

Programando em NCL Modelo Conceitual



Copyright © 2012 TeleMídia



1

Entidades Básicas

O quê? objetos de mídia

como? propriedades

onde? propriedades

quando? relacionamentos e relações

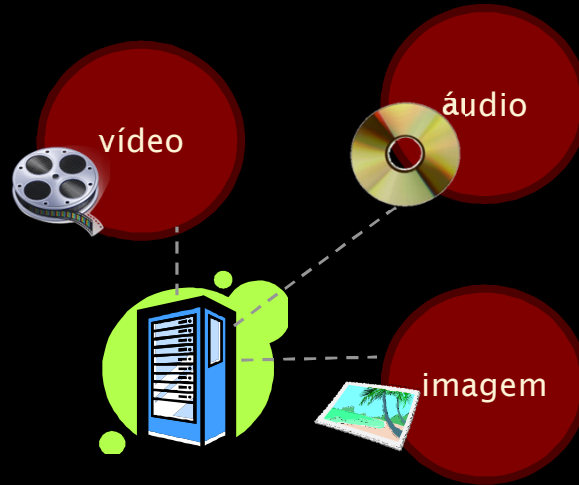


Copyright © 2012 TeleMídia

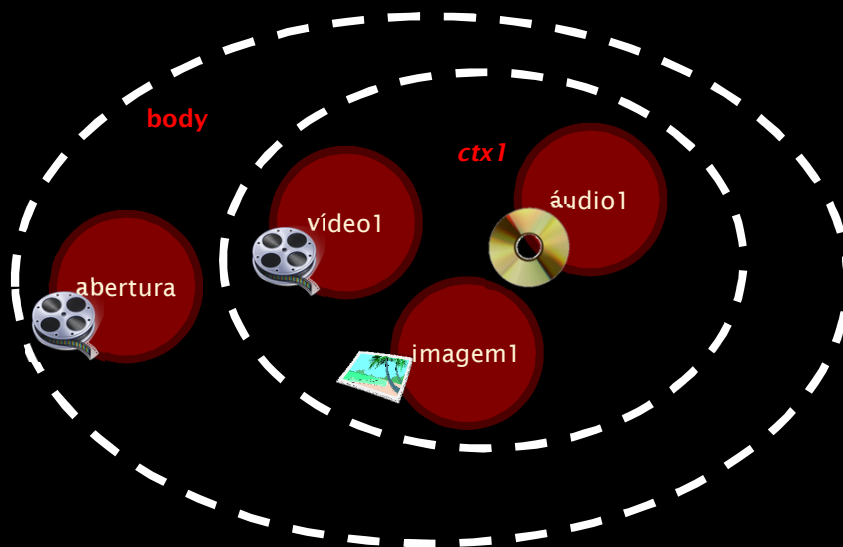


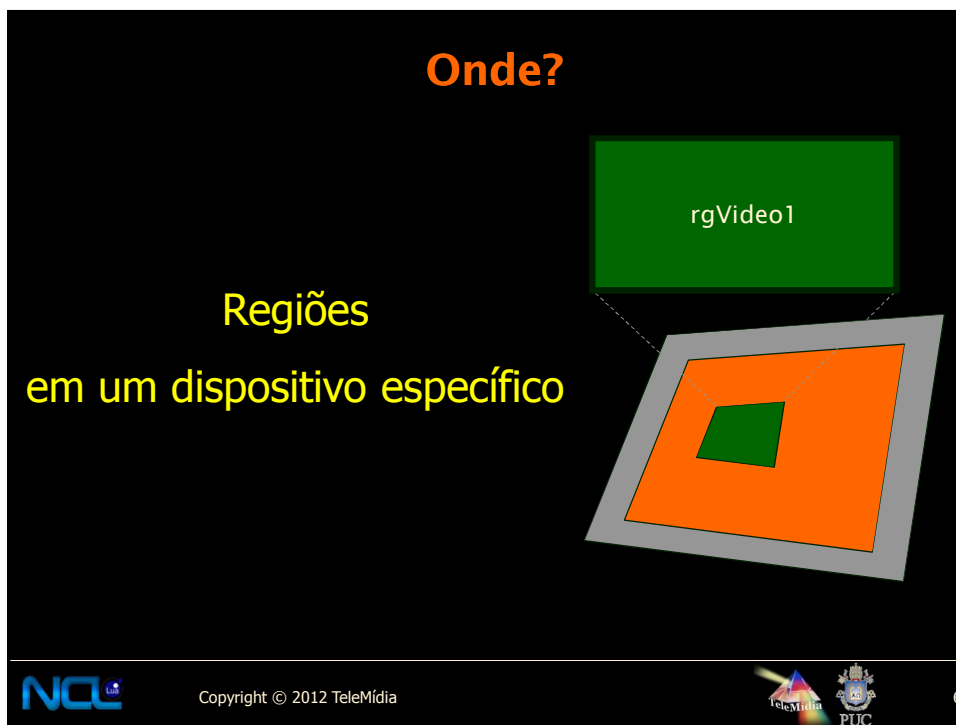
2

O quê? Objetos de Mídia

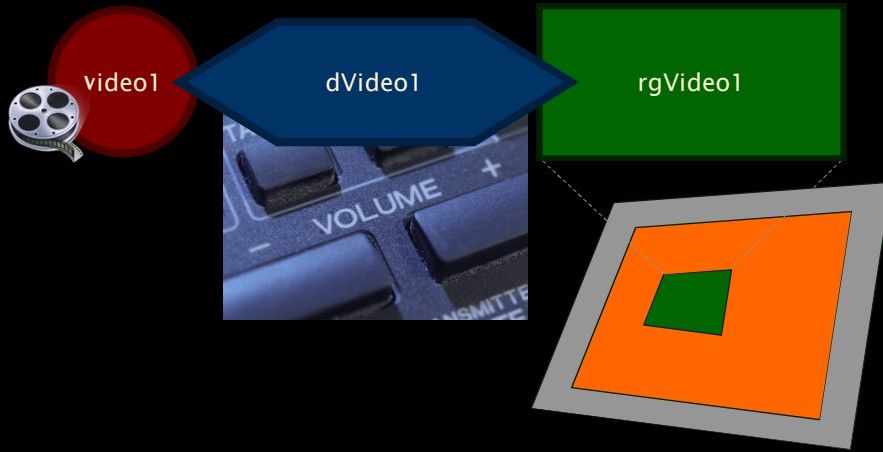


