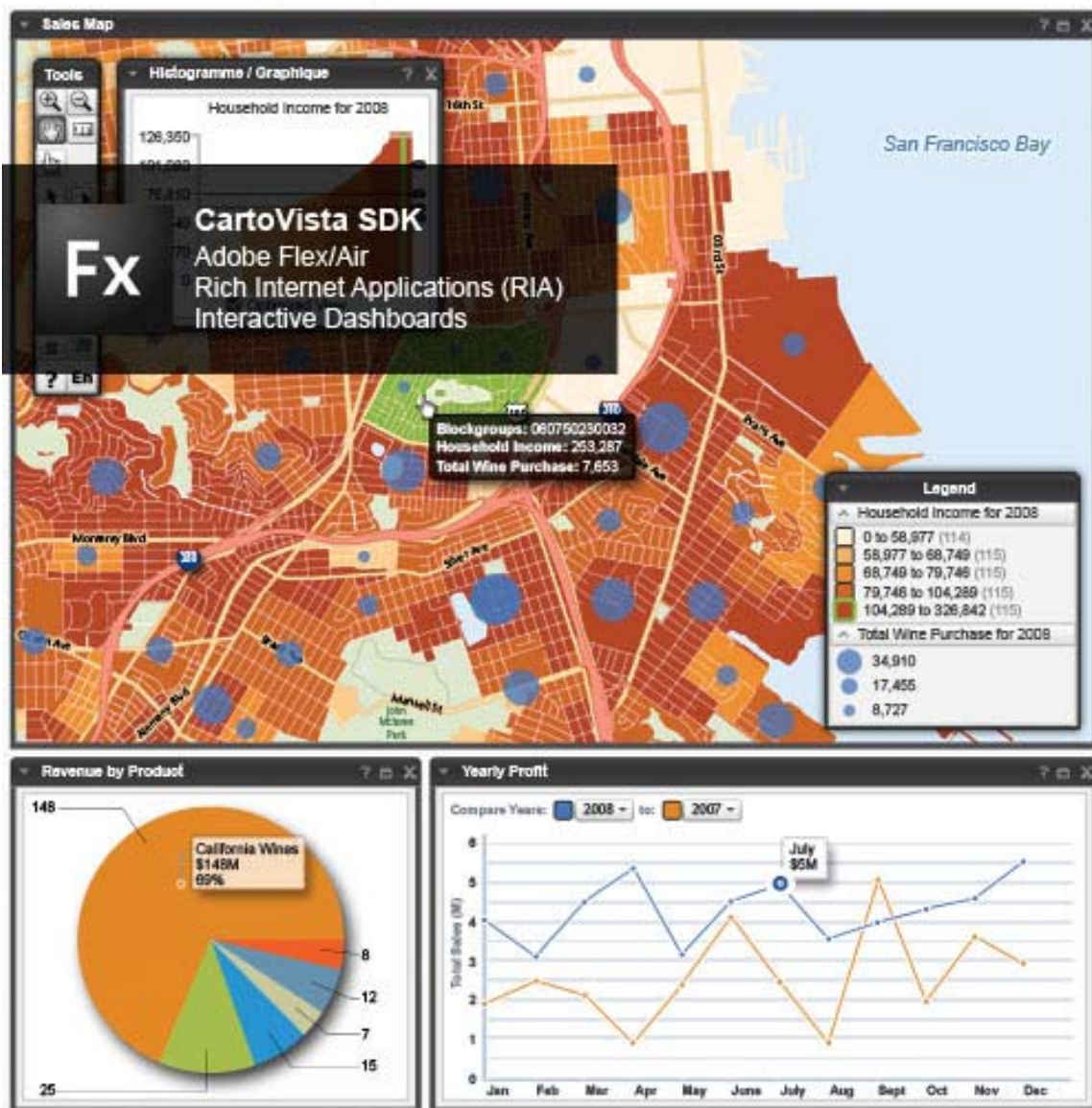




DBx GEOMATICS



CartoVista



<http://www.dbxgeomatics.com>

Table Of Contents

CartoVista Software Development Kit Help Content	1
Copyright Information	3
Getting Support for CartoVista	5
About CartoVista.....	7
About DBx GEOMATICS inc.....	9
Change History.....	11
Introduction	13
Building Dashboard Applications with CartoVista.....	17
Starting with the SDK.....	19
Functions.....	23
setLanguage Function.....	27
setSize Function	29
enableMouseEvents Function	31
setSWFMap Function	33
setRollOverColor Function	35
setThemeColor Function.....	37
setTitleVisibility Function.....	39
setSubTitleVisibility Function	41
setZoomAnimationVisibility Function.....	43
setToolTipDuration Function	45
setToolTipCallback Function.....	47
bgGetSelectedItemsID Function	49
bgGetSelectedItemsData Function.....	51
bgSetSelectedItems Function.....	53

