UNIVERSIDAD POLITECNICA DE VALENCIA

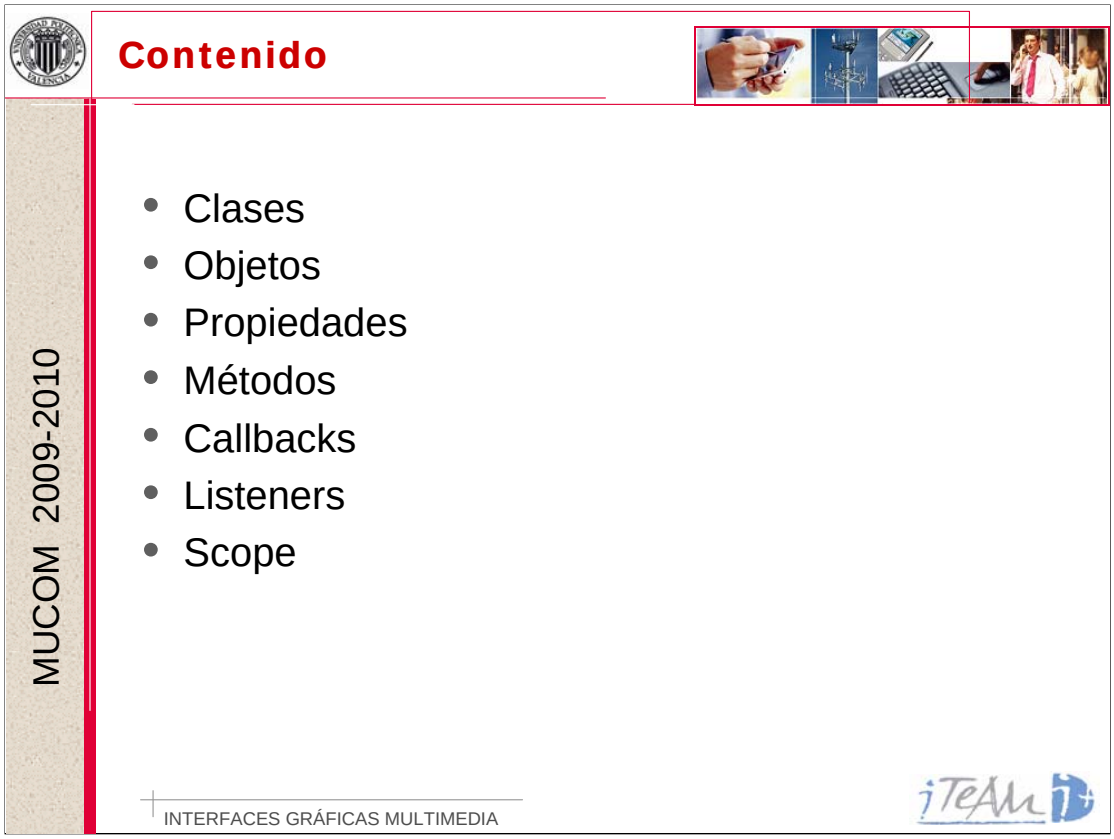
MASTER EN COMUNICACIONES Y DESARROLLO DE SERVICIOS MÓVILES

2009-2010

INTERFACES GRÁFICAS MULTIMEDIA

Conceptos de ActionScript

iTeAM Instituto de Telecomunicaciones y Aplicaciones Multimedia



En este tema se incidirá en algunos términos que permitirán conocer los fundamentos del lenguaje ActionScript, un lenguaje de programación orientada a objetos que puede funcionar en ocasiones como lenguaje de script. Esta es una forma de programación que cada vez tiene más adeptos, debido a las grandes posibilidades que supone unir los dos estilos de programación "modernos" – la estructurada y la orientada a objetos.

