

Linguagem NCL

Profa. Débora Christina Muchaluat Saade

debora@midia.com.uff.br

Linguagem NCL

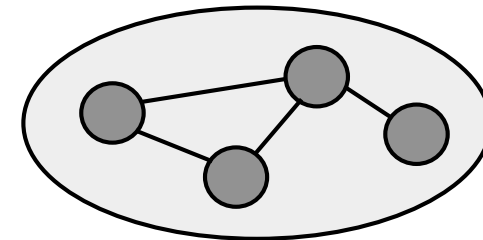
- ø **NCL – Nested Context Language**
 - *Baseada em XML*
- ø **NCL Versão 1.0 – [Antonacci 2000]**
- ø **NCL Versão 2.0 – [Muchaluat-Saade 2003]**
- ø **NCL Versão 3.0**
 - *Middleware declarativo GINGA-NCL*
 - *Padrão Brasileiro de TV Digital – Norma ABNT 2007*
 - *Padrão Internacional para IPTV – Norma ITU H.761 2009*
 - *Integração com linguagem LUA*

Linguagem NCL

- ø **É necessário um modelo conceitual hipermídia para expressar os componentes de um documento**
- ø **Modelo NCM – Nested Context Model**
 - *é usado pela linguagem NCL – Nested Context Language*
- ø **Componentes de um documento são representados pelas entidades do modelo**

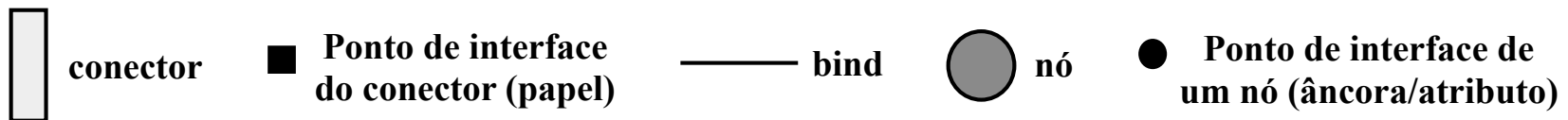
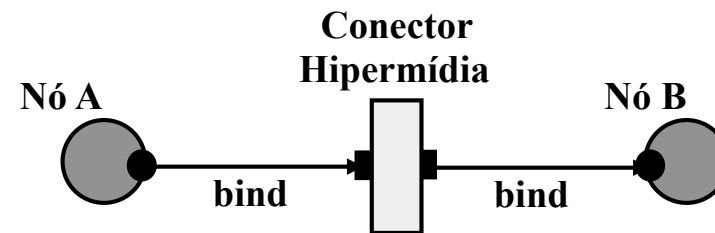
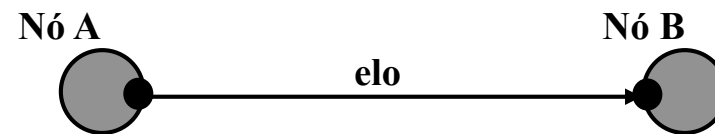
ð Principais entidades hipermídia

- *Blocos de informação (texto, áudio, vídeo, imagem ...)*
 - Nós (nodes)
 - *Âncoras (anchors)*
- *relacionamentos*
 - Elos (links)
 - *conectores*
 - nós de composição/composições/contextos (composite nodes/compositions/contexts)
 - *grupo de nós e grupo de elos*



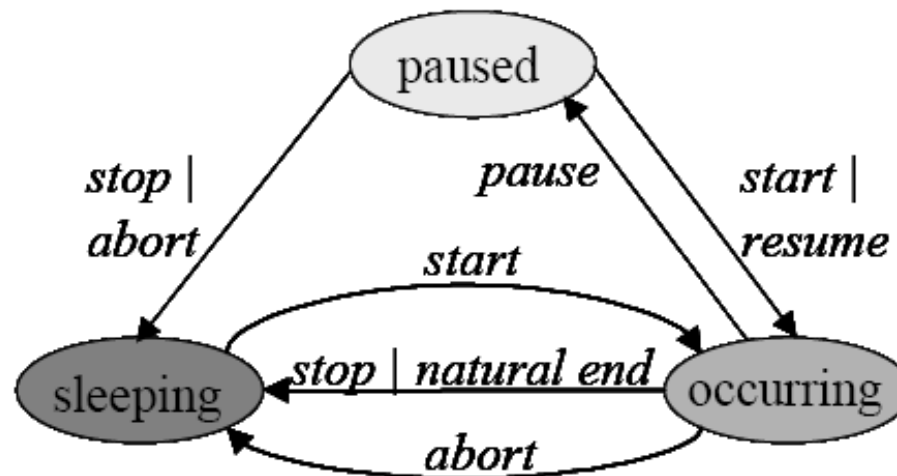
Conectores Hipermedia

- Elos divididos em dois elementos
 - *Definição da relação* => *conector hipermedia*
 - *Definição dos participantes* => *conjunto de binds*



Definição dos Papéis de um Conector

- Modelo de sincronização baseado em eventos
- Ponto de interface de um conector (papel) especifica o comportamento de um participante da relação
- Papéis são definidos baseados em uma máquina de estado de evento:
 - *presentation, selection, attribution*



- Módulo para definição de conectores em NCL
 - *relações de referência (interatividade) e de sincronização*

Exemplos de Conectores

ð Exemplos de Conectores:

ð Elo tradicional hipertexto

- *onKeySelectionStart*

ð Relações temporais de Allen

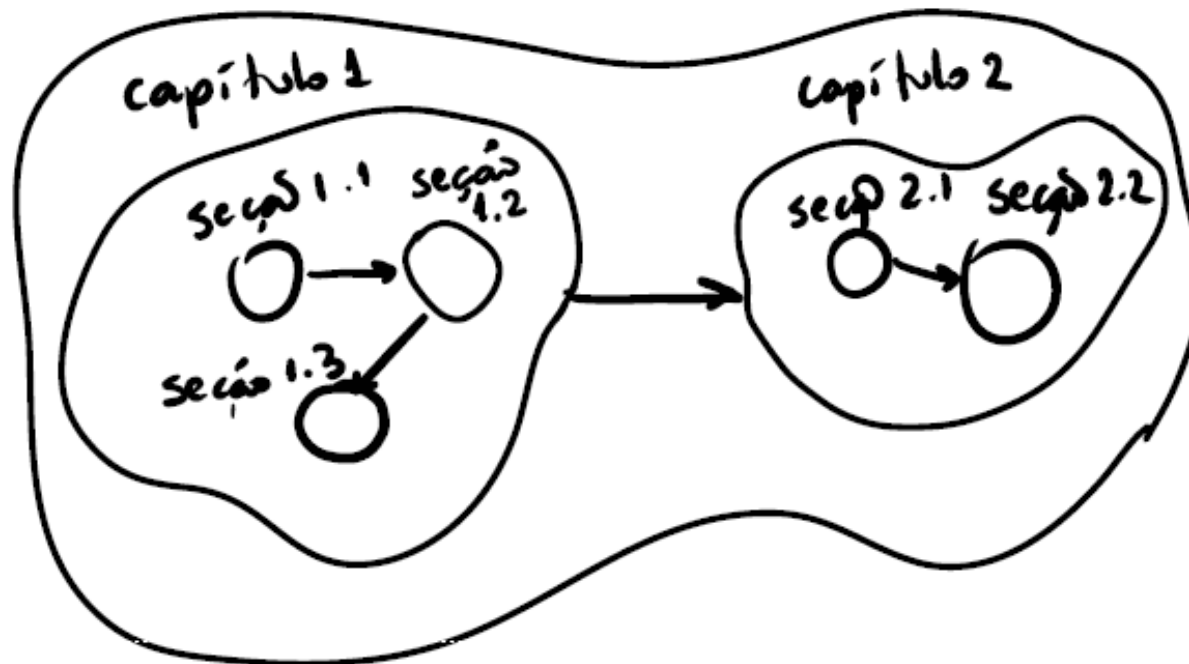
ð Relações multiponto

- *onKeySelectionStartNStopN*

| Relação de Allen | Ilustração | Conector Hipermídia |
|-----------------------------------|------------|-------------------------------------|
| x meets y y met by x | | onEndStart |
| x starts y y started by x | | onBeginStart |
| x finishes y y finished by x | | onEndStop |
| x before y y after x | | onEndStartDelay |
| X overlaps y y overlapped by x | | onBeginStartDelay |
| x during y y contains x | | onBeginStartDelay onEndStopDelay |
| x equals y | | onBeginStart onEndStop |

Nós de Contexto NCL

- **Nó de composição é chamado de nó de contexto**
 - *Grupo de nós e elos*



Estrutura de um documento NCL para TV digital

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

```
<ncl id="exemplo01"  
  xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
```

```
<head>
```

cabeçalho do documento

```
</head>
```

```
<body>
```

corpo do documento

```
</body>
```

```
</ncl>
```

Estrutura do Programa NCL

- ð **Elemento raiz (principal): <ncl>**
- ð **Principais atributos:**
 - *id*
 - *xmlns – namespace XML, que define o perfil de NCL usado:*
 - avançado EDTV (receptores full-seg)
 - básico BDTV (receptores one-seg)
 - *Não suporta animações, transições e metadados*
 - CausalConnector – usado para definição de base de conectores em um arquivo separado
- ð **Elementos filhos**
 - *cabeçalho <head> e*
 - *corpo <body>*