Estrutura: contextos





Como?

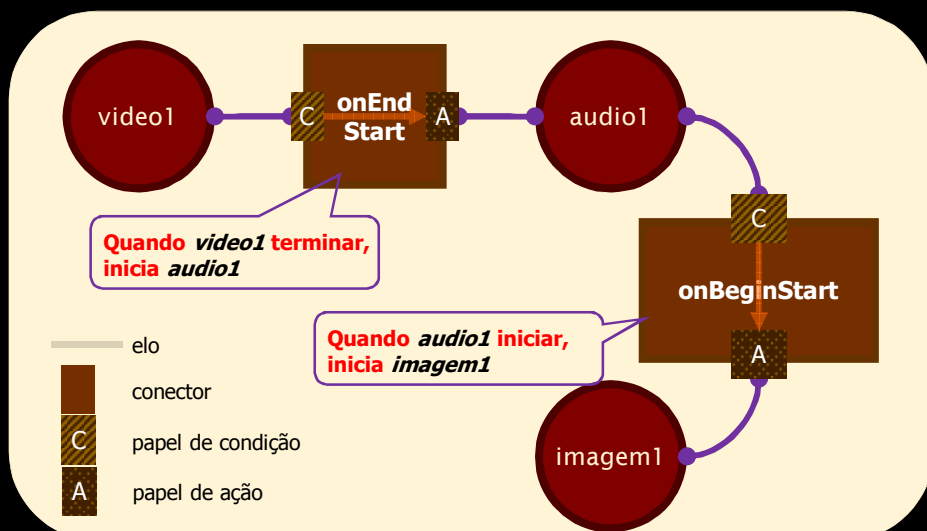


Copyright © 2012 TeleMídia



7

Quando? Elos e Conectores



Copyright © 2012 TeleMídia



8

Programando em NCL A Linguagem



Copyright © 2012 TeleMídia



9

Estrutura de um documento NCL - Perfil TV digital

```
<?xml version="1.0" encoding="ISO-8859-1"?>  
<ncl id="exemplo00"  
  xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
```

```
<head>
```