Contenido

MUCOM 2009-2010

- Clases ►
- Objetos
- Propiedades
- Métodos
- Eventos
- Callbacks
- Listeners
- Scope

Clases

MUCOM 2009-2010

- Datos que definen un objeto

  Flash (AS2) incluye ~65 clases de alto nivel.

  En Flash Lite (2.x) podemos emplear ~30 de estas clases.

`Array`	`Button`	`Color`	`Date`	`Key`
`LoadVars`	`Math`	`Mouse`	`MovieClip`	`Selection`
`SharedObject`	`Sound`	`Stage`	`String`	`System`
`TextField`	`TextFormat`	`Video`	`XML`	`XMLNode`

Una clase es un tipo de datos que define un objeto; como si fuera a una plantilla para crear objetos de cierto tipo. Las clases son el *backbone* de ActionScript. Flash incluye aproximadamente 65 clases de alto nivel que ofrecen desde tipos de datos básicos (`Array`, `Boolean`, `Date`, etc.) hasta errores y eventos personalizados, además de formas de cargar contenido externo (XML, imágenes, datos binarios, etc.). En Flash Lite podemos usar aproximadamente 30 de las clases de Flash.

Es posible escribir nuestras propias clases personalizadas e integrarlas en las aplicaciones, o incluso ampliar las clases de alto nivel y añadir otra funcionalidad o modificar alguna funcionalidad existente.

Contenido

MUCOM 2009-2010

- Clases
- Objetos ►
- Propiedades
- Métodos
- Eventos
- Callbacks
- Listeners
- Scope

Objetos

MUCOM 2009-2010

- Conjunto de datos con propiedades y métodos
  - (y eventos)

En ActionScript todo son objetos.

Algunas clases tienen asociados determinados eventos para sus objetos.

En AS2 existen los modificadores `public`, `private` y `static`.

En AS3, además, existen `protected` e `internal`.

Objeto Math		
propiedades	**métodos**	
`E`	`abs`	`log`
`LN10`	`acos`	`max`
`LN2`	`asin`	`min`
`LOG10E`	`atan`	`pow`
`LOG2E`	`atan2`	`random`
`PI`	`ceil`	`round`
`SQRT1_2`	`cos`	`sin`
`SQRT2`	`exp`	`sqrt`
	`floor`	`tan`

Objeto Stage		
propiedades	**métodos**	**eventos**
`align`	`addListener`	`onResize`
`height`	`removeListener`	
`scaleMode`		
`width`		

Un objeto es un conjunto de datos que contiene propiedades y métodos. En Flash todo son objetos, lo cual hace de él un entorno ideal para diseñadores y desarrolladores. Ejemplos de objetos son un campo de texto, un sonido e incluso la línea de tiempo o la Biblioteca.

Las clases, además, pueden desencadenar eventos; lo cual es muy útil en ciertas situaciones – por ejemplo, para importar una imagen desde un servidor Web nos puede interesar el estado de la descarga, o si hubo algún error de conexión.

En AS2 podemos asignar los siguientes modificadores tanto a las propiedades como a los métodos de clase:

- `public` Accesible desde fuera del ámbito de la clase.
- `private` Accesible solamente desde dentro de la propia clase.
- `static` Accesible desde clases que no se instancian (ej: `Math`).

En AS3 existen dos modificadores adicionales:

