

# Introducción Aplicaciones Declarativas (Ginga-NCL)

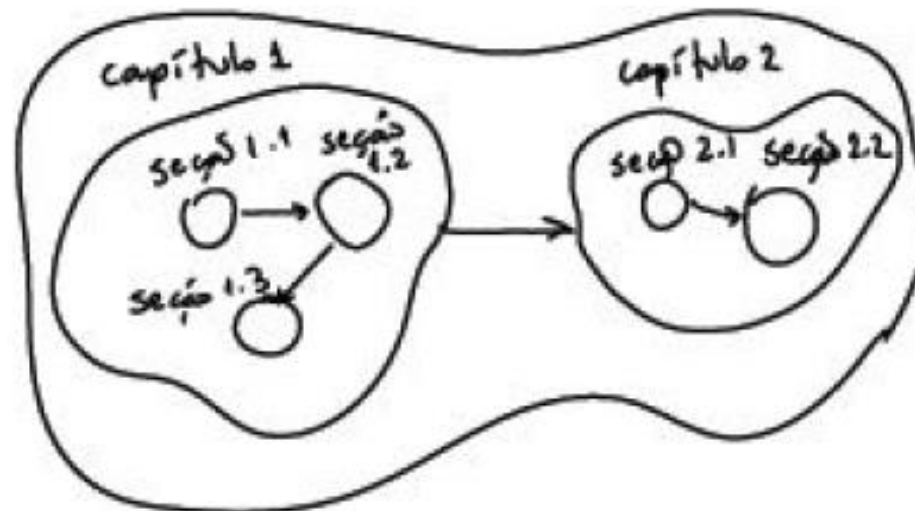
Mauricio Espinoza

# Modelo de contexto anidado (NCM)

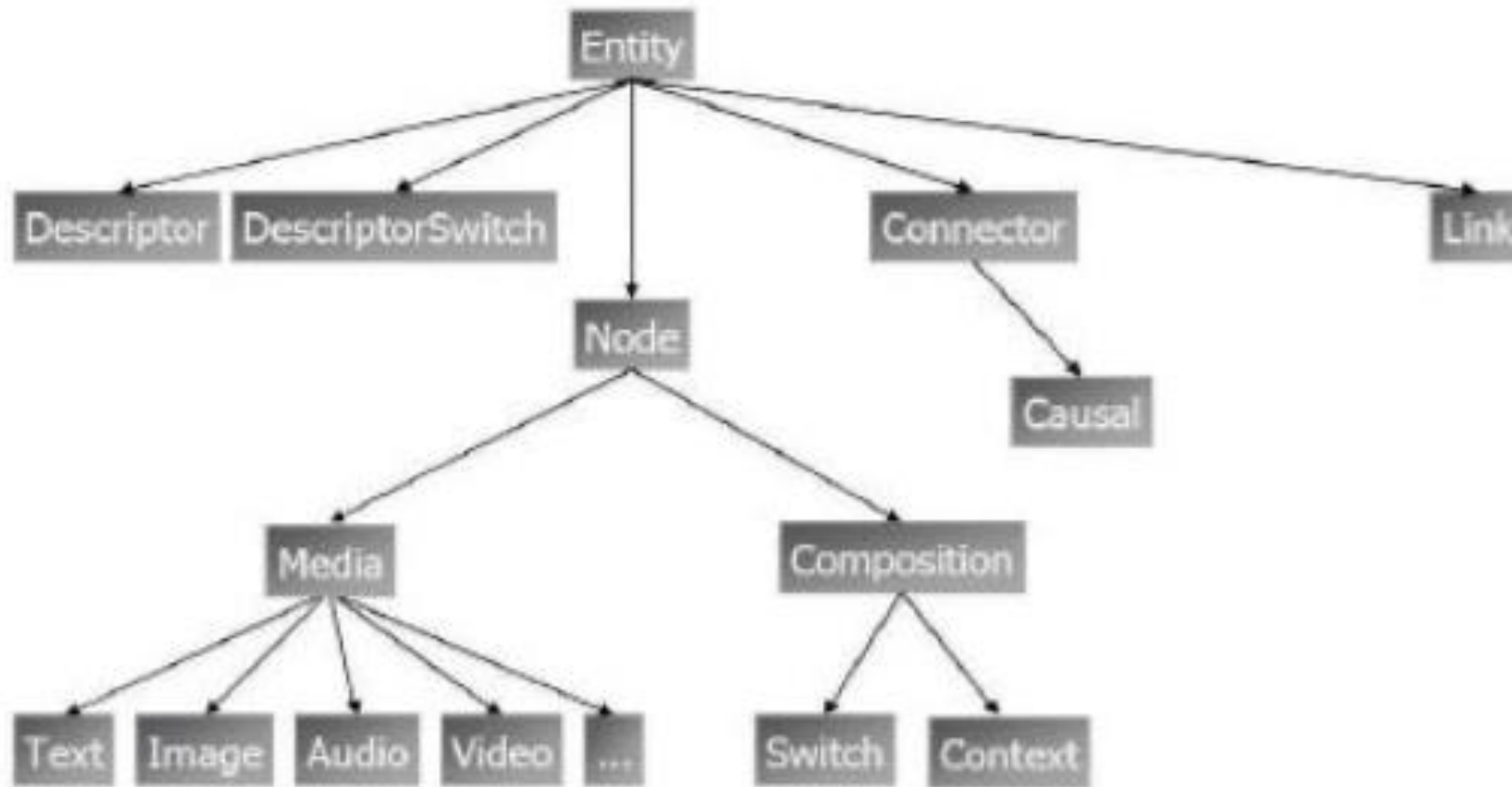
- NCM es un modelo
  - representación y manipulación de documentos hipermedia.
  - reglas y las operaciones
- documento hipermedia
  - nodos y enlaces (links), donde
  - cada nodo representa una media y
  - cada enlace representa una relación entre medias

