

GINGA-NCL

Segunda Parte

Recordar

- En la construcción de un programa NCL debemos tener en cuenta:
- Qué medios mostraremos? → <media>
- Dónde? → <region>
- Cómo? → <descriptor> o propiedades
- Cuándo? → <port>, conectores y enlaces
- Retomemos ejemplo previo: 01-region.ncl
- Hasta aquí hemos visto algo sobre cada uno de estos rótulos.

Despliegue de varios medios

```
<!-- regiones para los medios ->
<regionBase>
  <region id="rgvideo" width="70%" height="100%" zIndex="1"/>
  <region id="rgimg" width="30%" height="100%" left="70%" zIndex="2"/>
  <!-- resto de las regiones ->
</regionBase>
  <!--DESCRIPTOR-->
<descriptorBase>
  <descriptor id="descvideo" region="rgvideo"/>
  <descriptor id="descimg" region="rgimg"/>
  <!-- descriptores para cada region -->
</descriptorBase>

<!--CUERPO-->
<!--MULTIMEDIA-->
  <media id="video" src="../media/video1.mp4" descriptor="descvideo"/>
  <media id="img" src="../media/blue.jpg" descriptor="descimg"/>
  <!-- otros medios -->
  <!--PUERTAS-->
<port id="pentrada" component="video"/>
  <port id="p2" component="img"/>
```

Región anidada

```
<!--REGIÓN-->
```

```
<regionBase>
```

```
<region id="rgvideo" width="70%" height="100%" zIndex="1"/>
```

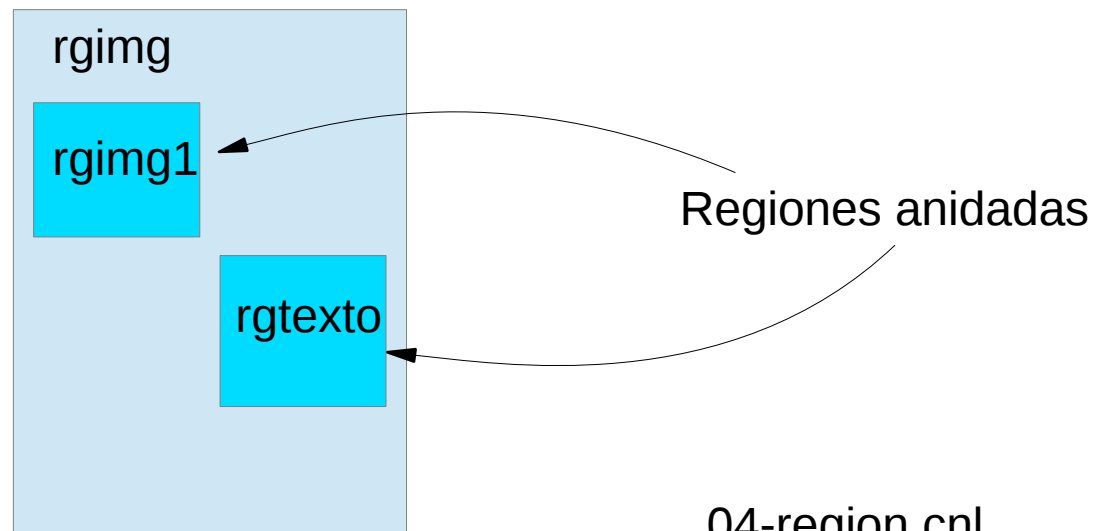
```
<region id="rgimg" width="30%" height="100%" left="70%" zIndex="1">
```

```
<region id="rgimg1" width="20%" height="20%" right="5%" top="5%" zIndex="2"/>
```

```
<region id="rgtexto" width="20%" height="20%" left="80%" top="40%" zIndex="2"/>
```

```
</region>
```

```
</regionBase>
```



04-region.cnl

Media posee propiedades para definir regiones

- Es posible prescindir de las regiones y usar propiedades de las regiones en su lugar.
- Usar lo más adecuado en cada caso.

```
<!--MULTIMEDIA-->
```

```
<media id="video" src="../media/video1.mp4">
```

```
<!--PROPIEDADES-->
```

```
<property name="width" value="100%"/>
```

```
<property name="height" value="100%"/>
```

```
<property name="zIndex" value="1"/>
```

```
</media>
```