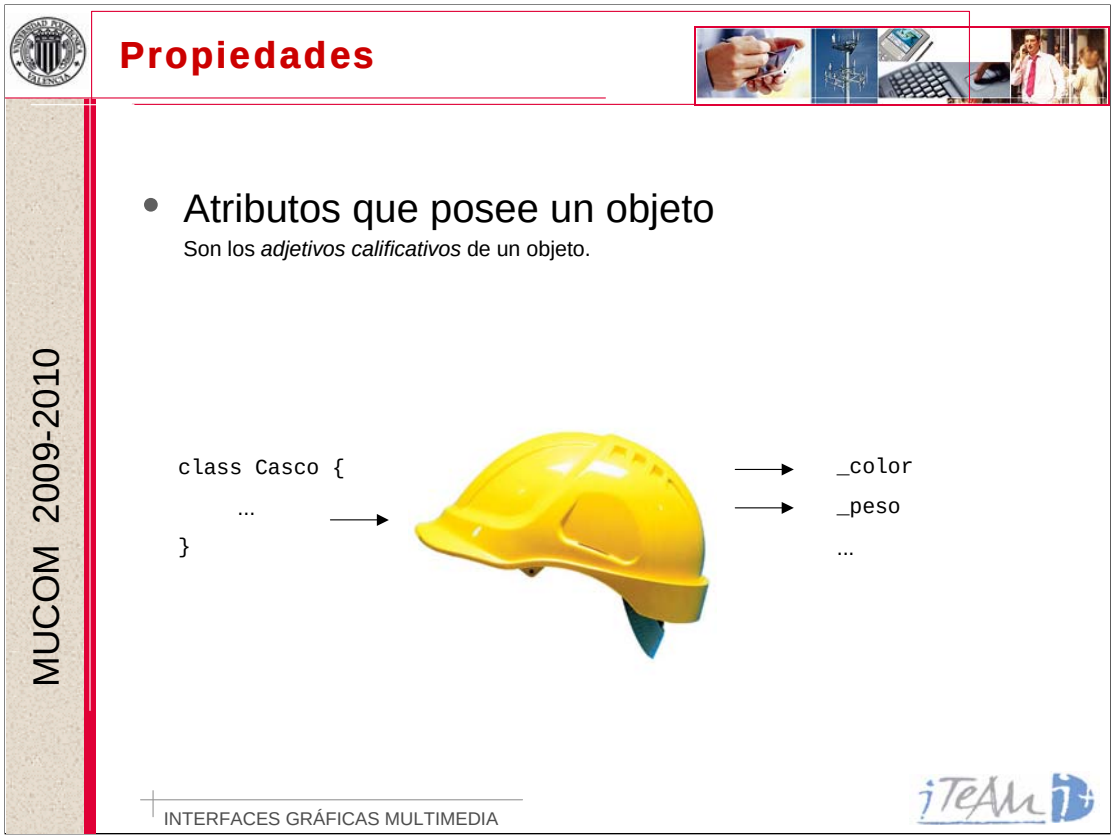
- `protected` Accesible desde desde clases heredadas.
- `internal` Accesible desde clases que están en el mismo `package`.

En AS2 los constructores de clases no llevan modificadores, pero en AS3 pueden ser `public` o `internal`.

Contenido

MUCOM 2009-2010

- Clases
- Objetos
- Propiedades ►
- Métodos
- Eventos
- Callbacks
- Listeners
- Scope



Una propiedad es un adjetivo calificativo de un objeto. Es un mero dato que proporciona información, como por ejemplo la coordenada vertical de un gráfico o la longitud de un vector.

Contenido

MUCOM 2009-2010

- Clases
- Objetos
- Propiedades
- Métodos ►
- Eventos
- Callbacks
- Listeners
- Scope

Métodos

- Funciones que realiza un objeto

  Son los *verbos* de un objeto.

```
class Casco {
     ...
}
```

```
colocar();
arrojar();
...
```


Un método es un verbo del objeto. Es una mera función encargada de que el objeto haga algo, como por ejemplo devolver un número aleatorio, dejar de reproducir una animación o borrar un campo de texto.

Contenido

MUCOM 2009-2010

- Clases
- Objetos
- Propiedades
- Métodos
- Eventos ►
- Callbacks
- Listeners
- Scope

Eventos

- Sucesos

  Definen lo que puede ocurrir con algún objeto.