bgGetIndicatorMetadata Function	55
setIndicatorsData Function	57
setSelectionModeAdditive Function	61
bgSetThematicLayer Function	63
bgGetThematicLayer Function	65
bgSetRangeTheme Function	67
bgGetIndicatorCategoryID Function	71
bgSetIndicatorCategoryByID Function	73
bgSetIndicatorByID Function	77
bgGetIndicatorID Function	79
bgRemoveIndicator Function	81
bgSetNumberOfRanges Function	83
bgGetNumberOfRanges Function	85
bgSetRangeThresholds Function	87
bgGetRangeThresholds Function	89
bgSetRangeColorScheme Function	91
bgGetRangeColorSchemeID Function	95
bgSetRangeColors Function	99
bgGetRangeColors Function	101
bgSetIndividualValuesColors Function	103
bgGetIndividualValuesColors Function	105
fgSetThematicLayer Function	107
fgGetThematicLayer Function	109
fgSetIndicatorCategoryByID Function	111
fgGetIndicatorCategoryID Function	115

fgSetIndicatorByID Function	119
fgGetIndicatorsID Function	121
fgRemoveIndicators Function	123
fgSetPieChartColors Function	125
fgGetPieChartColors Function	127
fgSetIndividualValuesColors Function	129
fgGetIndividualValuesColors Function	131
fgSetAtValue Function	133
fgGetAtValue Function	135
fgSetIndividualValuesSymbols Function	137
fgGetIndividualValuesSymbols Function	141
fgSetPieChartGraduation Function	143
fgGetPieChartGraduation Function	145
fgSetGraduateBy Function	147
fgGetGraduatedBy Function	149
fgSetGraduateScaling Function	151
fgGetGraduatedScaling Function	153
fgSetSymbolColor Function	155
fgGetSymbolColor Function	157
fgSetSymbolSize Function	159
fgSetSymbolColorRange Function	161
fgGetSymbolColorRange Function	163
fgSetOpacity Function	165
fgGetOpacity Function	167
fgSetSymbol Function	169

fgGetSymbol Function	171
layerSetVisibility Function	173
layerSetLabelling Function.....	175
layerSetOpacity Function	177
layerSetClassVisibility Function	179
layerSetClassLabelling Function.....	181
layerSetClassOpacity Function	183
getMapLayersInfo Function.....	185
getLayerClassInfo Function.....	187
getMapCoordinateSystemName Function	189
getMapCoordinateSystemCode Function	191
getMapUnits Function	193
getMapZoom Function	195
getMapWidth Function	197
getMapHeight Function	199
getMapCenter Function	201
mapZoomTo Function	203
mapRecenter Function	205
findObject Function	207
setObjectColor Function	209
disableLegend Function.....	213
disableSelectionWindow Function	215
setUIComponentVisibility Function.....	217
getUIComponentVisibility Function	219
setUIComponentPosition Function	221

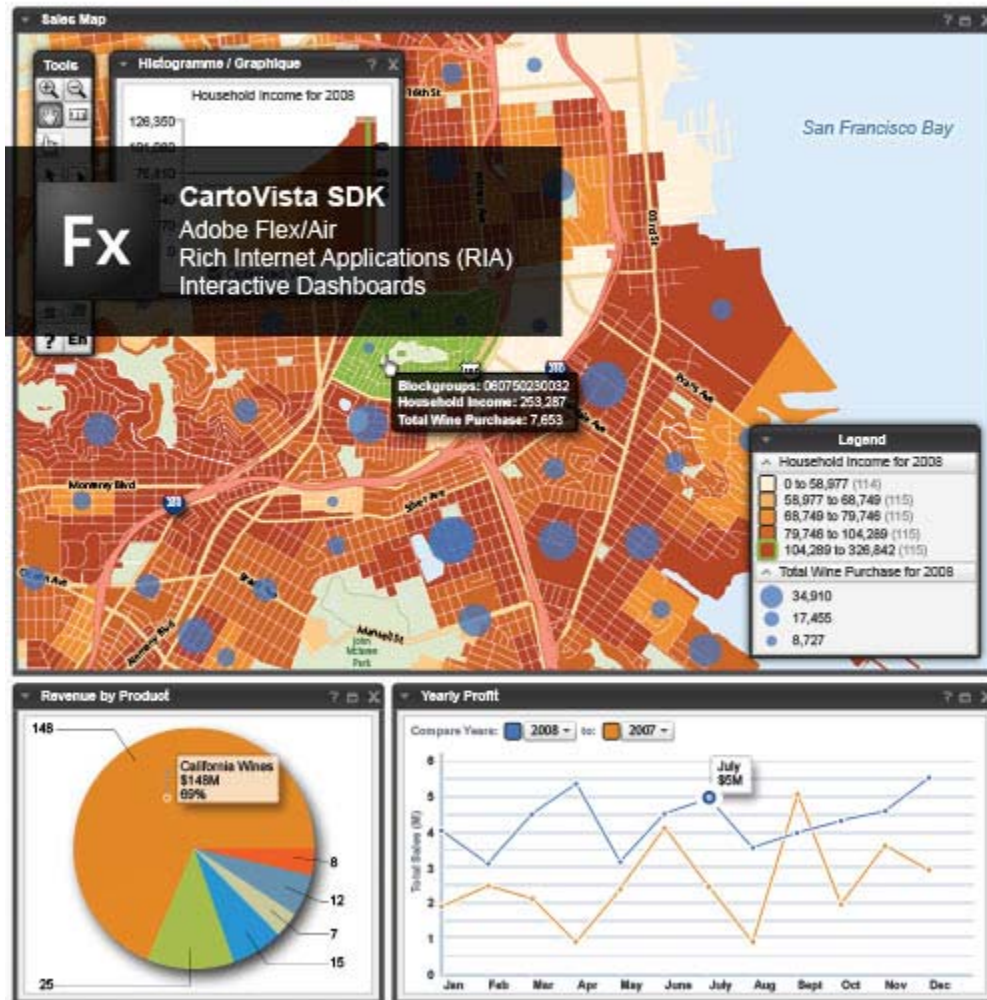
getUIComponentPosition Function	223
setActiveTool Function	225
getActiveTool Function	227
setPrintMode Function	229
Callback Functions (Events)	231
onCartoVistaLoadComplete Callback Function	233
onDataLoadComplete Callback Function	235
onCategoryFileLoadComplete Callback Function	237
onShowToolTip Callback Function	239
onSelectionChanged Callback Function	241
onInfoSelected Callback Function	243
CartoVistaMapConfiguration	247
IndicatorsTable	255
CartoVistaMapConfiguration XSD Schema	263
IndicatorsTable XSD Schema	269
CartoVista Configuration Parameters (FlashVars)	275
Index	283
Glossary	297