Entidades Básicas

ð **O que exibir?**

- *Nós*

ð **Onde exibir?**

- *Regiões*

ð **Como exibir?**

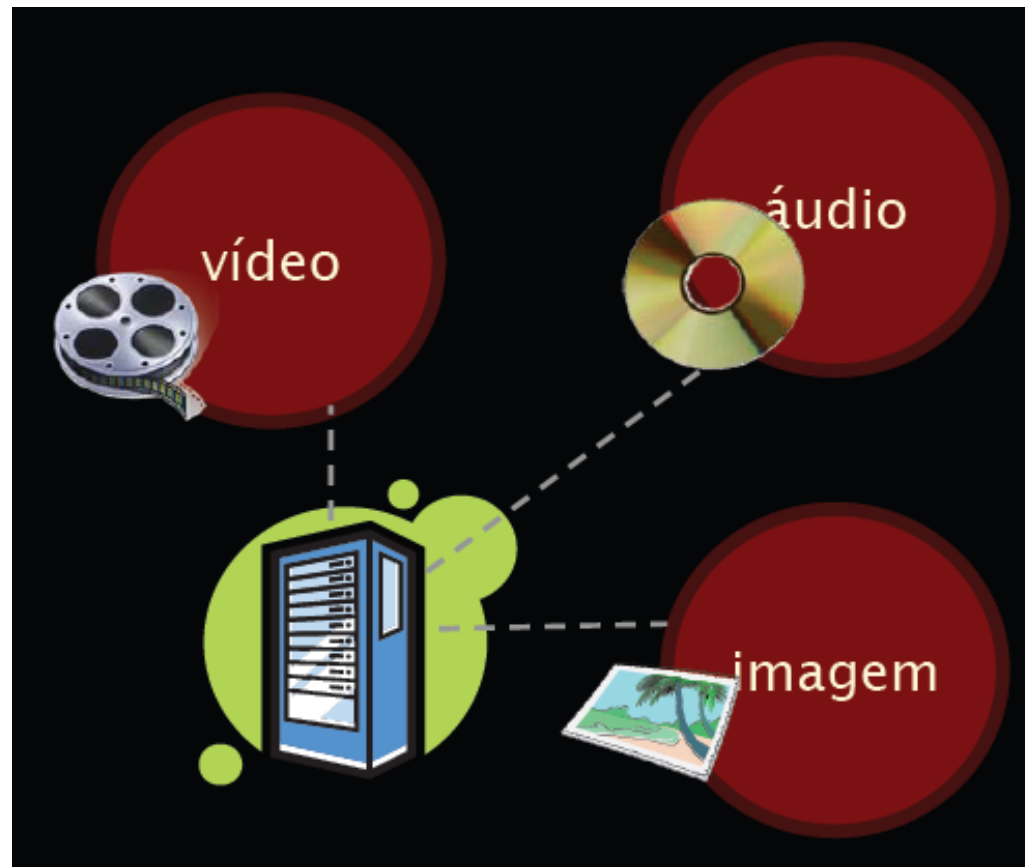
- *Descritores*

ð **Quando exibir?**

- *Elos e conectores*

O que exibir?

ð **Nós** – <media>



Nós de mídia

• **<media>**

• **Principais atributos:**

- *id* – *identificador único do nó no documento*
- *src* – *indica a URI do conteúdo do nó*
- *descriptor* – *indica o id do descritor (características de exibição do nó)*
- *type* – *define o tipo MIME (áudio, vídeo, texto, imagem, etc.)*

• **Um nó de mídia pode definir âncoras para marcar trechos do seu conteúdo**

- *<area begin="2s" end="4s"/>*

• **Um nó de mídia pode definir propriedades que serão atualizadas durante o programa**

- *<property name="soundLevel"/>*

Nós de mídia - exemplos

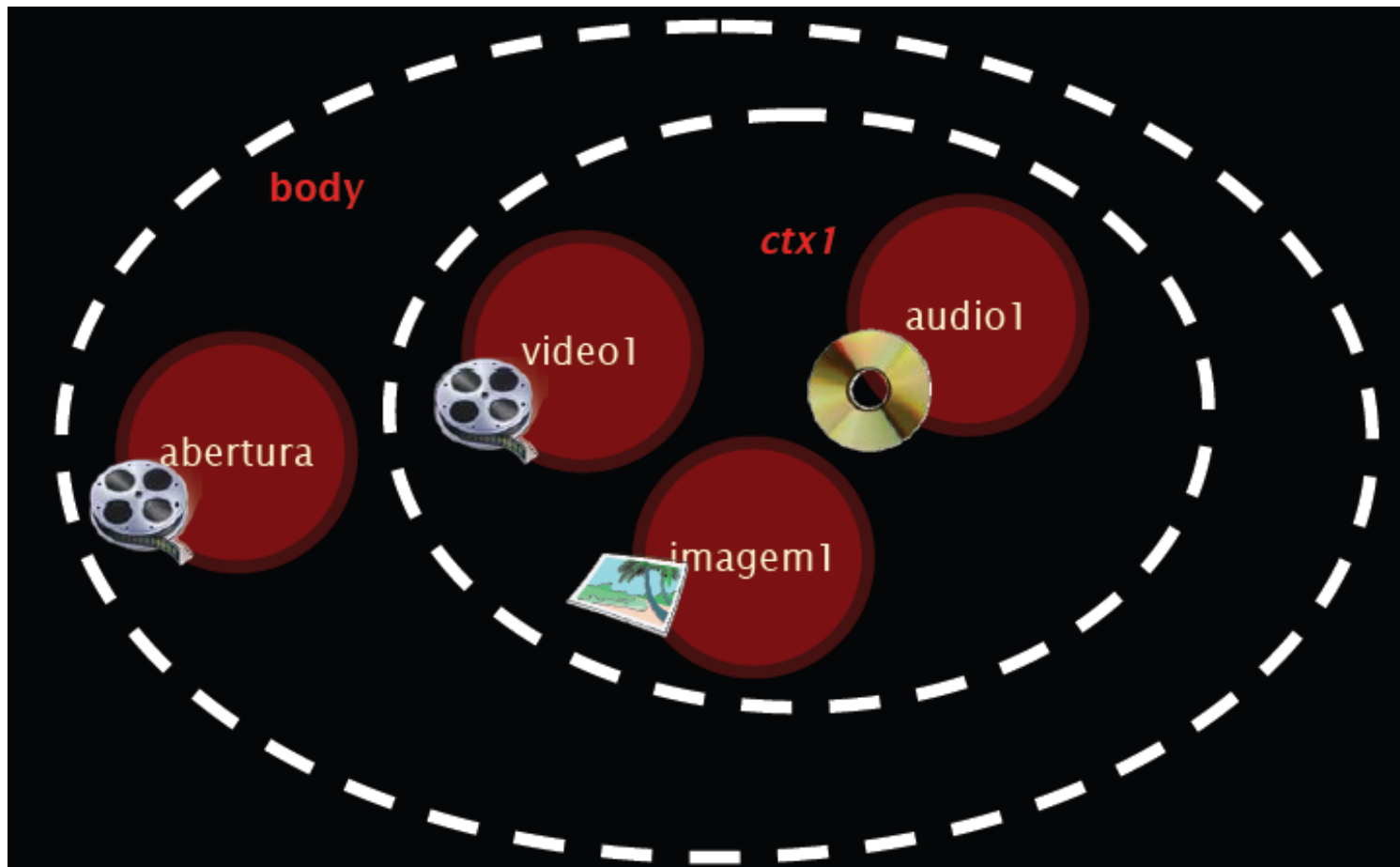
```
<media id="video" src="media/video.mp4" type="video/  
mp4" descriptor="dpVideo">  
  <area id="apresentaIcone" begin="20s" end="40s"/>  
  <area id="apresentaFigura" begin="50s"/>  
</media>
```

```
<media id="choro" src="choro.mp3"  
  descriptor="audioDesc">  
  <property name="soundLevel" value="1"/>  
</media>
```

O que exibir?

