

This article explains the use of the Flash Lite Menu component and API.

Contents

- [1 Introduction to the Menu Component](#)
- [2 Requirements](#)
- [3 Download](#)
- [4 Installation](#)
- [5 Preparations](#)
- [6 Structure](#)
 - ◆ [6.1 Assets](#)
 - ◆ [6.2 Skins](#)
 - ◆ [6.3 Inspectable Parameters](#)
- [7 Menu Component ActionScript API](#)
 - ◆ [7.1 Setting the assets path](#)
 - ◆ [7.2 Setting the skin path](#)
 - ◆ [7.3 Activating the component](#)
 - ◆ [7.4 Disabling the component](#)
 - ◆ [7.5 Insert Itens and Sub Itens](#)
 - ◆ [7.6 Manager Menu Itens](#)
 - ◆ [7.7 Change Label Alignment](#)
 - ◆ [7.8 Custom Open Sub Menu Animation](#)
 - ◆ [7.9 Events](#)

Introduction to the Menu Component

The Flash Lite Menu component is a flexible and scalable user interface component that allows Flash Lite developers to create mobile user interfaces easily, using Flash Lite.



Figure 1. Menu component.

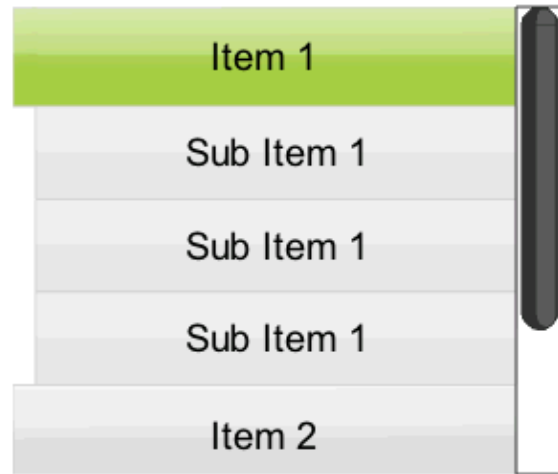


Figure 2. Menu component with Scrollbar.



Figure 3. Menu component over the skin.

Requirements

- ◇ Adobe Flash Professional CS3 or CS4
- ◇ Flash Lite 2.0 Player and above
- ◇ Touch Screen Device

Download

Menu component can be downloaded [here](#).

Installation

Installation of the Menu component is easy. Execute the components MXP file and follow the simple instructions in Adobe Extension Manager to complete the installation process. Restart Adobe Flash CS4 after installation.

Note: Component FLA and AS files should appear in the directories [INSTALL PATH]\Flash CS4\en\Configuration\Components\Forum Nokia, [INSTALL PATH]\Flash CS4\en\Configuration\Classes\com\forumnokia and [INSTALL PATH]\Flash CS4\en\Configuration\Classes\gs. In some setup environments, the files may appear under the wrong language directories (for example, fi instead of en). In such cases, the files should be manually moved to the correct directories as specified above.

Preparations

Menu_Component_for_Flash_Lite

1. Create a Flash Lite Project. Open the Component panel (Ctrl/Apple key + F7) and drag the Menu component onto the stage. Assign a unique instance name for the object using the properties panel (Ctrl/Apple key + F3).

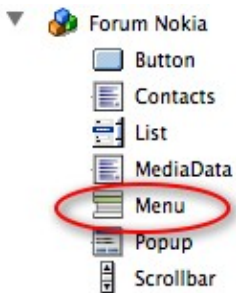


Figure 4. Nokia Components Tree

2. Forum Nokia Flash Lite UI components require the Flash Lite movie to be set to align to the top left corner of the device screen and not to scale. Some ActionScript code is needed to achieve this. The following code must be added to the first frame of the project:

```
fscommand2("DisableKeypadCompatibilityMode");
fscommand2("FullScreen", true);
fscommand2("SetQuality", "high");
Stage.scaleMode = "noScale";
Stage.align = "TL";
_focusrect = false;
```

