

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre

Cardozo, Sergio A. – Schwartz, Sebastián M.

Licenciatura en Ciencias de la Computación - Facultad de Ciencias Exactas, Físico-Químicas y Naturales - Universidad Nacional de Río Cuarto

Resumen. “CreaTV Digital” es una herramienta de fácil manejo destinada a desarrolladores de contenidos con una interface orientada a la producción de contenidos interactivos de Televisión Digital de manera gráfica, sin la necesidad de conocimientos detallados de los lenguajes NCL y Lua. La herramienta permite al usuario, por medio de una interfaz gráfica e intuitiva, realizar el contenido (programa/película junto con su aplicación interactiva) que será emitido y luego ésta se encargará de crear el código NCL correspondiente para poder ser ejecutado en el middleware Ginga. La herramienta ha sido desarrollada en el lenguaje C++ bajo la biblioteca multiplataforma Qt¹.

Palabras claves: TVD, Ginga, Lua, NCL, NCM, TDT.

1 Introducción

La Televisión Digital Terrestre (TDT)² [9] es una nueva técnica de difusión de las señales de televisión que promete sustituir a la televisión analógica actual.

Al recibir señal digital, es necesario tener un decodificador (Set-Top Box) que sea capaz de interpretar la señal y ejecutar aplicaciones *Ginga*³.

Ginga[10] es el nombre de la plataforma de ejecución de software para la televisión desarrollada por universidades Brasileñas y con importantes aportes desarrollados por el LIFIA de la Universidad Nacional de La Plata. Gracias a este middleware además de generar contenidos para televisión, podemos ejecutar las aplicaciones que acompañen a cada programa/película en un *Set-Top Box* (STB) [2].

La señal de TVD es enviada desde una central emisora a retransmisoras ubicadas en distintos lugares para así llegar a cada TV por medio del modelo de broadcasting. En la señal, además del video y audio, son enviados también los datos comprimidos (que representan aplicaciones) de tal forma que el Set-Top Box pueda procesarlos para mostrar la información en la TV como muestra la figura 1.

Ginga nos permite programar en los lenguajes declarativo e imperativo[4][5]. El lenguaje declarativo admitido por Ginga es NCL (*Nested Context Language*)[2] y el

¹ <http://www.qt.nokia.com>

² <http://www.tvdigitalargentina.gov.ar/tvdigital>

³ <http://www.ginga.org.br>

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 2 imperativo es el lenguaje llamado Lua⁴. Una de las grandes ventajas de Ginga es que es el único estándar internacional y también abierto.

El lenguaje declarativo NCL basado en el modelo conceptual NCM (*Nested Context Model*) [2] provee facilidades para especificar aspectos de interactividad, sincronismo espacial/temporal entre objetos de multimedia, adaptabilidad y soporte para múltiples dispositivos (TV, PC, Dispositivos Móviles), es decir construir aplicaciones.

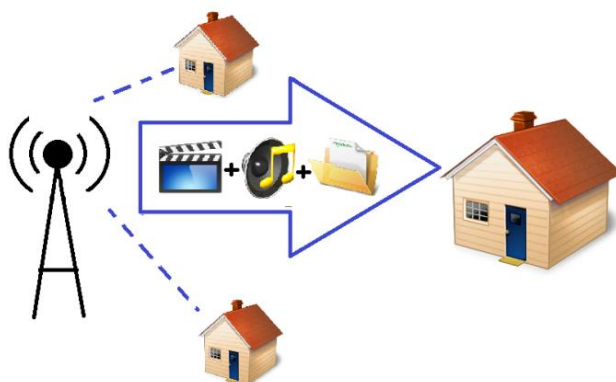


Fig. 1. Modelo de broadcasting de TVD

Si bien NCL no es un lenguaje complicado de aprender para personas en el ámbito de la programación, sería importante poder realizar aplicaciones para TDT sin la obligación de poseer conocimientos de programación, por lo que se realizó una herramienta de alto nivel con la cual se puede crear aplicaciones por medio de la interacción del usuario sin la necesidad de escribir código NCL.

Por ello desarrollamos “CreaTV Digital”⁵ para que el usuario pueda generar aplicaciones NCL mediante una interfaz gráfica e intuitiva sin la necesidad de conocer ni escribir código NCL. Además, debido al uso de esta herramienta, el encargado de desarrollar los contenidos podrá optimizar el tiempo de creación de aplicaciones. Por lo tanto, gracias a “CreaTV Digital”, los encargados de audiovisuales no requerirán tener un programador para la creación de aplicaciones para la televisión digital.

2 El modelo conceptual NCM

Todo lenguaje declarativo está basado en un modelo conceptual de datos. Un modelo conceptual debe representar los conceptos estructurales de datos, así como los eventos y las relaciones entre los datos. El modelo debe definir también las reglas estructurales y las operaciones sobre los datos para la manipulación y actualización de las estructuras.

⁴ <http://www.lua.org>

⁵ <http://code.google.com/p/creatvdigital> (código fuente junto con ejemplos de utilización)

Entidades básicas de NCM:

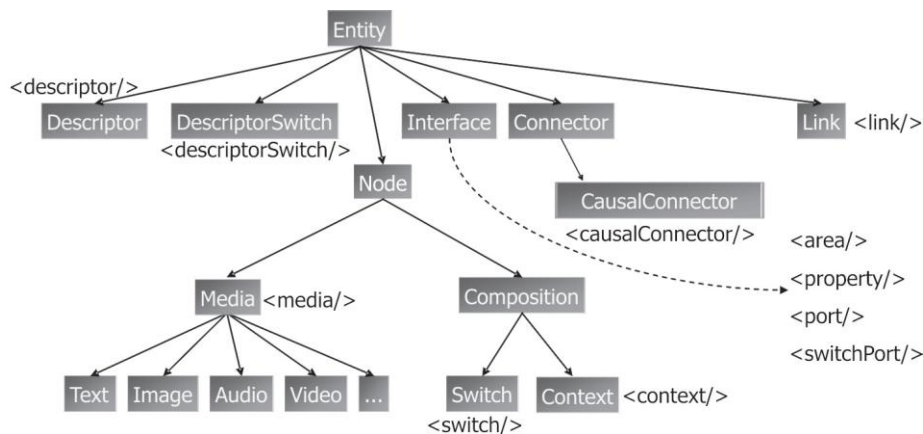


Fig. 2. Modelo Conceptual NCM básico junto con los elementos del lenguaje NCL.