# Modelo de contexto anidado (NCM)

- NCM dos tipos de nodos
  - contenido: audio, video, texto, aplicación
  - contexto: contiene otros nodos de composición, dar estructura documento
- NCM es el modelo subyacente al **NCL**



# Entidades básicas NCM

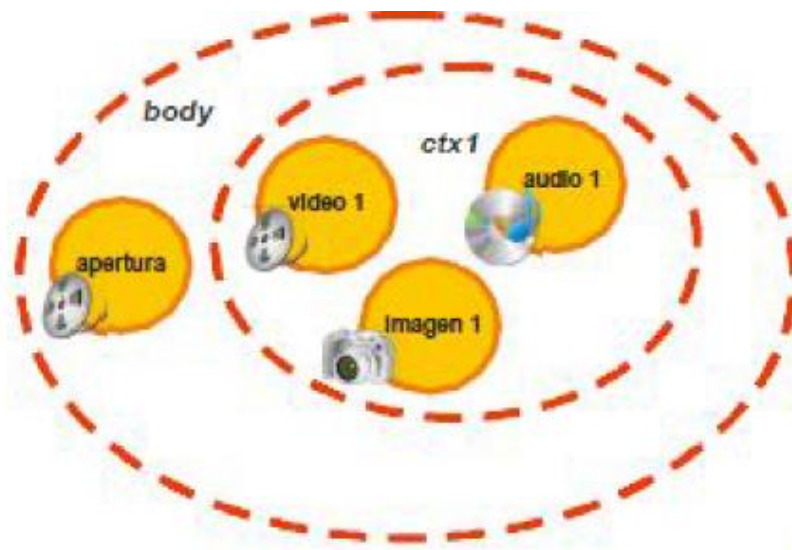


# Estructura de un documento Hipermedia.

¿Qué vamos a mostrar?	• Objetos Media
¿Dónde los vamos a mostrar?	• Regiones
¿Cómo los vamos a mostrar?	• Descriptores
¿Cuándo los vamos a mostrar?	• Links y Conectores
¿Cómo reusamos código?	• Contextos

# ¿Qué vamos a mostrar? - Objetos Media

- contenido audiovisual interactivo lo primero que se escoge
  - vídeos, imágenes, textos y otros medios
- En NCM, una media debe estar dentro de un nodo de composición, llamada contexto,

