

# Integrate Google Maps with Cognos 8

## An example implementation

Skill Level: Intermediate

[Rayudu VYSS \(rayudu.vecha@in.ibm.com\)](mailto:rayudu.vecha@in.ibm.com)  
System Software Engineer  
IBM

[Sridhar Munireddy \(smunired@in.ibm.com\)](mailto:smunired@in.ibm.com)  
Staff Software Engineer  
IBM

20 May 2010

The Google Maps application is commonly used by people around the world. Many prefer to use this application over other custom provided maps. Even though IBM® Cognos® has its own maps as a powerful feature, some users like to see their data integrated or represented using open source tools such as Google Maps. In this article, follow a step-by-step guide to see how to integrate Google Maps in Cognos reports, providing a way for you to represent tabular data in a spatial context.

## Introduction

As a Cognos user, you can now utilize the power of maps when representing your data and take advantage of open source technologies such as Google Maps. In this article, integrate Cognos data into Google Maps, and generate a report to represent tabular data in a spatial context. You can obtain a Google map view for any geography, choose the desired location, and open reports related to that particular location.

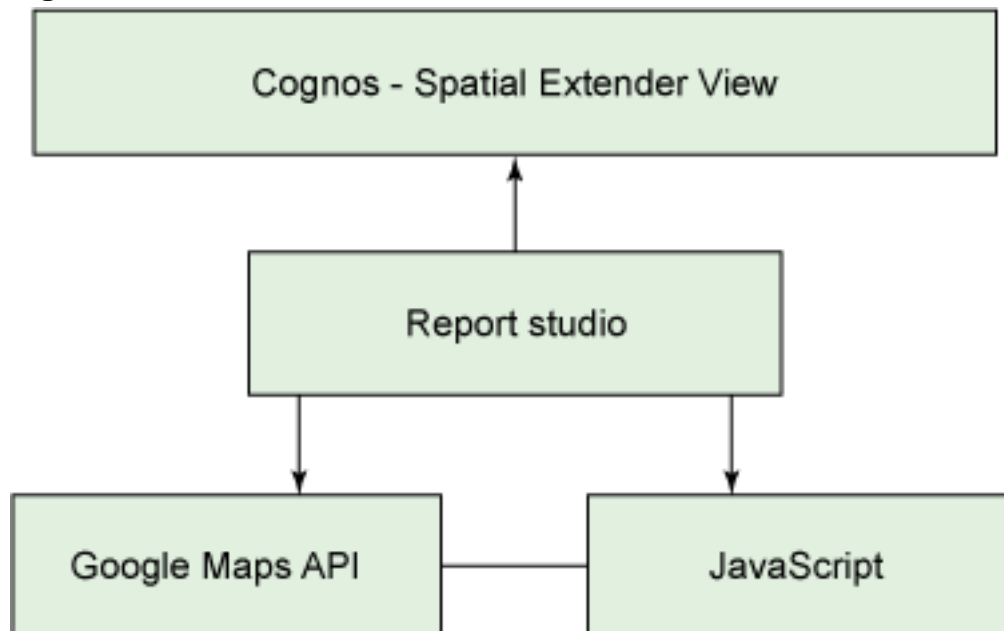
This article targets intermediate-level users who are Cognos report authors, and have experience in creating and designing reports based on the end-user requirements.

## Prerequisites

- Cognos BI Server 8.4 (on any operating system)
- Cognos report URL to be integrated into Google Maps

This article provides a step-by-step process for using the Google Maps API in the Cognos Report Studio. [Figure 1](#) illustrates the basic architecture of this process. It shows how the report objects in a Cognos 8 BI report, through the JavaScript, integrate with Google Maps using the Google Map API.

**Figure 1. Basic architecture**



## Generate a unique key for Google Maps API

The code and JavaScript used in this article (found in [Listing 1](#)) is based on the Google Maps API. You should read the terms and condition of the Google Maps API code before downloading the key and using this script.

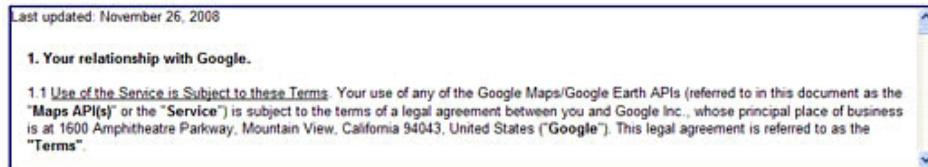
1. Log into your Google account.
2. Sign up for the Google Maps API using the following URL:  
<http://code.google.com/apis/maps/signup.html>  
You will be taken to the Google Maps API page, as shown in [Figure 2](#):  
**Figure 2. Signing up for Google Maps API**

### Sign Up for the Google Maps API

The Google Maps API lets you embed Google Maps in your own web pages. A single Maps API key is valid for a single "directory" or domain. See this [FAQ](#) for more information. You must have a [Google Account](#) to get a Maps API key, and your API key will be connected to your Google Account.