Cada uno de estos nodos posee un identificador, un contenido y un conjunto de anclas. El contenido de un nodo es formado por un conjunto de *unidades de información*. La noción exacta de lo que constituye una *unidad de información* es parte de la definición de nodo y depende de su especialización. Un ancla es un subconjunto de unidades de información de un nodo.

3 Aplicaciones de TV Digital

Las aplicaciones en la televisión digital pueden ser aquellas que no tienen relación con el programa que se está transmitiendo (ej.: la muestra del estado del clima en un programa de cocina), aplicaciones que tengan relación con el programa en transmisión (ej.: formación de los equipos durante la transmisión de un partido de fútbol) o pueden haber aplicaciones que necesiten de la interacción del usuario como por ejemplo un pequeño juego, preguntas y respuestas sobre el contenido de un programa educativo, etc.

El sistema de TV Digital utiliza un middleware que permite ejecutar estas aplicaciones interactivas dentro de un STB. Ese middleware es conocido como Ginga.

Ginga (Middleware Abierto del Sistema Brasileiro de TV Digital - SBTVD) nos permite correr aplicaciones sin importar qué STB tengamos actuando como una capa que se añade a los patrones de referencia de un sistema de TV digital.

El middleware abierto Ginga se subdivide en dos subsistemas principales interrelacionados que permiten el desarrollo de aplicaciones siguiendo dos paradigmas de programación diferentes. Dependiendo de las funcionalidades requeridas en cada proyecto de aplicación, un paradigma será más adecuado que otro. Estos dos subsistemas se llaman Ginga-J (para aplicaciones procedurales Java) y Ginga-NCL (para aplicaciones declarativas NCL e imperativas Lua) de los cuales mencionaremos el segundo.

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 4

3.1 Nexted Context Language (NCL)

NCL es un lenguaje declarativo basado en el modelo conceptual NCM, básicamente una aplicación XML. El lenguaje define claramente cómo los objetos media (elementos de contenido multimedial, es decir, los elemento a mostrar como por ejemplo videos, imágenes, sonidos, etc.) son estructurados y relacionados, en el tiempo y en espacio. Como es un lenguaje de marcado, no especifica los tipos del contenido de los objetos media de una aplicación.

3.2 Estructura del lenguaje NCL

La estructura general del lenguaje es la siguiente:

```
1. <?xml version="1.0" encoding="ISO-8859-1"?>
2. <ncl id="main"
   xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
3.   <head>
4.     region, descriptor, connector
5.   </head>
6.   <body>
7.     media, port, switch, link
8.   </body>
9. </ncl>
```

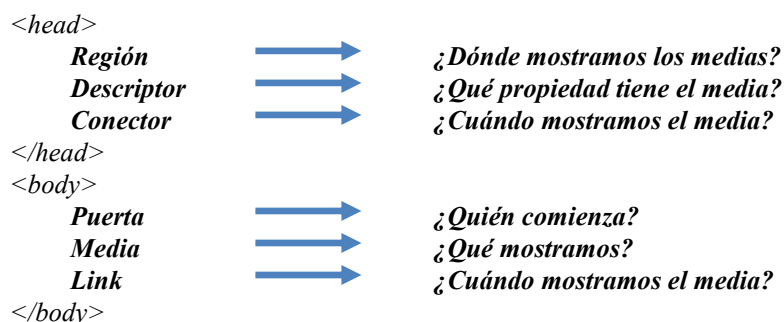
Como dijimos anteriormente, el lenguaje NCL es una aplicación XML, esto implica que debe tener un encabezado como se ve en la línea 1.

Luego se comienza con la estructura de la aplicación con el tag `<ncl>` con su respectivo atributo `id` y `xmlns` como lo indica la línea 2.

El elemento `<head>` contiene bases de elementos referenciados por el núcleo de la aplicación NCL (definido en el elemento `<body>`), como ser las regiones, los descriptores, las transiciones, los conectores y las reglas.

El elemento `<body>` contiene los elementos que definen el contenido de la aplicación propiamente dicha, tal como los objetos media, links, contextos y objetos switches.

Podemos graficar la estructura NCL de la siguiente forma:



“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 5

3.2.1 ¿Qué mostramos? – Media

Los objetos media o de contenido son representados por el elemento `<media>`. Como por ejemplo videos, audio, imágenes, texto, etc.

Ejemplo de la estructura de medias:

```
<media id="idImg" descriptor="idDesc" src="foto.jpg"/>
<media id="idVi" descriptor="idDesc2" src="video.mp4"/>
```

3.2.2 ¿Dónde mostramos los medias? – Región

Las regiones representan el área de la pantalla donde se mostrará el media. Éstas pueden ser indicadas en pixeles o porcentaje. Son representadas mediante el elemento `<region>`.

Ejemplo de la estructura de una región con un ancho y alto del 50% de la pantalla:

```
<region id="reg1" height="50%" width="50%" zIndex="1"/>
```

3.2.3 ¿Qué propiedad tiene el media? – Descriptor

Los descriptores especifican cómo los medias asociados a ellos serán exhibidos. Son representados mediante el elemento `<descriptor>`.

Ejemplo de la estructura de un descriptor que proporciona una transparencia del 50% a un media:

```
<descriptor id="idDesc" region="rgTVmenu">
  <descriptorParam name="transparency" value="50%" />
</descriptor>
```

3.2.4 ¿Quién comienza? – Puerta

Nos indica qué media iniciará al comenzar la aplicación. Son representados mediante el elemento `<port>`.

Ejemplo de la estructura de una puerta indicando que el media “*idImg*” comenzara al iniciar la aplicación.

```
<port id="pEntrada" component="idImg"/>
```

3.2.5 ¿Cuándo mostramos el media? – Conector/Link

Permite especificar acciones sobre objetos media a partir de la ocurrencia de eventos.

Los eventos que pueden llegar a ocurrir son:

- *onBegin*: Cuando la presentación de un media es iniciado.
- *onEnd*: Cuando la presentación de un media es terminado.
- *onAbort*: Cuando la presentación de un media es abortado.
- *onPause*: Cuando la presentación de un media es pausado.