3. Insert your menu hierarchy

```
//Params: (Label, Data)
myMenu.addRootItem("Item 1", 1);
myMenu.addRootItem("Item 2", "second");
myMenu.addRootItem("Item 3", 3);
myMenu.addRootItem("Item 4", {i:4});
myMenu.addRootItem("Item 5", [5]);

//Params: (Parent Index, Label, Data)
myMenu.addChildItemAt(0, "SubItem 1.1", 1.1);
myMenu.addChildItemAt(0, "SubItem 1.2", 1.2);
myMenu.addChildItemAt(0, "SubItem 1.3", 1.3);
myMenu.addChildItemAt(2, "SubItem 3.1", 3.1);
myMenu.addChildItemAt(2, "SubItem 3.2", 3.2);
myMenu.addChildItemAt(4, "SubItem 5.1", 5.1);
```

4. In order to catch the menu events, you need to add a listener to the menu onChange event:

```
var myChangeEvent:String = myMenu.onChangeEvent();
myMenu.addEventListener(myEvent, myListener);
function myListener(eventObject:Object):Void {
    if (eventObject.type == myChangeEvent) {
        trace("Item: "+eventObject.item);
        trace("Label: "+eventObject.label);
    }
}
```

Menu_Component_for_Flash_Lite

```
        trace("Data: "+eventObject.data);  
    }  
}
```

Structure

Assets

The asset of a menu component is composed of 2 different parts. Each asset has separate MovieClips for Up and Down states of the menu, which are organised in the following structure:

_assets

 Default

 Up

 RootLabel
 ChildLabel

 Down

 RootLabel
 ChildLabel

NOTE: Custom assets must follow the same folder structure, and linkage identifiers must be named accordingly.

Skins

The skin of a menu component is composed of 12 different parts for each state, up and down, as can be seen below. This structure enables the menu interface to be scalable and easily skinnable.

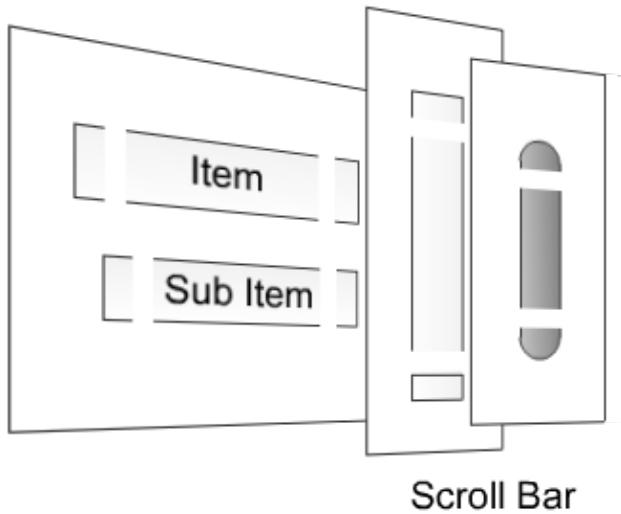


Figure 5. Skin Levels

Skins have separate MovieClips for Up and Down states of the Menu. The structure of these parts is defined below:

_skins

Default

Up

ScrollBack_up
ScrollBack_down
ScrollBack_center
Scroll_up
Scroll_down
Scroll_center
MenuRoot_right
MenuRoot_left
MenuRoot_center
MenuChild_right
MenuChild_left
MenuChild_center

Down

ScrollBack_up
ScrollBack_down
ScrollBack_center
Scroll_up
Scroll_down
Scroll_center
MenuRoot_right
MenuRoot_left
MenuRoot_center
MenuChild_right

Menu_Component_for_Flash_Lite

MenuChild_left
MenuChild_center

Note: Each new custom skin must follow the same folder structure, with linkage identifiers named accordingly.