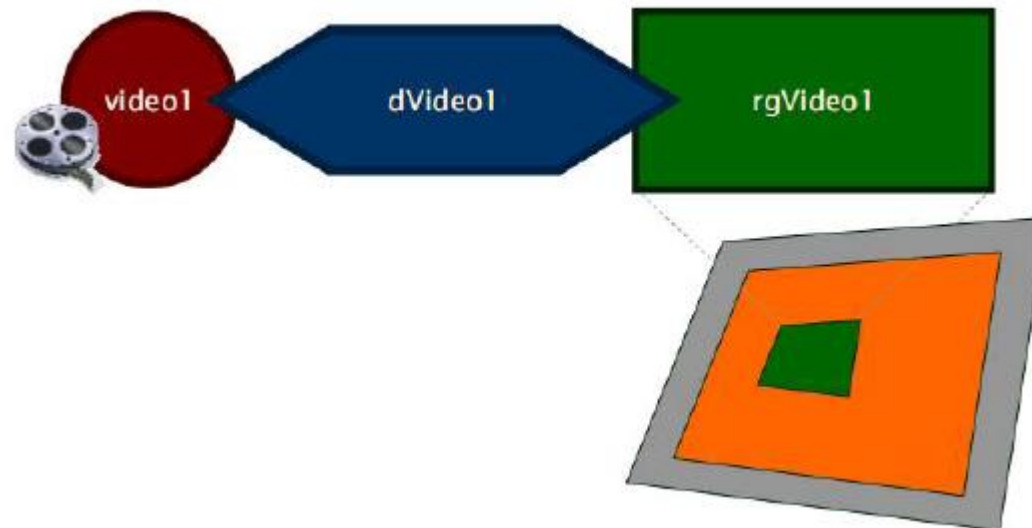


# ¿Dónde los vamos a mostrar? - Regiones

- Una región representa la posición y tamaño de la zona donde ciertos objetos media serán visualizados
- Una región sirve para inicializar la posición de los medios en una ubicación específica
- Una región representa el lugar en donde se podría presentar un nodo, pero no indica que nodo media será presentado en dicha región

# ¿Cómo los vamos a mostrar? - Descriptores

- Un descriptor puede detallar parámetros de representación de los nodos
  - región donde tendrá lugar la presentación de su volumen, su transparencia, tiempo de duración, entre otros.

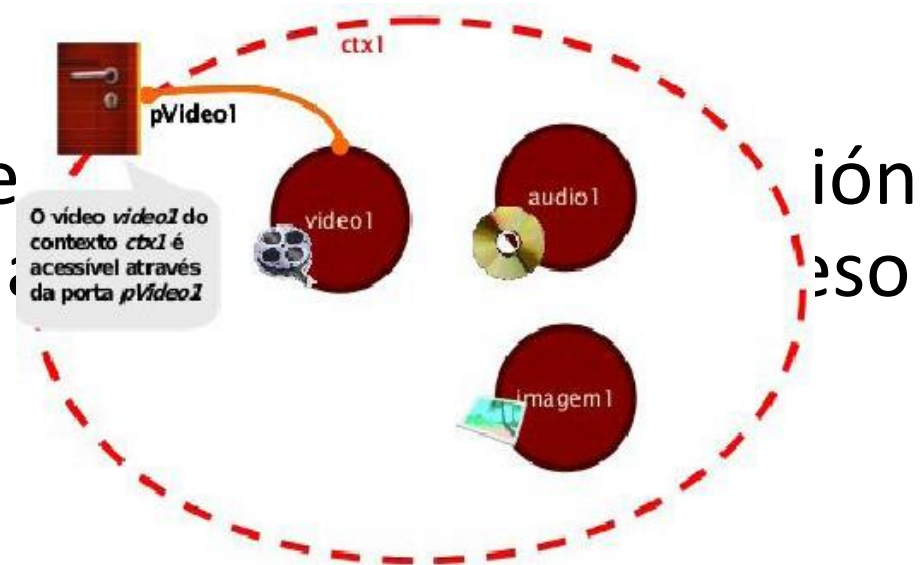




# ¿Cuándo los vamos a mostrar? - Links y Conectores

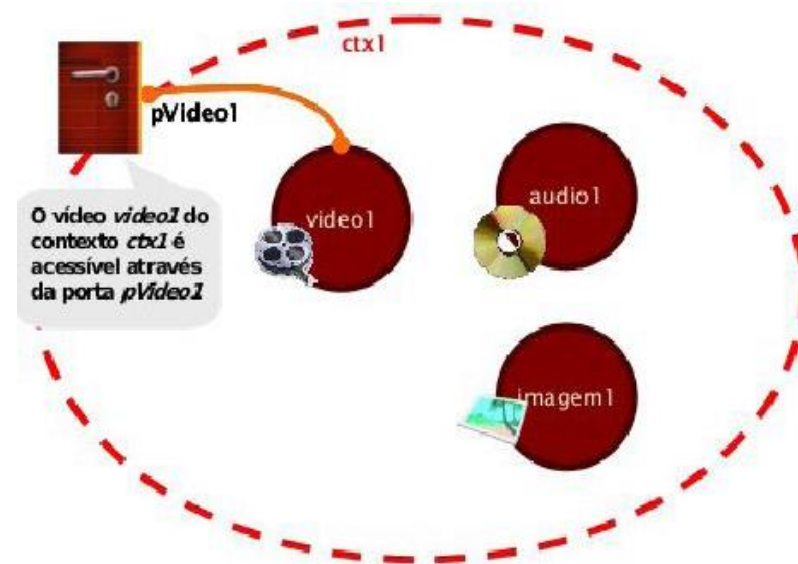
- Elemento **puerto**, define los nodos que serán presentados cuando un nodo contexto inicia.
  - Si hay más de un puerto en el contexto cuerpo, se abren todos en paralelo.
  - Los puertos dan acceso a los nodos

