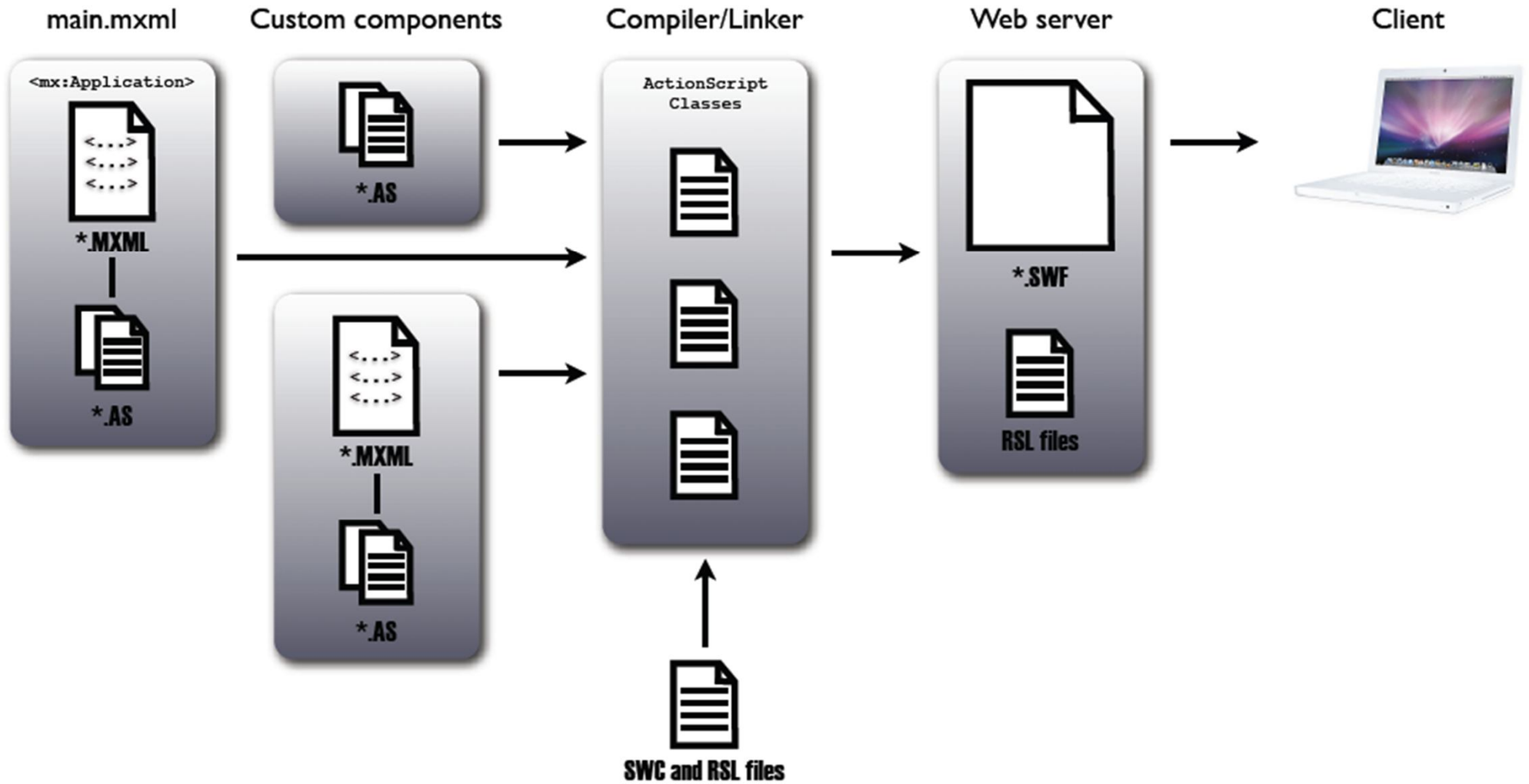


Programación con AS3

AS3



AS3

Sintaxis básica de AS3

Declaración de Variables

```
var x:Number;
```

Estructuras de control

```
if-else, for, for each (var i:int in array), for (var key:* in object({one:"ONE", two:"TWO"};)), while, do-while
```