cabeçalho do documento

```
</head>
```

```
<body>
```

corpo do documento

```
</body>
```

```
</ncl>
```



Copyright © 2012 TeleMídia



10

Planejando uma aplicação NCL

- visões
 - storyboard
 - estrutural
 - leiaute
 - temporal



Copyright © 2012 TeleMídia



11

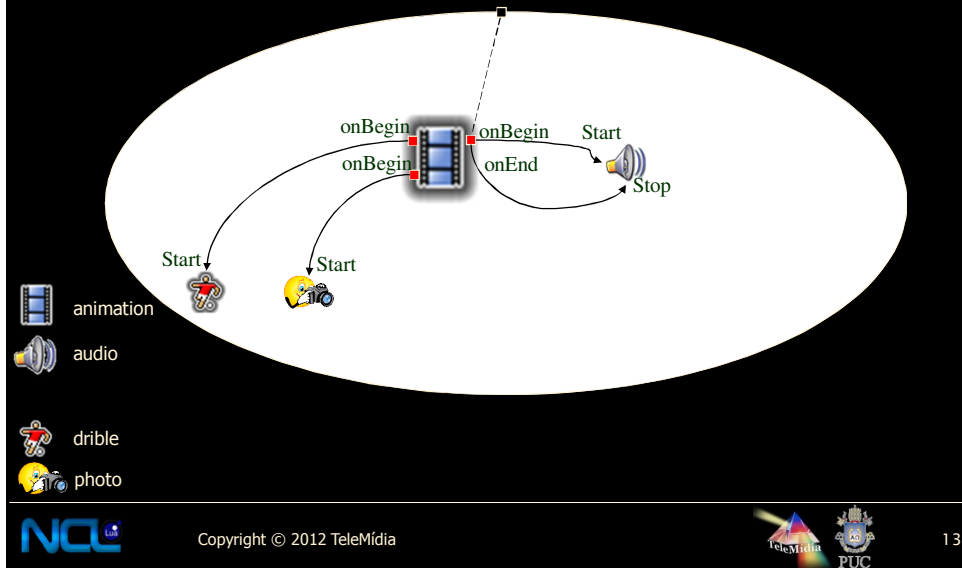
Exemplo 1 – Storyboard



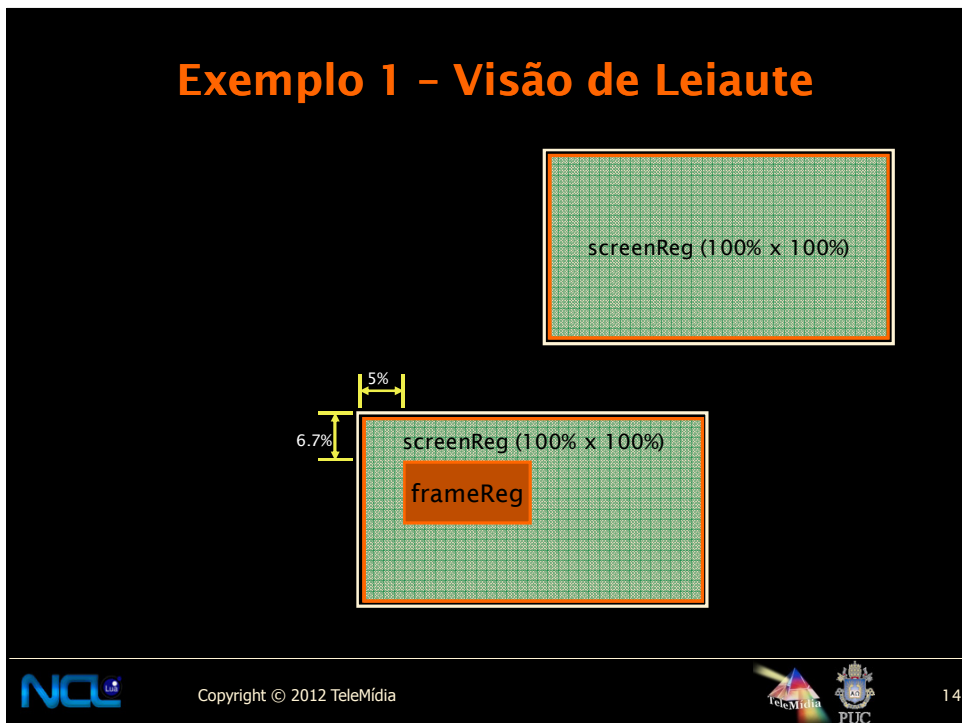
Copyright © 2012 TeleMídia



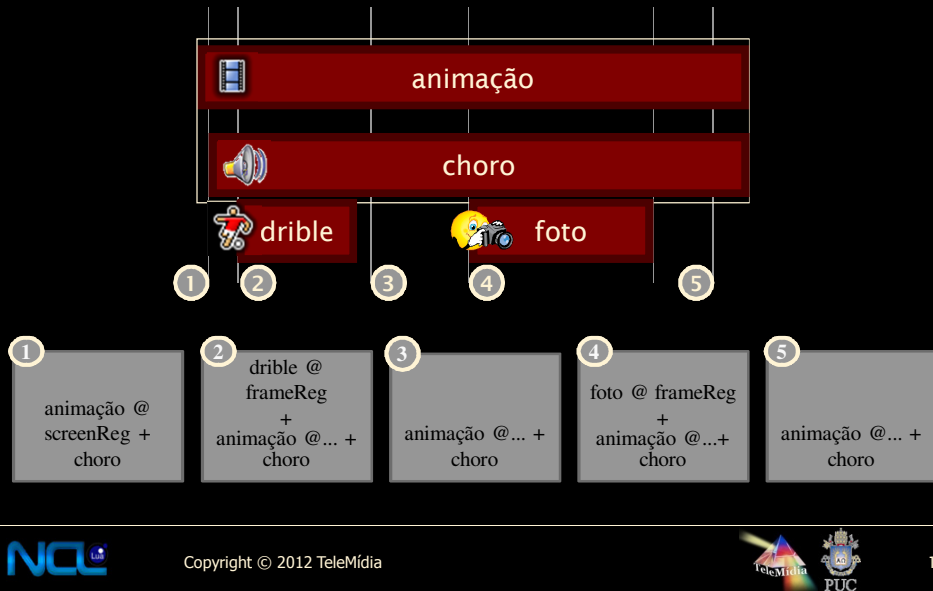
Exemple 1 – Visão Estrutural



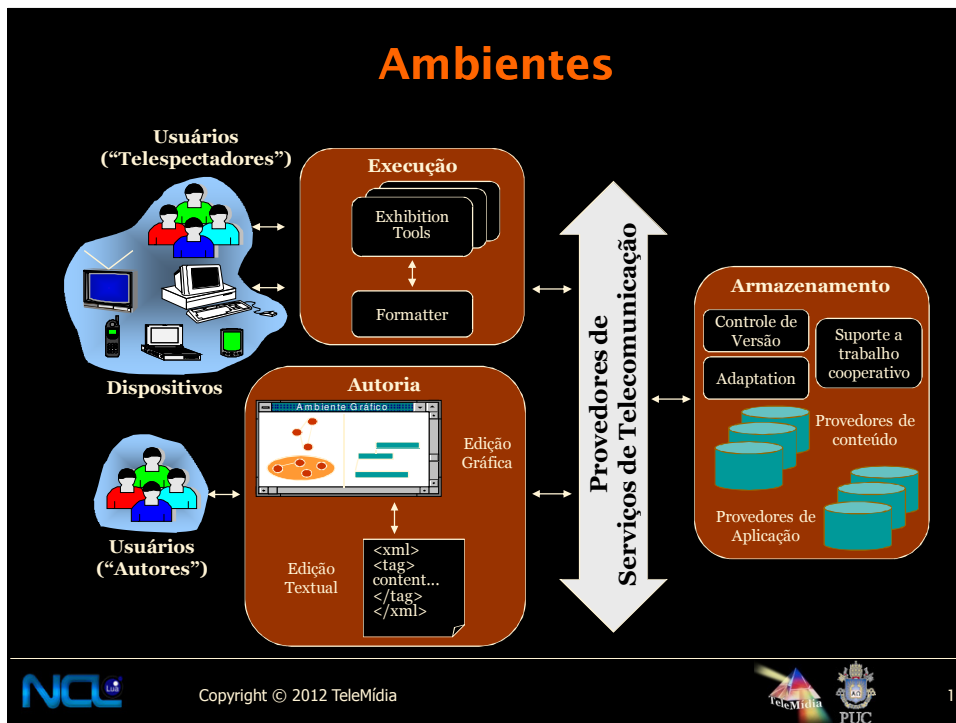
Exemplo 1 – Visão de Leiaute



Exemplo 1 - Visão Temporal



Ambientes



NCL Composer

The screenshot displays the NCL Composer interface with the following components:

- NCL Textual View:** XML code defining a structure with regions, media elements, and connectors.
- Structural View:** A hierarchical tree diagram of the NCL structure.
- Layout View:** A visual representation of the layout with regions labeled (rg1) through (rg4).
- Properties View:** A table showing attributes for the selected media element.
- Validator Plugin:** A list of error messages, including "Invalid data type..." for media, property, and link elements.

Attribute	Value
descriptor	
id	m1
instance	
refer	
src	
type	video



Copyright © 2012 TeleMídia



NCL Eclipse

The screenshot shows the Eclipse IDE with the following details:

- Editor:** XML code for a synchronization example without user interaction.
- Problems View:** A list of 4 errors related to the connector element.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Exemplo de sincronismo sem a interacao do usuario -->
<cncl id="sync" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <regionBase>
      <region id="backgroundReg" width="100%" height="100%" zIndex="1"/>
      <region id="screenReg" width="100%" height="100%" zIndex="2">
        <region id="frameReg" left="5%" top="6.74%" width="10.5%" height="10.5%" zIndex="3"/>
      </region>
    </regionBase>
    <descriptorBase>
      <descriptor id="backgroundBase" region="backgroundReg"/>
      <descriptor id="screenDesc" region="screenReg"/>
      <descriptor id="photoDesc" region="frameReg" explicitDur="5s"/>
      <descriptor id="audioDesc"/>