- Los links no definen todas las relaciones entre los nodos y la interactividad; requiere el uso de conectores



# ¿Cuándo los vamos a mostrar? - Links y Conectores

- Los puertos son necesarios para dar acceso a los nodos



# Lenguaje NCL

- lenguaje declarativo XML basado NCM
- lenguaje de script Lua → variables
- A diferencia de XHTML o HTML tiene una separación estricta entre el contenido y la estructura de un documento

# Estructura de un documento NCL.

- Todo contenido de un documento NCL está definido dentro del elemento <ncl>
- Su estructura está dividida
  - cabecera (head)
  - cuerpo del texto (body)
- Documento NCL debe incluir obligatoriamente
  - instrucciones de procesamiento

```
<?xml versión="1.0" encoding="ISO-8859-1" ?>
```

```
<ncl id="qualquer string" xmlns="http://www.ncl.org.br/NCL3.0/profileName">
```

# Estrutura de um documento NCL.

Encabezado de arquivo NCL	1. <?xml version="1.0" encoding="ISO-8859-1"?> 2. <ncl id="exemplo01" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ncl.org.br/NCL3.0/EDTVProfile http://www.ncl.org.br/NCL3.0/profiles/NCL30EDTV.xsd">	1
Encabezado de programa	3. <head>	
Base de regiones	4. <regionBase> 5. <!-- regiões da tela onde as mídias são apresentadas --> 6. </regionBase>	2
Base de descritores	7. <descriptorBase> 8. <!-- descritores que definem como as mídias são apresentadas --> 9. </descriptorBase>	3
Base de conectores	10. <connectorBase> 11. <!-- conectores que definem como os elos são ativados e o que eles disparam --> 12. </connectorBase>	8
Cuerpo de programa	13. </head>	
Punto de entrada al programa	14. <body> 15. <port id="plnicio" component="ncPrincipal" interface="ilnicio"/>	5
Contenido de programa	16. <!-- contextos, nós de mídia e suas âncoras, elos e outros elementos --> 17. </body>	6 7 4
Termino	18. </ncl>	

# Herramientas

- Hay dos formas de usar Ginga en una PC:
  - instalación nativa
  - máquina virtual con Ginga pre-instalado
- máquina virtual se puede simular dentro del sistema ya existente (Windows, Linux, MacOS)
  - facilidad para poner a funcionar el ambiente
  - difícil manejar archivos entre dos S.O.
- instalación de forma nativa permite ejecutar Ginga en el sistema operativo existente
  - mejora la velocidad de la PC y simplicidad en el manejo de archivos.
  - proceso de instalación es mucho más complicado

# Herramientas

- Desarrollo de aplicaciones interactivas Ginga se utilizan dos tipos de herramientas:
  - De desarrollo
  - De presentación.

# Herramientas de desarrollo

- Es posible crear programas interactivos en cualquier editor XML
- Factores para escoger herramientas
  - nivel de conocimientos de programación
  - complejidad de la aplicación interactiva
- Composer.
- Eclipse NCL.