Here are some highlights from the terms for those of you who aren't lawyers:

- **There is no limit on the number of page views you may generate per day using the Maps API.** See this [FAQ](#) for more information.
- **There is a limit on the number of geocode requests per day.** See this [FAQ](#) for more information.
- **The Maps API does not include advertising.** If we ever decide to change this policy, we will give you at least 90 days notice via the [announcements lists](#).
- If you use other APIs in conjunction with the Maps API, you should also review the terms for the other APIs. Note in particular that the [GoogleBar](#) in the JS Maps API uses the AJAX Search API, and that API has its own terms.
- **Your service must be [freely accessible](#) to end users.** To use Google mapping technology in other types of applications, please use [Google Maps API Premier](#). See this [FAQ](#) for more information.
- **You may not alter or obscure the logos or attribution on the map.**
- You must [indicate whether your application is using a sensor](#) (such as a GPS locator) to determine the user's location.
- You may use the API (except for the Static Maps API) in websites or in software applications. For websites, please sign up with the URL where your implementation can be found. For other software applications, please sign up with the URL of the page where your application can be downloaded.
- Google will upgrade the APIs periodically. To be notified of updates, please subscribe to the [announcements lists](#).
- Remember that we reserve the right to suspend or terminate your use of the service at any time, so please read through the [FAQ](#) and [forum posts](#) to decide whether your site meets the Terms of Use before you begin API integration.



I have read and agree with the terms and conditions ([printable version](#))

My web site URL:

Tip: Signing up a key for `http://yourdomain.com` is usually the best practice, as it will work for all subdomains and directories. See this [FAQ](#) for more information.

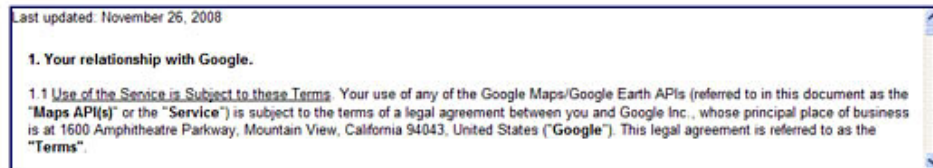
3. Enter the IP address of the server where the reports will be running, as illustrated in Figure 3:  
**Figure 3. Signing up for Google Maps API, and generating the API key**

## Sign Up for the Google Maps API

The Google Maps API lets you embed Google Maps in your own web pages. A single Maps API key is valid for a single "directory" or domain. See this [FAQ](#) for more information. You must have a [Google Account](#) to get a Maps API key, and your API key will be connected to your Google Account.

Here are some highlights from the terms for those of you who aren't lawyers:

- **There is no limit on the number of page views you may generate per day using the Maps API.** See this [FAQ](#) for more information.
- **There is a limit on the number of geocode requests per day.** See this [FAQ](#) for more information.
- **The Maps API does not include advertising.** If we ever decide to change this policy, we will give you at least 90 days notice via the [announcements lists](#).
- If you use other APIs in conjunction with the Maps API, you should also review the terms for the other APIs. Note in particular that the [GoogleBar](#) in the JS Maps API uses the AJAX Search API, and that API has its own terms.
- **Your service must be freely accessible to end users.** To use Google mapping technology in other types of applications, please use [Google Maps API Premier](#). See this [FAQ](#) for more information.
- **You may not alter or obscure the logos or attribution on the map.**
- You must [indicate whether your application is using a sensor](#) (such as a GPS locator) to determine the user's location.
- You may use the API (except for the Static Maps API) in websites or in software applications. For websites, please sign up with the URL where your implementation can be found. For other software applications, please sign up with the URL of the page where your application can be downloaded.
- Google will upgrade the APIs periodically. To be notified of updates, please subscribe to the [announcements lists](#).
- Remember that we reserve the right to suspend or terminate your use of the service at any time, so please read through the [FAQ](#) and [forum posts](#) to decide whether your site meets the Terms of Use before you begin API integration.