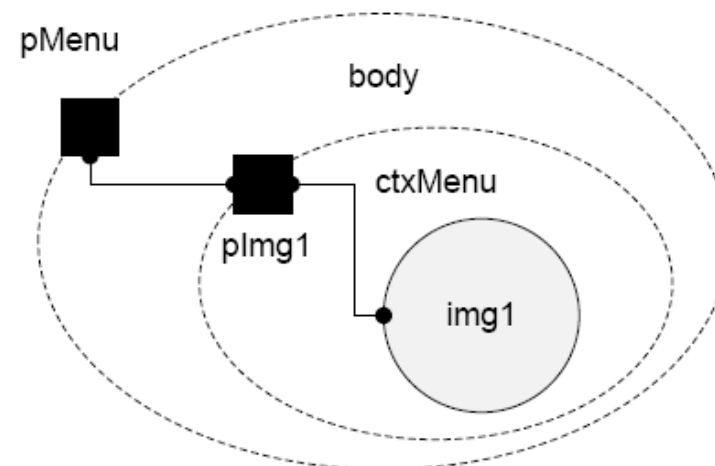
ð Estrutura do documento

- *Nós de contexto* – `<body>` e `<context>`



Nós de Contexto

- ø <context>
- ø Principal atributo:
 - *id – identificador único do nó no documento*
- ø Um nó de contexto define um conjunto de nós, que podem ser outros contextos aninhados, e elos entre seus componentes



Nós de Contexto - Exemplo

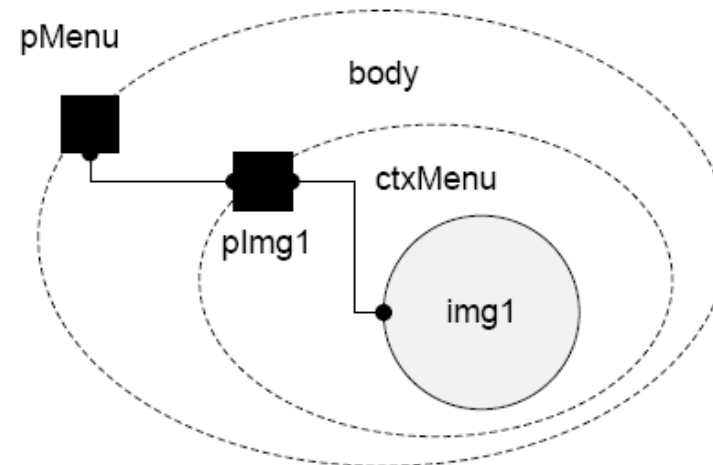
<body>

<context id="ctxMenu">

<media id="img1"/>

</context>

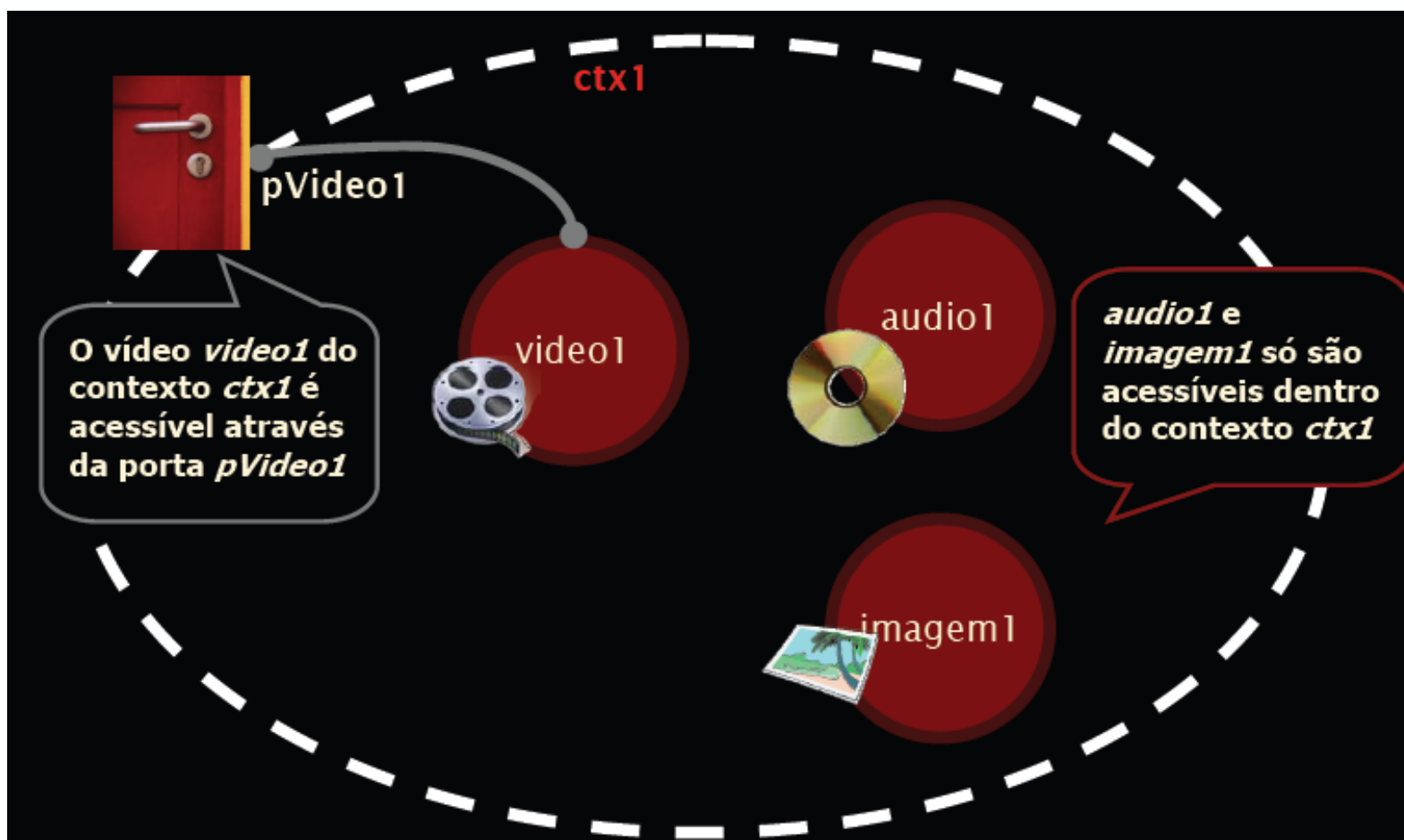
</body>



O que exibir?

ð Como entrar/sair de um contexto?

- *Portas* - *<port>*

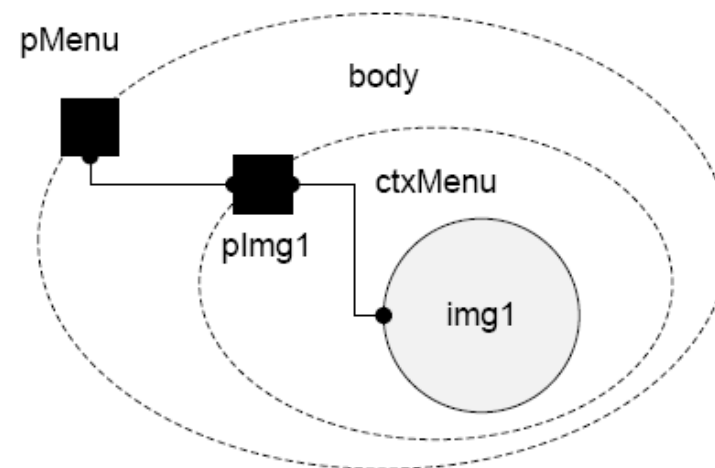


Nós de Contexto - Portas

ð <port>

ð Principais atributos:

- *id* – *identificador único do nó no documento*
- *component* – *indica o id do componente interno*
- *interface* – *indica uma porta/âncora/atributo do componente interno*



Portas - Exemplo

<body>

*<port id="pMenu" component="ctxMenu"
interface="pImg1" />*

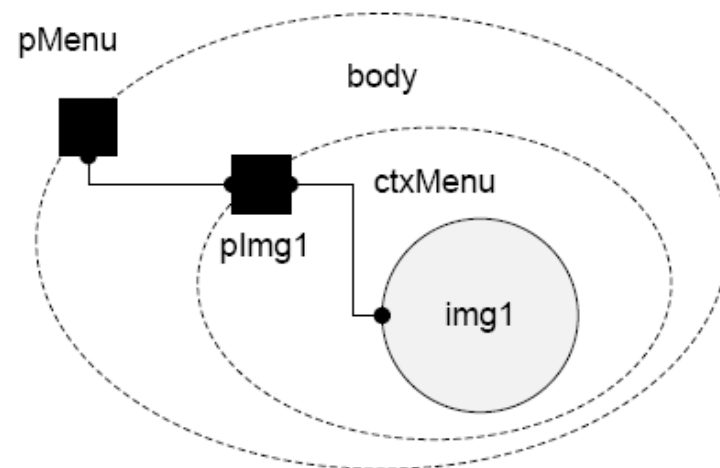
<context id="ctxMenu">

<port id="pImg1" component="img1" />

<media id="img1" />

</context>

</body>



Tipo especial de Nó

ð Nó para guardar variáveis – tipo *settings*

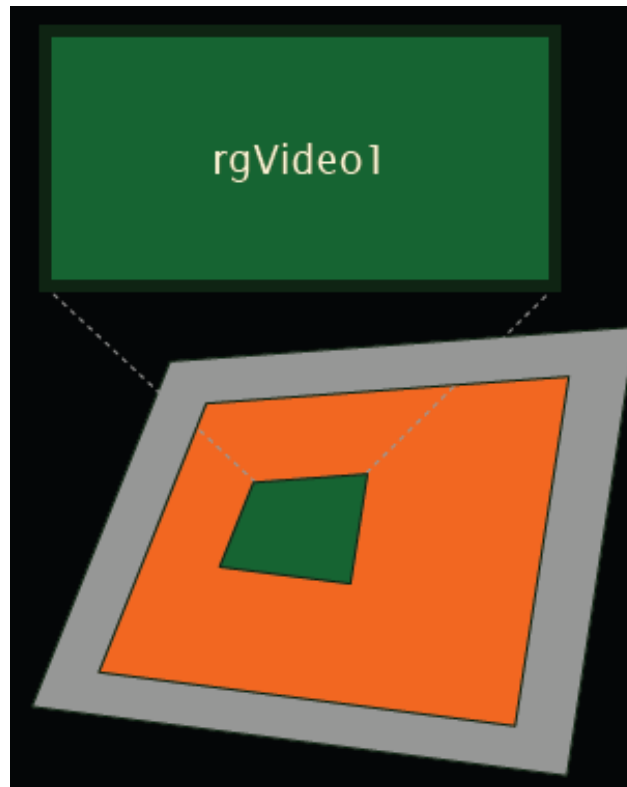
- *Programa pode manipular variáveis durante a execução*

```
<media type=" application/x-ginga-settings" id="nodeSettings">  
  <property name="idioma" />  
</media>
```

Onde exibir?