      <descriptor id="dribbleDesc" region="frameReg"/>
    </descriptorBase>
    <connectorBase>
      <importBase documentURI=".../causalConnBase.ncl" alias="conExt"/>
    </connectorBase>
  </head>
  <body>
    <port id="entry" component="background"/>
    <media id="background" src=".../media/background.png" descriptor="backgroundDesc"/>
    <media id="animation" src=".../media/animGar.mp4" descriptor="screenDesc">
  </body>
</cncl>
  
```

Description	Resource	Path	Line
Papel <var> nClO definido no elemento <connector (conExt:forBegin/arStart),	07settings.nd	Livro TV/Exemplos	1
Papel <var> nClO definido no elemento <connector (conExt:forBegin/arStart),	08animation.nd	Livro TV/Exemplos	1
Papel <var> nClO definido no elemento <connector (conExt:forBegin/arStart),	09animation.nd	Livro TV/Exemplos	1
Papel <var> nClO definido no elemento <connector (conExt:forBegin/arStart),	10menu.nd	Livro TV/Exemplos	1



Copyright © 2012 TeleMídia



18

Set-top Box Virtual Ginga-NCL

NCL
Nested Context Language

Your Ginga-NCL Set-top box is ready for action!

Quick hints:

- From your host operating system, use SSH to open a text console. The STB's IP address is shown below.
- User is "root"; Initial password is "telemidia"
- Upload your NCL applications and media to the /misc/ncl30 directory, via SFTP or SCP
- Use the /misc/launcher.sh script to run your NCL application
Example: /misc/launcher.sh /misc/ncl30/sample03/sample03.ncl
- Use the following keymap:
- **Have fun!**

IP:192.168.127.129

To direct input to this virtual machine, press Ctrl+G.

TeleMídia
middleware
Ginga
TV Interativa se faz com Ginga!

VMware Player



Copyright © 2012 TeleMídia



19

Ginga4Windows



Copyright © 2012 TeleMídia



Linguagem de Marcação XML

- TAG ou Elemento XML (<tag>...</tag> OU <tag />):
 - <media id="video1" descriptor="dVideoWholeScreen">
</media>
 - <media id="video1" descriptor="dVideoWholeScreen"/>
- Atributo (formato: atributo="valor")
 - id
 - descriptor
- Valor de atributo (entre aspas "")
 - "video1"
 - "dVideoWhole Screen"



Copyright © 2012 TeleMídia



21

Construindo uma aplicação NCL

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