Declaración de funciones

```
[virtual | override | final] [public | protected | private] [static]  
function nombre_funcion([nombreParámetro:TipoParametro[= "DefaultValue"]]*):[Tipo  
retorno|void]//los parametros obligatorios deben ir agrupados al principio
```

Impresión en la línea de comandos

```
trace("el mensaje es" + msj);
```

Importación de paquetes y/o clases

```
import (nombre_paquete|nombre clase)
```

AS3

Declaración de clases

```
[dynamic] [public | internal] [final] class className  
[ extends superClass ]  
[ implements interfaceName[, interfaceName... ] ] {  
    // definición  
}
```

- public – La clase es visible para todos
- dynamic – Las instancias de la clase pueden ser modificadas dinámicamente, por ejemplo se les puede agregar propiedades en tiempo de ejecución.
- final – La clase no puede ser extendida.
- internal – La clase está disponible, para cualquier clase que se encuentre en su mismo paquete.

(Los archivos mxml son en si mismos también una clase.)

AS3

Getters and Setters

```
function get propertyName():returnType  
{  
    // deben contener un return  
}
```

```
function set propertyName(value:*) :void  
{  
    // no return  
}
```

AS3

Declaración de Interfaces

```
interface InterfaceName [extends InterfaceName]
```

No se pueden definir interfaces en MXML

- Contienen solo declaraciones de método (operaciones)
- Las definiciones de métodos no contienen modificadores de acceso(public,...)
- La implementación de métodos debe ser marcada como public
- No declaran propiedades pero si se pueden agregar getters and setters

AS3

Paquetes

```
package packageName {  
class SomeClassName {  
}  
}
```

- Organizan jerárquicamente un conjunto de clases dándole un nombre a la agrupación.
- Permite dar un nombre a una agrupación de clases para luego poder importarlas refiriéndose a ellas con el nombre de su paquete.

AS3

Importación de contenido

```
import packageName.ClassName;  
import packageName.*;
```

- Permite que se posible utilizar elementos definidos en otras clases y paquetes.

Existe también otra directiva denominada include que incluye el contenido de un archivo externo como parte del script, el contenido incorporado mediante include es invocado en tiempo de compilacion.

AS3

MetaData

- Proven información al compilador de como serán usados los componentes
- Se puede anotar a una clase entera o a un atributo de una clase.
- Se pueden crear metadatas y anotar componentes con ella pero se debe compilar con la opción *keep-as3-metadata*.
- Se puede utilizar la Api de Refleciont para operar sobre la metadata.

AS3

MetaData

```
[ArrayElementType("com.miclase.data.Entity")]  
public var items : Array;
```

```
[Bindable]  
public var data : Object;
```

```
package com.view {  
    [DefaultProperty("items")]  
    public class CustomComponent  
        extends UIComponent  
    {  
        public var items : Array;  
    }  
}
```

Sin propiedad por defecto

```
<view:CustomComponent>  
    <view:items>  
        <mx:Object index="1"/>  
  
        ...  
  
        <mx:Object index="N"/>  
    </view:items>  
</view:CustomComponent>
```

Con propiedad por defecto

```
<view:CustomComponent>  
    <mx:Object index="1"/>  
  
    ...  
  
    <mx:Object index="N"/>  
</view:CustomComponent>
```

AS3

MetaData

```
[Embed(source="assets/animation.swf")]
public var embeddedMovie : Class;

[Event(name="mouseover", type="flash.events.MouseEvent")]
[Event(name="mouseout", type="flash.events.MouseEvent")]
public class CustomComponent {}
//Ide Code hinting

[InstanceType("mx.controls.Label")]
public var topRow : IDeferredInstance;
```

AS3

MetaData

```
[RemoteClass]
public class EmployeeInformation {}
Registra la clase para preservar el tipo de información cuando la instancia es
serializada por amf
```

```
[RemoteClass(alias="com.server.EmployeeInfo")]
public class EmployeeInformation {}
Representa un objeto del servidor desde la aplicación cliente
```

```
[Transient]
public var isEmployeeSessionOpen : Boolean;
Annotates properties that should be skipped
when synchronizing with remote object
mapped with [RemoteClass]
```

```
[Style(name="shadowAngle", type="int", inherit="yes")]
public class CustomComponent extends UIComponent {}
```

AS3

Display Object

- Es la clase encargada de contener y visualizar elementos.
- Existe una jerarquía de elementos (instancias directas e indirectas)
- Cada transformación que sea realizada sobre un display padre es automáticamente realizada también en todos sus descendientes.
- El método `addChild(x:DisplayObject)` permite agregar un nuevo nodo x como hijo directo de el display object sobre el cual se ejecuto esta operación.
- El último nodo agregado será el que tenga mayor índice z.