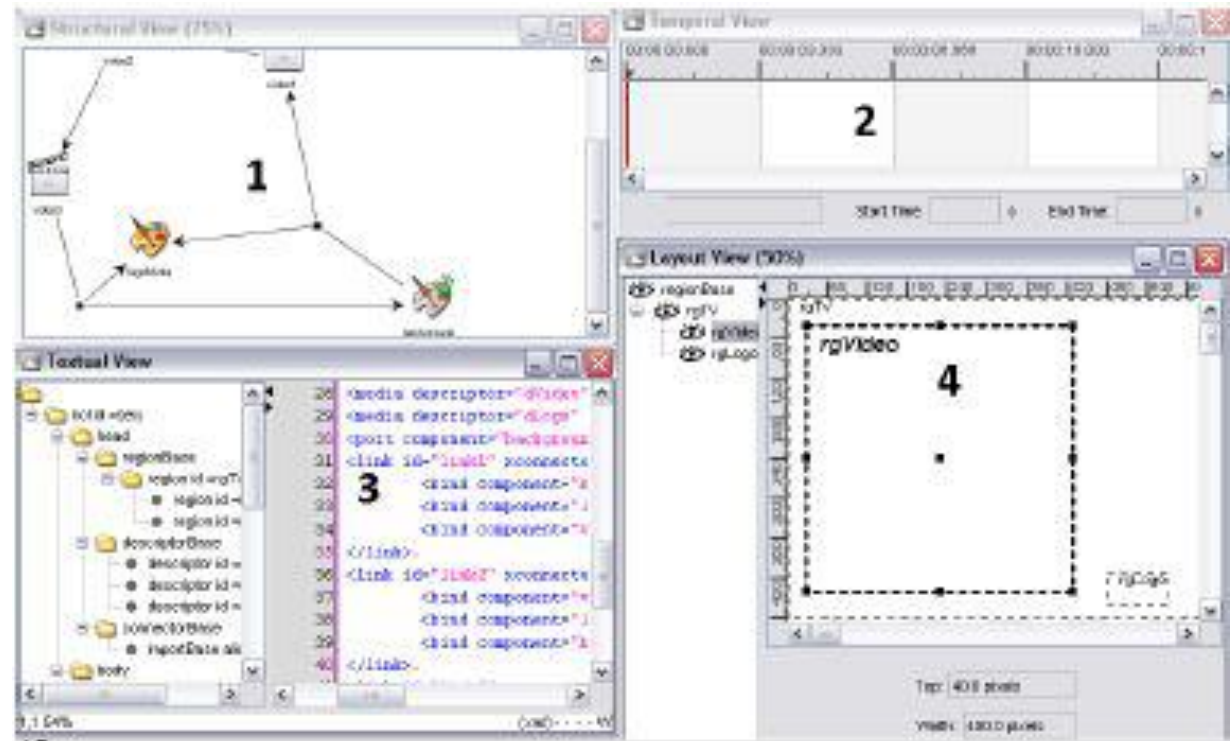


# Composer

- herramienta de software libre
- facilita y agiliza la construcción de documentos NCL para TV digital interactiva
- Laboratorio de Telemidia del departamento de informática de la PUC-Rio.
- versión actual de composer permite al usuario trabajar con 4 tipos de visiones

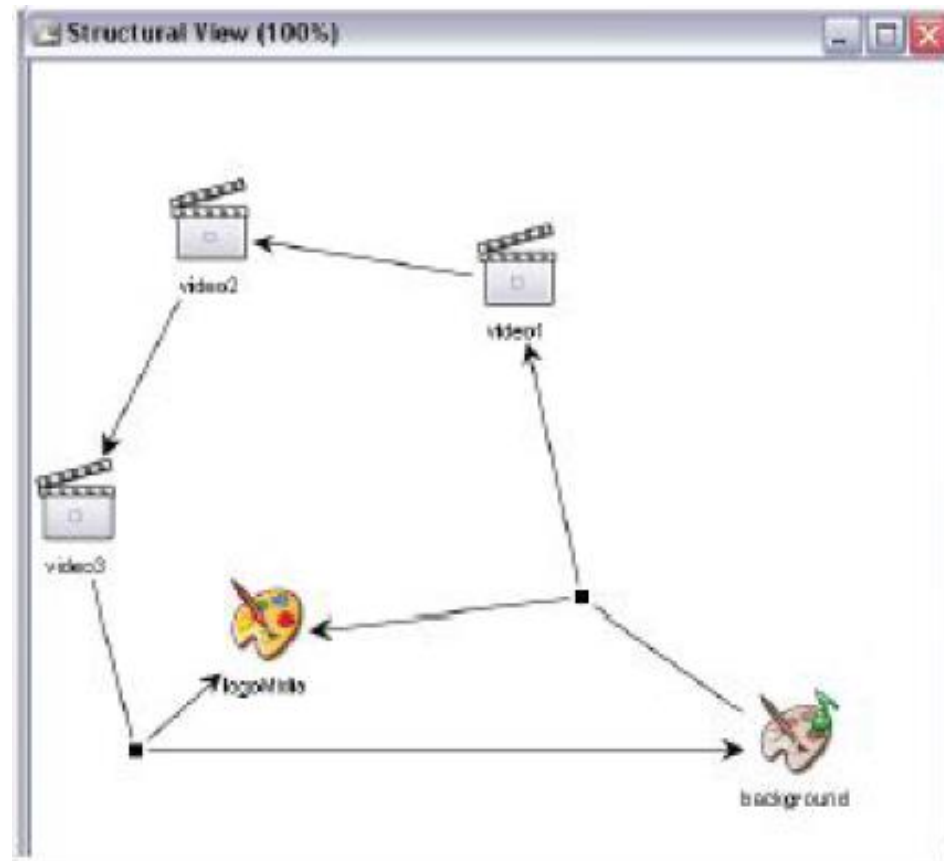
# Composer

1. Visión Estructural.
2. Visión Temporal.
3. Visión de Diseño o Esquema
4. Visión Textual



# Visión Estructural

- crear una estructura lógica de los aplicativos NCL
- crear, editar y eliminar composiciones, objetos de medios y enlaces.