Inspectable Parameters

Inspectable parameters help customise the Menu component from the Component Inspector panel (Shift + F7). All inspectable parameters can also be controlled via ActionScript with the component APIs.

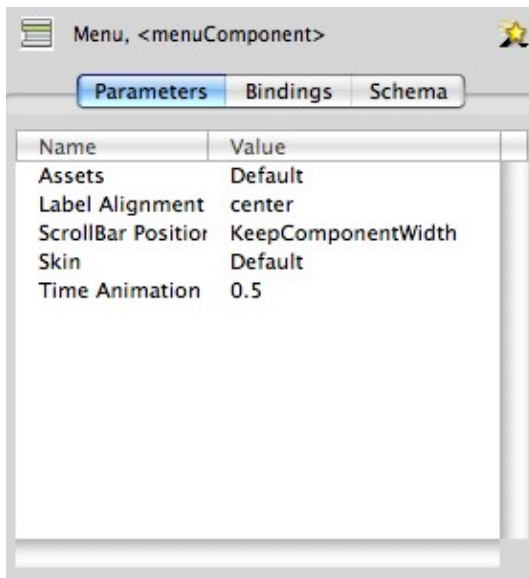


Figure 6. Component Inspector

Parameter	Description	Value
Skin	Defines the menu skin linkage identifier path. For instance, for the <i>mySkin</i> value, the menu would search for the skin components under the <i>Menu.mySkin.*</i> linkage path.	Skin identifier string
Assets	Defines the menu asset linkage identifier path. For instance, for the <i>myAssets</i> value, the component would search for the assets under the <i>Menu.myAssets.*</i> linkage path.	Skin identifier string
Label Alignment	Defines the text alignment of the label asset.	center/left/right

Menu_Component_for_Flash_Lite

ScrollBar Position	Defines the scroll bar position when it is visible.	KeepComponentWidth, ExpandComponentWidth, OverComponentSkin
Time Animation	Defines the time of open submenu animation.	Number

Menu Component ActionScript API

Setting the assets path

```
public function setAsset( path:String ): Void;
```

Sets a new path for the assets. Reconstructs the component.

Setting the skin path

```
public function setSkin( path:String ): Void;
```

Sets a new path for the skin. Reconstructs the component.

Activating the component

```
public function enableComponent(): Void;
```

Makes the component react to user input.

Disabling the component

```
public function disableComponent(): Void;
```

Makes the component stop reacting to user input.

Insert Itens and Sub Itens

```
public function addRootItem(label:String, data):Void;
```

Insert one Root Item on Menu Component.

```
public function addChildItemAt(rootIndex:Number, label:String, data):Void;
```

Insert the subItem that has the index given by the parameter.

NOTE: Each Item and Sub Item has an attribute data that can store any information to be used by the user

Manager Menu Itens

public function setSelectedItem(item:MovieClip):Void;

Set selected Item.

public function getSelectedItem():MovieClip;

Get selected Item.

public function unSelectItem():Void;

Unselect the selected Item

Change Label Alignment

public function alignLabelField(newPosition:String):Void;

Change the Itens and subItens label alignment.

Custom Open Sub Menu Animation

setEasingFunction(func:Function):Void

Set a custom easing function. You can use any easing equation written in ActionScript 2. Has many easing equations in [TweenLite Package](#)

Events

public function onChangeEvent():String;

Calls the menu component?s change event.

public function onPressEvent():String;

Calls the menu component's press an item event.

public function onReleaseItemEvent():String;

Insert Itens and Sub Itens

Menu_Component_for_Flash_Lite

Calls the menu component items release an item event.

public function onStartItemAnimationEvent():String

Calls the menu component's start animation event.

public function onStopItemAnimationEvent():String;

Calls the menu component's stop animation event.

public function onStartDragScrollBarEvent():String;

Calls the menu component's start drag scroll bar event.

public function onStopDragScrollBarEvent():String;

Calls the menu component's stop drag scroll bar event.