AS3

Eventos

MouseEvent (Down, Up, Move, Click...)

KeyboardEvent (Down, Up, ...)

Para manejar los eventos se deben registrar y asignar una función que los controle, lo que comúnmente se denomina handler:

Registrar evento

`addEventListener(Evento:String, nombreFunciónManejadora)`

Dejar de registrar el evento

`removeEventListener(Evento:String, nombreFunciónManejadora)`

AS3

Eventos

`event.stopPropagation();`

Durante cualquier fase se puede detener y hacer que el evento no se siga propagando en los distintos display object de una jerarquía, permite al `currentNode` que ejecute sus listeners.

`event.stopImmediatePropagation();`

No permite ejecutar ningún otro event listeners en la ejecución del `currentNode`

`event.preventDefault();`

Muchos eventos tienen asociados comportamientos por defecto, éste comportamiento puede ser anulado utilizando este método.

AS3

Eventos

Propiedades

`Bubbles:Boolean`

Indica cuando un evento es del tipo Bubble.

`Cancelable:Boolean`

Indica cuando el comportamiento de un evento puede ser cancelado.

`currentTarget:Boolean`

Es el objeto que esta actualmente procesando el evento mediante algún eventListener.

AS3

Eventos

Propiedades

`eventPhase:uint`

Es la actual fase en el flujo del evento:

- `EventPhase.CAPTURING_PHASE`
- `EventPhase.AT_TARGET`
- `EventPhase.BUBBLING_PHASE`

`target:Object`

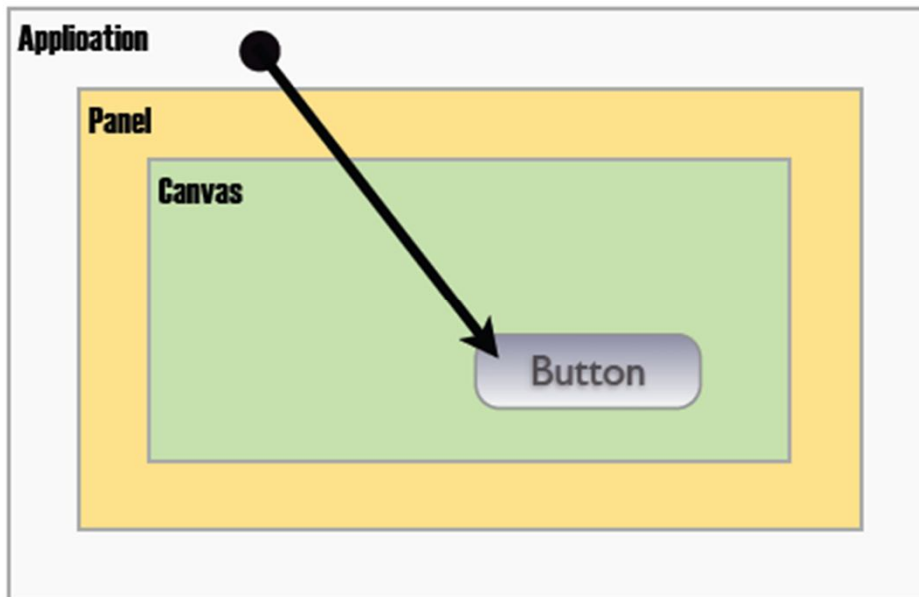
Es el nodo de la display list que contiene el despachador del evento.

`type:String`

Es el nombre del evento.