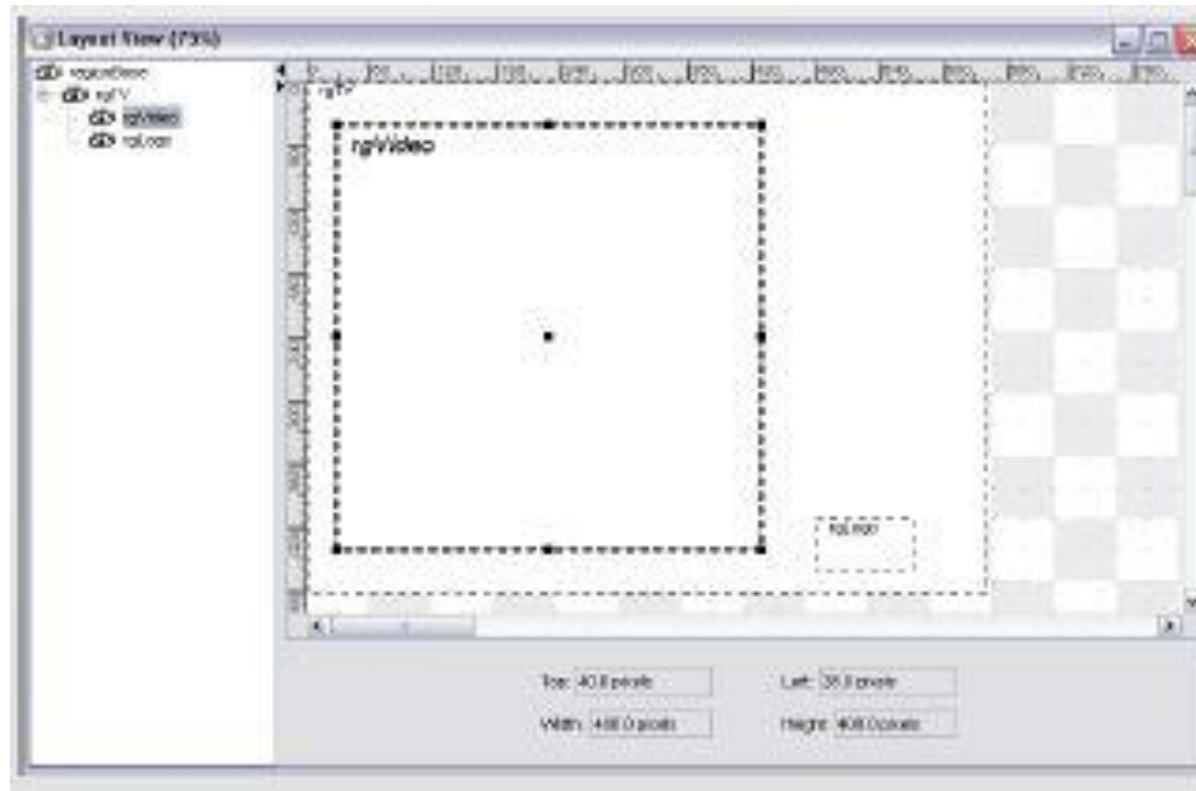
# Visión Temporal

- relaciones temporales para la presentación del documento en la línea del tiempo
- anclas transitorias en audio, vídeo e imágenes.