Cambio de font de texto usando propiedades

```
<media id="img3" src="../../../media/TEXT01.txt" >
  <!--PROPIEDADES-->
  <property name="width" value="20%"/>
  <property name="height" value="20%"/>
  <property name="left" value="80%"/>
  <property name="top" value="40%"/>
  <property name="fontColor" value="yellow"/>
  <property name="fontSize" value="30"/>
  <property name="zIndex" value="2"/>
</media>
```

Otras propiedades

- Los medios poseen múltiples propiedades.
- Se pueden listar usando <control> <espacio> en eclipse luego de parámetro name en propiedades.
- Por ejemplo aquí también se puede indicar la duración de un medio.

```
<media id="img3" src="../media/TEXT01.txt" >  
  <!--PROPIEDADES-->  
  <property name="width" value="20%"/>  
  <property name="height" value="20%"/>  
  <property name="left" value="80%"/>  
  <property name="top" value="40%"/>  
  <property name="fontColor" value="yellow"/>  
  <property name="fontSize" value="30"/>  
  <property name="zIndex" value="2"/>  
</media>
```

¿Dónde? Especificado en regiones y ¿cómo? en propiedades

- Ver ejemplo 7.
- Notar que hay varias formas para responder las preguntas Dónde y Cómo.
- Dónde: en regiones o propiedades
- Cómo: en descriptores o propiedades
- Ejemplo 8, simple sobre transparencia.

Efecto de transición de objetos

- A través del rótulo <transition> podemos definir transiciones Como “fade”, “barWipe”, etc.
- Ver ejemplo 09-descriptor.ncl

Cuándo se muestra un medio?

- <port> para partir.
- En otros casos podemos usar **conectores**: Relaciona dos o varios elementos multimedia e identifica cuándo ellos serán presentados como respuesta a una acción.
- Ejemplo: Al momento de empezar un elemento multimedia, cambiar el tamaño de otro elemento.
- Para esto usamos conectores y enlaces.
- Primero debemos definir el conector, el cual indicamos la condición bajo la cual una acción debe ser hecha.
- Luego usamos los conectores creando enlaces donde indicamos los medios relacionados a las condiciones y acciones del conector.

Conectores-enlaces

```
<!--CONECTORES en head-->
```

```
<connectorBase>
```

```
  <causalConnector id="OnBeginStart">
```

```
    <simpleCondition role="onBegin" />
```

```
    <simpleAction role="start" />
```

```
  </causalConnector>
```

```
</connectorBase>
```

```
<!-- uso de conectores en enlaces puestos en body-->
```

```
<link xconnector="OnBeginStart" id="limg">
```

```
  <bind role="onBegin" component="video"/>
```

```
  <bind role="start" component="img1"/>
```

```
</link>
```

Ejemplo 10-conector_enlace.ncl y 11-conector_enlace.ncl






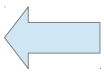
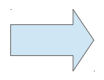


Los tipos de conector a usar se pueden importar desde un archivo

- Ver ejemplo `12-conector_enlace.ncl`
- En este caso le damos un alias y luego nos referimos a los conectores allí definidos con:
“alias”#”conector”

Sincronismo

- Podemos definir el rótulo <area> “anchor” (ancla) Permite definir puntos relativos a un medio. Así otro puede iniciarse sincrónicamente con un punto del primero.
- Usamos rótulo<area> en media y luego en enlace.
- Ver ejemplo 13-sincronismo.ncl
- Es posible enlazar varios medios con un enlace.
- Ver ejemplo 14-control_enlaceN.ncl

Control Remoto

- RED F1 
- GREEN F2 
- YELLOW F3 
- BLUE F4 
- MENU F5 
- INFO F6 
- ENTER 
- LEFT 
- RIGHT 
- UP 
- DOWN 

Reaccionando ante el control remoto

```
<link xconnector="conector#onKeySelectionNStartNStopN" id="amarillo">  
  <bind role="onSelection" component="imyyellow"/>  
  <linkParam name="keyCode" value="YELLOW"/>  
  <bind role="start" component="img1"/>  
  <bind role="stop" component="img"/>  
  <bind role="stop" component="img2"/>  
  <bind role="stop" component="img3"/>  
</link>
```

```
<link xconnector="conector#onKeySelectionNStartNStopN" id="rojo">  
  <bind role="onSelection" component="imyred"/>  
  <linkParam name="keyCode" value="RED"/>  
  <bind role="start" component="img2"/>  
  <bind role="stop" component="img"/>  
  <bind role="stop" component="img1"/>  
  <bind role="stop" component="img3"/>  
</link>
```