AS3

Eventos



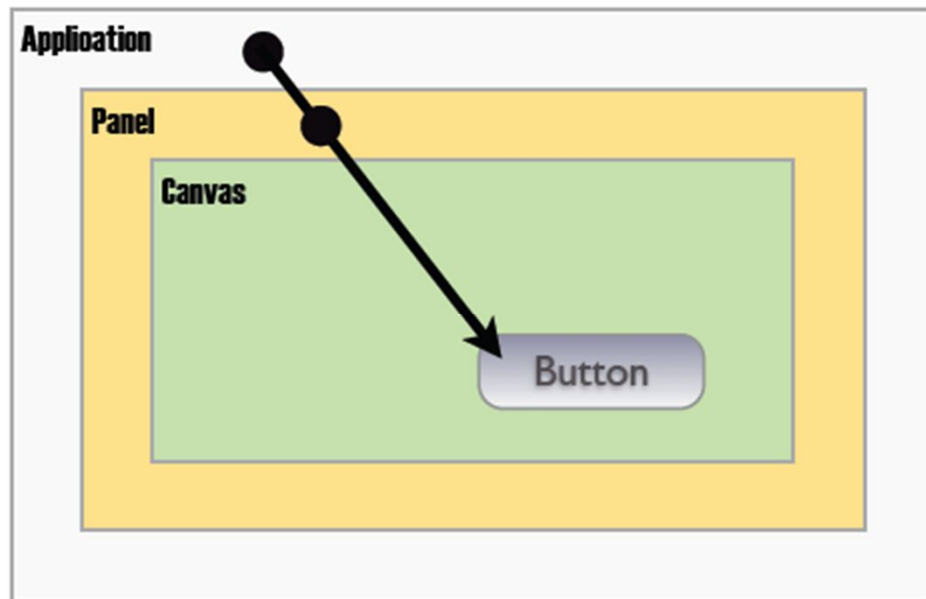
```
eventPhase = EventPhase.CAPTURING_PHASE;
```

```
target = button;
```

```
currentTarget = application;
```

AS3

Eventos



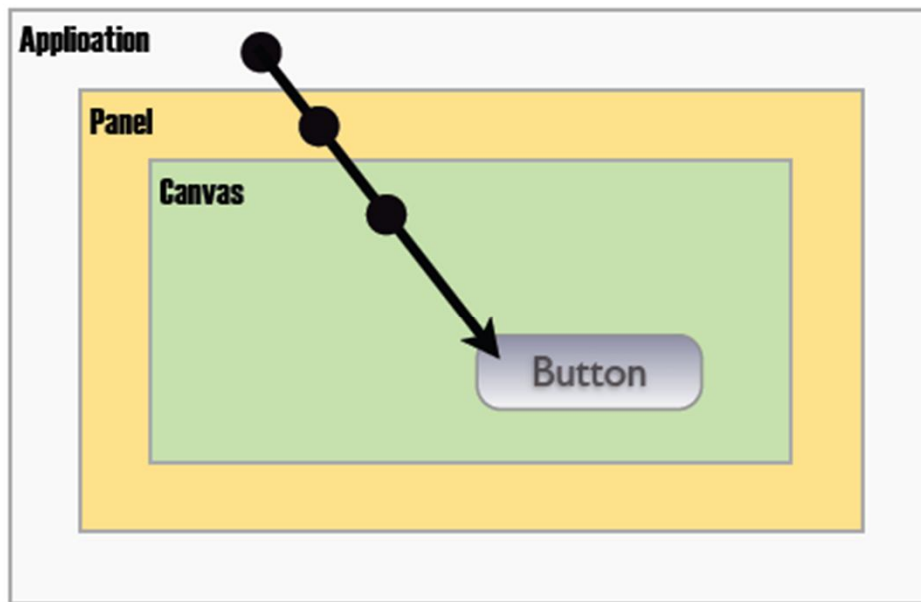
```
eventPhase = EventPhase.CAPTURING_PHASE;
```

```
target = button;
```

```
currentTarget = panel;
```

