Custom Infoboxes</title>  <meta http-equiv="Content-Type" content="text/html; charset=utf-8">  <!-- Add a reference to the Bing Maps Control -->  <script charset="UTF-8" type="text/javascript" src="http://ecn.dev.virtualearth.net/mapcontrol/mapcontrol.ashx?v=7.0"></script>    <!-- jQuery reference. Used to ensure cross browser support. -->  <script type="text/javascript" src="http://code.jquery.com/jquery-1.5.2.min.js"></script>  <script type="text/javascript">  var map, customInfobox;  function GetMap() {  map = new Microsoft.Maps.Map(document.getElementById("myMap"), { credentials: "Your Bing Maps Key" });  //Register and load the Custom Infobox Module  Microsoft.Maps.registerModule("CustomInfoboxModule", "scripts/V7CustomInfobox.min.js");  Microsoft.Maps.loadModule("CustomInfoboxModule", { callback: function () {  //Create an instance of the custom infobox control  customInfobox = new CustomInfobox(map);  }  });  //Define custom properties for the pushpin class.  //This is an example of one way to assign metadata to a shape.  Microsoft.Maps.Pushpin.prototype.title = null;  Microsoft.Maps.Pushpin.prototype.description = null;  // Retrieve the location of the map center  var center = map.getCenter();  // Add a pin to the center of the map  var pin = new Microsoft.Maps.Pushpin(center);  //Add some mock info about the pin  pin.title = "Pin 1";  pin.description = "This is the infobox for Pin 1.<br/><span style='color:red;'>Here is some custom HTML</span>";  // Add a handler for the pushpin click event.  Microsoft.Maps.Events.addHandler(pin, 'click', displayInfobox);  // Add the pushpin and info box to the map  map.entities.push(pin);  }  function displayInfobox(e) {  var latlong = e.target.getLocation();  //Define the layout contents in the infobox  var html = ["<div style='padding:10px'>"];  html.push("<b>", e.target.title, "</b><br/>");  html.push(e.target.description, "<br/>");  html.push("</div>");  //Display Infobox  customInfobox.show(latlong, html.join(''));  }  </script>  </head>  <body onload="GetMap();">  <div id='myMap' style="position:relative; width:100%; height:500px;"></div>  </body>  </html> |