CartoVista Software Development Kit Help Content



CartoVista



SDK Version: **1.15**

Last Update: **September 10, 2010**

Welcome to the CartoVista SDK Developer Guide.

This documentation is designed to answer all of your questions about using the CartoVista SDK.

NOTE: This version of the SDK (1.15) is compatible with version **2.1.9** or higher of the CartoVista Flash Viewer. To see which version of the CartoVista viewer you have, click on the CartoVista logo at located on the bottom-right of the map.



Copyright Information



DBx GEOMATICS

CartoVista is a registered trademark of DBx GEOMATICS inc. The CartoVista software, including all related files and documentation is protected by copyright. Please refer to the CartoVista End User License Agreement for more information on the terms of use of the software.

Adobe *Flash* and the Adobe Flash Player are registered trademarks of [Adobe Systems Incorporated](#).



Getting Support for CartoVista

Contacting Technical Support

If you encounter problems working with CartoVista, our technical support specialists can help. Technical support includes referrals to documentation, assistance with error messages and suggestions for causes of error messages.

Technical Support is available in North America Monday - Friday from 8:00am - 4:00pm EST, excluding holidays. Please remember to include your serial number or contract number when contacting technical support.

Phone (toll free in North America): 1.866.772.2660

Phone: 819.772.2000

Fax: 819.772.2001

Email: support@dbxgeomatics.com



About CartoVista

The CartoVista engine is an innovative web-based mapping solution to share and publish statistical data.

In a few mouse clicks you can display meaningful data through a robust *Flash* -based interface. A digital *dashboard* like CartoVista helps share information and analysis capabilities within your organization.



With a comprehensive list of interactive features, CartoVista combines the best in *Vector* -based web mapping with state-of-the-art thematic display and visualization. For more information on the CartoVista engine please visit <http://www.cartovista.com>



About DBx GEOMATICS inc.

DBx GEOMATICS is a company specialized in the development of interactive web mapping applications. We offer web-base products and services for the development of innovative, user-friendly and effective geo-spatial applications. DBx GOEOMATICS create innovative mapping solutions that synthesize and analyze spatial information.



DBx GEOMATICS

The company's head office is located at:

DBx GEOMATICS inc.
37 St-Joseph Boulevard
Second Floor,
Gatineau, Quebec, Canada
J8Y 3V8

For more information on our products and professional services, please visit our web site at <http://www.dbxgeomatics.com>.



Change History

Version 1.15 - August 31, 2010

New Functions

- [setSelectionModeAdditive Function](#)
- [setToolTipCallBack Function](#)
- [setObjectColor Function](#)

New Functions (callback)

- [onCategoryFileLoadComplete Callback Function](#)
- [onShowToolTip Callback Function](#)
- [onInfoSelected Callback Function](#)

Version 1.1 - December 13, 2009

New thematic mapping functions (get)

Background

- [bgGetThematicLayer Function](#)
- [bgGetIndicatorCategoryID Function](#)
- [bgGetIndicatorID Function](#)
- [bgGetNumberOfRanges Function](#)
- [bgGetRangeThresholds Function](#)
- [bgGetRangeColorSchemeID Function](#)
- [bgGetRangeColors Function](#)
- [bgGetIndividualValuesColors Function](#)

Foreground

- [fgGetThematicLayer Function](#)
- [fgGetIndicatorCategoryID Function](#)
- [fgGetIndicatorsID Function](#)
- [fgGetPieChartColors Function](#)
- [fgGetIndividualValuesColors Function](#)
- [fgGetAtValue Function](#)
- [fgGetIndividualValuesSymbols Function](#)
- [fgGetPieChartGraduation Function](#)
- [fgGetGraduatedBy Function](#)
- [fgGetGraduatedScaling Function](#)
- [fgGetSymbolColor Function](#)
- [fgGetSymbolColorRange Function](#)
- [fgGetOpacity Function](#)
- [fgGetSymbol Function](#)

