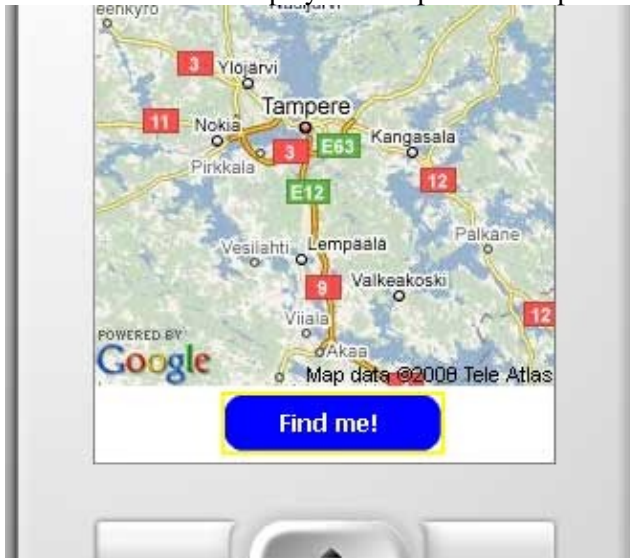




This article shows how to display current phone GPS position using [Flash Lite](#), [KuneriLite](#), and [Google Static](#)



Maps.

## Contents

- [1 Prerequisites](#)
  - ◆ [1.1 Get your own Google Maps API key](#)
  - ◆ [1.2 Download and install KuneriLite](#)
- [2 Create the Flash Lite application](#)
  - ◆ [2.1 Create your Flash Lite movie](#)
  - ◆ [2.2 Enter ActionScript code](#)
- [3 Test your Flash Lite application](#)
  - ◆ [3.1 Test on PC](#)
  - ◆ [3.2 Test on a real device](#)
- [4 Resources](#)

## Prerequisites

### Get your own Google Maps API key

NOTE: Usage of this code with the free Google Maps API Key breaks Google's [Terms and Conditions](#) (section 10.8). You should purchase an Enterprise License if you wish to use the Google Maps API as shown in this example.

## Displaying\_GPS\_position\_using\_Google\_Maps\_images\_in\_Flash\_Lite

To use Google Maps services, you need a Google Maps API key. If you do not have one, you can sign up for a key here: <http://code.google.com/apis/maps/signup.html>

## Download and install KunerLite

KunerLite is a toolkit that extends Flash Lite capabilities, allowing applications to access native Symbian OS functionalities, such as file writing or reading GPS data.

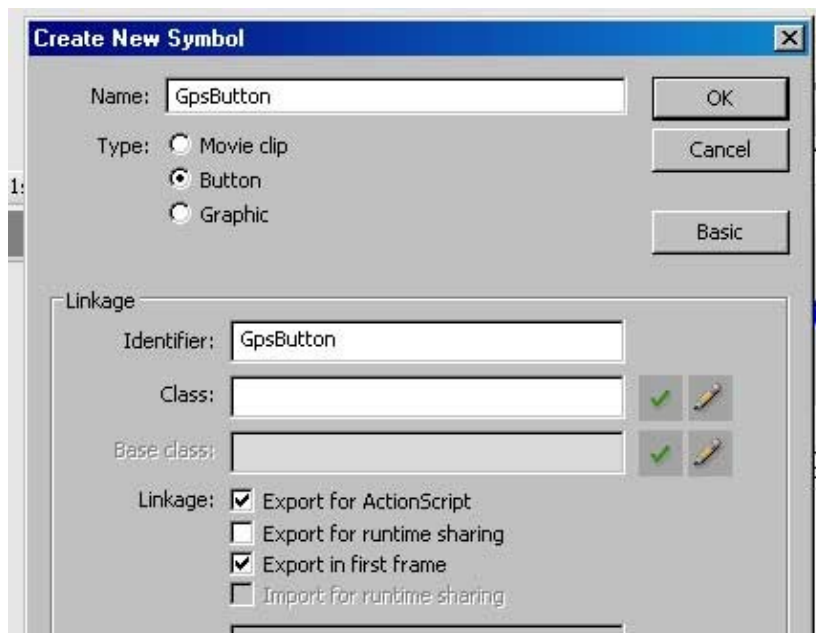
To proceed in this tutorial, download and install KunerLite from the [KunerLite download page](#).