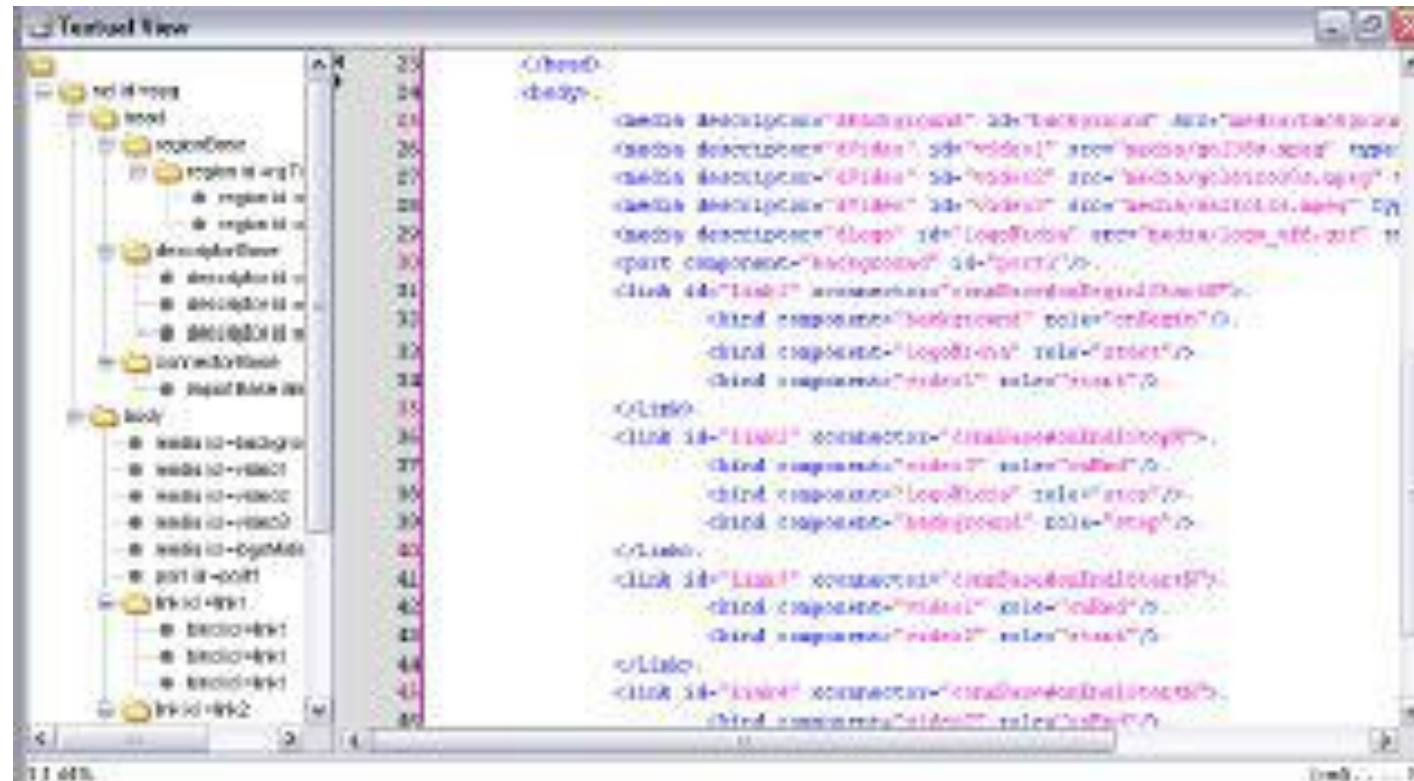
# Visión de Diseño o Esquema

- creación y configuración de zonas donde los medios serán presentados.
- crear sub-regiones o regiones superpuestas.



# Visión Textual

- presenta el código NCL en donde el usuario puede crear directamente el código NCL como un editor de texto común