```
<ncl id="exemplo00" xmlns="http://www.ncl.org.br/NCL3.0/  
EDTVProfile">
```

```
<head>
```

cabeçalho do documento

1

```
</head>
```

```
<body>
```

corpo do documento

2

```
</body>
```

```
</ncl>
```

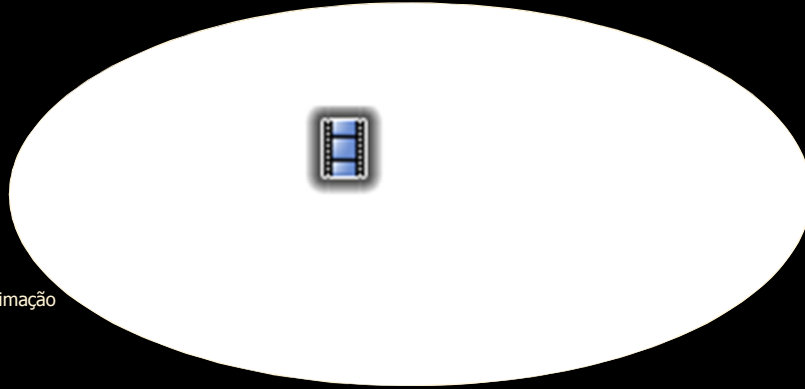


Copyright © 2012 TeleMídia



22

Exemplo 1 – Visão Estrutural



 animação

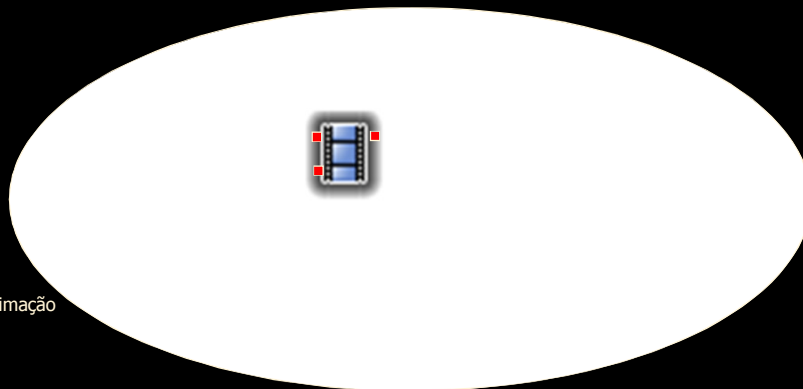
Exemplo 1

```
<body>
```

```
<media id="animation" src="../../media/animGar.mp4" >
```

Esquema	Parte específica do esquema	Uso
file:	//file_path/#fragment_identifier	Para arquivos locais
http:	//server_identifier/file_path/#fragment_identifier	Para arquivos remotos buscados pelo canal de interatividade usando o protocolo http
https:	//server_identifier/file_path/#fragment_identifier	Para arquivos remotos buscados pelo canal de interatividade usando o protocolo https
rtsp:	//server_identifier/file_path/#fragment_identifier	Para fluxos (<i>streams</i>) obtidos pelo canal de interatividade usando o protocolo rtsp
rtp:	//server_identifier/file_path/#fragment_identifier	Para fluxos (<i>streams</i>) obtidos pelo canal de interatividade usando o protocolo rtp
ncl-mirror:	//media_element_identifier	Para um fluxo de conteúdo idêntico a um que esteja em apresentação por um outro elemento de mídia
sbtvd-ts:	//program_number.component_tag	Para fluxos elementares recebidos pelo fluxo de transporte (TS)

Exemplo 1 – Visão Estrutural

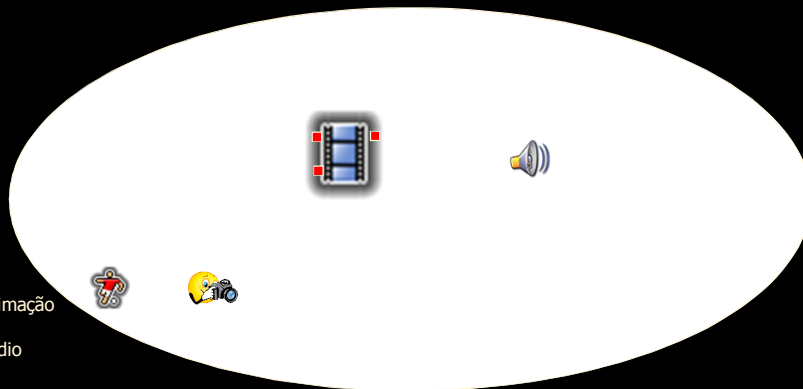


animação

Exemplo 1

```
<body>  
  
{  
  <media id="animation" src="../media/animGar.mp4" >  
    <area id="segDrible" begin="12s"/>  
    <area id="segPhoto" begin="41s"/>  
    <property name="width" value="100%"/>  
    <property name="height" value="100%"/>  
  </media>  
}  
  
