Series 40: SVG resources localization

This article explains how to localize SVG resources in java midlet application. This is the next article after Game development for Series 40 : adding touch support in wiki article series Rapid game development for Series 40 with NetBeans Visual Designer and Game Builder.

Introduction

Attractive main menu can be created with use of Scalable 2D vector graphics. SVG menu has lot of advantages than any other approach:

1. it is scalable -- means you support any device automatically
2. it is pure artwork and potentially looks nice
3. it is compact and easier in implementation than Canvas use for graphic menu
4. you can get access to SVG elements and receive notification from them upon user interaction

In this article we are going to explore direct access to SVG elements in SVG document for text localization. Using one graphic presentation for range of languages that is amazing opportunity to speed-up development!

Before starting

Before going deep in mobile software localization problems, please go through amazingly clear and proved-to-be working tutorial How to create localized text resources created by grahamhughes . Having that localization approach in hand all we need is somehow to access SVG element by ID and alter it.

How to access SVG element in SVG document

SVG document is a XML based document follows scheme declared and maintained by W3C consortium. Every element can be given a string ID. java SVG supporting package allows to find element by ID. To alter its value trait "#text" is used. Please see function implementation below.

```
void localize(GameMIDlet midlet, String ID, int localizedID) {
    SVGElement text = (SVGElement) svgDocument.getElementById(ID);
    if(text != null){
        text.setTrait("#text", midlet.locText[localizedID]);
    }
}
```

This function can be found in TouchSVGMenu class implementation of fifteenpuzzle application project source code. In the code fragment `midlet.locText[localizedID]` call gets string resource from a resource array, instantiated in the midlet.

How to detect device language and select proper resource

Query the current locale of the device using "microedition.locale" attribute :

```
String locale = System.getProperty("microedition.locale");
```

the result string follows ISO_639-1 standard. With help of How to create localized text resources tutorial let's build real application, again, all referenced materials are in How to create localized text resources and fifteenpuzzle application project source code. Please find directory "tools" here where you can see text localization resources sources for languages English, Finnish and Russian, resource conversion tool and its source code.

Summary

This article demonstrates advantages of using Scalable Graphics in Series 40 applications. Please see screen shot below taken with Nokia Remote Devices service and fifteenpuzzle puzzle game application project

main screen localization English

main screen localization Finnish

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.