Other functions

- [setToolTipDuration Function](#)
- [getMapWidth Function](#)
- [getMapHeight Function](#)
- [getUIComponentVisibility Function](#)
- [getUIComponentPosition Function](#)
- [getActiveTool Function](#)

Version 1.0 - June 30 2009

- First Release of the SDK



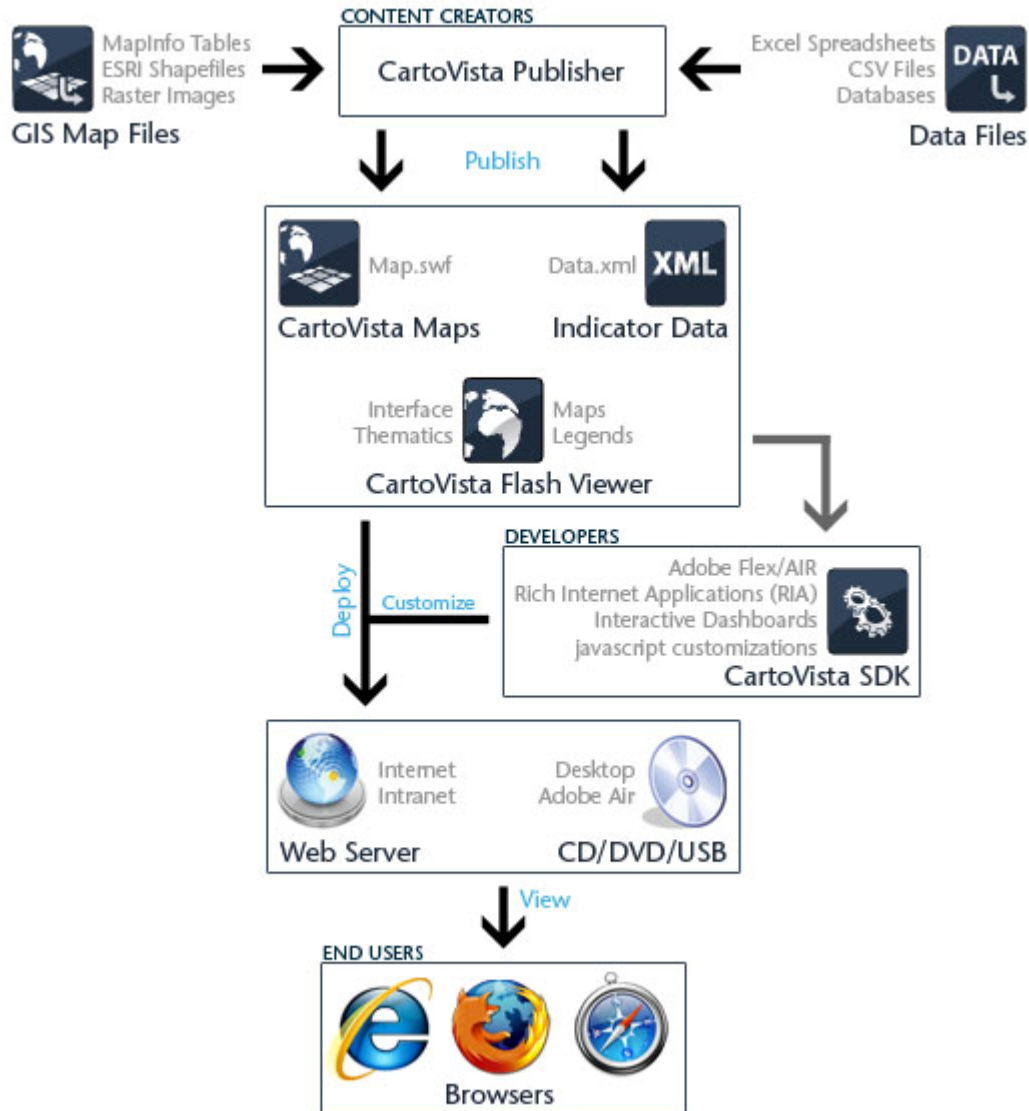
Introduction

Description

The CartoVista SDK is a set of standard Application Programming Interface (API) functions for developers to properly extend the CartoVista solution with technologies such as Adobe Flex or standard HTML scripting (Javascript).

For example, using the API, developers can implement their own methods of defining a CartoVista thematic map. A Flex developer could easily develop a form with a drop down list where the user can pick a predefined theme. Using the API, the developer can configure every thematic parameters that correspond to the user choice.

CartoVista SDK Architecture



Extending and Customizing CartoVista

In particular, the Adobe Flex technology includes a wealthy set of components to create rich internet applications. The SDK can also be used to leverage Flex and extend CartoVista. The SDK and Flex combination greatly simplifies the development of custom dashboard application. The SDK API functions can be setup to respond to user actions such as selecting objects form the map. This allows implementing powerful complementary graphs, charts and windows as part of a CartoVista deployment.