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 6

- *onResume*: Cuando la presentación de un media es reanudado.
- *onSelection*: Cuando una tecla relacionada al media es presionada.

Las acciones disparadas al ocurrir un evento son:

- *start*: Inicia la presentación del media asociado.
- *stop*: Termina la presentación del media asociado.
- *abort*: Aborta la presentación del media asociado.
- *pause*: Pausa la presentación del media asociado.
- *resume*: Reanuda la presentación del media asociado.
- *set*: Establece un valor de una propiedad asociada al media.

3.3 Lua

Ginga, además del lenguaje declarativo NCL, admite el lenguaje imperativo Lua. El lenguaje es usado normalmente para permitir que una aplicación sea extendida o adaptada a través de *scripts*. Lua es un lenguaje de fácil aprendizaje, que combina sintaxis procedural con declarativa, con pocos comandos primitivos. Por lo tanto, comparado con otros lenguajes, posee implementación ligera y extensible. Otra de las características es que posee un Garbage-collection, un sistema dinámico de tipos y un alto grado de portabilidad, pudiendo ser ejecutada en diversas plataformas, tales como computadores personales, celulares, consolas de video juego, etc.

3.4 NCL y Lua

Para poder integrar el lenguaje Lua al ambiente de la TV Digital y NCL, Lua fue extendido con nuevas funcionalidades. Como por ejemplo, un elemento NCLua precisa comunicarse con un documento NCL para saber cuando su objeto *<media>* es iniciado por un link.

Aquí nombraremos las funcionalidades que agrega la biblioteca NCLua a Lua: *event, canvas, settings, persistent*.

Para llamar a una aplicación Lua desde NCL, se manipula al elemento Lua como cualquier otro objeto media:

```
<body>
  <media id="id" src="file.lua" descriptor="idD"/>
</body>
```

4 Modelo sobre “línea de tiempo” creado

Para graficar las aplicaciones NCL se utilizan gráficos espaciales (para representar las regiones donde se mostrarán los objetos), gráficos temporales (para representar el tiempo de vida de un objeto) y gráficos estructurales (para representar las acciones ejecutadas por las ocurrencia de un evento).

Para un diseñador de contenidos, le es más fácil trabajar con gráficos espaciales y temporales (dado a que es de su rutina) que con grafos de estados (o gráficos estructurales). Por lo que hacer una aplicación medianamente grande y representarla

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 7 con un gráfico de estados, es complicado y engorroso de entender. Por ello hemos tratado de evitar la utilización de gráficos de estados y en cambio, los combinamos con el gráfico de línea de tiempo generando así un nuevo modelo más fácil de entender.

4.1 Elementos del modelo









Elementos del diagrama	Detalle
	Media
	Puertas (Ports)
	Línea de Tiempo con terminación Natural (cuando termina el media)
	Indicador de interactividad.
	Línea de tiempo de media con variabilidad de inicio según momento de interacción
[Marca de inicio de intervalo de tiempo (área-ancla)
]	Marca de fin de intervalo de tiempo (área-ancla)
	Botón del control remoto
	Switch
	Componente Default del switch

Fig. 3. Elementos del modelo creado

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 8

- Medias: Los elementos medias del lenguaje NCL lo representamos mediante distintas imágenes dependiendo del tipo (video, imagen, texto, etc.).
- Puerta: Señala el media que comienza la aplicación.
- Línea de tiempo con terminación natural: Nos indica el tiempo de vida del media con la longitud de la línea.
- Indicador de interactividad: Nos muestra el intervalo de tiempo en el que está disponible la iteración del control remoto.
- Línea de tiempo de media con variabilidad de inicio según momento de interacción: Nos indica el rango de variabilidad de tiempo posible en el cual el media pueda iniciar y terminar, es decir, indica el menor tiempo en el que el media puede empezar y el máximo tiempo que el media pueda terminar (según la intervención del usuario presionando una tecla del control remoto).
- Marca de inicio de intervalo de tiempo (área-ancla): Representa el inicio de un subsegmento de tiempo dentro de una línea de tiempo.
- Marca de fin de intervalo de tiempo (área-ancla): Representa el fin de un subsegmento de tiempo dentro de una línea de tiempo.
- Botón del control remoto: Imagen representando una tecla del control remoto.
- Switch: Indica distintas posibilidades de elección según regla a cumplir.
- Componente default del switch: Indica el media por defecto del switch.

Comparación entre el modelo estructural y el modelo construido de una aplicación NCL⁶:

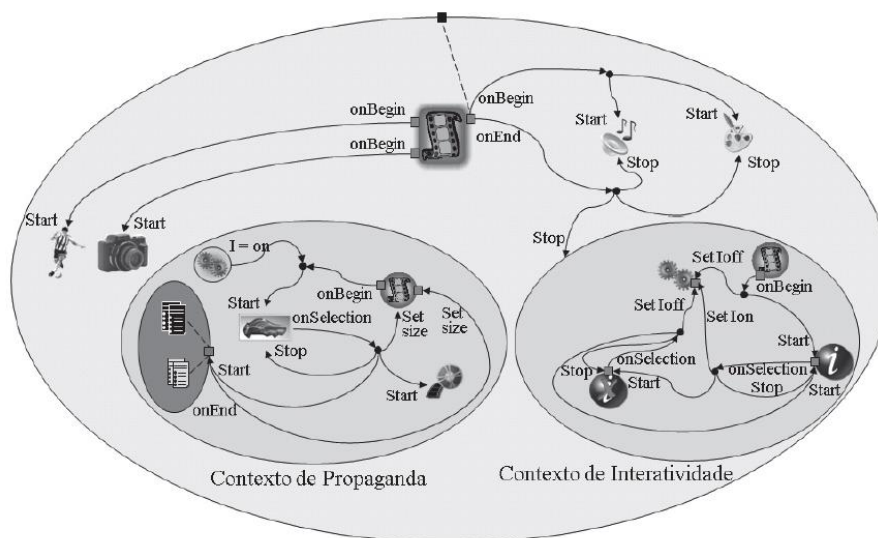


Fig. 4. Modelo estructural

⁶ Ejemplo tomado del libro: Soares, Luis Fernando Gomez y Simone Diniz, Junqueira Barbarosa. Promamdo em NCL: Desenvolvimento de Aplicações para middleware Ginga, TV Digital e WEB. Rio de Janeiro : Elsevier, 2009