I have read and agree with the terms and conditions ([printable version](#))

My web site URL:

Tip: Signing up a key for `http://yourdomain.com` is usually the best practice, as it will work for all subdomains and directories. See this [FAQ](#) for more information.

4. Read the terms and conditions, and select the check box to agree with the terms and conditions (shown in [Figure 3](#)).
  5. Click on the **Generate API Key** button, as shown in [Figure 3](#), to generate a unique key. The unique key that is generated (see [Figure 4](#)) is only valid for the server that you entered earlier to generate the key, and it will be used in the script for working with the Google Maps API. Please remember that the key is only valid for a single Cognos server IP.
- Figure 4. Generated key**

## Sign Up for the Google Maps API

### Thank You for Signing Up for a Google Maps API Key!

Your key is:

```
ABQIAAAA-f503TdYod-I1sBbEyXn7BQo8trvBvRdgG00DhAEvRr3V1nTuhRDV8_xmFg2j7X1505cRLPYG6cQIA
```

**Note:** for more information on the API key system, consult <http://code.google.com/apis/maps/faq.html#keysystem>.

How you use your key depends on what Maps API product or service you use. Your key is valid for use within the entire family of Google Maps API solutions. The following examples show how to use your key within the Maps API product family.

### JavaScript Maps API Example

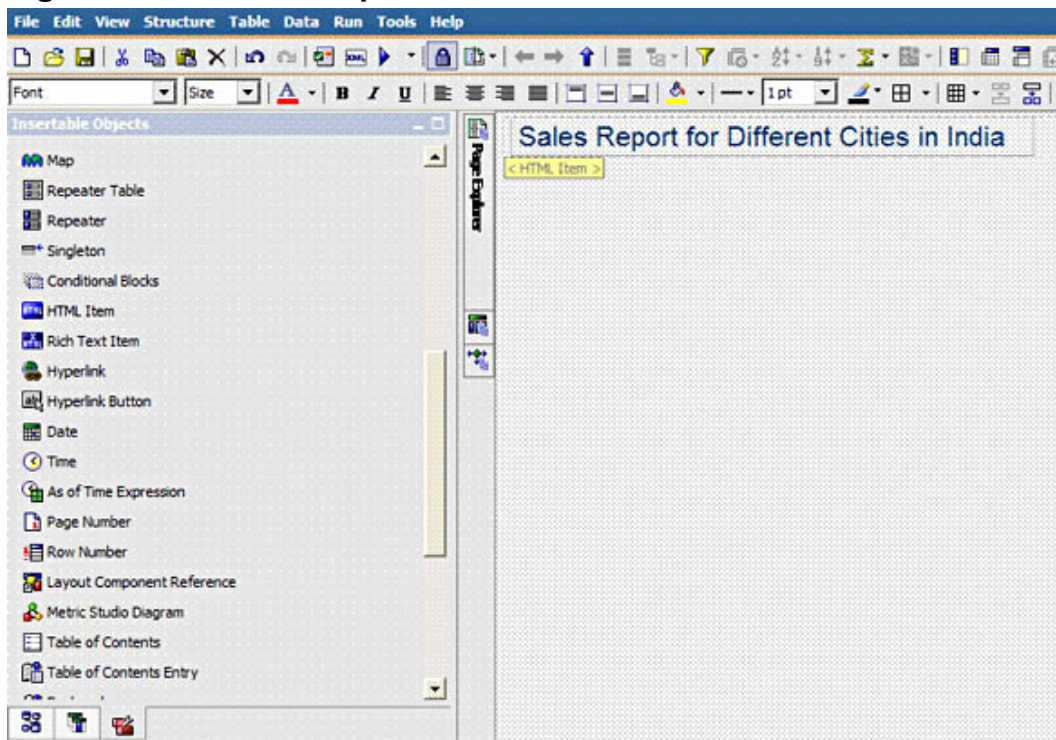
Within the JavaScript Maps API, place the key within the script tag when you load the API:

```
...  
// Note: you will need to replace the sensor parameter below with either an explicit true or false value.  
<script src="http://maps.google.com/maps?file=api&v=2&sensor=true_or_false&key=ABQIAAAA-f503TdYod-I1sBbEyXn7BQo8trvBvRdgG00DhAEvRr3V1nTuhRDV8_xmFg2j7X1505cRLPYG6cQIA">  
...  
</script>
```

See [Loading the Maps API](#) in the JavaScript Maps API documentation for more information.