Generate the data for display in CartoVista

The documentation section of the XML data from CartoVista in this document provides the technical details needed to generate data for display in CartoVista.

This section is useful for users who want to integrate dynamic data to CartoVista module without publication.

The documentation contains all the elements and attributes in addition to XML Schemas XSD.

Other configuration settings CartoVista 2.0 (FlashVars)

[This section](#) will present various parameters Flash (FlashVars) that allow CartoVista also be configured without using the SDK API functions. It should be noted that the use FlashVars variables is limited. To enrich CartoVista it is preferable to use the SDK API functions.



Building Dashboard Applications with CartoVista

Flex is a free, open source framework for building highly interactive, expressive web applications that deploy consistently on all major browsers, desktops, and operating systems. It provides a modern, standards-based language and programming model that supports common design patterns. MXML, a declarative XML-based language, is used to describe user interface (UI) layout and behaviors, and ActionScript™ 3, a powerful object-oriented programming language, is used to create client logic. Flex also includes a rich component library with more than 100 proven, extensible UI components for creating rich Internet applications (RIAs), as well as an interactive Flex application debugger.

RIAs created with Flex can run in the browser using Adobe Flash® Player software or on the desktop on Adobe AIR™, the cross-operating system runtime. This enables Flex applications to run consistently across all major browsers and on the desktop. And using AIR, Flex applications can now access local data and system resources on the desktop. Both Flash Player and Adobe AIR are available as a free download on <http://www.adobe.com>.

CartoVista can be embedded in any Flex application, allowing to integrate advanced thematic mapping capabilities. With the SDK, Flex developers can customize the map displays and develop high quality dashboard windows.

CartoVista 2.0 - SWC Component

The SDK API includes a CartoVista 2.0 SWC component that allows you to easily integrate card CartoVista in Flex and benefit the environment for development with ease. For more information on the implementation of the component Adobe Flex Builder 3.0, see the section [Starting with the API](#).

Configuring CartoVista with Flash variables and no scripting

The viewer can also CartoVista be parameterized and configured for loading with Flash variables (FlashVars). For more information, see [Configuration Settings For CartoVista \(FlashVars\)](#).



Starting with the SDK

SDK Documentation

Each function of the SDK API presents a brief description and details of parameters and values required. Each parameter is documented to understand the use and discover its usefulness. In addition to the description of parameters and possible values each function is also a concrete example of code with Adobe Flex and Javascript.

Developing with Adobe Flex Builder

Please follow the following steps to use the SDK API in Flex CartoVista Builder 3.0:

1) Install the *CartoVista.swc* component

To develop with Adobe Flex Builder, you must first install the component ***CartoVista.swc*** in the library folder of your Flex project. Each project Flex includes a directory named ***libs***. This directory contains the SWC components. Please copy the file *CartoVista.swc* to this directory.

2) Setting up the script containing HTML (*index.template.html*)

To use the SDK API, it is necessary to set up a script Standard (Javascript) that allows CartoVista to communicate with Flex and exhibit Certain functions (Online help for example). In your Flex project, the directory *html-template* contains a file named *index.template.html* which is the HTML output (container) of the Flex application. It is necessary to add the following script.

JS

HTML Contents (Javascript)

SCRIPT:

```

<script src="Scripts/RoboHelp_CSH.js" type="text/javascript"></script>
<script language="JavaScript" type="text/javascript">
<!--
////////////////////////////////////
//This function gets called by CartoVista when it is loaded....
var CartoVistaIsReady = false;
function onCartoVistaLoadComplete() {
    CartoVistaIsReady = true;
    thisMovie("${swf}").onCartoVistaLoadComplete();
}
////////////////////////////////////
//This function gets called by CartoVista when the map is loaded....
function onDataLoadComplete() {
    thisMovie("${swf}").onDataLoadComplete();

```

```

}
////////////////////////////////////
//This function gets called by CartoVista everytime the selection changes...
function onSelectionChanged(selectedRecords) {
    thisMovie("${swf}").onSelectionChanged(selectedRecords);
}

////////////////////////////////////
//This function gets called by CartoVista everytime an object is click with the info tool...
function onInfoSelected(selectedItemId, selectedClassId, selectedLayerId) {
    thisMovie("${swf}").onInfoSelected(selectedItemId, selectedClassId, selectedLayerId);
}

////////////////////////////////////
//This function gets called by CartoVista when a new category file is loaded....
function onCategoryFileLoadComplete() {
    thisMovie("${swf}").onCategoryFileLoadComplete();
}

////////////////////////////////////
//This function gets called by CartoVista every time a tooltip should be shown and the
//callback has been activated with setToolTipCallBack(true)
function onShowToolTip(selectedItemId) {
    thisMovie("${swf}").onShowToolTip(selectedItemId);
}

////////////////////////////////////
var pageReady = false;
function init() {
    pageReady = true;
}
function isFlashReady() {
    if (thisMovie("${swf}")) {
        return (pageReady && CartoVistaIsReady);
    }
    else {
        return false;
    }
}
function thisMovie(movieName) {
    if (navigator.appName.indexOf("Microsoft") != -1) {
        return window[movieName];
    }
    else {
        return document[movieName];
    }
}
// -->
</script>

```

HTML BODY:

```
<body scroll="no" onLoad="init()">
```

3) Hello World Map Example

The example below shows how to include a single card in Flex using CartoVista component.

