Archived:Accessing system resources with PySymbian on S60 3rd Edition devices

Aquivado: Este artigo foi arquivado, pois o conteúdo não é mais considerado relevante para se criar soluções comerciais atuais. Se você achar que este artigo ainda é importante, inclua o template {{ForArchiveReview|*escreva a sua justificativa*}}.

All PySymbian articles have been archived. PySymbian is no longer maintained by Nokia and is not guaranteed to work on more recent Symbian devices. It is not possible to submit apps to Nokia Store.

Introduction

A common problem in developing applications in Flash Lite or WRT is the poor access to S60 resources. These languages don't have access to GPS, Bluetooh, Contacts, SMS, etc in the S60 3rd Edition.

In this post we'll show a Flash Lite application which can access S60 3rd Edition internal resources using PySymbian.

Solution

This was done using the architecture and implementation showed in the following Archived:How to access Symbian resources in WRT or Flash Lite, using PySymbian.

Like was showed in the post, a client application has access to S60 internal resources throught HTTP requests. In ActionScript 2.0 (which is used in Flash Lite 2.x and 3.x) a HTTP request can be executed using the LoadVars class. The basic use of this class is showed below:

```
var lv:LoadVars = new LoadVars();
lv.onLoad = function(success:Boolean){
//Actions when data is loaded.
}
lv.sendAndLoad("http://www.iana.org/domains/example/",lv,"GET");
```

The parameters of sendAndLoad method are:

- *http://www.iana.org/domains/example/* - The url from which will load the data
- lv - Object with the data to send to the URL
- "GET" - The method to send the data, can be "GET" or "POST"

We'll use this class to access S60 internal resources when using the code showed in this post to access this S60 resources.

Code example

In the below code we show how to access the GPS position using this solution.

```
//Fullscreen
fscommand2("FullScreen",true);

//Create the load vars variable
var lv:LoadVars = new LoadVars();
//Create a MovieClipLoader object to load the google static map
var loader:MovieClipLoader = new MovieClipLoader();
//Generate a API_KEY and set here.
var apiKey =
'ABQIAAAAJq0bIh_b5seBuaS5dG3MyBTgIMgsbASCAp5eBSzKUN4OpxTWjxQe25ul58fmeLTrIy1RiBxMJ2gJ5w';
//When the GPS position is retrieved. The following method is called.
lv.onLoad = function(success:Boolean){
 if(success){
                //Set the UI components with the retrieved data
                //The retrieved data when you're accessing 'get_position' service are
the 'latitude' and 'longitude' of the device.
  latitude_txt.text = lv["latitude"].substring(0,8);
  longitude_txt.text = lv["longitude"].substring(0,8);
  var url:String = 'http://maps.google.com/staticmap?
center='+trim(lv["latitude"].substring(0,8))+','+trim(lv["longitude"].substring(0,8))+'&zoom=10&size=220x200&maptype=mobile&key='+apiKey;
                //Load a map with your position in the center
  loader.loadClip(url,map_mc);
 }else{
  trace("Error");
 }
}
//Start the work when press the 'LOAD MAP' button.
loadMap_btn.onPress = function(){
        //The url gived in sendAndLoad method has access to the self device (127.0.0.1,
to the port 5004 (This is the default port of the ServerPython) and the service
'get_position' (Retrieves the GPS position of the default module)
 lv.sendAndLoad("http://127.0.0.1:5004/get_position",lv,"GET");
}
```

Remember that you have to start 'spython' application before the Flash Lite access S60 resources.