ð Regiões - <region>

- *Indicam área do dispositivo de exibição que será usada para apresentar algum nó*

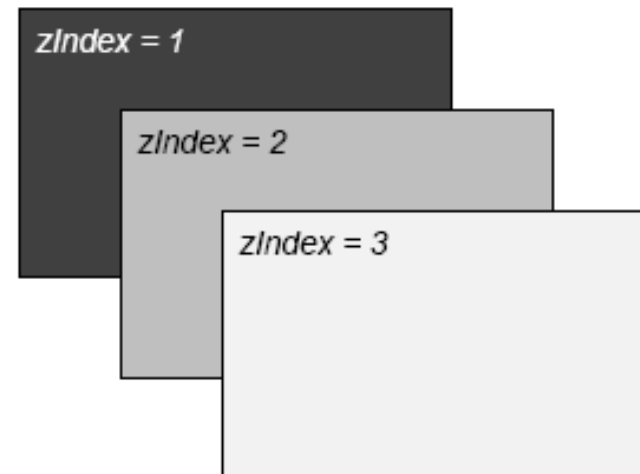
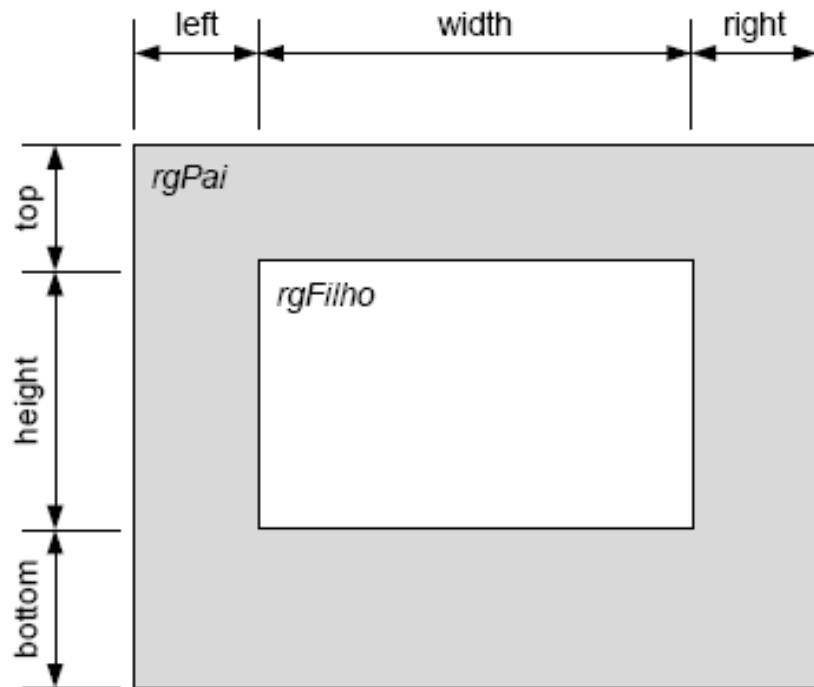


Regiões

- **<regionBase> e <region>**
- **Uma base de regiões agrupa regiões de um mesmo dispositivo de exibição**
- **Pode definir várias bases (TV, celular)**
- **Programa NCL define base de regiões no cabeçalho ou importa base existente**
- **Principais atributos de região:**
 - *id – identificador único*
 - *Coordenadas top, left, height, width, right, bottom*
 - *zIndex – indica sobreposição de regiões*

Regiões

ø atributos



Regiões - Exemplos

<regionBase>

*<importBase alias="regMenu" documentURI="..."
region="..." />*

<region id="TV">

*<region id="rgVideoTV" top="5%" left="5%" width="50%"
height="50%" />*

*<region id="rgImg" top="100" left="100" width="200"
height="200" />*

</region>

<region id="Palm" device="systemScreen(1)">

*<region id="rgVideoPalm" top="0" left="0" width="100%"
height="100%" />*

</region>

</regionBase>

Como exibir?

ð Descritores - <descriptor>

- *Indicam as características de exibição de um nó*



Descritores

- ø <descriptor> e <descriptorBase>
- ø Pode definir várias bases de descritores
- ø Programa NCL define base de descritores no cabeçalho ou importa base existente
- ø Principais atributos de descritor:
 - *id* – *identificador único*
 - *region* – *id de uma região existente*
 - *explicitDur* – *duração explícita*
 - *soundLevel* – *volume de exibição*
 - *Pode definir parâmetros adicionais a serem modificados durante a exibição*

Descritores - Exemplos

<descriptorBase>

<importBase alias="descMenu" documentURI="..." />

<descriptor id="dVideo" region="rgVideoTV"/>

*<descriptor id="dImg" region="rgImg"
explicitDur="4s">*

<descriptorParam name="transparency" value="30%" />

</descriptor>

</descriptorBase>

Descritores

- ð **Facilitam a navegação por teclas**
- ð **Implementação de um menu de opções**
- ð **Atributos:**
 - *focusIndex*
 - *moveLeft, moveRight, moveUp, moveDown*
 - *focusBorderColor, focusBorderWidth*
 - *etc.*

Descritores - Exemplos

<descriptorBase>

*<descriptor id="dMenu1" region="rgMenu1"
focusIndex="1" moveUp="3" moveDown="2"
focusBorderColor="white" focusBorderWidth="3"/>*

*<descriptor id="dMenu2" region="rgMenu2"
focusIndex="2" moveUp="1" moveDown="3"
focusBorderColor="white" focusBorderWidth="3"/>*

*<descriptor id="dMenu3" region="rgMenu3"
focusIndex="3" moveUp="2" moveDown="1"
focusBorderColor="white" focusBorderWidth="3"/>*

</descriptorBase>

Como exibir um nó?

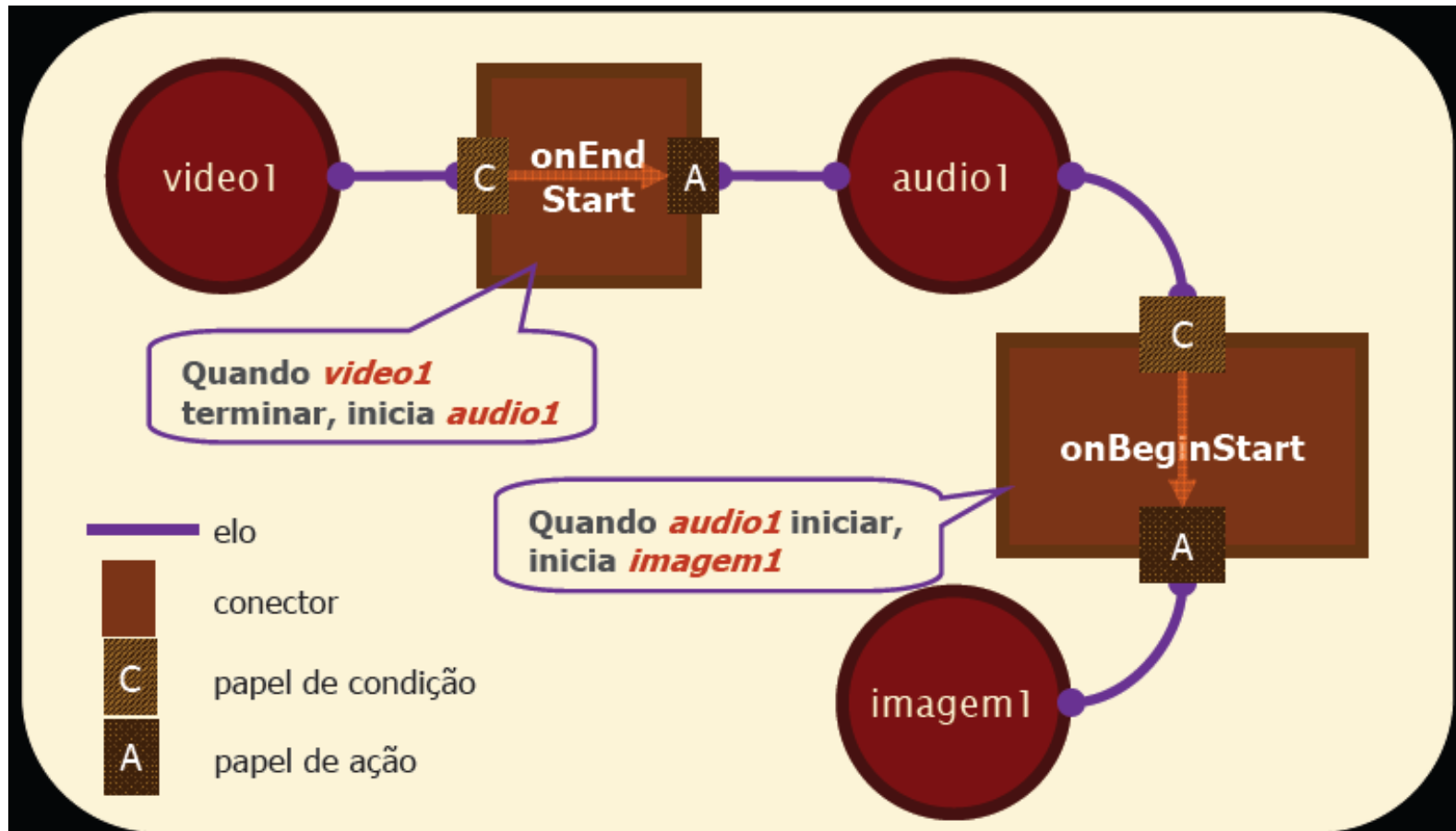
• Todo nó deve ser associado a um descritor, que pode ter uma região associada

- *<region id="rgVideo1" .../>*
- *<descriptor id="dVideo1" region="rgVideo1"/>*
- *<media id="video1" descriptor="dVideo1"/>*



Quando exibir?