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 9

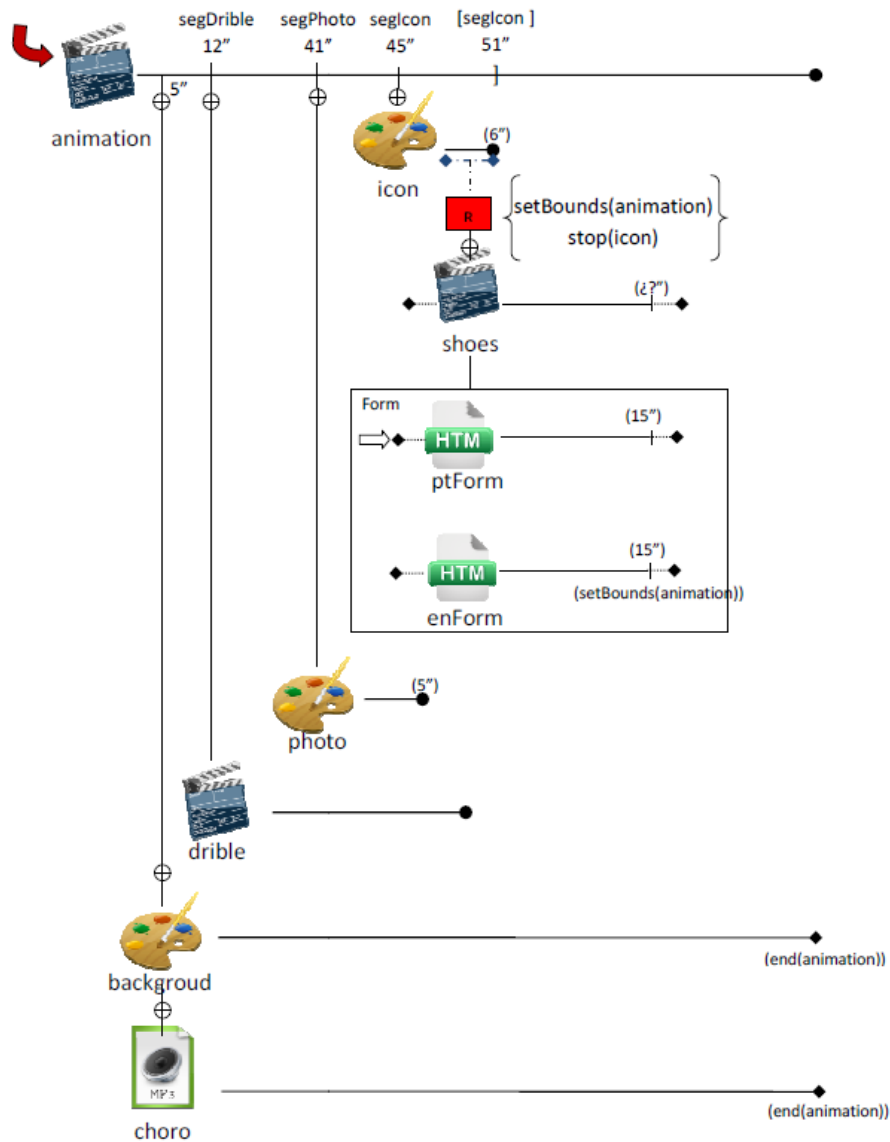


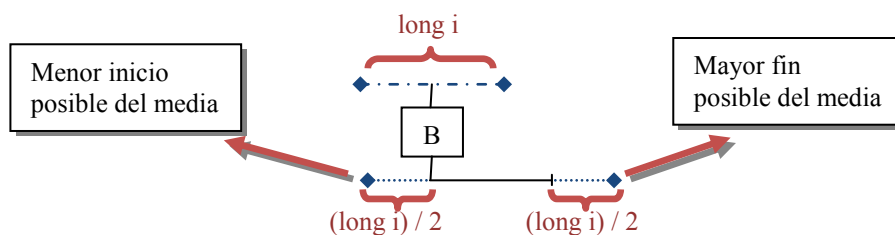
Fig. 5. Ejemplo del modelo creado.

Analizando las figuras de los dos modelos podemos ver que es más sencillo e intuitivo representar una aplicación mediante el modelo creado, vemos que es más fácil ubicarnos en el tiempo y realizar relaciones entre los medias.

En la figura 4 vemos al media “animation” como puerta de entrada, a partir del cual al transcurrir el tiempo inician otros medias. El media “icon” posee la interacción

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 10 indicando que al presionar el botón rojo del control remoto se iniciará el media “shoes”.

El media “shoes” está representado con una “Línea de tiempo de media con variabilidad de inicio según momento de interacción”. La línea punteada nos indica la longitud del tiempo de la interactividad. Esto es para indicar que la aparición y finalización del media puede variar en ese intervalo de tiempo. Para mayor claridad presentamos un ejemplo:



5 CreaTV Digital



“CreaTV Digital” fue creado con el principal objetivo de simplificar la creación de contenidos a las personas encargadas de audio-visuales en TVD.

Al utilizar la herramienta para la creación de aplicaciones para la TVD evitamos la necesidad de poseer conocimiento de programación. Evitamos que los desarrolladores de contenido realicen la “tediosa” tarea de escritura de código para una aplicación dándole el beneficio que esto implica como ser reducción del tiempo al tipearla, aprendizaje de los lenguajes de programación, errores de programación, optimización de código, modularización, etc.

Como la producción de contenido necesita tener un buen diseño para el atractivo visual, la herramienta facilita la ubicación en el espacio de los objetos mediante una sección de dibujo de regiones pudiendo visualizar de antemano en qué sitio específico será mostrado un media. Además, es mas intuitivo para el usuario el manejo de la vida de los objetos mediante una línea de tiempo y realizando relaciones entre ellas.

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 11

5.1 Estructura de la herramienta

Al iniciar “CreaTV Digital” creamos un proyecto nuevo indicándole su nombre, una breve descripción, y la posible resolución del dispositivo en la que será mostrada la aplicación.

