Changing Title Pane Text in QML applications

This article explains how to dynamically set the Title Pane text in a QML Application where StatusBar component is used from Qt Quick Components for Symbian.

Description

This article targets the applications that are targeting Symbian Platform written in QML and uses Qt Quick Components for Symbian. As defined by the Symbian Design Guidelines all the Symbian applications are advised to have Status Bar visible. This gives us the opportunity to use the Title Pane of the `StatusBar` to show context-dependent text according to the state of our application.

The interesting part

The `StatusBar` component contained in Qt Quick Components is not the real native status bar. It is just a custom component made to look and act exactly like the native Status Bar. In most cases no one will notice that fact. But when it comes to controlling it, a normal thought would be to use the Symbian C++ Title Pane API but playing around for a while you will see that nothing happens no matter what...this is the main difference.

The resolution is quite simple. Use the `StatusBar` component as if it was any other component and show on it the information you want like you would on an other QML element. This leads use to the following code:

Using Window element as base:

```
import QtQuick 1.1
import com.nokia.symbian 1.1

Window {
    property alias pageStack: stack
    property alias title: statusPaneTitle.text

    StatusBar {
        id: sbar
        anchors.right: parent.right
        anchors.left: parent.left
        anchors.top: parent.top

        Rectangle
        {
            anchors.left: parent.left
            anchors.leftMargin: 2
            anchors.verticalCenter: parent.verticalCenter
            width: sbar.width - 140
            height: parent.height
            clip: true;
            color: "#00000000"

            Text{
                id: statusPaneTitle
                anchors.verticalCenter: parent.verticalCenter
                maximumLineCount: 1
                x: 0
                color: "white"
            }

            Rectangle{
                width: 25
                anchors.top: parent.top
                anchors.bottom: parent.bottom
                anchors.right: parent.right
                rotation: -90

                gradient: Gradient{
                    GradientStop { position: 0.0; color: "#00000000" }
                    GradientStop { position: 1.0; color: "#ff000000" }
                }
            }
        }
    }

    Item {
        id: contentArea

        anchors.top: sbar.bottom
        anchors.bottom: tbar.top
        anchors.left: parent.left
        anchors.right: parent.right
        Item {
            id: contentItem
            anchors.fill: parent
            PageStack {
                id: stack
                anchors.fill: parent
                toolBar: tbar
            }
        }
    }

    ToolBar {
        id: tbar

        width: parent.width
        anchors.bottom: parent.bottom
        anchors.right: parent.right
        anchors.left: parent.left
    }

    Component.onCompleted: {
```


(C) Copyright Nokia 2013. All rights reserved.

```
        pageStack.push(Qt.createComponent("MainPage.qml"))
    }
}
```

Using PageStackWindow element as base:

```
import QtQuick 1.1
import com.nokia.symbian 1.1

PageStackWindow {
    id: window
    initialPage: MainPage {tools: toolBarLayout}
    showStatusBar: false
    showToolBar: true

    property alias title: statusPaneTitle.text

    StatusBar {
        id: sbar
        anchors.right: parent.right
        anchors.left: parent.left
        anchors.top: parent.top

        Rectangle
        {
            anchors.left: parent.left
            anchors.leftMargin: 2
            anchors.verticalCenter: parent.verticalCenter
            width: sbar.width - 140
            height: parent.height
            clip: true;
            color: "#00000000"

            Text{
                id: statusPaneTitle
                anchors.verticalCenter: parent.verticalCenter
                maximumLineCount: 1
                x: 0
                color: "white"
            }

            Rectangle{
                width: 25
                anchors.top: parent.top
                anchors.bottom: parent.bottom
                anchors.right: parent.right
                rotation: -90

                gradient: Gradient{
                    GradientStop { position: 0.0; color: "#00000000" }
                    GradientStop { position: 1.0; color: "#ff000000" }
                }
            }
        }
    }

    ToolBarLayout {
        id: toolBarLayout
        ToolButton {
            flat: true
            iconSource: "toolbar-back"
            onClicked: window.pageStack.depth <= 1 ? Qt.quit() : window.pageStack.pop()
        }
    }
}
```

Create a Rectangle element in the Status Bar containing a Text element that will hold the title. The width of the Rectangle is set accordingly to create boundaries for the title so that the Status Bar information will not get overlapped. This is why the **clip** property is set to true. To remain consistent with the looks of the native Title Pane we place a Gradient element at the end of the rectangle to make the clipped text fade out (in case it is too long).

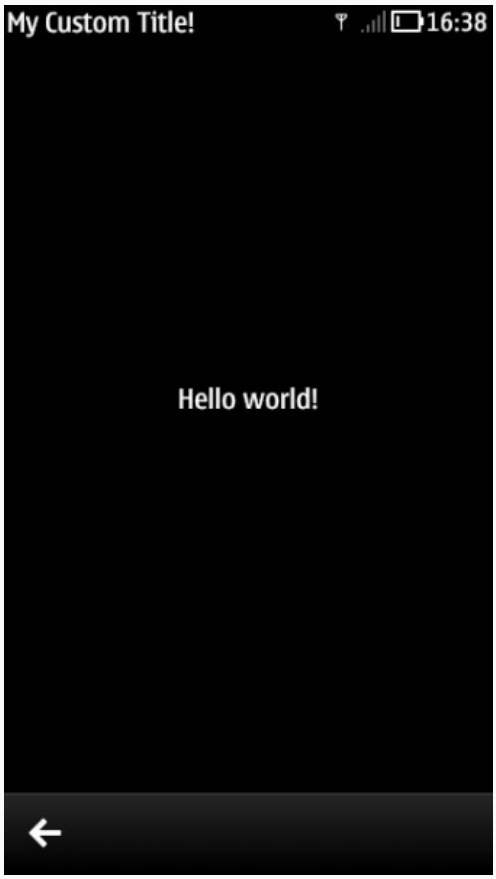
Now you are ready at any time and any point of your code to just do the following and you are done!

```
title = "My Custom Title!"
```

Ok, we haven't implemented the animation of the native Title Pane has (when long title is set) but for the time it is something minor and trivial.

Screenshots

(C) Copyright Nokia 2013. All rights reserved.



favoritas37 16:53, 30 July 2012 (EEST)

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.