</body>
```

Exemplo 1 – Visão Estrutural



animação

áudio

drible

foto

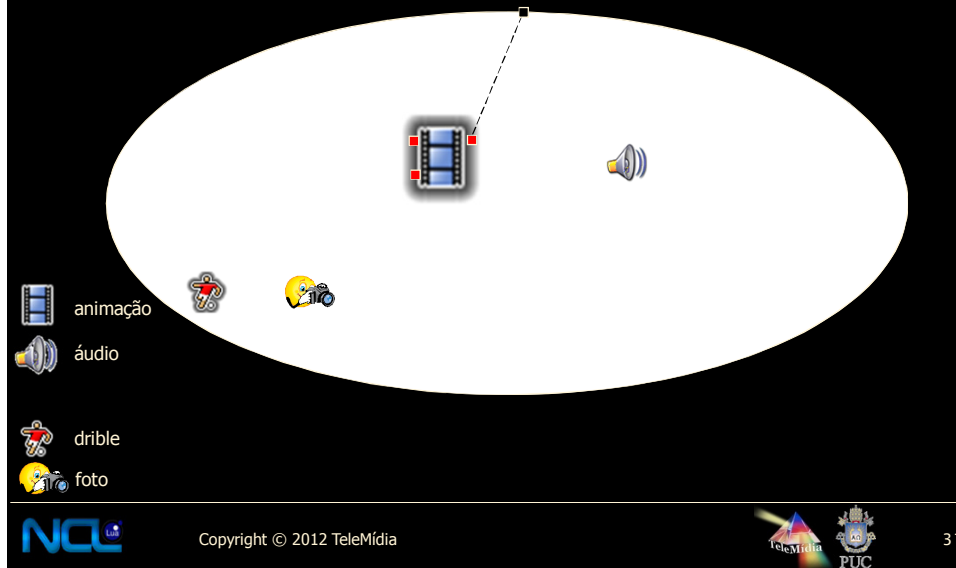
Exemplo 1

```
<body>  
  
{  
  <media id="animation" src="../media/animGar.mp4" >  
    <area id="segDrible" begin="12s"/>  
    <area id="segPhoto" begin="41s"/>  
    <property name="width" value="100%"/>  
    <property name="height" value="100%"/>  
    <property name="zIndex" value="2"/>  
  </media>  
  <media id="choro" src="../media/choro.mp3"/>  
  <media id="drible" src="../media/drible.mp4">  
    <property name="left" value="5%"/>  
    <property name="top" value="6.7%"/>  
    <property name="width" value="18.5%"/>  
    <property name="height" value="18.5%"/>  
    <property name="zIndex" value="3"/>  
  </media>  
}
```

Exemplo 1

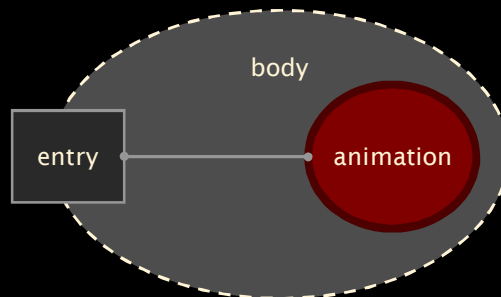
```
{  
  <media id="photo" src="../media/photo.png">  
    <property name="left" value="5%"/>  
    <property name="top" value="6.7%"/>  
    <property name="width" value="18.5%"/>  
    <property name="height" value="18.5%"/>  
    <property name="zIndex" value="3"/>  
    <property name="explicitDur" value="5s"/>  
  </media>  
}
```

Exemplo 1 – Visão Estrutural



Exemplo 1 – Contexto *body* e port

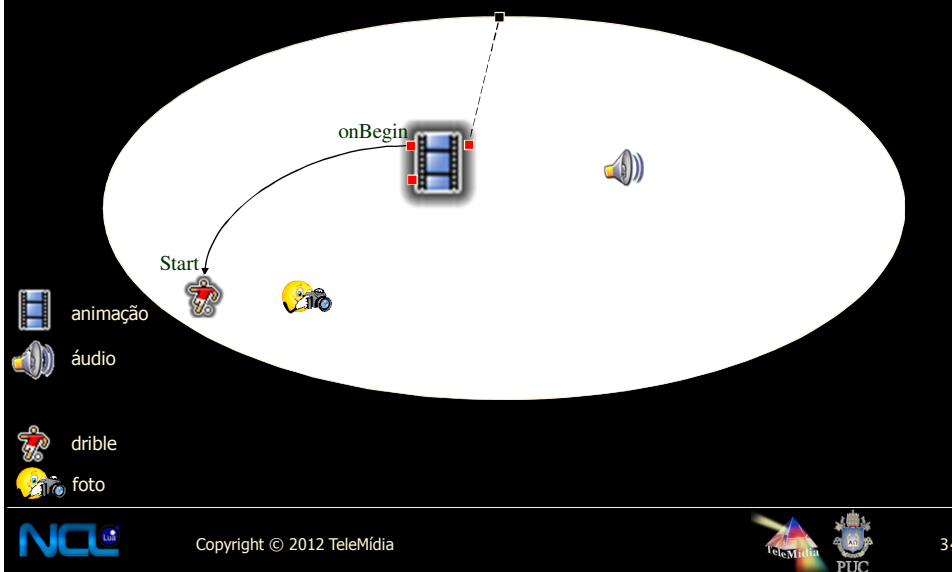
```
<body>  
  <port id="entry" component="animation" />  
  ...  
</body>
```



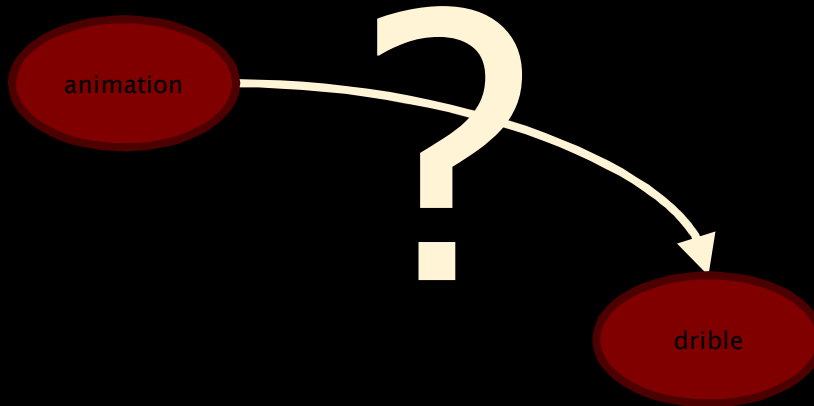
Exemplo 1