La herramienta posee una pantalla principal que está dividida en dos secciones importantes. Una de ellas es el área espacial y la otra es el área temporal.

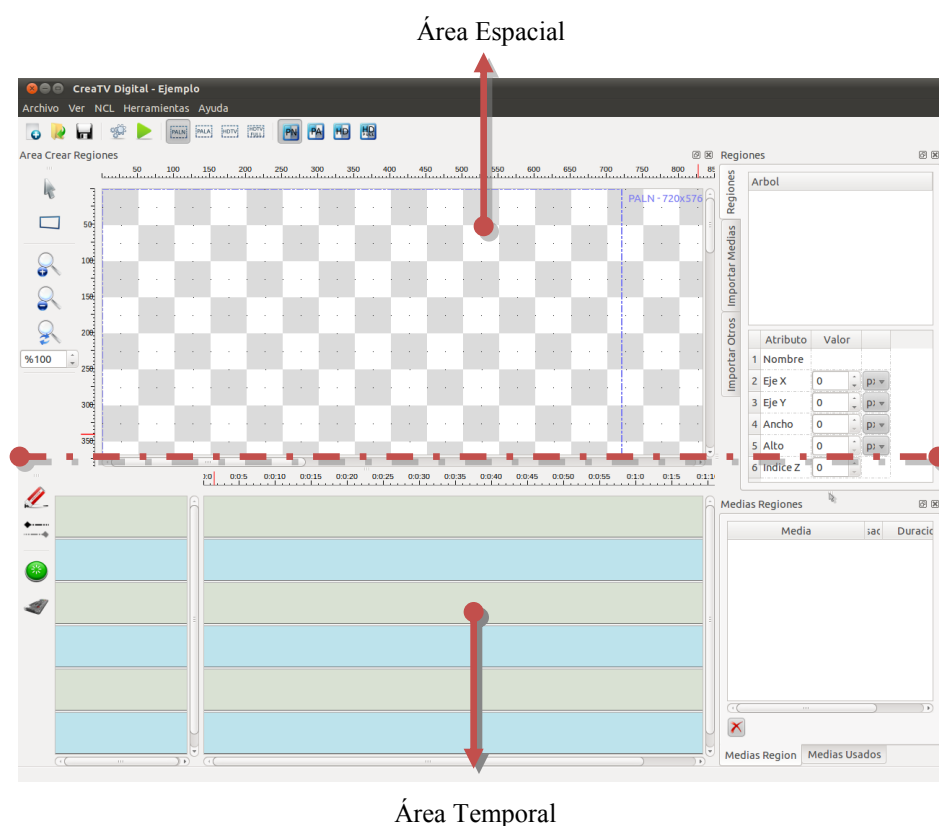


Fig. 6. Secciones principales de la herramienta

En el área espacial nos encargamos de buscar los archivos que utilizaremos en el proyecto. Una vez hecho esto, podemos crear regiones gráficamente guiándonos mediante las reglas y ubicarlas en el lugar deseado mediante el uso del mouse, o especificado directamente sus atributos sobre una tabla, esto nos permite asignar la ubicación exacta a la región, también podemos utilizar la herramienta del zoom para lograr una delicada ubicación de las regiones en el espacio. En el área de dibujo de regiones se encuentran marcas que delimitan el tamaño de pantalla del dispositivo seleccionado para así trabajar mejor ubicándonos en el área de trabajo.

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 12

Una vez creadas las regiones, son importados los archivos a utilizar y los arrastramos a la región donde serán mostrados.

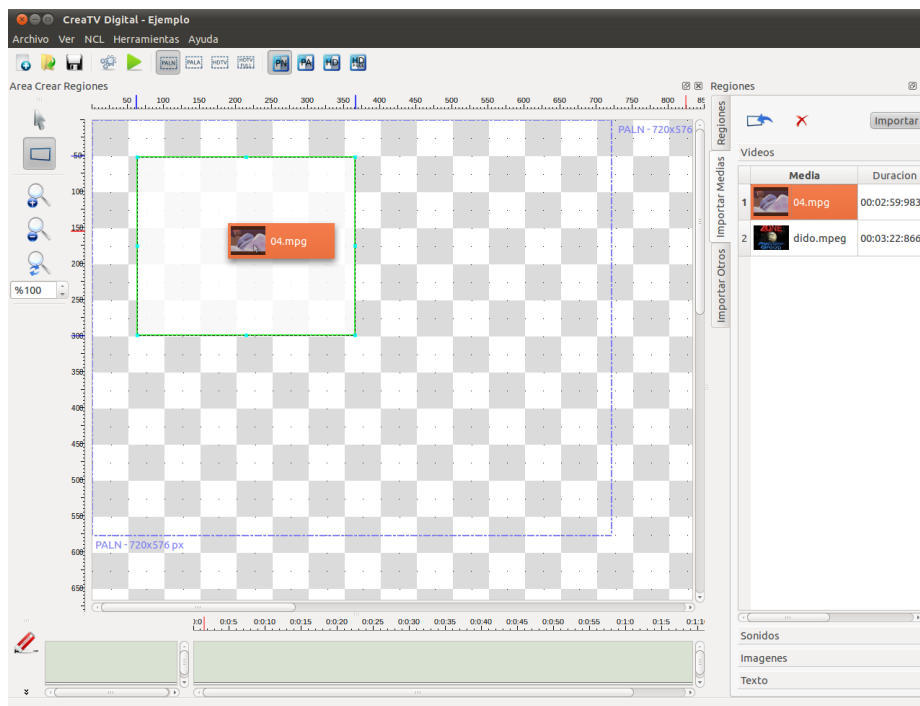


Fig. 7. Sección área espacial.

En la sección de área temporal acomodamos los medias usados en las regiones, arrastrándolos hacia el “área de dibujo”. De esta forma indicamos en qué momento de la aplicación se mostrará el media, ya sea por un evento o porque es la puerta del programa.

Tenemos la posibilidad también de asignar interactividad a cada media (seleccionando el botón deseado del control remoto) y asociarle una acción a realizar. Al comienzo y finalización de los medias se pueden asociar las acciones que se necesiten en la aplicación.