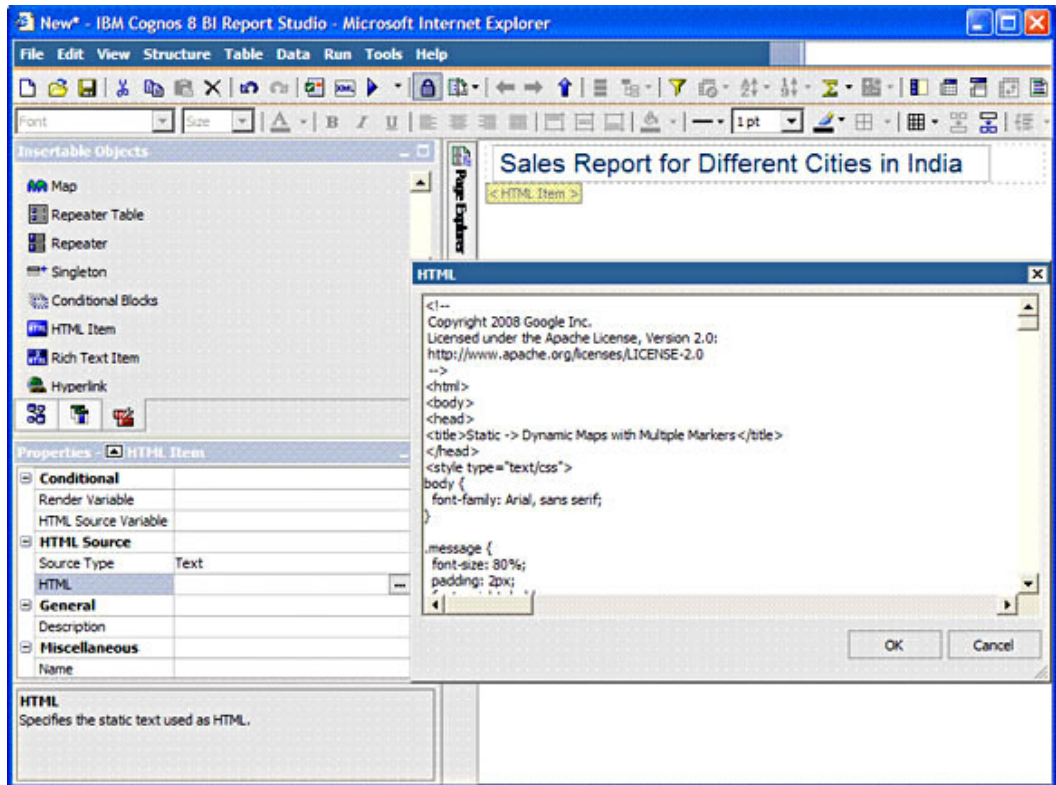
- 6. Create a new report from Report Studio, and drag an HTML item on to the report page, as shown in Figure 5:

**Figure 5. Create new report**

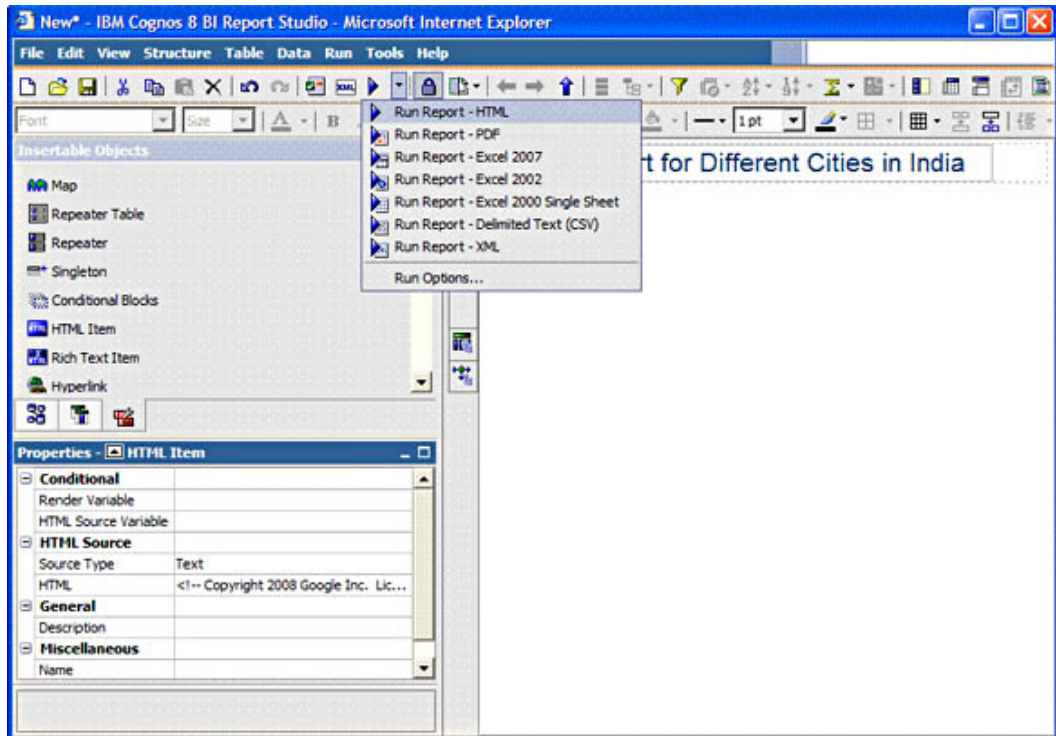


- 7. Edit the HTML item by adding the JavaScript shown in Listing 1 to the HTML source.