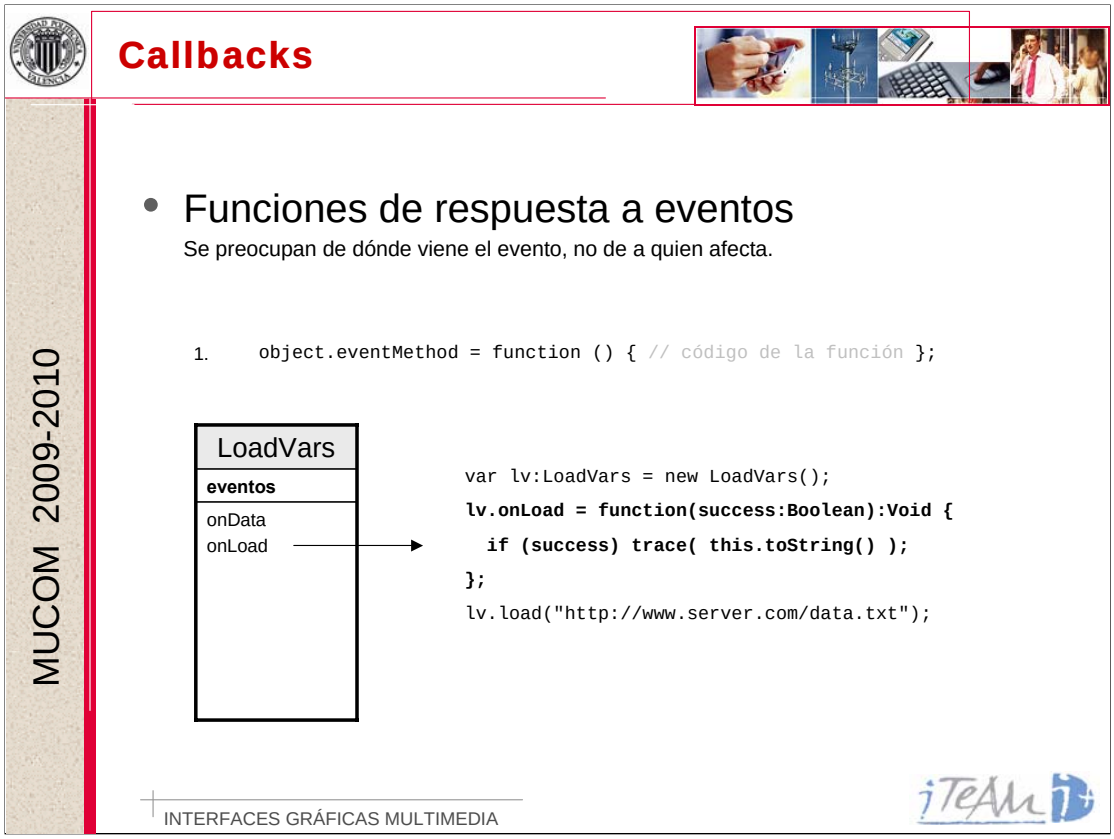
Clase LoadVars		
propiedades	**métodos**	**eventos**
contentType loaded	addRequestHeader decode getBytesLoaded getBytesTotal load send sendAndLoad toString	onData onLoad

Un evento es un suceso. Cuando ocurre algo con algún objeto, el evento nos informa y actúa en consecuencia si así lo hemos programado. Ejemplos de eventos son pulsar una tecla, entrar en un fotograma o mover el ratón.

Contenido

MUCOM 2009-2010

- Clases
- Objetos
- Propiedades
- Métodos
- Eventos
- Callbacks ►
- Listeners
- Scope



Un callback es una función de respuesta que se aplica directamente a un evento. Se preocupa de dónde viene el evento, no de a quien afecta. Un ejemplo de callback es este pseudo-código:

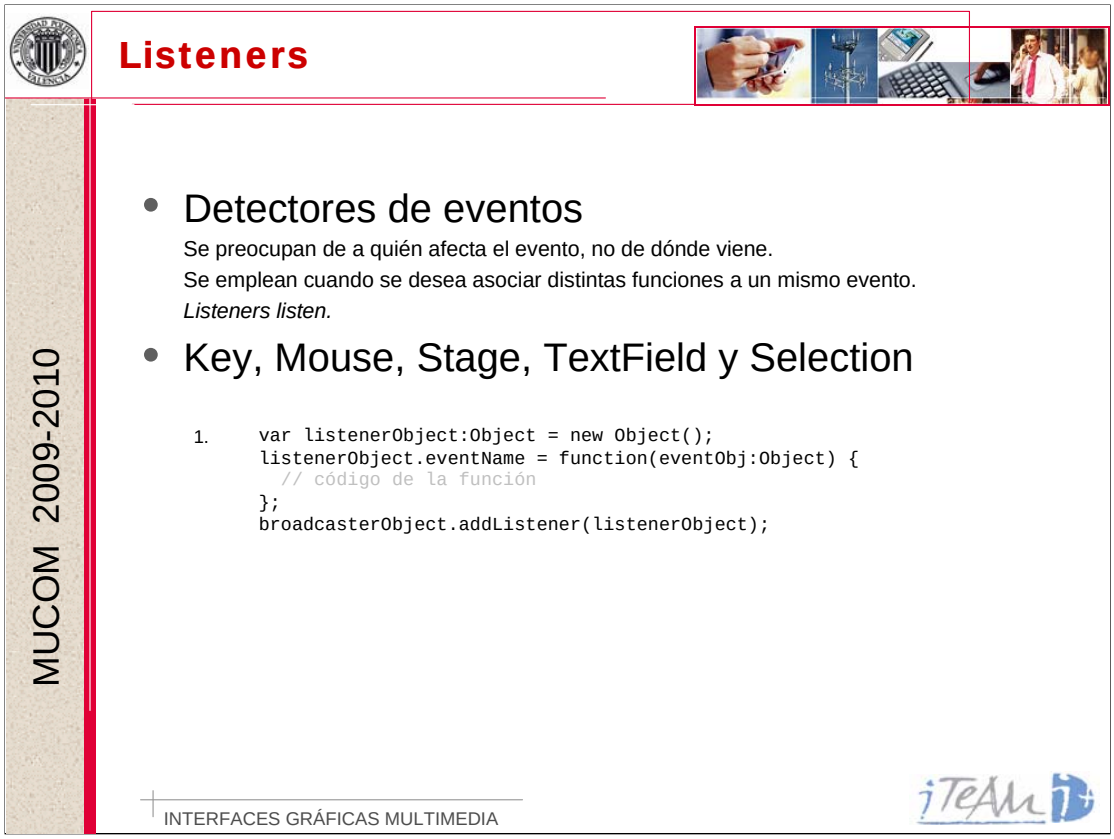
cuando ocurra el evento { hacer algo (callback) }

Normalmente un callback también se refiere al conjunto de la función + el evento sobre el que actúa, aunque estrictamente es sólo la función de respuesta.

Contenido

MUCOM 2009-2010

- Clases
- Objetos
- Propiedades
- Métodos
- Eventos
- Callbacks
- Listeners ►
- Scope



Un listener es un dato que asocia un evento a un objeto difusor (broadcaster). Se preocupa de a quién afecta el evento, no de dónde viene. En este pseudo-código se observa su diferencia con un callback:

crear un listener que reciba el evento x

cuando ocurra el evento x { ejecutar callback }

asociar el listener a un objeto broadcaster (por ejemplo al teclado)