Cada media tiene la posibilidad de modificar la forma en que se van a mostrar en pantalla, es decir, setear los atributos propios de su tipo. Esto lo podemos realizar al ocurrir un evento, o modificar las propiedades al media como default.

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware GINGA de Televisión Digital Terrestre 13

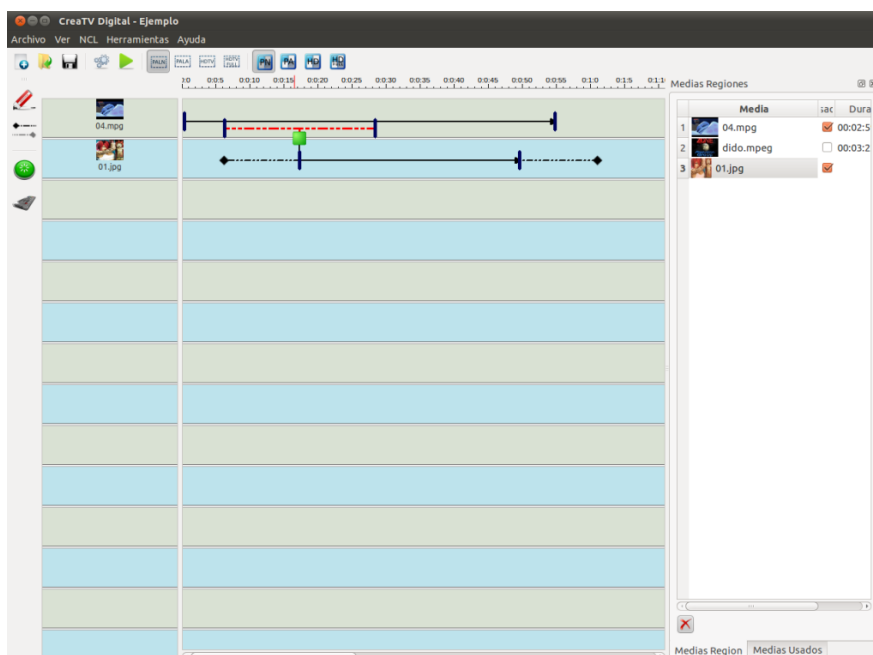


Fig. 8. Sección área temporal

Una vez creada la aplicación, podemos seleccionar el botón para generar la aplicación NCL. Esto creará automáticamente todo el código NCL que representará al proyecto creado. Luego de esto podremos ejecutar la aplicación para ver los resultados obtenidos con el botón correspondiente.

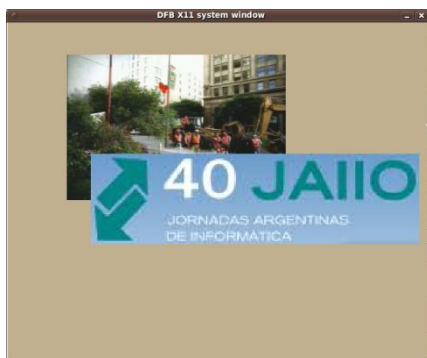


Fig. 9. Aplicación NCL corriendo bajo Ginga

En el apéndice A podemos ver un ejemplo concreto de creación de una aplicación NCL y su ejecución utilizando “CreaTV Digital”. Para más información ingresar al link <http://code.google.com/p/creatvdigital/> (aquí se podrá encontrar videos de uso de la herramienta, el código fuente, ejemplos, etc.)

6 Conclusión y futuras extensiones

Uno de los objetivos más importantes planteados al comienzo del proyecto fue el de crear un modelo gráfico de línea de tiempo para la representación de la vida de los elementos de aplicaciones NCL, la interactividad con el televidente y los eventos con sus acciones ocurridos a lo largo de la aplicación. El objetivo fue cubierto gracias al modelo de línea de tiempo construido.

Luego, al implementar este modelo construyendo una herramienta de fácil uso e intuitiva para personas sin conocimiento de programación los objetivos personales fueron satisfactoriamente cumplidos logrando implementar a “CreaTV Digital”. Esta herramienta podrá ser de gran ayuda para mucha gente relacionada con la creación de contenidos audiovisuales dado que no existía alguna en su tipo.

Como futuras extensiones a la herramienta podemos nombrar la de realizar una implementación completa del modelo de línea de tiempo que no ha podido hacerse por cuestiones de tiempo. Estas extensiones serían las de implementar la creación de switchs, contextos, áreas, optimización de la generación de código ncl, realización de módulo para visualizar diagrama de estados generado automáticamente, editor de código NCL para personas con conocimientos en el lenguaje, reproductor de vista previa de medias, editor de código Lua para personas con conocimiento del lenguaje, implementación de contextos, entre otras.

7 Referencias

1. **Balaguer, Federico and Isasmendi, Leonardo.** *TV Digital – Ginga NCL.*
2. **Soares, Luiz Fernando Fernando Gomes.** *Construindo Programas Audiovisuais Interativos Utilizando a NCL 3.0 e a Ferramenta Composer.*
3. **Soares, Luis Fernando Gomez and Simone Diniz, Junqueira Barbarosa.** *Promando em NCL: Desenvolvimento de Aplicações para middleware Ginga, TV Digital e WEB.* Rio de Janeiro : Elsevier, 2009.
4. **Sant'Anna, Francisco and Soares Neto, Carlos de Salles.** *Desenvolvimento de Aplicações Declarativas para TV Digital no Middleware Ginga com Objetos Imperativos Lua.* Fortaleza, Brasil : s.n., 2009.
5. **Sant'Anna, Francisco, Soares Neto, Carlos de Salles and Azevedo, Roberto Gerson de Albuquerque.** *Declarative Applications Development for Digital TV middleware Ginga using Imperative NCLua Objects.* 2011.
6. **Soares Neto, Carlos de Salles and Junqueira Barbosa, Simone Diniz.** *Declarative Applications Development for Interactive Digital TV.* 2011.
7. **Gomes Soares, Luiz Fernando.** *Introduction to Digital TV and to Ginga-NCL.* 2009.
8. **Junqueira Barbosa, Simone Diniz and Gomes Soares, Luiz Fernando.** *JAI2008 Tutorial - Interactive Digital TV in Brazil is made up with Ginga.* 2008.
9. **Ginga-NCL: Declarative Middleware for Multimedia IPTV Services.** **Gomes Soares, Luiz Fernando and Ferreira Moreno, Marcio.** 2010.

