KYMERA
SYSTEMS

Map Panel Module

User Guide

1 800 470 2302
www.kymerasystems.com

Table of Contents

Introduction

The Kymera Map Panel module allows users to embed maps in their Ignition projects with the ability to add custom map markers with notes or embedded expressions, as well as zoom in and out and use a variety of map providers.

Usage

After installing the module you will find the Kymera Map Panel component on the Display tab of the Component Palette. Simply drag the component on to a window to begin.

Initially, the map panel has no defined map points. To add map points to the map panel, add them to the Data Points dataset. The dataset has the following columns:

String Name	The id of the map point. Commonly the location name of the map point.
Double Latitude	The latitude of the map point location.
Double Longitude	The longitude of the map point location.
String Expression	An expression to display text on the label of the map point.
String Expression Type	The type of expression on the map point. Can be one of 2 types, ‘text’ or ‘expression’. A ‘text’ expression displays the text defined in the expression column. An ‘expression’ expression displays the result of an Ignition Expression Function defined in the Expression column.
String Icon Path	The path to the image to use for the map point.
String Icon Hover Path	The path to the image to use when the mouse is hovered over the map point.

To zoom in on the map panel, scroll up on the mouse wheel or double click the left mouse button.

To zoom out on the map panel, scroll down on the mouse wheel or double click the right mouse button.

To move the map around, click and hold the mouse while moving around.

Properties

Data

Max Tile Cache Size

The maximum amount of cached tiles.
Scripting name `maxCacheSize`
Data type int

Data Points

The map points to draw on the map
Scripting name `dataPoints`
Data type Dataset

Appearance

Animated

Use animation to draw the maps when zooming?
Scripting name `animated`
Data type boolean

Zoom Level

The zoom level of the map panel, 1 being fully zoomed out.
Scripting name `zoom`
Data type int

Debug

Debug

Draws borders around the individual map panel tiles.
Scripting name `debug`
Data type boolean

Scripting : Map Panel

The following scripting functions are available on the map panel.

setTileServer()

Description

Sets the tile server to be used by the map panel.

Syntax

```
setTileServer(url, maxZoom)
```

Parameters

String `url` – The URL to the tile server.
int `maxZoom` – The max zoom allowed on the map panel.

Returns

nothing

Scope

Client

Examples

The following snippet switches the map panel to use Map Quest’s Open Street Map.

```
mapPanel = event.source.parent.getComponent('MapPanel')
mapPanel.setTileServer(‘http://a.tile.openstreetmap.org/’, 20)
```

getDataPoints()

Description

Gets the data points representing the map markers on the Map Panel from the Map Panel Data Points property.

Syntax

```
getDataPoints()
```

Parameters

none

Returns

Dataset – The data points representing the map markers.

Scope

Client

Examples

```
mapPanel = event.source.parent.getComponent('MapPanel')
mapPoints = mapPanel.getDataPoints()
```

setDataPoints()

Description

Sets the map points to display on the map. The dataset must have the following columns:

- Name
- Latitude
- Longitude
- Expression Type
- Expression
- Icon Path
- Icon Hover Path

Syntax

```
setDataPoints(dataPoints)
```

Parameters

Dataset `dataPoints` – A dataset of map points to display on the map.

Returns

nothing

Scope

Client

Examples

The following snippet adds a pink pin to the map for the City of Edmonton, Alberta. It shows a shadow when hovered over, and displays the name “Edmonton” when clicked.

```
mapPanel = event.source.parent.getComponent('MapPanel')
headers = ["Name", "Latitude", "Longitude", "Expression Type", "Expression", "Icon Path",
        "Icon Hover Path"]
data = []
data.append(["Edmonton", 53.55, -113.49, "text", "Edmonton", "MapPanel/pink.png",
        "MapPanel/shadow-pink.png"])
mapPoints = system.dataset.toDataSet(headers, data)
mapPanel.setDataPoints(mapPoints)
```

isAnimated()

Description

Returns the status of the Animated property of the Map Panel.

Syntax

```
isAnimated()
```

Parameters

none

Returns

boolean – True if the Animated property is checked.