## Create the Flash Lite application

### Create your Flash Lite movie

In this example, Flash Lite 2.1 is used. However, porting it to other (older or newer) Flash Lite versions is quite straightforward. So, after you have created an empty Flash Lite movie, do as follows:

- Create a button by going to **Insert > New Symbol...** and entering these properties:
  - ◆ Enter **GpsButton** as the name.
  - ◆ Mark the **Export for ActionScript** and **Export in first frame** check boxes.



- **Design the button.** You can, for example, place a big "Find me!" label on it.
- After you have designed the button, place it in the movie root, in the lower part of the stage as in the attached screenshot, and give **startButton** as the Instance Name.



### Enter ActionScript code

In the movie root, create a new layer called **Actions**, and open its ActionScript editor. First define the properties:

```
// Enter your api key here  
var apiKey = 'API_KEY';
```

```
//If you use the non-commercial version of Kunerilite, you do not need to change this  
var kuneriPath = 'http://127.0.0.1:1001/Basic/';
```

Next, define some useful functions to be used in the code:

```
//This function is called when some Kunerilite-related errors occur  
function kuneriError(error:String)  
{  
    trace("Kunerilite error: " + error);  
}  
  
//This function will do all calls to Kunerilite servers  
//and call the given handler passing response values as the argument  
function kuneriLoad(url, handler)  
{  
    var loader:LoadVars = new LoadVars();  
  
    loader.addEventListener(Event.COMPLETE, function()  
{  
    (this).callHandler(handler);  
});  
    loader.load(url);  
}
```

Create your Flash Lite movie

## Displaying\_GPS\_position\_using\_Google\_Maps\_images\_in\_Flash\_Lite

Next, code the button-related logic. When the user presses **startButton**, the application should:

- **Start the GPS.**
- **Retrieve the current GPS position.**
- **Display a map** centered in the retrieved GPS position.

To get full information about the Kunerilite GPS plug-in, check the related Wiki page:[http://wiki.kunerilite.net/index.php?title=GPS\\_plugin](http://wiki.kunerilite.net/index.php?title=GPS_plugin) The GPS is started when the gpsButton is pressed, using the **start klCommand**:

```
startButton.onPress = function()
{
    kuneriLite.callPath + 'GPS?klCommand=start', gpsStarted);
}
function gpsStarted(res:LoadVars)
{
    if(res.klError == 0 || res.klError == -11)
    {
        trace("GPS started");

        (kunerilite.callPath + 'GPS?klCommand=read', gpsDataRead);
    }
    else
    {
        ("Error: Starting GPS!");
    }
}
```

The `gpsStarted()` handler will:

- Check that there are no errors (`klError = 0`) or if the GPS is already started (`klError = -11`). For a full list of errors associated with the GPS plug-in, check the Kunerilite Wiki page:

[http://wiki.kunerilite.net/index.php?title=GPS\\_plugin](http://wiki.kunerilite.net/index.php?title=GPS_plugin)

- If there is an error in starting the GPS, call the `kuneriError()` function defined above.
- If the GPS is started correctly, it will make a second call to Kunerilite, this time to retrieve the current GPS position (`klCommand=read`)

The second call to Kunerilite will call the `gpsDataRead()` handler defined below:

```
function gpsDataRead(res:LoadVars)
{
    if(res.klError == 0)
    {
        var lat = res.klPosLatitude;
        var lng = res.klPosLongitude;

        trace("POSITION: " + lat + ", " + lng);

        (lat, lng).loadMap
    }
    else
    {
        ("Error: Starting GPS!");
    }
}
```