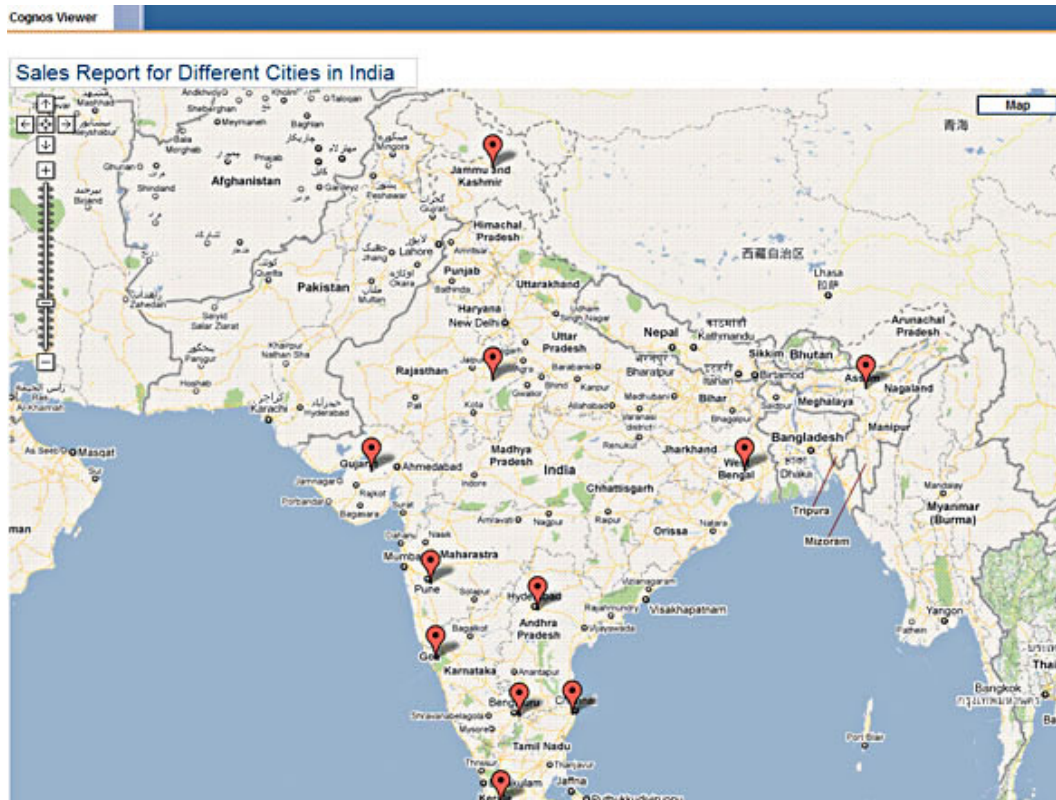
**Figure 6. Edit HTML source**



8. Run the Cognos report. Select **Run Report - HTML** from the menu drop down.
- Figure 7. Run report**



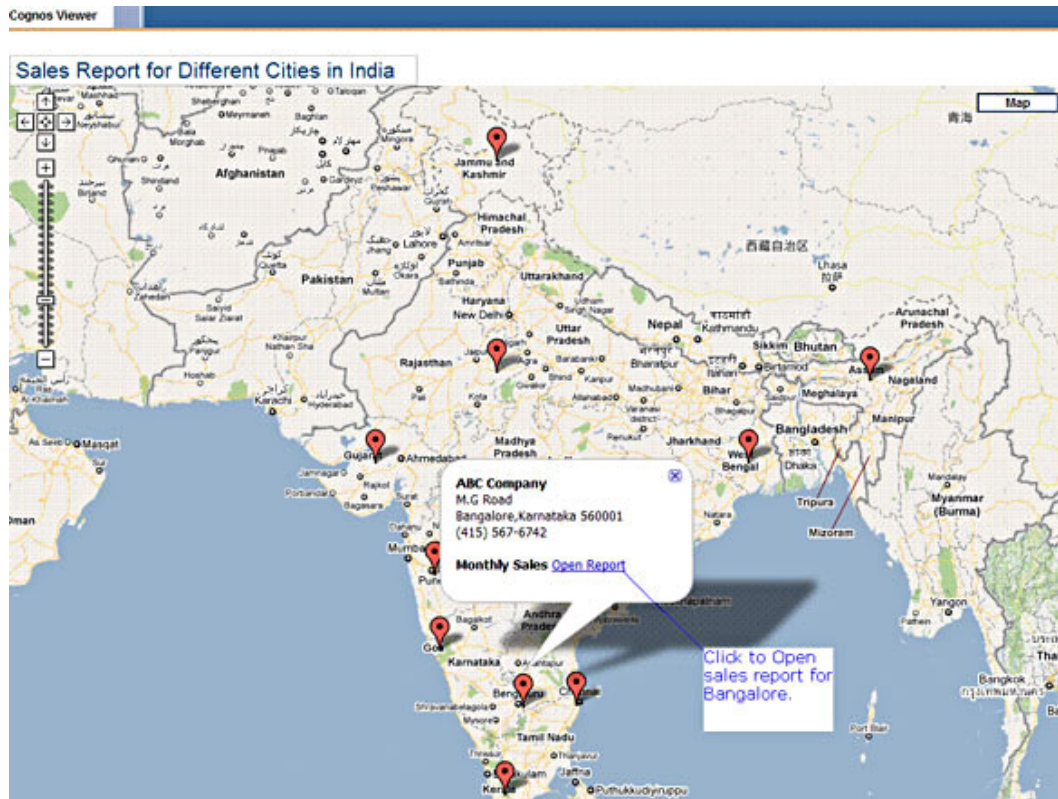
- 9. After running the report, you see the India map, as shown in Figure 8:  
**Figure 8. Map showing business locations in various cities**



The pre-populated red markers (or balloons) in the report represent the cities where various branches of the ABC Company are located. These cities are added in the script. You can add more cities or delete cities based on requirements.

10. Within the Google map, you can select the city for which you wish to see the sales report. Click on the red marker, and a pop-up window opens, showing which city was clicked, the address of that company, and a link to open the report.

**Figure 9. Business city selected**



11. When you click on the link to open the report, a new window opens, showing the monthly sales figures for that particular city. The city name is passed as a prompt value to the sales report, which is linked in the script. In this case, since we clicked on the Bangalore location, it shows the sales report for Bangalore, as illustrated in Figure 10:

**Figure 10. Sales report for business city Bangalore**



Cognos Viewer About

Sales Report for Bangalore from Jan 2009 to June 2009.

| Product Code | January      | February     | March       | April       | May         | June         | Total |
|--------------|--------------|--------------|-------------|-------------|-------------|--------------|-------|
| Product A    | 1000         | 1200         | 1100        | 1000        | 1100        | 1400         | 6800  |
| Product AB   | 800          | 900          | 1000        | 800         | 1000        | 1800         | 6300  |
| Product AC   | 1000         | 2000         | 1000        | 1200        | 1200        | 1200         | 7600  |
| Product AD   | 800          | 2000         | 1000        | 900         | 800         | 1100         | 6600  |
| Product AE   | 1000         | 2000         | 1000        | 800         | 900         | 800          | 6500  |
| Product AH   | 2000         | 2000         | 1000        | 1200        | 1100        | 800          | 8100  |
| Product AK   | 1000         | 2000         | 1000        | 1100        | 1200        | 900          | 7200  |
| Product BB   | 1600         | 2000         | 1000        | 900         | 1200        | 800          | 7500  |
| Product BC   | 1000         | 2000         | 1000        | 1100        | 900         | 1200         | 7200  |
| <b>Total</b> | <b>10200</b> | <b>16100</b> | <b>9100</b> | <b>9000</b> | <b>9400</b> | <b>10000</b> |       |

Likewise, it will show the sales report for other cities when selected.