Scope

Client

Examples

```
mapPanel = event.source.parent.getComponent('MapPanel')
animated = mapPanel.isAnimated()
```

setAnimated()

Description

Sets or clears the Animated property of the Map Panel.

Syntax

```
setAnimated(animated)
```

Parameters

boolean `animated` – The state of the Animated property to set.

Returns

nothing

Scope

Client

Examples

The following snippet enables animation when drawing the maps if it is turned off.

```
mapPanel = event.source.parent.getComponent('MapPanel')
animated = mapPanel.isAnimated()
if animated == 0:
       mapPanel.setAnimated(1)
```

setGpsLocation()

Description

Centers the map around a set of latitude and longitude coordinates.

Syntax

`setGpsLocation`(lat, lon)

Parameters

double `lat` – The latitude of the map coordinate to center the map on.

double `lon` – The longitude of the map coordinate to center the map on.

Returns

nothing

Scope

Client

`setGpsLocation`(gpsLocation)

Parameters

Point2D.Double `gpsLocation` – A point containing the latitude and longitude of the map coordinate to center the map on.

Returns

nothing

Scope

Client

Examples

The following property change code executes on a table containing cities with their latitudes and longitudes to center the map on the city selected and zoom in.

```
if event.propertyName == "selectedRow" and event.newValue != -1:
        lat = event.source.data.getValueAt(event.newValue,"Latitude")
        lon = event.source.data.getValueAt(event.newValue,"Longitude")
        event.source.parent.getComponent('MapPanel').setZoom(9)
        event.source.parent.getComponent('MapPanel').setGpsLocation(lat,lon)
```

Using setGpsLocation(gpsLocation)

```
if event.propertyName == "selectedRow" and event.newValue != -1:
        from java.awt.geon import Point2D
        lat = event.source.data.getValueAt(event.newValue,"Latitude")
        lon = event.source.data.getValueAt(event.newValue,"Longitude")
        centerPoint = Point2D.Double(lat,lon)
        event.source.parent.getComponent('MapPanel').setZoom(9)
        event.source.parent.getComponent('MapPanel').setGpsLocation(centerPoint)
```

getGpsLocation()

Description

Returns the GPS coordinates at the center of the map panel.

Syntax

```
getGpsLocation()
```

Parameters

none

Returns

Point2D.Double – An x and y coordinate representing the center of the map panel.

Scope

Client

Examples

The following code snippet gets the x and y coordinate of the center of the map panel.

```
mapPanel = event.source.parent.getComponent('MapPanel')
centerPoint = mapPanel.getGpsLocation()
lat = centerPoint.x
lon = enterPoint.y
```

getZoom()

Description

Returns the current zoom level of the map.

Syntax

```
getZoom()
```

Parameters

none

Returns

int – The current zoom level.

Scope

Client

Examples

```
mapPanel = event.source.parent.getComponent('MapPanel')
print mapPanel.getZoom()
```

setZoom()

Description

Sets or clears the Animated property of the Map Panel.

Syntax

```
setZoom(zoom)
```

Parameters

int zoom – The zoom level to set.

Returns

nothing

Scope

Client

Examples

The following snippet zooms the map out as far as possible.

```
mapPanel = event.source.parent.getComponent('MapPanel')
mapPanel.setZoom(1)
```

getMapBounds()

Description

Returns the latitude and longitude of the top left corner, plus the width and height of the visible map.

Syntax

```
getMapBounds()
```

Parameters

none

Returns

Rectangle2D – A rectangle containing an x and y coordinate, plus the height (h) and width(w).

Scope

Client

Examples

The following snippet gets the location, height and width of the visible map.

```
mapPanel = event.source.parent.getComponent('MapPanel')
mapBounds = mapPanel.getMapBounds()
latitude = mapBounds.x
longitude = mapBounds.y
width = mapBounds.w
height = mapBounds.h
```

Scripting : Map Points

The following scripting functions are available on the map panel to interact with map points.

getIconPath()

Description

Returns the path to the icon used by the current map point.