AS3

Eventos



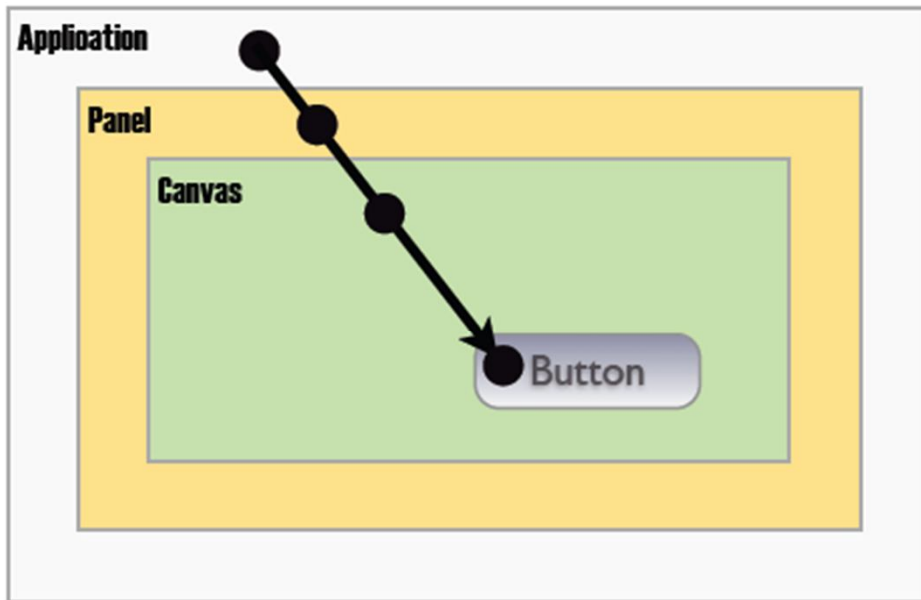
```
eventPhase = EventPhase.CAPTURING_PHASE;
```

```
target = button;
```

```
currentTarget = canvas;
```

AS3

Eventos



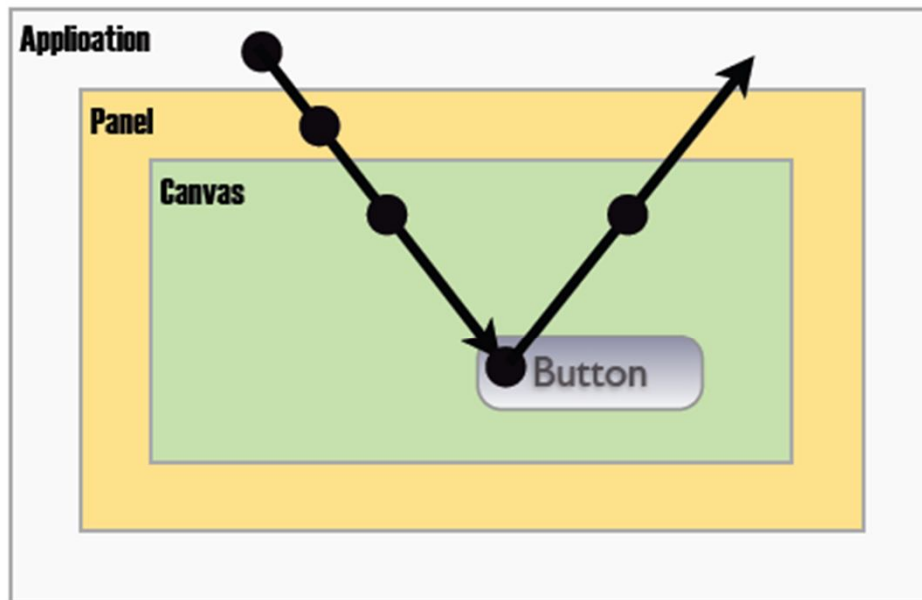
```
eventPhase = EventPhase.AT_TARGET;
```

```
target = button;
```

```
currentTarget = button;
```