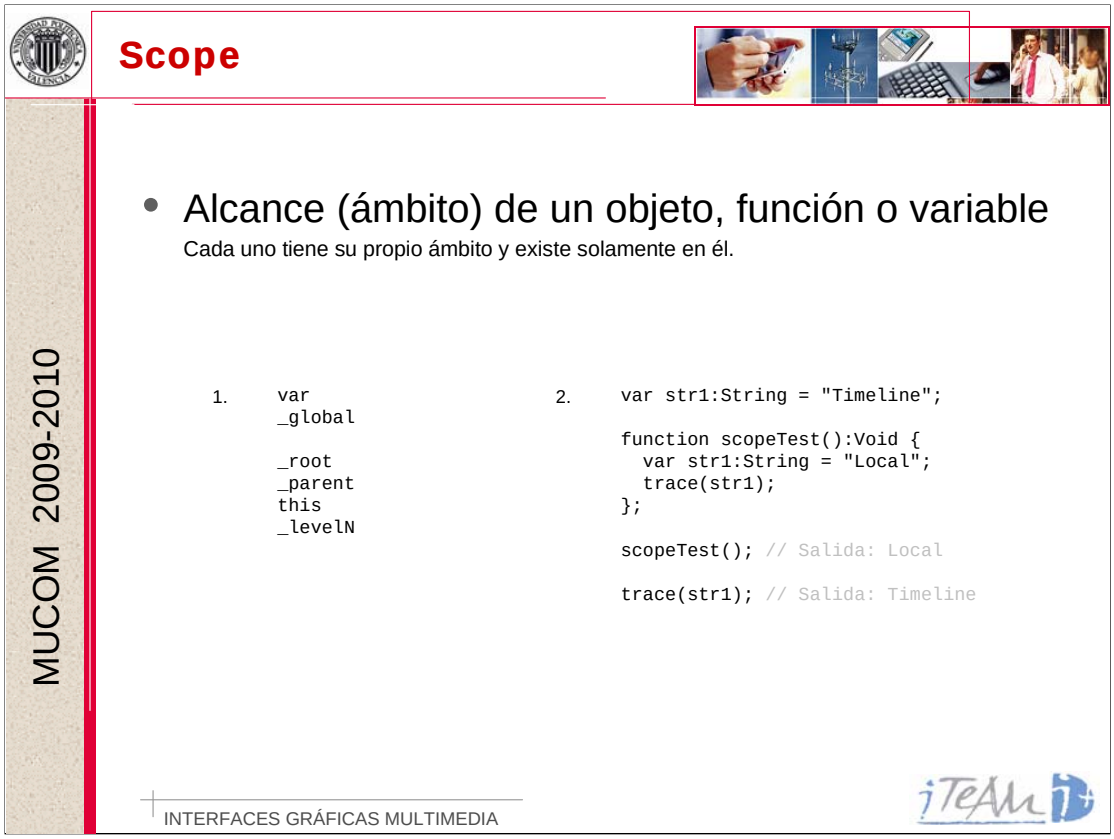
Los listeners se emplean principalmente cuando se desea asociar distintas funciones a un mismo evento sin que entren en conflicto – ya que al emplear un callback estamos asignando directamente una única función a un determinado evento.

Los eventos deben pertenecer al objeto difusor que va a recibir los listeners. Los objetos difusores más importantes son `Key` (teclado), `Mouse` (stylus), `Stage` (escenario), `TextField` (texto) y `Selection` (foco de atención).

Contenido

MUCOM 2009-2010

- Clases
- Objetos
- Propiedades
- Métodos
- Eventos
- Callbacks
- Listeners
- Scope ►



El *scope* es el alcance o ámbito de una variable, función u objeto – las instancias son también objetos. Cada uno tiene su propio ámbito y existe solamente en él. Por eso es muy importante conocer el ámbito, sobre todo de las variables. Por ejemplo, si queremos que al pulsar un botón se mueva un gráfico, la función que actúa sobre ese botón debe saber dónde se encuentra el gráfico. A modo de estructura de directorios, estos son los scopes más utilizados en Flash:

- `_root` La película principal (`/`)
- `_parent` Objetos que se se encuentras en un nivel superior (`../`)
- `this` Objeto actual (`.`)
- `_levelN` Película SWF cargada en el nivel `N`
- `_global` Ámbito global, cualquier variable es accesible desde cualquier nivel