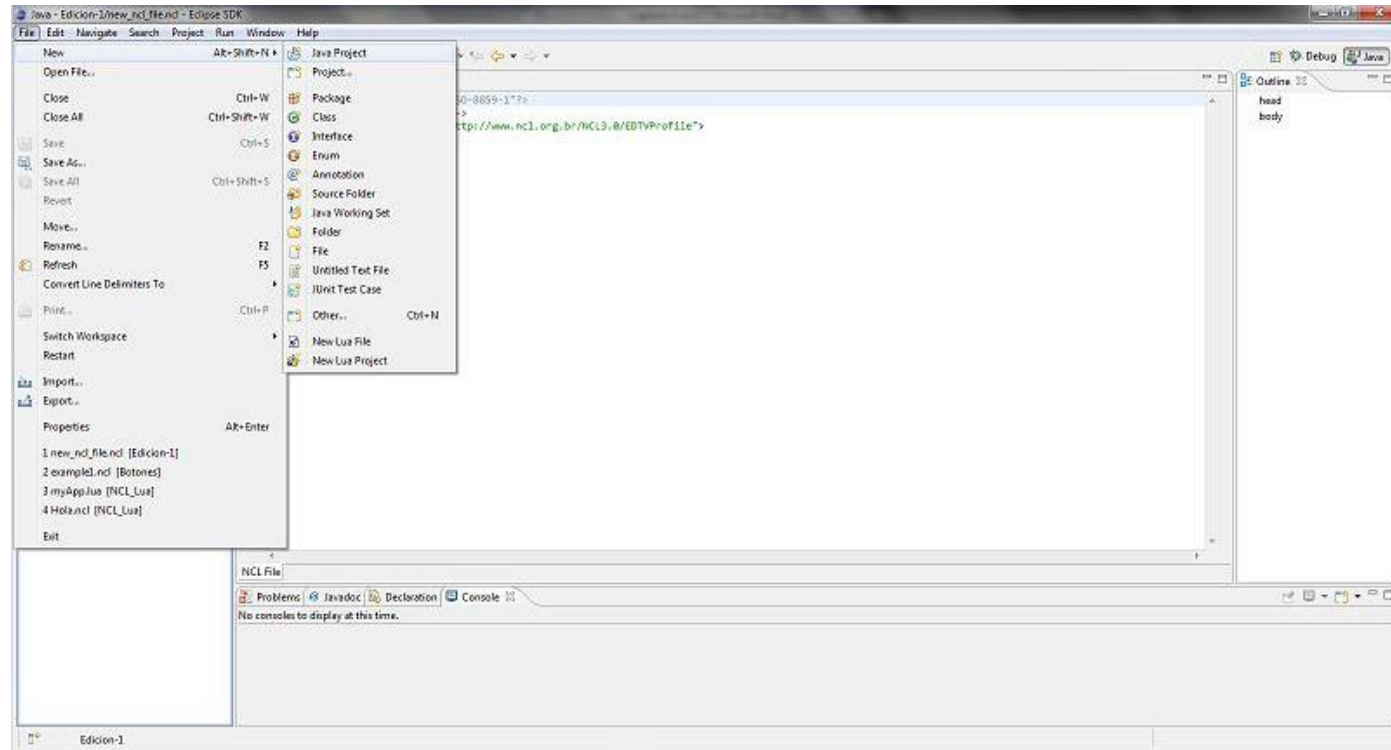


# Entorno Composer

- herramienta muy inestable?
- utiliza GINGA Emulador para mostrar los documentos NCL
- falta de soporte para el lenguaje Lua?
- problemas con las disposiciones de los componentes de interfaz gráfica?
- poca especificación de los errores encontrados?

# Eclipse NCL.

- Universidad Federal do Maranhão desarrolló NCL-Eclipse, un editor XML/NCL
- integración con otras herramientas como Lua Eclipse





# Herramientas de presentación

- Para la visualización de las aplicaciones en entorno Ginga-NCL, existen dos aplicativos más usadas:
  - Emulador Ginga NCL.
  - Set Top Box Virtual.

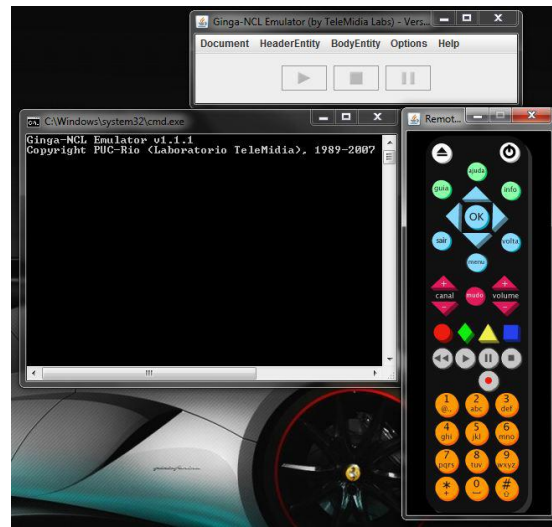
# Emulador Ginga NCL

- herramienta más fácil de usar y más accesible
- ejecutar acciones mediante el control remoto interactivo que es activado con el mouse.
- instalada la Máquina Virtual de Java (JVM)

# Emulador Ginga NCL

## Limitaciones

- No hay soporte para la transparencia
- no posee soporte para el lenguaje Lua.
  - está incrustado en la herramienta composer
  - puede ser instalado como plugin de Eclipse IDE



# Virtual SetTopBox

- implementación C++ del middleware Ginga
- emulada en una máquina virtual para VMWare
- simula el ambiente de presentación de aplicaciones declarativas
- posee un mejor rendimiento y un entorno más parecido
- para ejecutar una aplicación se debe abrir una conexión utilizando el protocolo SSH

# Virtual SetTopBox

- posee soporte de ejecución de los script en LUA
- soporte a la transparencia.

## Requisitos de Hardware:

- Arquitectura Intel.
- Promedio Core Duo.
- Memoria RAM recomendado de 2 GB.
- Placa Aceleradora de Vídeo con 64Mb.
- Disco duro con 10 GB (depende de la cantidad de vídeos almacenados).

# Virtual SetTopBox

## Requisitos de Software:

- Depende del Sistema Operativo en uso:  
Windows XP, Linux, Mac OS X.
- Software de virtualización:
  - VMWare Player (Windows o Linux),
  - VMWare Workstation (Windows/Linux),
  - VMWare Fusion (Mac OS X).