AS3

Eventos



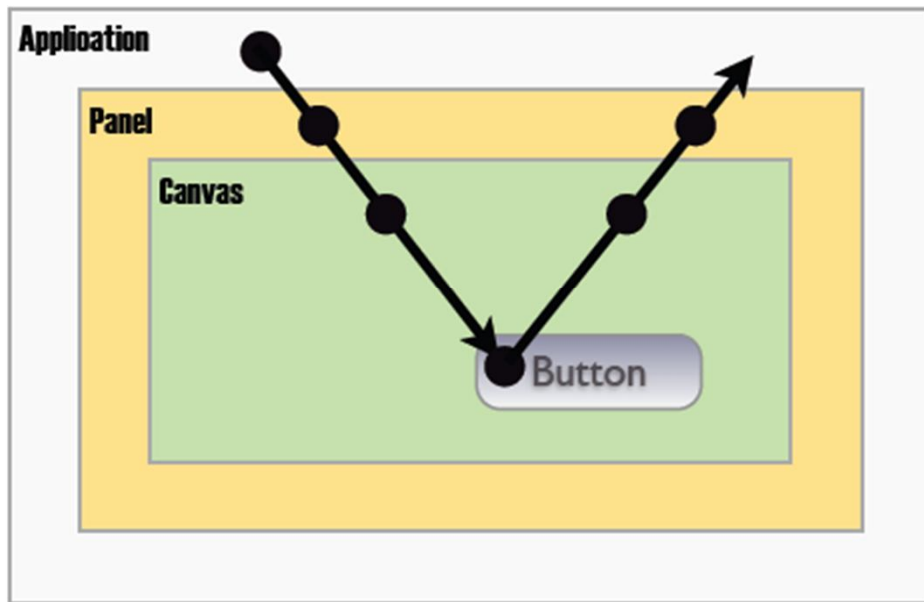
```
eventPhase = EventPhase.BUBBLING_PHASE;
```

```
target = button;
```

```
currentTarget = canvas;
```

AS3

Eventos



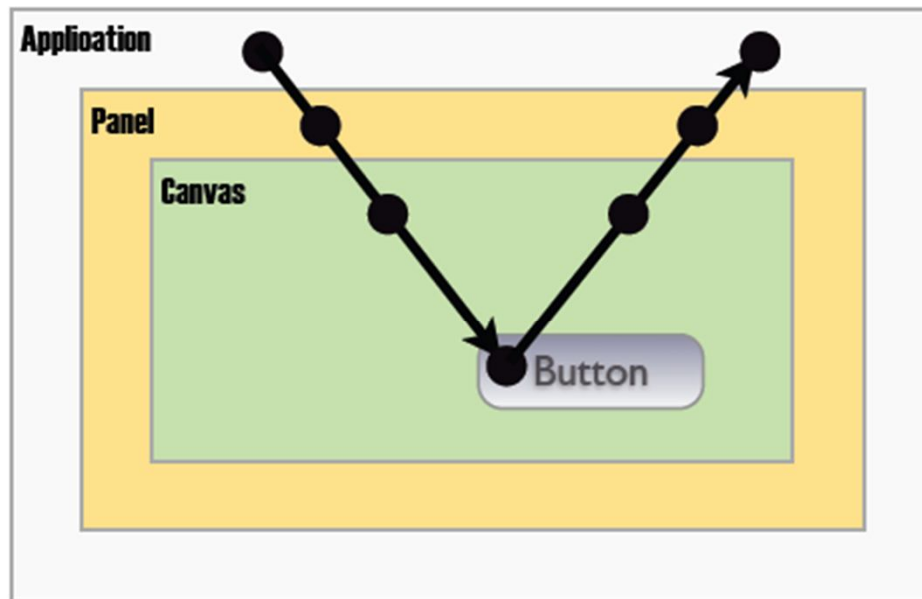
```
eventPhase = EventPhase.BUBBLING_PHASE;
```

```
target = button;
```

```
currentTarget = panel;
```

AS3

Eventos



```
eventPhase = EventPhase.BUBBLING_PHASE;
```

```
target = button;
```

```
currentTarget = application;
```

AS3

Animaciones (MovieClip)

Propiedades básicas

`x,y,height, width,rotation, scaleX, scaleY,mouseX,mouseY,alpha,visible`

Para dibujar en un clip utilizamos la clase Graphics

`lineStyle,moveTo,lineTo,beginFill,endFill, clear.`

Detección de colisiones

`hitTestObject(x:MovieClip) //controla si se superponen las bounding box de los dos movie clips`

`hitTestPoint(x,y,flag:Boolean)
//controla si hay colicion en determinado pixel , si es falsa el chequeo es contra la bundingbox`

Animación

`goToAndPlay(nro_de_frame, "nombre escena")`

`stop()`

AS3

Ejercicios

Investigar las siguientes clases y librerías:

SharedObject

Mask

Timer

TransitionManager

as3corelib(investigar en particular para el manejo de formatos de imagenes)

AS3

Ejercicios

Investigar el manejo de xml con AS3, en especial utilizando E4X

Bibliografía

Essential Action Script 3, O'Reilly

Fin