```
<body>
  <port id="entry" component="animation"/>
  <media id="animation" src="../media/animGar.mp4" >
    <area id="segDrible" begin="12s"/>
    <area id="segPhoto" begin="41s"/>
    <property name="width" value="100%"/>
    <property name="height" value="100%"/>
    <property name="zIndex" value="2"/>
  </media>
  <media id="choro" src="../media/choro.mp3"/>
  <media id="drible" src="../media/drible.mp4">
    <property name="left" value="5%"/>
    <property name="top" value="6.7%"/>
    <property name="width" value="18.5%"/>
    <property name="height" value="18.5%"/>
    <property name="zIndex" value="3"/>
  </media>
```

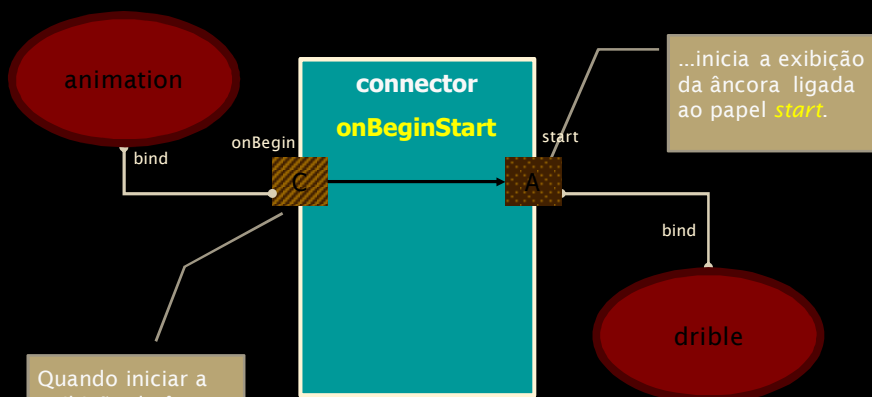
Exemplo 1 – Visão Estrutural



Relacionamento entre Objetos



Connector *onBeginStart*



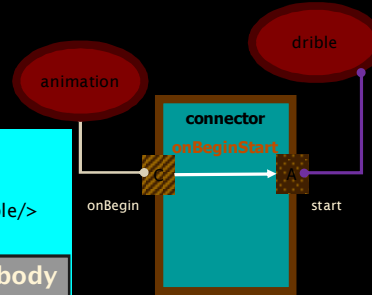
Connector *onBeginStart*

```
<causalConnector id="onBeginStart">
  <simpleCondition role="onBegin"/>
  <simpleAction role="start" max="unbounded" qualifier="par"/>
</causalConnector>
```

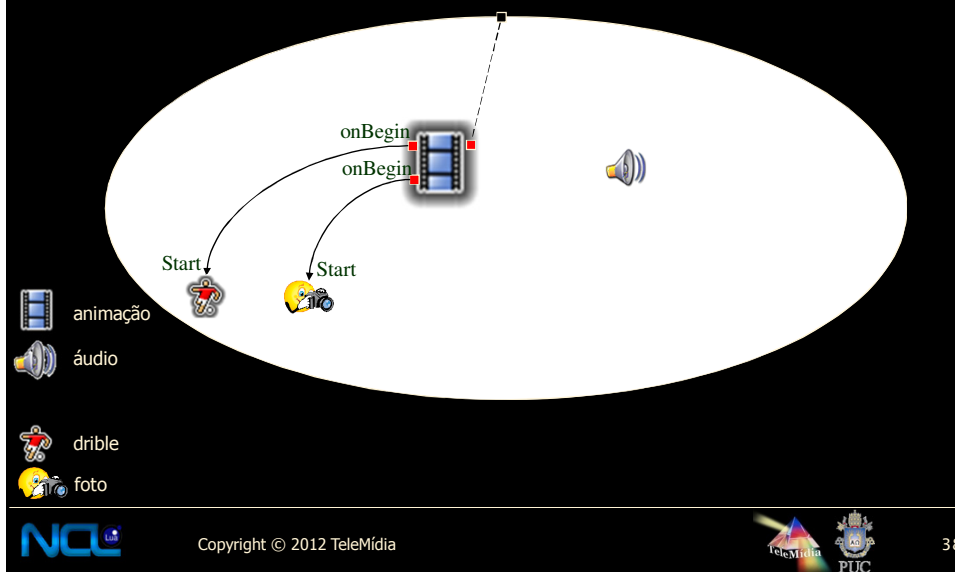
Elo que utiliza o conector *onBeginStart*:
Iniciando o vídeo *drible*

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

na seção *body*



Exemplo 1 – Visão Estrutural



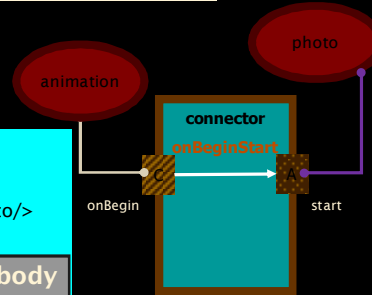
Connector *onBeginStart*