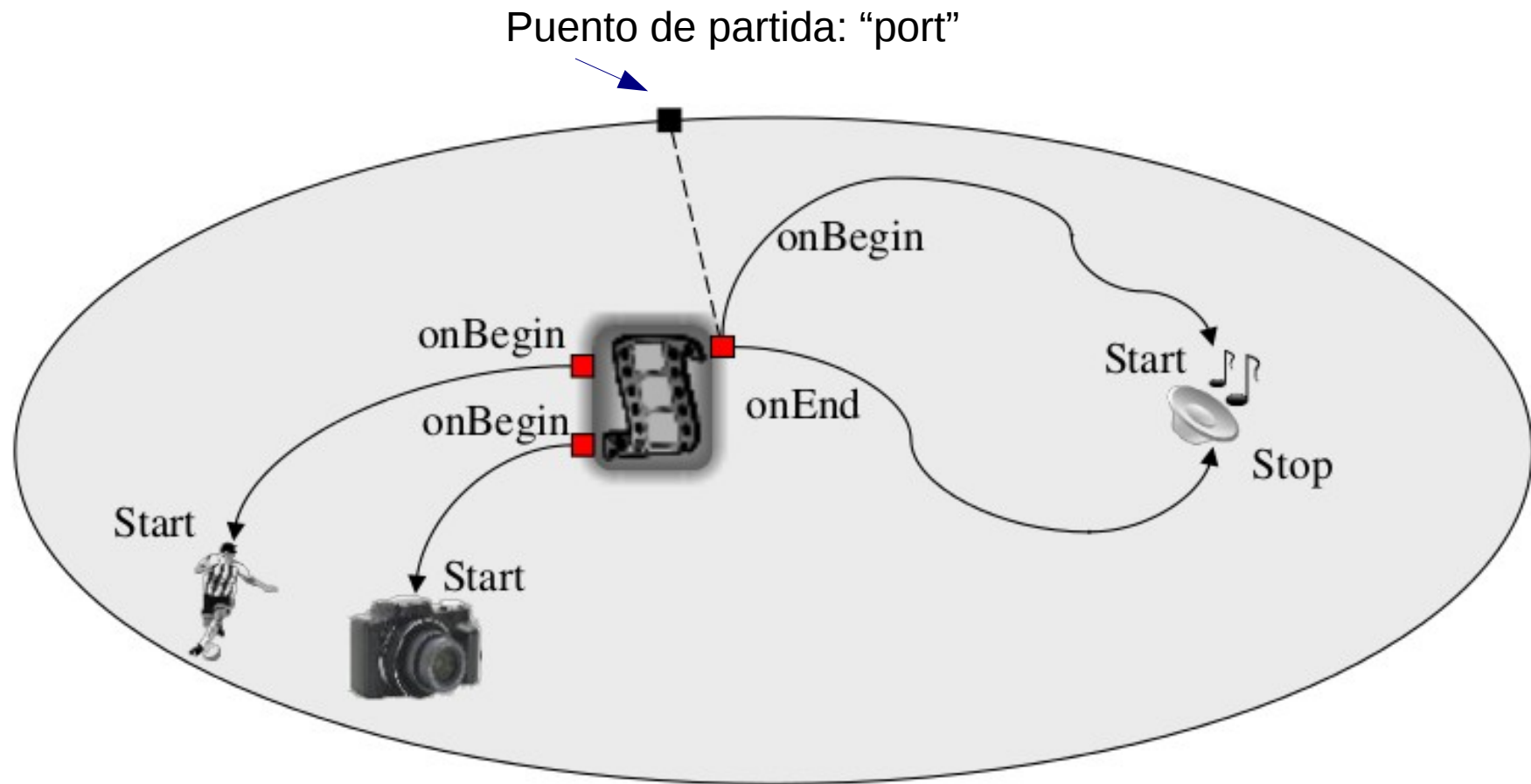
Redimensionamiento

```
<media id="video" src="../../media/video1.mp4" descriptor="dvideo">  
  <property name="bounds"/>  
</media>
```

....

```
<link xconnector="conector#onKeySelectionSetNStopN">  
  <linkParam name="keyCode" value="GREEN"/>  
  <bind role="onSelection" component="bgreen"/>  
  <bind role="stop" component="bgreen"/>  
  <bind role="set" component="video" interface="bounds">  
    <!--REDIMENSIONAR PANTALLA-->  
    <bindParam name="var" value="25%,25%,50%,50%"/> <!-- f,t,w,h-->  
  </bind>  
</link>
```


Visión estructural de una aplicación



Sólo como referencia, no corresponde al programa previo.