### Listing 1. Sample JavaScript for integrating Cognos reports into Google Maps

```

<!-- Copyright 2008 Google Inc. Licensed under the Apache License, Version 2.0:
http://www.apache.org/licenses/LICENSE-2.0 -->
<html><body><head>
<title>Static -> Dynamic Maps with Multiple Markers</title>
</head>
<style type="text/css">

.message {
font-size: 80%;
padding: 2px;
font-weight: bold;
text-align: center;
width: 550px;
height: 450px;
background-color: #ffcc00;
border: 1px solid white;
position: relative;
}

/**
 * This is where we add the city names (any location) to be marked on the Google map
 * In the script, we have just mentioned only one city along with the latitude and
 * longitude details for that city. This can also be done dynamically, which is not
 * in the scope of this article.
 */

```

```

</style>
<script type="text/javascript">
var businesses = [
  {name: 'ABC Company',
  street: 'M.G Road',
  city: 'Bangalore',
  state: 'Karnataka',
  zip: '560001',
  country: '',
  phone: '(415) 567-6742',
  lat: 12.811801316582619, lng: 77.783203125}];

var map = null;
var bounds = null;
var currentMarker = null;
var mapDiv = null;
var containerDiv = null;
var clickedX = 0;
var clickedY = 0;
var isLoaded = false;

/**
 * Called after script is asynchronously loaded in.
 * Creates the GMap2, GMarker objects and performs actions according to
 * what the user did to trigger the map load (search, zoom, click etc).
 */

function loadMap() {
  if (GBrowserIsCompatible()) {
    mapDiv.style.background = '#fff';
    mapDiv.style.cursor = '';
    map = new GMap2(mapDiv, {logoPassive: true});
    bounds = new GLatLngBounds();
    for (var i = 0; i < businesses.length; i++) {
      bounds.extend(new GLatLng(businesses[i].lat, businesses[i].lng));
    }
    var latSpan = bounds.toSpan().lat();
    map.setCenter(bounds.getCenter(), map.getBoundsZoomLevel(bounds));
    map.addControl(new GLargeMapControl());
    //map.setZoom(2);
    var mapControl = new GMapTypeControl();
    map.addControl(mapControl);

    var newBounds = map.getBounds();
    var newLatSpan = newBounds.toSpan().lat();
    if (latSpan/newLatSpan > .90) { map.zoomOut(); }

    for (var i = 0; i < businesses.length; i++) {
      var marker = createMarker(i);
      var latlng = marker.getLatLng();
      var pixel = map.fromLatLngToDivPixel(latlng);
      if (Math.abs(pixel.x - clickedX) < 12 && Math.abs(pixel.y - clickedY) < 20) {
        //GEvent.trigger(marker, 'click');
      }
      map.addOverlay(marker);
    } } }

function zoomToAll() {
  map.setCenter(bounds.getCenter(), map.getBoundsZoomLevel(bounds));
}

/**
 * Creates a marker for the given business.
 * @param {Number} ind
 * @return {GMarker}
 */

function createMarker(ind) {

```

```

var value= 103037;
var business = businesses[ind];
var marker = new GMarker(new GLatLng(business.lat, business.lng));
/**
 * When the user clicks on the city, 'Business City' name is passed as a parameter
 * to the Cognos Report prompt. The URL in the 'href' below is the sales report
 * created in Cognos for different cities in India. Which will be shown when clicked
 * on the 'Open Report' for a particular city.
 */

GEvent.addListener(marker, 'click', function() {
    marker.html = ['<b>', business.name, '</b><br>',
                  business.street, '<br>', business.city, ',',
                  business.state, ' ', business.zip, '<br>',
                  business.phone, '<br><br>', '<b> Monthly Sales </b> ',

                  '<a href="http://9.26.96.75/cognos8/cgi-bin/cognos.cgi?b_action=cognosViewer&
    ui.action=run&ui.object=CAMID(%22BLUEPAGES%3au%3auid%3d735027744%2cc%3din%2cou%3
    dbluepages%22) %2ffolder[%40name%3d%27My%20Folders%27]%2ffolder[%40name%3d%27S%
    260P% 20Dashboards%27]%2ffreport[%40name%3d%27Sales%20Figures-%20Product%20
    -Code-quantity%27]&ui.name=Sales%20Products-%20Cities-%20India%20-Sales2Q2009
    &p_ordernumberp='+business.city+'&run.prompt=false" target="_blank ">
    Open Report</a>'].join('');

    currentMarker = marker;    (City Name passed to the sales report when selected)
    marker.openInfoWindowHtml(marker.html);
});
return marker;
}

/**
 * Formats business info into a URL-friendly version for maps url.
 * @param {Object} business
 * @return {String}
 */

function formatAddressForMaps(business) {
    var address = business.street + ' ' + business.city + ' ' + business.state + ' '
    + business.zip;
    return escape(address.replace(' ', '+'));
}

function _cel(elementType, id) {
    var element = document.createElement(elementType);
    element.id = id;
    return element;
}

/**
 * Loads in the Maps API script. This is called after some sort of user interaction.
 * The script loads asynchronously and calls loadMap once it's in.
 */
function loadScript() {
    if (!isLoading) {
        isLoading = true;
        var div = document.createElement('div');
        div.className = 'message';
        div.innerHTML = 'Loading...';
        div.style.left = (1200/2 - 53) + 'px';
        div.style.top = 1200/2 + 'px';
        mapDiv.appendChild(div);
        var script = document.createElement('script');
        script.type = 'text/javascript';
        script.src = 'http://maps.google.com/maps?file=api&v=2' +
        '&async=2&callback=loadMap&key=ABQIAAAA-f503TdYcd-IlsBbEyXn7BQ
        o8trwBvRdgGoOdHAeWrR3V1nTuhRDV8_xmPg2j7Xi50StRiPYG6cQIA';
        document.body.appendChild(script);
    }
}