Adobe Flex Sample MXML Application

```
<?xml version="1.0" encoding="utf-8"?>
<mx:Application xmlns:mx="http://www.adobe.com/2006/mxml" layout="absolute" xmlns:cartovista="co
<mx:Script>
<![CDATA[
import com.dbx.cartovista.CVLanguages;
private function onClick():void {
CartoVista.setRollOverColor(0x00FFFF);
CartoVista.setLanguage(CVLanguages.French);
}
]}>
</mx:Script>
<mx:Panel x="10" y="10" width="675" height="648" layout="absolute" id="CVPanel">
<cartovista:CartoVista x="0" y="0" width="100%" height="100%" id="CartoVista"/>
</mx:Panel>
<mx:Button x="870" y="30" label="Button" click="onClick()"/>

</mx:Application>
```

Terms and references to the Flex documentation examples

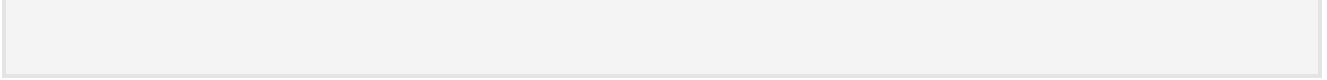
The sample code in the Flex SDK API functions are created by considering that the name of the component (id) from CartoVista in the MXML application is "CartoVista".

Adobe Flex

//CartoVista is the name of the component in the Flex project, as in the MXML code:

```
<cartovista:CartoVista x="0" y="0" width="100%" height="100%" id="CartoVista"/>
```

```
//Call the function to specify the English language
CartoVista.setLanguage(CVLanguages.English);
```





Functions

NOTE: *New SDK functions for this version of the CartoVista Viewer (2.1.9) are identified in the following tables with a blue background.*

Interface and Global Functions

The following functions are used to set global CartoVista settings and configuration options.

Function	Description
setLanguage	Set the interface language.
setSize	Set the map size.
enableMouseEvents	Enable or disable mouse events.
setSWFMap	Load a new swf map.
setRollOverColor	Set the rollover color for thematic layers.
setTitleVisibility	Set the title on or off.
setSubTitleVisibility	Set the subtitle on or off.
setZoomAnimationVisibility	Set the zoom animation visibility.
setToolTipDuration	Set the tooltip display duration (in seconds)

Data and Selections Functions

The data and selection functions are used to get information about the state of the selection in a CartoVista map. Use these functions to get the list of selected objects, data and indicator metadata.

Function	Description
bgGetSelectedItemsID	Get a list of the currently selected objects ID.
bgGetSelectedItemsData	Gets the attribute data for the current selection.
bgSetSelectedItems	Set the selection in CartoVista.
bgGetIndicatorMetadata	Get a list of the current indicator metadata (Indicator data type, description, source, etc.).
setIndicatorData	Set the indicator in CartoVista. This method permits sending a table of data instead of working with the XML data.
setSelectionModeAdditive	Sets the selection mode to be additive or not

Thematic Mapping Functions

The thematic mapping functions allow controlling every parameter of the CartoVista thematic analysis. You can use these functions to build your own thematics interfaces. For example, you may want to create a drop-down menu of pre-defined themes. When the user selects an item from the list box, CartoVista API calls are made to build a specific background and foreground theme.

Background Theme Functions

Function	Description
bgSetThematicLayer	Set the active thematic layer for the background.
bgGetThematicLayer	Get the current thematic layer for the background
bgSetRangeTheme	Sets the ranges thresholds and colors in one function call.
bgSetIndicatorCategoryByID	Set the indicator category to load for the background theme.
bgGetIndicatorCategoryID	Get the current indicator category for the background.
bgSetIndicatorByID	Set the indicator to display in the background.
bgGetIndicatorID	Get the indicator currently displayed in the background
bgRemoveIndicator	Remove indicator.
bgSetNumberOfRanges	Set the number of ranges in the background theme.
bgGetNumberOfRanges	Get the number of ranges in the background theme.
bgSetRangeThresholds	Set the range thresholds.
bgGetRangeThresholds	Get the current range thresholds
bgSetRangeColorScheme	Set the CartoVista color scheme to use for ranges.
bgGetRangeColorSchemeID	Get the CartoVista color scheme in use for ranges.
bgSetRangeColors	Set custom range colors.
bgGetRangeColors	Get the custom range colors.
bgSetIndividualValuesColors	Set individual values colors.
bgGetIndividualValuesColors	Get the color values in use for an individual values theme.