ð Elos e conectores



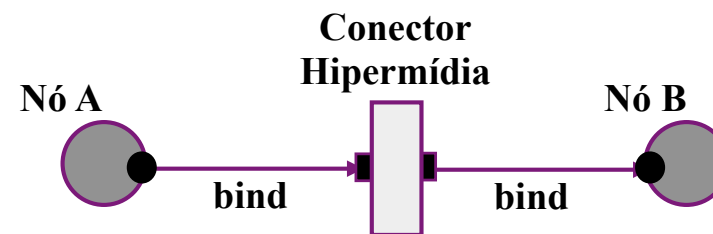
Quando exibir?

ð Como criar um relacionamento entre nós?



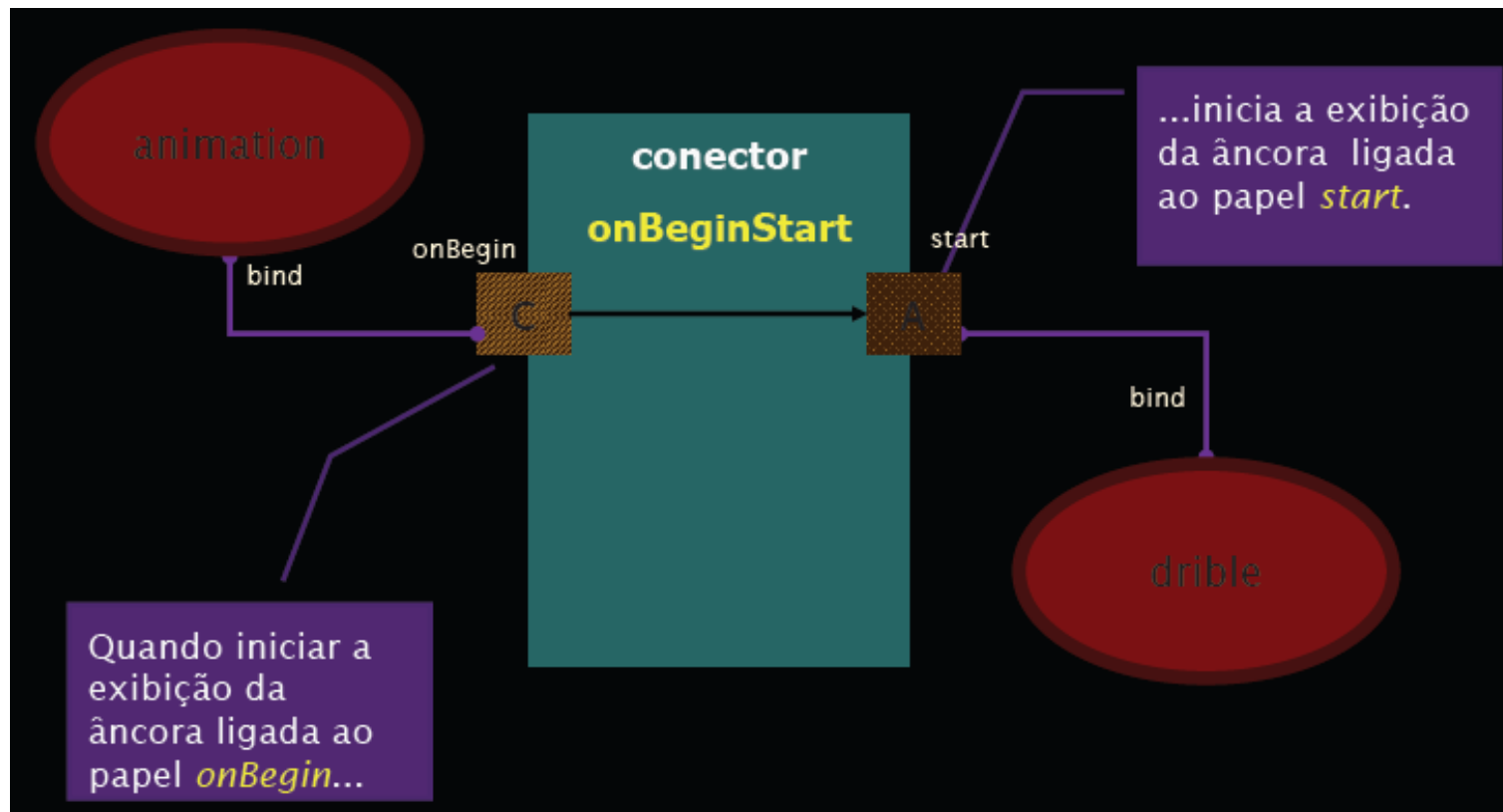
Como criar um relacionamento entre nós?

- Relacionamentos são representados por elos
- Elos divididos em dois elementos
 - *Definição da relação* => *conector hipermídia*
 - *Definição dos participantes* => *conjunto de binds*



Quando exibir?

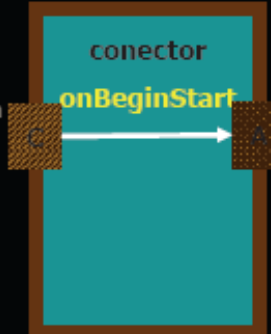
ð Conector onBeginStart



Quando exibir?

ð Como definir o conector onStart?

```
(em causalConnBase.ncl)
<causalConnector id="onBeginStart">
  <simpleCondition role="onBegin"/>
  <simpleAction role="start" />
</causalConnector>
```



The diagram shows a teal rectangular box representing a connector. On the left side, there is a brown textured port labeled 'onBegin'. On the right side, there is another brown textured port labeled 'start'. A white arrow points from the 'onBegin' port to the 'start' port. Inside the box, the text 'conector' is at the top, and 'onBeginStart' is written in yellow in the center.

```
(em exemplo01.ncl)
<connectorBase>
  <importBase documentURI="causalConnBase.ncl" alias="conEx"/>
</connectorBase>
```

na seção *head*

Conectores

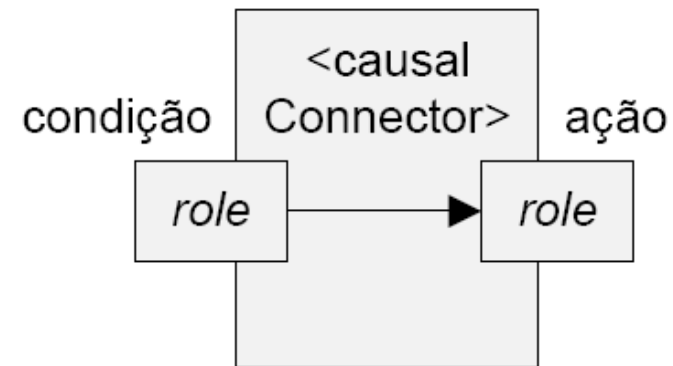
ð <causalConnector> e <connectorBase>

ð Principais atributos:

- *id*

ð Elementos filhos:

- *Papéis de condição:*
 - simpleCondition/compoundCondition
- *Papéis de ação:*
 - simpleAction/compoundAction
- *connectorParam*

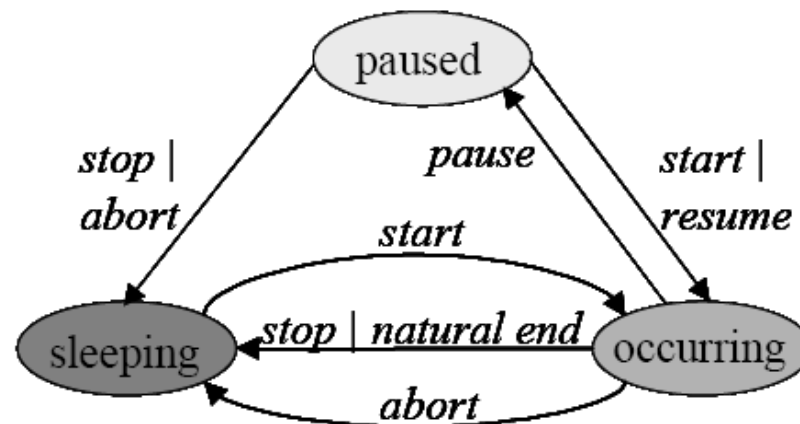


Conectores

δ Papéis de condição simples

δ Principais atributos:

- *role* – nome do papel
- *eventType* – tipo do evento:
 - presentation, selection, attribution, preparation
- *transition* – transição da máquina de estados
- *delay* – retardo
- *key* – tecla do controle remoto (só para seleção)
- *min, max* – cardinalidade de binds do papel
- *qualifier* – and, or (só quando cardinalidade > 1)



Conectores

ð Valores predefinidos por palavras reservadas para *role* em *simpleCondition*:

- **Evento de apresentação (*presentation*)**

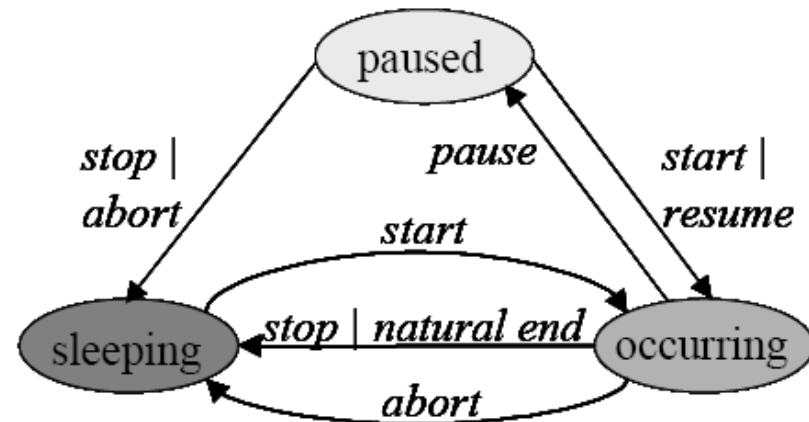
- *onBegin*
- *onEnd*
- *onAbort*
- *onPause*
- *onResume*

- **Evento de seleção (*selection*)**

- *onSelection*

- **Evento de Atribuição (*attribution*)**

- *onBeginAttribution*
- *onEndAttribution*

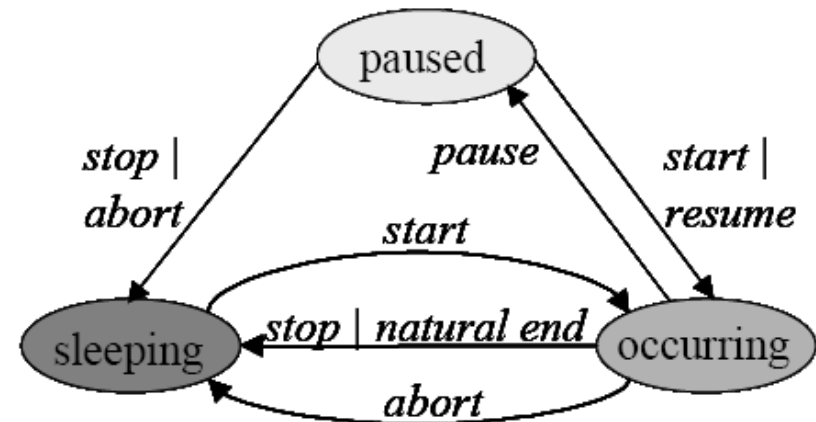


Conectores

ð Valores predefinidos por palavras reservadas para *role* em *simpleCondition*:

- **Evento de preparação (*preparation*)**

- *onBeginPreparation*
- *onEndPreparation*
- *onAbortPreparation*
- *onPausePreparation*
- *onResumePreparation*



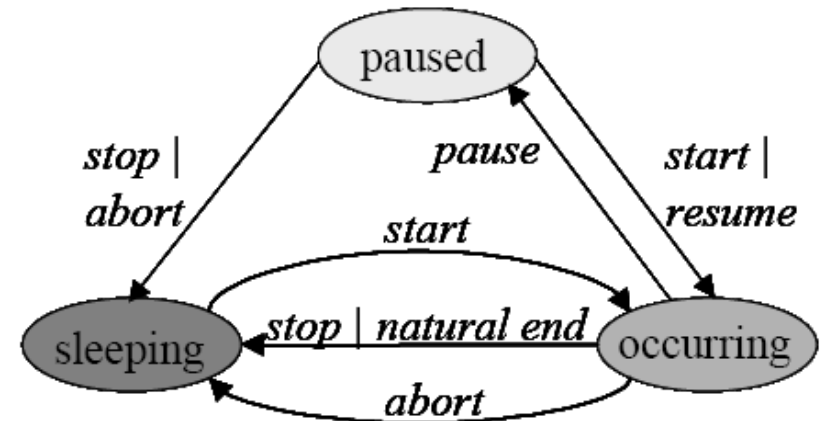
- **O evento de preparação instancia o exibidor de conteúdo da mídia (player) e buferiza o início de seu conteúdo para evitar atrasos na apresentação**

Conectores

δ Papéis de ação simples

δ Principais atributos:

- *role* – nome do papel
- *eventType* – tipo do evento:
 - presentation, selection, attribution, preparation
- *actionType* – ação da máquina de estados
- *delay* – retardo
- *value* – valor (só para atribuição)
- *min, max* – cardinalidade de binds do papel
- *qualifier* – par (quando cardinalidade > 1)



Conectores

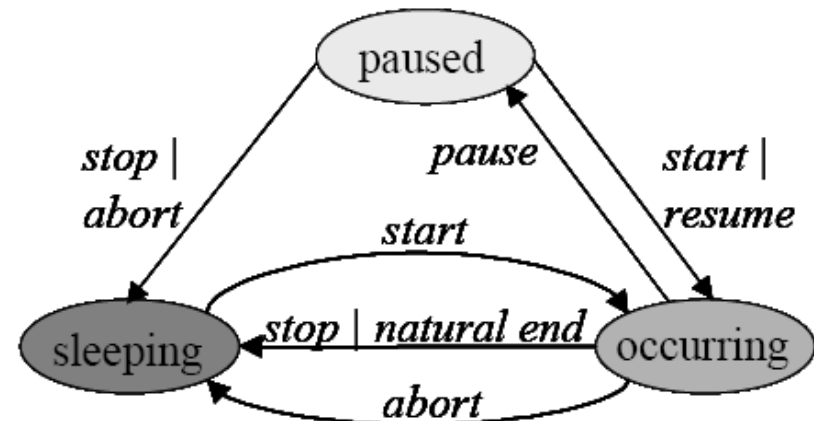
ð Valores predifinidos por palavras reservadas para *role* em `simpleAction`:

- **Evento de apresentação (*presentation*)**

- *start, stop, abort*
- *pause, resume*

- **Evento de Atribuição**

- (*attribution*)
- *set*



- **Evento de preparação (*preparation*)**

- *startPreparation, stopPreparation, abortPreparation*
- *pausePreparation, resumePreparation*

Quando exibir?

ð Como definir o conector onStart?

- *<causalConnector>* e *<connectorBase>*

```
(em causalConnBase.ncl)
<causalConnector id="onBeginStart">
  <simpleCondition role="onBegin"/>
  <simpleAction role="start" />
</causalConnector>
```



```
(em exemplo01.ncl)
<connectorBase>
  <importBase documentURI="causalConnBase.ncl" alias="conEx"/>
</connectorBase>
```

na seção head

Conectores de Interatividade

- ð **Exemplo de conector com seleção de qualquer tecla do controle remoto:**

```
<causalConnector id="onKeySelectionStartN">
```

```
<connectorParam name="vKey" />
```

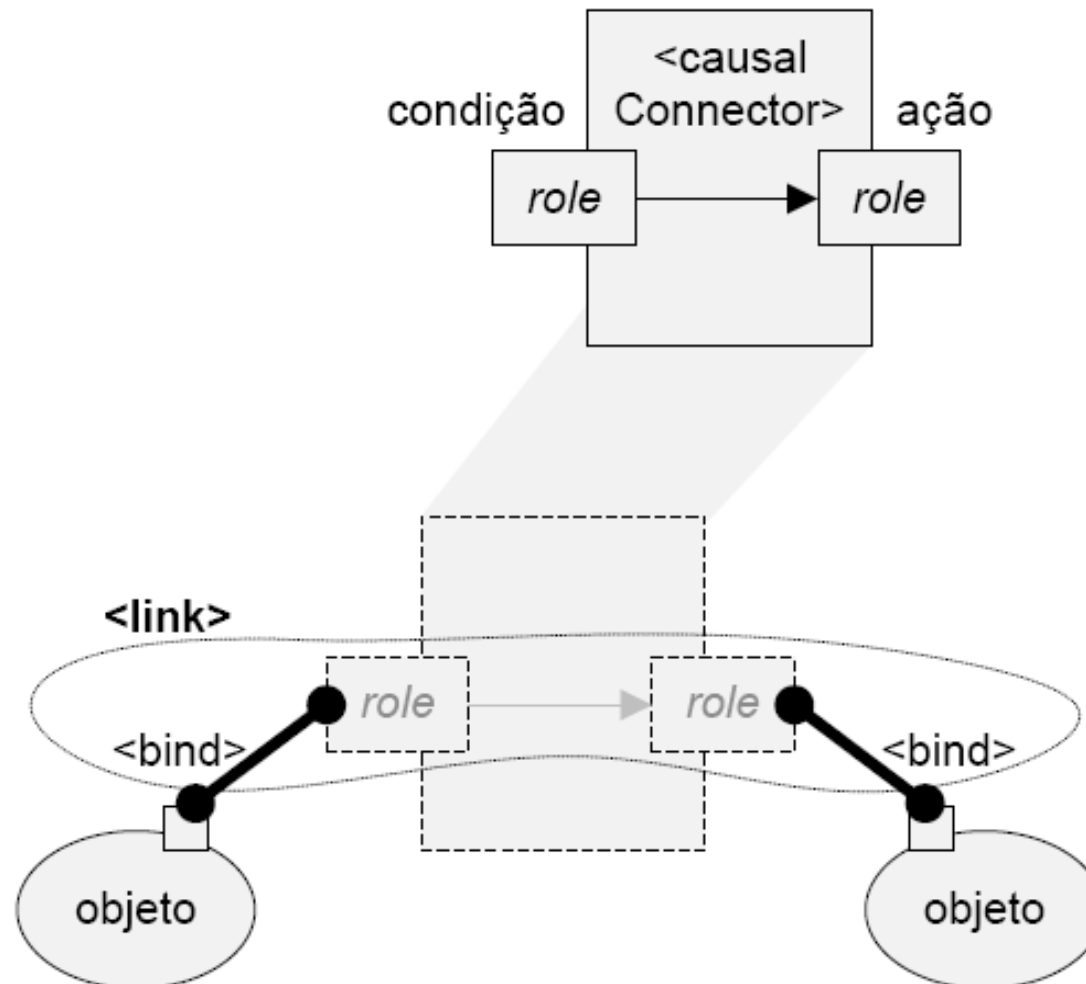
```
<simpleCondition role="onSelection" key="$vKey" />
```

```
<simpleAction role="start" max="unbounded"  
  qualifier="par" />
```

```
</causalConnector>
```