“CreaTV Digital”: Herramienta para la creación de aplicaciones NCL para el middleware
GINGA de Televisión Digital Terrestre 15

10. *Towards the Convergence of Digital TV Systems*. **Gomes Soares, Luiz Fernando and Ferreira Moreno, Marcio**. 2010.

11. *As Múltiplas Possibilidades do Middleware Ginga*. **Gomes Soares, Luiz Fernando**. 2008.

12. **Balaguer, Federico and Isasmendi, Leonardo**. *Desarrollo de Aplicaciones NCL para TV Digital*.

13. —. *Connectors y links*.

14. —. *Reuso a partir de contextos y switches*.

15. —. *Codificación en Lua*.

Apéndice A: Ejemplo de generación de una aplicación de TVD interactiva usando "CreaTV Digital".

1 Ejemplo de uso de la aplicación junto con su código generado

Las imágenes a continuación muestran un ejemplo completo de creación de un proyecto, generación de código y por último la ejecución en Ginga.

Mostraremos un video principal ocupando toda la pantalla y la imagen de un botón en la esquina inferior derecha al inicio de la aplicación. Cuando el usuario presione el botón azul del control remoto, aparecerá en la esquina superior derecha el video miniatura. Además cuando termine el tiempo de vida de la imagen del botón, el video en miniatura será detenido.

Como video principal (puerta) importaremos al archivo de tipo video "04.mpg", el media de tipo imagen representando al botón será el archivo "boton.png". Para el video en miniatura, usaremos el archivo "dido.mpeg".

Crearemos una región ocupando toda la pantalla para el video principal (alto=100%, ancho=100%, ejex= 0px, ejey=0px), un región para el botón y una para el video en miniatura.

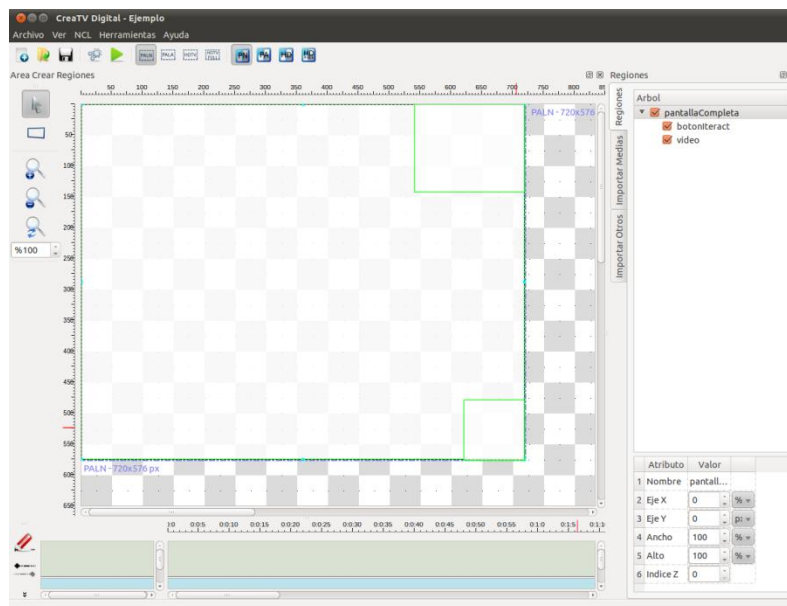


Fig. 10. Sección área espacial con tres regiones creadas

Apéndice A:
Ejemplo de generación de una aplicación de TVD interactiva usando "CreaTV Digital".
II

Una vez asignados los medias a las regiones, en la sección de área temporal, arrastraremos "04.mpg" a la sección de dibujo de líneas de tiempo, a continuación arrastraremos el media "bonton.png" de manera que coincida el inicio del video principal con el de la imagen. Luego, con la herramienta de creación de interactividad abarcaremos todo el media y seleccionaremos el botón azul.

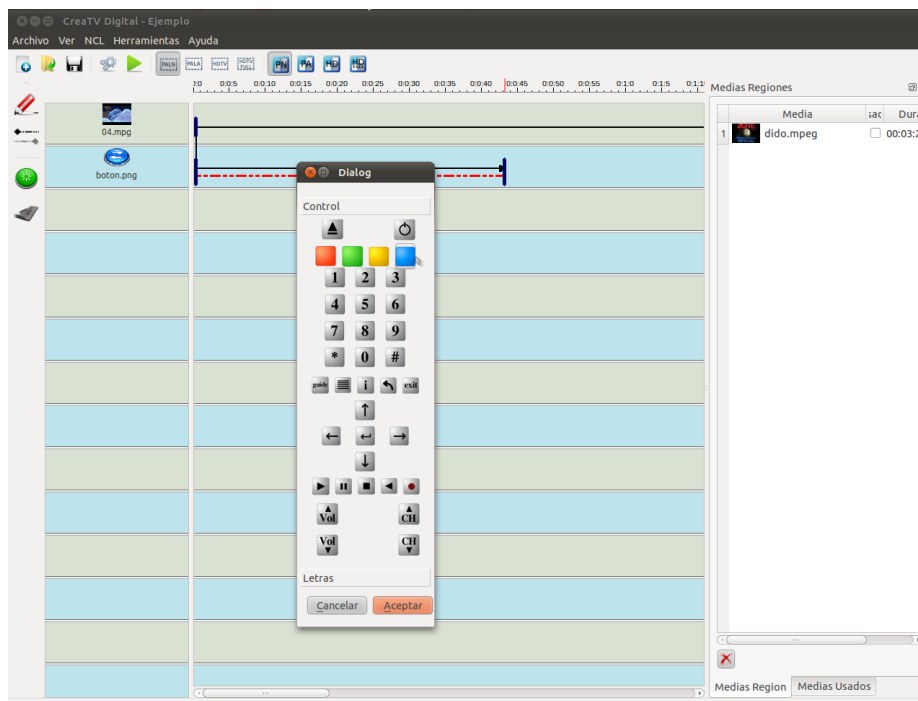


Fig. 11. Sección área temporal mostrando el control remoto para la selección de un botón

Luego de seleccionar el botón azul, se verá una línea punteada de color rojo con un botón azul debajo de la línea de tiempo del media, mostrando la sección de interactividad del media. En este ejemplo, la interactividad ocupa toda la vida del media, aunque podría ocurrir que necesitemos hacer la interactividad sobre una determinada sección de tiempo de vida del media.