Foreground Theme Functions

Function	Description
fgSetThematicLayer	Set the active thematic layer for the foreground.
fgGetThematicLayer	Get the current thematic layer for the foreground.
fgSetIndicatorCategoryByID	Set the indicator category to load for the foreground theme.
fgGetIndicatorCategoryID	Get the current indicator category for the foreground.
fgSetIndicatorByID	Add an indicator in the foreground theme (Symbol or Pie Chart).
fgGetIndicatorsID	Get the list of indicators displayed in the foreground.
fgRemoveIndicators	Remove one or more indicators from the foreground.
fgSetPieChartColors	Set the pie chart theme colors.
fgGetPieChartColors	Get the pie chart theme colors.
fgSetIndividualValuesColors	Set the symbol individual values colors.
fgGetIndividualValuesColors	Get the colors and values used in the foreground theme.
fgSetAtValue	Set the symbol or pie chart graduation size value (At value).
fgGetAtValue	Get the symbol or graduation size value for the current foreground theme.

fgSetIndividualValuesSymbols	Set the symbols to associate with values on a symbolic theme.
fgGetIndividualValuesSymbols	Get the individual values symbols associated with a symbolic theme.
fgSetPieChartGraduation	Set pie chart to graduated or not.
fgGetPieChartGraduation	Get the current pie chart graduation.
fgSetGraduateBy	Set the symbol or pie chart graduation method.
fgGetGraduatedBy	Get the current symbol or pie chart graduation method.
fgSetGraduateScaling	Set the graduation Scaling.
fgGetGraduatedScaling	Get the current graduation scaling.
fgSetSymbolColor	Set the symbol color for symbolic theme.
fgGetSymbolColor	Get the symbol color for symbolic theme.
fgSetSymbolSize	Set the symbol size
fgSetSymbolColorRange	Set the symbol color ranges (from one color to another).
fgGetSymbolColorRange	Get the symbol color ranges (from one color to another).
fgSetOpacity	Set the foreground theme opacity.
fgGetOpacity	Get the foreground theme opacity
fgSetSymbol	Set the symbol shape for a symbolic theme.
fgGetSymbol	Get the symbol shape for a symbolic theme.

Layer Functions

Function	Description
layerSetVisibility	Set a map layer visible or not.
layerSetLabelling	Set a map layer labels visible or not.
layerSetOpacity	Set a map layer opacity.
layerSetClassVisibility	Set a map layer class visible or not.
layerSetClassLabelling	Set a map layer class labels visible or not.
layerSetClassOpacity	Set a map layer class opacity.
getMapLayersInfo	Get information about map layers
getLayerClassInfo	Get information about classes for a specific map layer

Map Navigation and Object Functions

Function	Description
getMapCoordinateSystemName	Get the name of the coordinate system used in the map.
getMapCoordinateSystemCode	Get the code (EPSG) of the coordinate system used in the map.
getMapUnits	Get the map distance units.
getMapZoom	Get the current map zoom (in the map distance units).
getMapCenter	Get the map center X and Y coordinate.
getMapWidth	Get the map width in pixels
getMapHeight	Get the map height in pixels

mapZoomTo	Zoom to a specific point and scale on the map.
mapRecenter	Recenter the map based on point coordinates (in the map coordinate system).
findObject	Find and zoom to a named object on the map.
setObjectColor	Sets the color of an object on the map

Window Functions

Function	Description
disableLegend	Set the legend visible or not.
disableSelectionWindow	Set the selection window visible or not.
setUIComponentVisibility	Set a particular CartoVista user interface component (legend, histogram, layer control, toolbar, etc.) visible or not.
getUIComponentVisibility	Get a particular CartoVista user interface component visibility setting (legend, histogram, layer control, toolbar, etc.)
setUIComponentPosition	Set the position (x,y) of a particular CartoVista component (legend, histogram, layer control, toolbar, etc.)
getUIComponentPosition	Get the position (x,y) of a particular CartoVista component (legend, histogram, layer control, toolbar, etc.)
setActiveTool	Set the active tool in the CartoVista toolbar
getActiveTool	Get the active tool in the CartoVista toolbar.
setPrintMode	Activate or deactivate the print mode for CartoVista windows and dialogs.



setLanguage Function

Category

[Interface and Global Functions](#)

Description

This function sets the CartoVista interface language.

Parameters: 1

1) *cvLanguageString (String)*

Values

String (Javascript)	Enum (Flex)	Description
"en"	CVLanguages.English	English
"fr"	CVLanguages.French	French
"es"	CVLanguages.Spanish	Spanish
"de"	CVLanguages.Deutsch	German

Examples

Fx *Adobe Flex*

```
//Call the function to set the language to french  
CartoVista.setLanguage(CVLanguages.French);
```

JS *Javascript*

```
//Call the function to set the language to french  
CartoVista.setLanguage("fr");
```




setSize Function

Category

[Interface and Global Functions](#)

Description

This function sets the CartoVista width and height dimensions in pixels. This is useful to get proper rendering with the Flex <swfloader> element. This function should be called in Flex whenever the swfloader container is resized.