Syntax

```
getIconPath()
```

Parameters

none

Returns

String – The icon path of the map point's icon.

Scope

Client

Examples

The following snippet in a mouseClicked event on the map panel displays the icon path of the map point when clicked on.

```
if event.mapPoint != None:
	print event.mapPoint.getIconPath()
```

setIconPath()

Description

Sets the icon path used by the current map point.

Syntax

```
setIconPath(iconPath)
```

Parameters

String `iconPath` – The path to the icon to set on the map point.

Returns

nothing

Scope

Client

Examples

The following snippet changes the map point icon to a pink map marker.

```
if event.mapPoint != None:
        event.mapPoint.setIconPath('MapPanel/pink.png')
```

getIconHoverPath()

Description

Returns the path to the icon used by the current map point when the mouse hovers over it.

Syntax

```
getIconHoverPath()
```

Parameters

none

Returns

String – The icon path of the map point’s hover icon.

Scope

Client

Examples

The following snippet in a mouseClicked event on the map panel displays the icon path of the map point’s hover icon when clicked on.

```
if event.mapPoint != None:
        print event.mapPoint.getIconHoverPath()
```

setIconHoverPath()

Description

Sets the icon path used by the current map point when the mouse hovers over it.

Syntax

```
setIconHoverPath(iconHoverPath)
```

Parameters

String `iconHoverPath` – The path to the icon to set on the map point when the mouse hovers over it.

Returns

nothing

Scope

Client

Examples

The following snippet changes the map point hover icon to a pink map marker with a shadow.

```
if event.mapPoint != None:
        event.mapPoint.setIconHoverPath('MapPanel/shadow-pink.png')
```

getName()

Description

Returns the name of the map point.

Syntax

```
getName()
```

Parameters

none

Returns

String – The name of the map point.

Scope

Client

Examples

The following snippet in a mouseClicked event on the map panel displays the name of the map point when clicked on.

```
if event.mapPoint != None:
	print event.mapPoint.getName()
```

getExpression()

Description

Returns the expression defined for the map point.

Syntax

```
getExpression()
```

Parameters

none

Returns

String – The map point’s expression.

Scope

Client

Examples

The following snippet in a mouseClicked event on the map panel displays the expression of the map point when clicked on.

```
if event.mapPoint != None:
        print event.mapPoint.getExpression()
```

setExpression()

Description

Sets an expression for the map point.

Syntax

setExpression(expression, expressionType**)**

Parameters

String `expression` – The expression to set on the map point.
String `expressionType` – The type of expression. Can be 'text' or 'expression'.

Returns

nothing

Scope

Client

Examples

The following snippet displays the current time on the map point.

```
if event.mapPoint != None:
	event.mapPoint.setExpression('now(1000)', 'expression')
	event.mapPoint.setShowLabel(1)
```

isShowLabel()

Description

Returns whether the label is showing on the map point or not.

Syntax

```
isShowLabel()
```

Parameters

none

Returns

boolean – True if the label is showing on the map point.

Scope

Client

Examples

The following snippet in a mouseClicked event on the map panel checks if the label is showing on the map point before attempting to display it

```
if event.mapPoint != None:
        if event.mapPoint.isShowLabel() == 0:
                event.mapPoint.setShowLabel(1)
```

setShowLabel()

Description

Shows or hides the map point's label.

Syntax

```
setShowLabel(showLabel)
```

Parameters

boolean `showLabel` – True to display the label, 0 to hide it.

Returns

nothing

Scope

Client

Examples

The following snippet hides the map points label if it is showing.

```
if event.mapPoint != None:
        if event.mapPoint.isShowLabel() == 1:
                event.mapPoint.setShowLabel(0)
```

getEvaluatedExpression()

Description

Evaluates the expression on the map point and returns the value.

Syntax

```
getEvaluatedExpression()
```

Parameters

none

Returns

String – The output of the map point expression

Scope

Client

Examples

The following snippet in a mouseClicked event on the map panel displays the current time from the map point expression 'now(1000)'.

```
if event.mapPoint != None:
	print event.mapPoint.getEvaluatedExpression()
```