Apéndice A:
Ejemplo de generación de una aplicación de TVD interactiva usando "CreaTV Digital".
III

Nos quedaría arrastrar el media, que será presentado como miniatura, hacia el botón azul y asignarle al final de la vida de tiempo del media “botón.png” la acción de detener al media “dido.mpeg”.

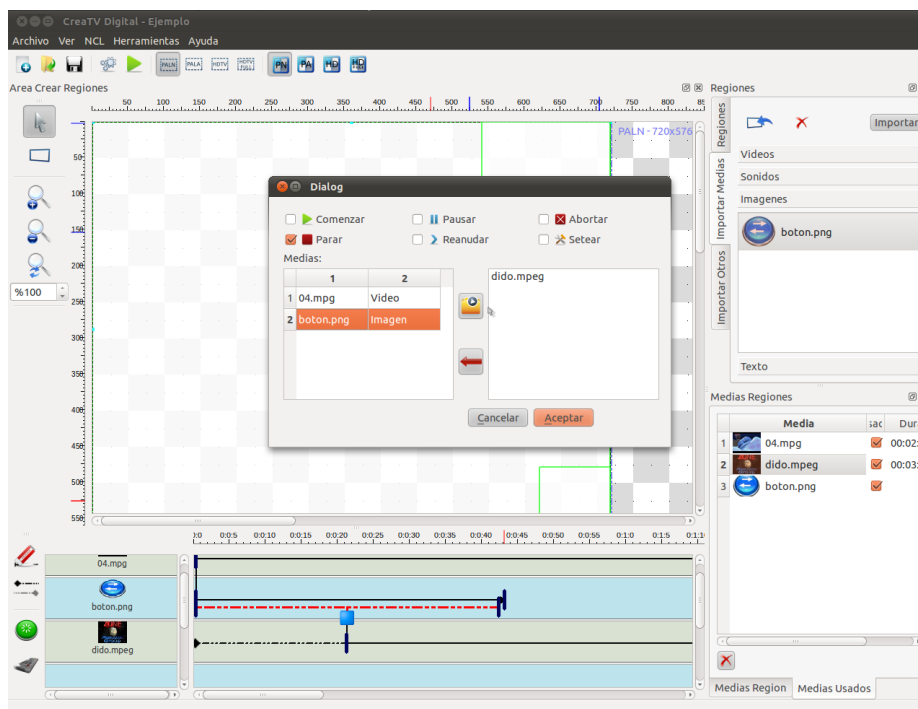


Fig. 12. Asignación a “botón.png” la acción de detener a “dido.mpeg”

Luego de arrastrar el media “dido.mpeg” sobre el botón azul veremos, además de la línea de tiempo del media, una línea punteada negra al comienzo (para indicar la variabilidad de inicio según interacción) y al final de la línea de tiempo del media (graficando la variabilidad de fin según interacción).

Apéndice A:

Ejemplo de generación de una aplicación de TVD interactiva usando "CreaTV Digital".

IV

Ahora generamos el código NCL y lo ejecutamos con el botón de ejecución.



Fig. 13. Aplicación NCL corriendo bajo Ginga donde se muestra el video principal y el botón

Por último verificamos si la interactividad funciona apretando el botón azul para que aparezca el video en miniatura en la esquina superior derecha de la pantalla.



Video miniatura "dido.mpeg"

Imagen botón "boton.png"

2 Código generado por la aplicación

```

1. <?xml version="1.0" encoding="ISO-8859-1"?>
2. <ncl id="Cardozo-Schwartz"
   xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">

3.   <head>
4.     <regionBase>
5.       <region id="pantallaCompleta" left="0" top="0"
   height="100%" width="100%" zIndex="0">

6.         <region id="botonInteract" left="620" top="478"
   height="100" width="100" zIndex="1">

7.           </region>
8.           <region id="video" left="75%" top="0"
   height="25%" width="25%" zIndex="1">

9.             </region>
10.          </region>
11.        </regionBase>
12.      <descriptorBase>
13.        <descriptor id="descriptor-04.mpg"
   region="pantallaCompleta">

14.          </descriptor>
15.          <descriptor id="descriptor-dido.mpeg"
   region="video">

16.            </descriptor>
17.            <descriptor id="descriptor-boton.png"
   explicitDur="42.8s" region="botonInteract">

18.              </descriptor>
19.            </descriptorBase>
20.          <connectorBase>
21.            <importBase documentURI=
   "../connector/causalConnBase.ncl" alias="conEx"/>

22.          </connectorBase>
23.        </head>

```

```

24. <body >
25.     <port id="Port-04.mpg" component="04.mpg"/>
26.     <media id="04.mpg" src="../04.mpg"
      type="video/mpeg" descriptor="descriptor-04.mpg">
27.     </media>
28.     <media id="dido.mpeg" src="../dido.mpeg"
      type="video/mpeg" descriptor="descriptor-dido.mpeg">
29.     </media>
30.     <media id="boton.png" src="../boton.png"
      descriptor="descriptor-boton.png">
31.         <area id="areal" begin="0s" end="42.8s"/>
32.     </media>
33.     <link id="Lboton.png"
      xconnector="conEx#onKeySelectionStart">
34.         <bind role="onSelection" component="boton.png"
      interface="areal">
35.             <bindParam name="keyCode" value="BLUE"/>
36.         </bind>
37.         <bind role="start" component="dido.mpeg">
38.         </bind>
39.     </link>
40.     <link id="LonBeginboton.png"
      xconnector="conEx#onBeginStart">
41.         <bind role="onBegin" component="04.mpg">
42.         </bind>
43.         <bind role="start" component="boton.png">
44.         </bind>
45.     </link>
46.     <link id="LStopdido.mpeg"
      xconnector="conEx#onEndStop">
47.         <bind role="onEnd" component="boton.png">
48.         </bind>
49.         <bind role="stop" component="dido.mpeg">
50.         </bind>
51.     </link>
52. </body>
53. </ncl>

```