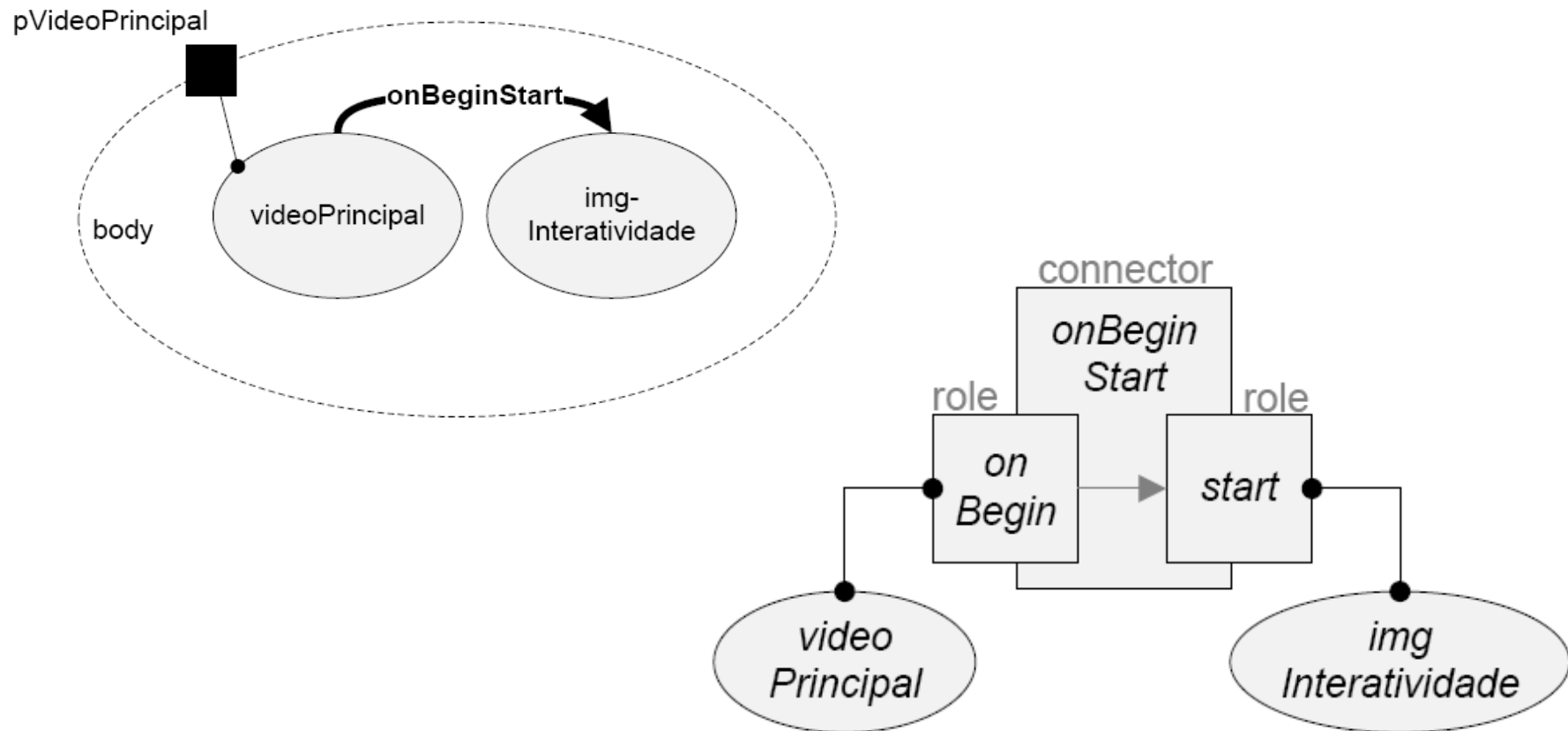
Quando exibir?

ð Como definir um elo entre nós?



Quando exibir?

ð Como definir um elo entre nós?



Elos

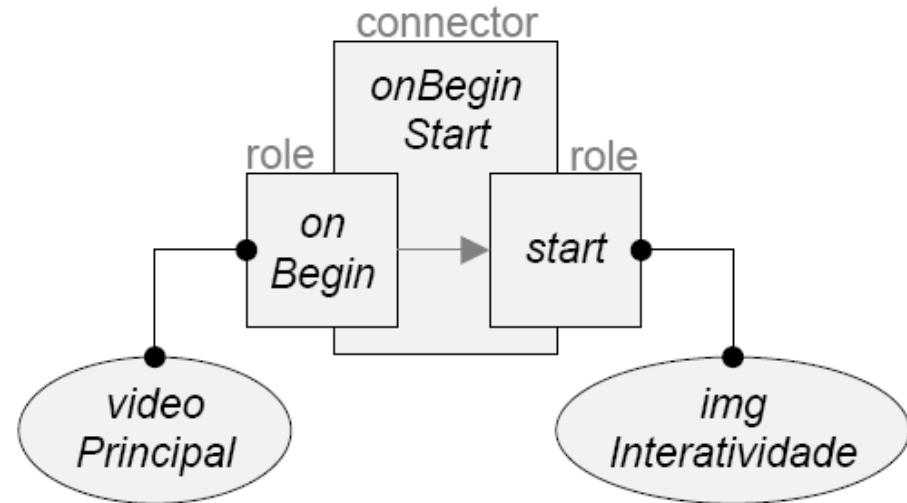
ð <link>

ð Principais atributos:

- *id*
- *xconnector*

ð Elementos filhos:

- *bind*
 - atributos: *role*, *component*, *interface*
 - *bindParam* (atributos *name*, *value*)
- *linkParam*
 - atributos: *name*, *value*



Quando exibir?

ð Como definir o elo entre os nós?

```
<causalConnector id="onBeginStart">  
  <simpleCondition role="onBegin"/>  
  <simpleAction role="start" />  
</causalConnector>
```

conector no arquivo
causalConnBase.ncl

```
<connectorBase>  
  <importBase documentURI="causalConnBase.ncl" alias="conEx"/>  
</connectorBase>
```

na seção head

...inicia a animação.

```
<link xconnector="conEx#onBeginStart">  
  <bind component="animation" interface="segDrible"  
    role="onBegin">  
  <bind component="drible" role="start">  
</link>
```

na seção body



Elos de Interatividade

ð Exemplo de elo com seleção da tecla vermelha:

```
<link xconnector="onKeySelectionStartN">
```