Enter ActionScript code

## Displaying\_GPS\_position\_using\_Google\_Maps\_images\_in\_Flash\_Lite

This handler, as above, checks if there are any errors raised by KumeriLite and if not, extracts latitude and longitude coordinates from the response's `klPosLatitude` and `klPosLongitude` properties. After this, it calls the `loadMap()` function that actually loads the **static map** image.

```
function loadMap(lat:Number, lng:Number)
{
var mapClip:MovieClip = _root.createEmptyMovieClip('mapClip', _root.getNextHighestDepth());

    mapClip.
    mapClip.

var mapWidth = 240;
var mapHeight = 280;

var loader:MovieClipLoader = new MovieClipLoader();

var mapUrl:String = 'http://maps.google.com/staticmap?center=' +
    ',' + lng lat&format=jpg&zoom=8&size=' +
    'x' +mapWidth+ 'x' +mapHeight + '&key=' + apiKey;

    loader.
}
}
```

The above function:

- Attaches a **new empty movie clip** to the movie root.
- Places it to coordinates (0,0).
- Uses a `MovieClipLoader` to load a 240x280 pixel map image in JPEG format in the empty clip.

After this, you can actually test your Flash Lite movie.

# Test your Flash Lite application

## Test on PC

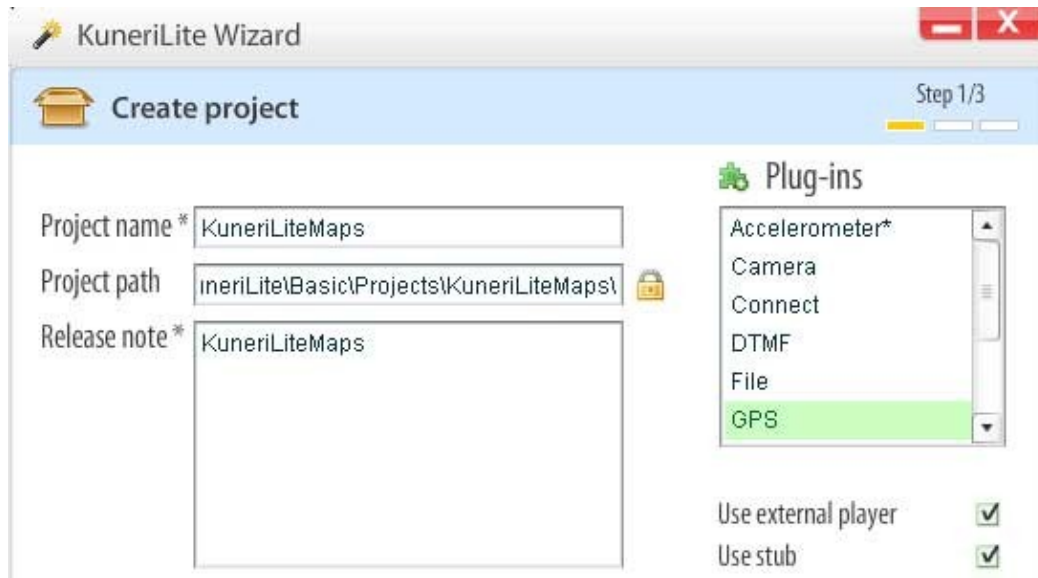
To test your application without deploying it on a real device:

- Start the **KumeriLite emulator** with the default settings (port: 1001, key: Basic).
- Start your **FlashLite movie**.
- Press **Find Me!** and wait for your image to load (the GPS position is, of course not real when testing on the emulator :))

For more information about the KumeriLite emulator, see [KumeriLite Emulator](#)

## Test on a real device

To test your application on a real device, package your SIS application using the **KumeriLite Wizard**:



- Export your FlashLite movie.
- Create a new KunerLite project.
- Enter the application name and other data. Select *GPS* from the available plug-ins.
- Select the "Use external player" option if you are developing for a development player (2.x or 3.x) and would like to launch the application using one of those players.
- It is also recommended to always select the "Use stub" option.
- Select the exported SWF as project **Main SWF**.

For more information about KunerLite Wizard, see the [KunerLite Wizard Beginner's Guide](#).

## Resources

- The FLA source of this tutorial is available for download here: [Media:KunerGpsMap.zip](#)
- For further information about using Google Maps in mobile applications, read this article: [How to use Google Maps data in mobile applications](#)