Parameters: 2

1) *width (Number)*

2) *height (Number)*

Examples



Adobe Flex

```
//Call the function to set the size of the CartoVista map.  
CartoVista.setSize(800,600);
```



Javascript

```
//Call the function to set the size of the CartoVista map.  
CartoVista.setSize(800,600);
```




enableMouseEvents Function

Category

[Interface and Global Functions](#)

Description

This function disables or enables mouse events in CartoVista. This is useful for Flex applications what need to display a component on top of the map and remove the events.

Parameters: 1

1) *cvEnabled (Boolean)*

Examples

Fx *Adobe Flex*

```
//Call the function to disable mouse events in CartoVista  
CartoVista.enableMouseEvents(false);
```

JS *Javascript*

```
//Call the function to disable mouse events in CartoVista  
CartoVista.enableMouseEvents(false);
```




setSWFMap Function

Category

[Interface and Global Functions](#)

Description

This function sets the SWF map to load in CartoVista

Parameters: 1

1) *cvSWFMap (String)*

The name/path to the SWF map to load.

Examples



Adobe Flex

```
//Call the function to change the map  
CartoVista.setSWFMap("World.swf");
```



Javascript

```
//Call the function to change the map  
CartoVista.setSWFMap("World.swf");
```




setRollOverColor Function

Category

[setRollOverColor Function](#)

Description

This function sets the rollover color for thematics.

Parameters: 1

1) cvColor (Number)

A color expressed in hexadecimal.

Examples



Adobe Flex

```
//Call the function to set the rollover color  
CartoVista.setRollOverColor(0xff00ff);
```



Javascript

```
//Call the function to set the rollover color  
CartoVista.setRollOverColor(0xff00ff);
```




setThemeColor Function

Category

[Interface and Global Functions](#)

Description

This function sets the CartoVista interface theme color. For best results, a dark color should be used.

Parameters: 1

1) cvColor (Number)

A color expressed in hexadecimal.

Examples



Adobe Flex

```
//Call the function to set the theme color  
CartoVista.setThemeColor(0xff00ff);
```



Javascript

```
//Call the function to set the theme color  
CartoVista.setThemeColor(0xff00ff);
```




setTitleVisibility Function

Category

[Interface and Global Functions](#)

Description

This function sets the map title visible or not.

Parameters: 1

1) *cvVisible (Boolean)*

Examples

Fx *Adobe Flex*

```
//Call the function to set the title invisible  
CartoVista.setTitleVisibility(false);
```

JS *Javascript*

```
//Call the function to set the title invisible  
CartoVista.setTitleVisibility(false);
```




setSubTitleVisibility Function

Category

[Interface and Global Functions](#)

Description

This function sets the map subtitle visible or not.

Parameters: 1

1) *cvVisible (Boolean)*

Examples

Fx *Adobe Flex*

```
//Call the function to set the subtitle invisible  
CartoVista.setSubTitleVisibility(false);
```

JS *Javascript*

```
//Call the function to set the subtitle invisible  
CartoVista.setSubTitleVisibility(false);
```




setZoomAnimationVisibility Function

Category

[Interface and Global Functions](#)

Description

This function sets the animation on or off then the user zoom on the map. The animation scales the map vector objects when the user is zooming.

Parameters: 1

1) *cvZoomAnimation (Boolean)*

Examples

Fx *Adobe Flex*

```
//Call the function to disable the zoom animation  
CartoVista.setZoomAnimationVisibility(false);
```

JS *Javascript*

```
//Call the function to disable the zoom animation  
CartoVista.setZoomAnimationVisibility(false);
```




setToolTipDuration Function

Category

[Interface and Global Functions](#)

Description

This function is used to set the number of seconds a tooltip should be displayed when the end user mouse over the objects on the map.

Parameters: 1

cvDuration (Number)

The duration in seconds.

Examples

Adobe Flex

```
//Call the function to set the tooltip duration to 2.5 seconds  
CartoVista.setToolTipDuration(2.5);
```

Javascript

```
//Call the function to set the tooltip duration to 2.5 seconds  
CartoVista.setToolTipDuration(2.5);
```




setToolTipCallback Function

Category

[Interface and Global Functions](#)

Description

Enables the custom display of infotips. When enabled, an event is dispatched via the [onShowToolTip callback Function](#).

Parameters: 1

1) *cvSetToolTipCallback* (*Boolean*)

Samples

Fx *Adobe Flex*

```
//Enable the tooltip callback...  
CartoVista.setToolTipCallback(true);
```

JS *Javascript*

```
//Enable the tooltip callback...  
CartoVista.setToolTipCallback(true);
```




bgGetSelectedItemsID Function

Category

[Data and Selections Functions](#)

Description

This function returns an array containing the ID of the currently selected map objects in the background.

Parameters: None

Returns

cvSelectedItemsIDArray

Examples



Adobe Flex

```
//First declare an array to receive the data
var cvSelectedItemsID:Array;

//Call the function to get the list of ID's for the current selection in the array
cvSelectedItemsID=CartoVista.bgGetSelectedItemsID();
```



Javascript

```
//First declare an array to receive the data
var cvSelectedItemsID= new Array();

//Call the function to get the list of ID's for the current selection in the array
cvSelectedItemsID=CartoVista.bgGetSelectedItemsID();
```