CS 116 – Action Script Event Handling

Elie Abi Chahine

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Event Handling

• Most programs support user interaction of some sort— whether it's something as simple as responding to a mouse click or something more complex, such as accepting and processing data entered into a form. Any such user interaction with your program is considered an event

• An event-handling system allows programmers to respond to user input and system events in a convenient way.

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Event Handling

• In ActionScript 3.0, each event is represented by an event object, which is an instance of the Event class or one of its subclasses. An event object not only stores information about a specific event, but also contains methods that facilitate manipulation of the event object. For example, when Flash Player detects a mouse click, it creates an event object (an instance of the MouseEvent class) to represent that particular mouse click event.

•You can “listen” for event objects in your code using event listeners. Event listeners are the functions or methods that you write to respond to specific events. To ensure that your program responds to events, you must add event listeners either to the event target or to any display list object that is part of an event object’s event flow.

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Event Handling

- Any time you write event listener code, it follows this basic structure (elements in red are placeholders you'd fill in for your specific case):

```
function eventResponse(eventObject_:EventType):void
{
	// Actions performed in response to the event go here.
}


eventTarget.addEventListener(EventType.EVENT_NAME, eventResponse);
```

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Event Handling

•The previous code does two things.

- ➢ First, it defines a function, which is the way to specify the actions that will be performed in response to the event.
- ➢ Next, it calls the addEventListener() method of the source object, in essence “subscribing” the function to the specified event so that when the event happens, the function’s actions are carried out.

• When the event actually happens, the event target checks its list of all the functions and methods that are registered as event listeners.

• It then calls each one in turn, passing the event object as a parameter.

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Handling Events

Example:

Let's start with a simple example. We have our ship on screen which is an instance of the Ship class(extended from MovieClip).
Everytime the mouse clicks on the ship, we want to display a sentence in the output window. Write the following in your .fla file:

```
function DisplayOnMouseClick(eventObject_:MouseEvent):void
{
	trace("clicked the mouse on the character");
}

var mcShip:Ship = new Ship();
mcShip.Initialize(150,200,100); /* This is a function that we created to initialize all the
                                   ship's properties at once in the beginning */
stage.addChild(mcShip)
mcShip.addEventListener(MouseEvent.CLICK, DisplayOnMouseClick);
```

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Event Handling

If we know in advance that we need this functionality for every ship we create, then the best way to go is to put it inside our Ship class.

```
/* put this function inside our class */
function DisplayOnMouseClick(eventObject_:MouseEvent):void
{
	trace("clicked the mouse on the character");
}

/* add this line of code inside our Initialize function*/
addEventListener(MouseEvent.CLICK, DisplayOnMouseClick);
```

Note: Just in case you want to import the Mouse Events functionalities

import flash.events.MouseEvent;

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Events Types

• We have many types of Events in ActionScript, and each one of them contains many functionalities.

• I'm going to cover the most important ones now, if you are interested you can check out the rest.

- ➢ Event
- ➢ MouseEvent
- ➢ KeyboardEvent

NB: The Mouse and Keyboard Events won't run unless the object is in focus. (we will have examples on this topic but I wanted to mention it more than once)

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Event

The "Event" object handler contains functionalities like:

• ***ENTER_FRAME:*** *Will run the function assigned every frame(every game loop)*

•***EXIT_FRAME:*** *Will run the function assigned every time it exits a frame*

•***ADDED:*** *Will run the function assigned every time an object is added as a child*

•***ADDED_TO_STAGE:*** *Will run the function assigned every time an object is added as a child to the stage.*

•***REMOVED:*** *Will run the function assigned every time an object is removed*

•***REMOVED_FROM_STAGE:*** *Will run the function assigned every time an object is removed from the stage*

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

MouseEvent

The “Event” object handler contains functionalities like:

• ***CLICK:*** *Will run the function assigned every time the mouse is clicked.*

•***MOUSE_DOWN:*** *Will run the function assigned every time the mouse button is down.*

•***MOUSE_UP:*** *Will run the function assigned every time the mouse button is up.*

•***MOUSE_MOVE:*** *Will run the function assigned every time the mouse moves.*

•***MOUSE_OVER:*** *Will run the function assigned every time the mouse is over.*

•***MOUSE_WHEEL:*** *Will run the function assigned every time the mouse wheel is used.*

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

KeyboardEvent

The “Event” object handler contains functionalities like:

• ***KEY_DOWN:*** *Will run the function assigned every time a key is down.*

•***KEY_UP:*** *Will run the function assigned every time a key is up.*

Note: **You can’t use KeyboardEvent inside your class or added to a MovieClip since the object needs to be in focus so that the listener will catch the event. That is why we add the keyboard events to the stage since it is always in focus.**

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Getting the attributes

Just in case you want to get the list of attributes when using the dot "." operator, You will have to import the following files:

Events	→	import flash.events.Event;
MouseEvents	→	import flash.events.MouseEvent;
KeyboardEvents	→	import flash.events.KeyboardEvent;
All of the above at once	→	import flash.events.*

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Exercice 1

Make the ship always rotate clockwise every frame

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Solution

```
/* put this function inside our class */
public function Update(e_:Event):void
{
	rotation++;
}

/* add this line of code inside our Initialize */
addEventListener(Event.ENTER_FRAME, Update);
```

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Keyboard Input

- As I mentioned in a previous slide, you can’t use KeyboardEvent inside your class (added to a MovieClip) since the object needs to be in focus so that the listener will catch the event.

- That is why we add the keyboard events to the stage since it is always in focus.

- Inside the keyboard event function, we need to check which key is pressed. For that we will need every key’s code to check with.

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

Keyboard Input

```
function input(ke_:KeyboardEvent)
{
	if(ke_.keyCode == Keyboard.UP) /* to get the Keyboard attribute list you need to
	{                                 import flash.ui.keyboard */
		mcShip.MoveUp();
	}

	if(ke_.keyCode == Keyboard.DOWN)
	{
		mcShip.MoveDown();
	}

	if(ke_.keyCode == Keyboard.LEFT)
	{
		mcShip.MoveLeft();
	}

	if(ke_.keyCode == Keyboard.RIGHT)
	{
		mcShip.MoveRight();
	}
}
stage.addEventListener(KeyboardEvent.KEY_DOWN, input);
```

Copyright © 2011, DigiPen Institute of Technology. All Rights Reserved

The End ☺