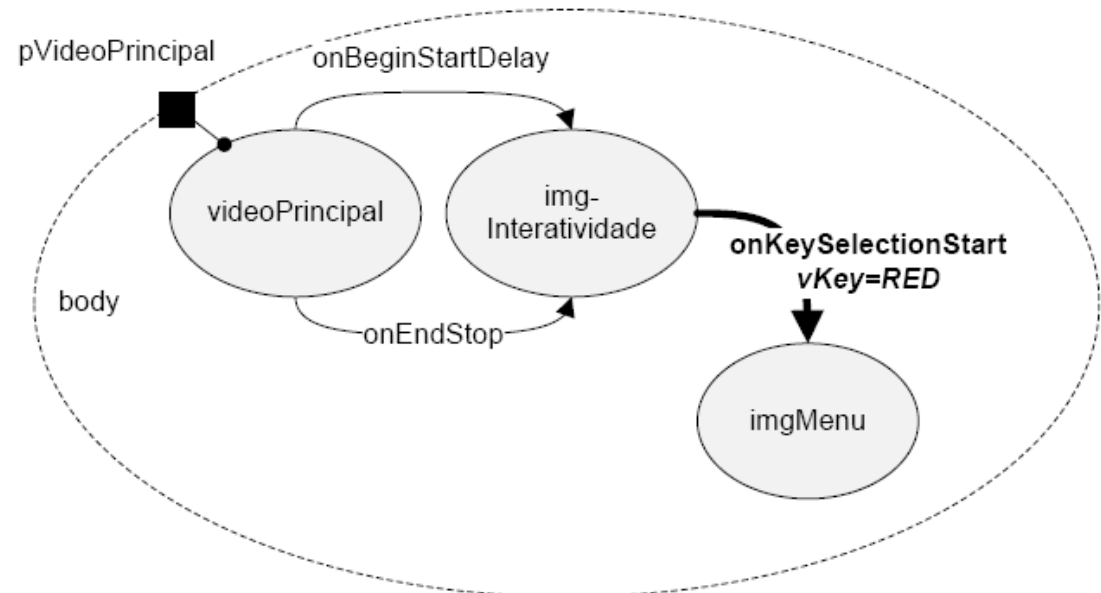
```
  <bind role="onSelection" component="imgInteratividade">
```

```
    <bindParam name="vKey" value="RED"/>
```

```
  </bind>
```

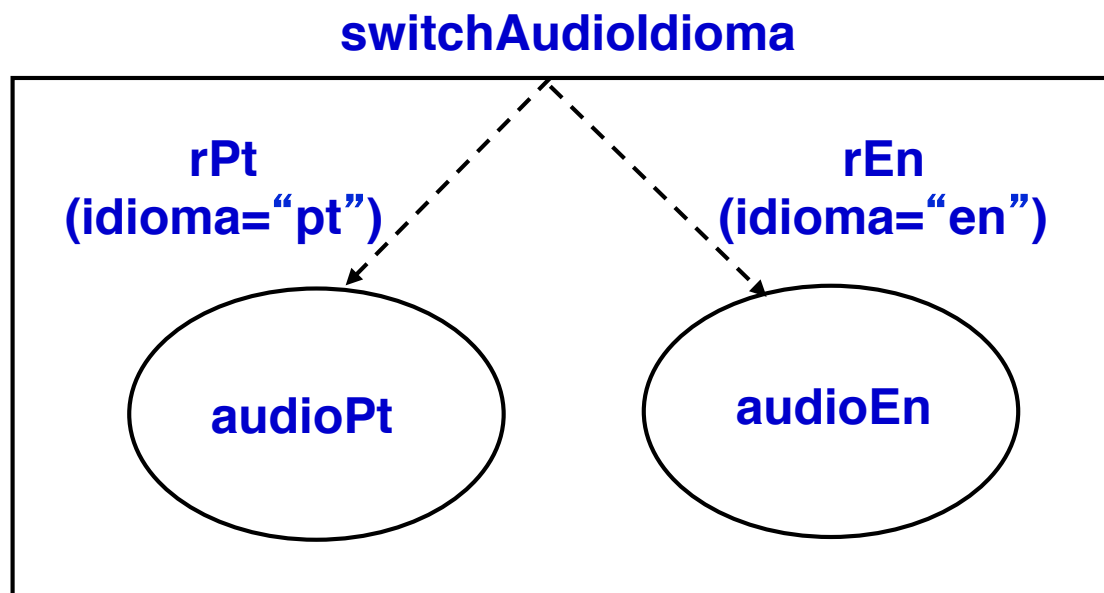
```
  <bind role="start" component="imgMenu"/>
```

```
</link>
```



Adaptação do conteúdo de um programa

ð Regras e switches (alternativas de conteúdo)



ð Nó para guardar variáveis – tipo *x-ginga-settings* ou *x-ncl-settings*

```
<media type=" application/x-ginga-settings" id="nodeSettings">
  <property name="idioma" />
</media>
```

Adaptação do conteúdo de um programa

ð Regras e base de regras – cabeçalho

- *<rule> e <ruleBase>*

```
<ruleBase>
  <rule id="rEn" var="idioma" comparator="eq" value="en" />
  <rule id="rPt" var="idioma" comparator="eq" value="pt" />
</ruleBase>
```

– atributo comparator:

- *eq, ne, gt, lt, gte, lte*

- ***Regras compostas:***

– <compositeRule>

– atributo operator:

- *and, or*

Adaptação do conteúdo de um programa

ð Switches – corpo do documento

- *<switch>*
- *Elementos filhos:*
 - bindRule
 - *Atributos*
 - rule, constituent
 - defaultComponent (atributo *component*)
 - Nós constituintes (*media, context, switch*)

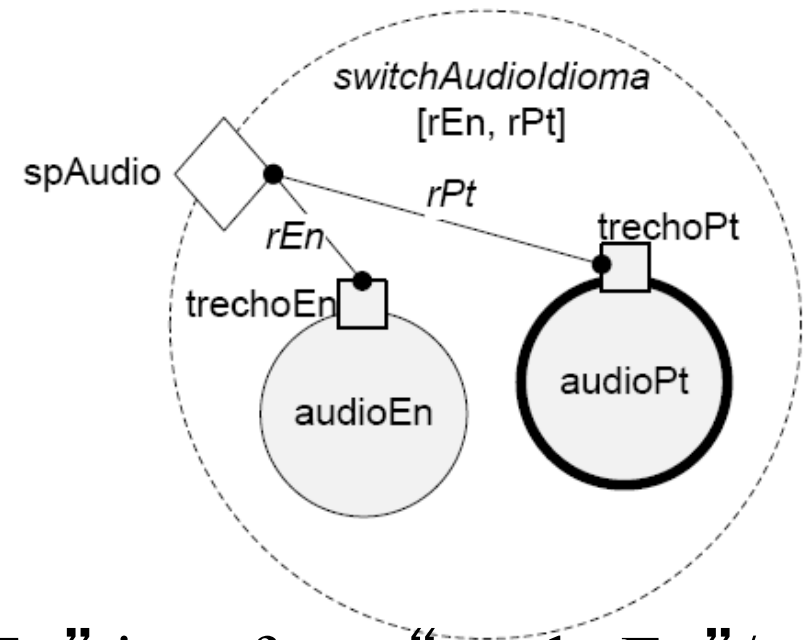
```
<switch id="switchAudioldioma">
  <bindRule rule="rEn" constituent="audioEn" />
  <bindRule rule="rPt" constituent="audioPt" />

  <media type="audio" id="audioEn" src="media/audioEn.mp3" descriptor="dAudio1" />
  <media type="audio" id="audioPt" src="media/audioPt.mp3" descriptor="dAudio1" />
</switch>
```

Portas em um switch

- Para seleccionar uma interface de um constituinte do switch

- *<switchPort>*



- Exemplo:

- *<switchPort id="spaudio">*

- *<mapping component="audioEn" interface="trechoEn"/>*

- *<mapping component="audioPt" interface="trechoPt"/>*

- *</switchPort/>*

- *<defaultComponent component="audioPt"/>*

ð Outras facilidades:

- *Definição de metadados*
- *Reúso*
- *Definições de animações e transições*
- *Programação para múltiplos dispositivos*
- *Comandos para edição ao vivo*
- *Integração com objetos Lua*

Bibliografia

- ø **Normas ABNT – Padrão Brasileiro de TV Digital**
 - *ABNT NBR 15606-2:2018, Televisão digital terrestre - Codificação de dados e especificações de transmissão para radiodifusão digital, Parte 2: Gíngua-NCL para receptores fixos e móveis - Linguagem de aplicação XML para codificação de aplicações*
- ø **Programando em NCL 3.0,**
 - *Luiz Fernando Soares e Simone Barbosa, Ed. Campus, 2ª. Edição, 2012.*
 - *<http://www.telemidia.puc-rio.br/sites/telemidia.puc-rio.br/files/Programando%20em%20NCL%203.0.pdf>*
- ø **Introdução às Linguagens NCL e Lua: Desenvolvendo Aplicações Interativas para TV Digital, Minicurso da SEMENGE 2009,**
 - *R. Carvalho, J. Santos, J. Damasceno, J. Silva, D. Muchaluat Saade,*
 - *<http://www.midiacom.uff.br/gtvd/files/apostila.pdf>*
- ø **TV Digital no Brasil se faz com Gíngua – Fundamentos, Padrões, Autoria Declarativa e Usabilidade,**
 - *Simone Barbosa e Luiz Fernando Soares, Capítulo 3, Livro de Minicursos da Jornada de Atualizações em Informática (JAI) 2008, Editora PUC-Rio, RJ, 2008.*
- ø **Construindo Programas Audiovisuais Interativos Utilizando a NCL 3.0 e a Ferramenta Composer,**
 - *Neto, Soares, Rodrigues e Barbosa, DI/PUC-Rio, 2007.*
- ø **Programming in LUA,**
 - *Roberto Ierusalimschy, 2ª. Edição, Lua.Org, 2006.*

TV Interativa se faz com GINGA!!!

ð Mais informações sobre NCL:

- [*www.ginga.org.br*](http://www.ginga.org.br)
- [*www.ncl.org.br*](http://www.ncl.org.br)
- [*clube.ncl.org.br*](http://clube.ncl.org.br)
- [*www.softwarepublico.org.br*](http://www.softwarepublico.org.br)
- [*www.forumsbtvd.org.br*](http://www.forumsbtvd.org.br)