```

```

/**
 * Sets up the gadget by setting CSS and click events.
 */
function loadMapGadget() {
  containerDiv = document.getElementById('container');
  mapDiv = document.getElementById('map');

  mapDiv.onclick = function (e) {
    clickedX = (window.event && window.event.offsetX) || e.clientX;
    clickedY = (window.event && window.event.offsetY) || e.clientY;
    loadScript();
  };

  mapDiv.style.cursor = 'pointer';
  var urlString = ['http://maps.google.com/staticmap?markers='];
  var markerString = [];
  for (var i = 0; i < businesses.length; i++) {
    markerString.push(businesses[i].lat + ',' + businesses[i].lng + ',red');
  }
  urlString.push(markerString.join('|'));
  urlString.push('&size=1200x800');
  urlString.push('&key= ABQIAAAA-f503TdYcd-I1sBbEyXn7BQo8trwBvRdgGoODhAEwRr
3V1nTuhRDV8_xmPg2j7Xi50StRiPYG6cQIA -----> Key
');
  mapDiv.style.background = 'url(\'\' + urlString.join(\'\') + '\')';
}
</script>
</head>
<body onload="loadMapGadget();" >
<div id="container">
  <div id="map" style="width: 1200; height: 800; overflow:hidden"></div>
</div>
</body>
</html>

```

## Conclusion

This article provided an example to show you how to use Google Maps in Cognos to link a sales report into a Google map for different cities in India. You should now know how to use the Google Maps API in Cognos reports to provide a more interactive user experience.

# Resources

## Learn

- [Google Mapd API documentation](#): Get an overview of new functionality and features as they are released on Google Maps. The documentation includes more examples apart from this article.
- [Google Maps Latitude, Longitude Popup](#): Find the latitude and longitude for any location in the Google map.
- [Cognos area on developerWorks](#): Get the resources you need to advance your Cognos skills.
- [developerWorks Information Management zone](#): Learn more about Information Management. Find technical documentation, how-to articles, education, downloads, product information, and more.
- Stay current with [developerWorks technical events and webcasts](#).

## Get products and technologies

- [JavaScript: Adding Google Maps to Cognos reports](#): Download a sample that shows how to add Google Maps to Cognos reports.
- Build your next development project with [IBM trial software](#), available for download directly from developerWorks.

## Discuss

- [Participate in the discussion forum for this content](#).
- Participate in [developerWorks blogs](#) and get involved in the [My developerWorks community](#); with your personal profile and custom home page, you can tailor developerWorks to your interests and interact with other developerWorks users.

## About the authors

### Rayudu VYSS



Rayudu Vecha has been working in India Software Lab for the past 3.5 years as part of Platform Technology Center team. Rayudu has a good working knowledge of DB2 Enterprise Server Edition, Cognos, and Optim.

## Sridhar Munireddy



Sridhar Munireddy has worked in the India Software Labs under Information Management for the past four years. He has worked with DB2, focusing on replication and federation technology, and now currently works on DB2 Health Advisor Service. He has hands-on experience with various reporting technologies, including Jasper Reports, BIRT, Alphablox, Crystal Reports, and IBM Cognos.