```
<causalConnector id="onBeginStart">
  <simpleCondition role="onBegin"/>
  <simpleAction role="start" max="unbounded" qualifier="par" />
</causalConnector>
```

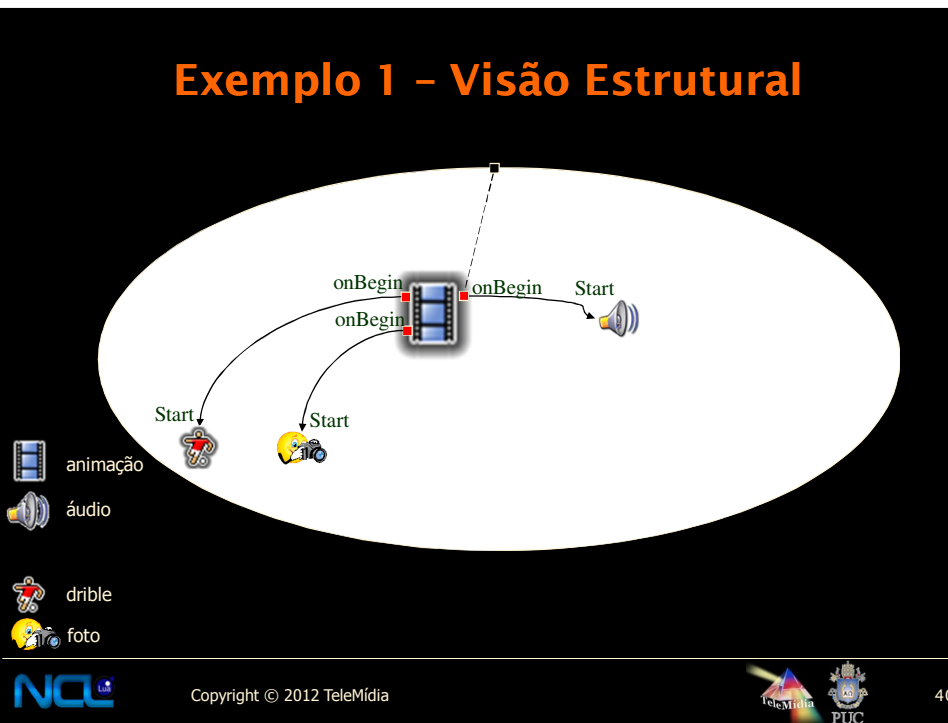
Elo que utiliza o conector *onBeginStart*:
Iniciando a imagem da *foto*

```
<link xconnector="onBeginStart">
  <bind role="onBegin" component="animation"
        interface="segPhoto"/>
  <bind role="start" component="photo" />
</link>
```

na seção *body*



Exemplo 1 – Visão Estrutural



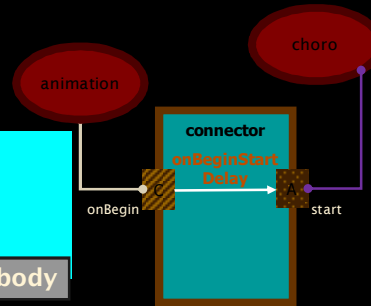
Connector *onBeginStart_delay*

```
<causalConnector id="onBeginStart_delay">
  <simpleCondition role="onBegin"/>
  <simpleAction role="start" delay="5s" max="unbounded" qualifier="par"/>
</causalConnector>
```

Elo que utiliza o conector *onBeginStart*:
Iniciando o áudio *choro*

```
<link xconnector="onBeginStart_delay">
  <bind role="onBegin" component="animation"/>
  <bind role="start" component="choro" />
</link>
```

na seção body



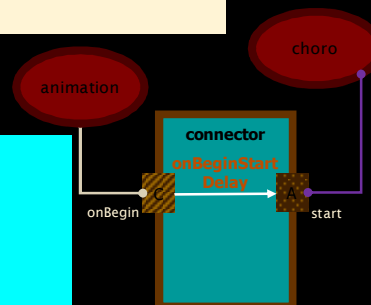
Connector *onBeginStart_delay*

```
<causalConnector id="onBeginStart_delay">
  <connectorParam name="xyz"/>
  <simpleCondition role="onBegin"/>
  <simpleAction role="start" delay="$xyz" max="unbounded" qualifier="par"/>
</causalConnector>
```

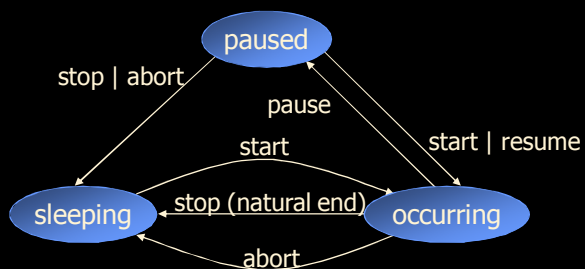
Elo que utiliza o conector *onBeginStart*:
Iniciando o áudio *choro*

```
<link xconnector="onBeginStart_delay">
  <bind role="onBegin" component="animation"/>
  <bind role="start" component="choro">
    <bindParam name="xyz" value="5s"/>
  </bind>
</link>
```

na seção body



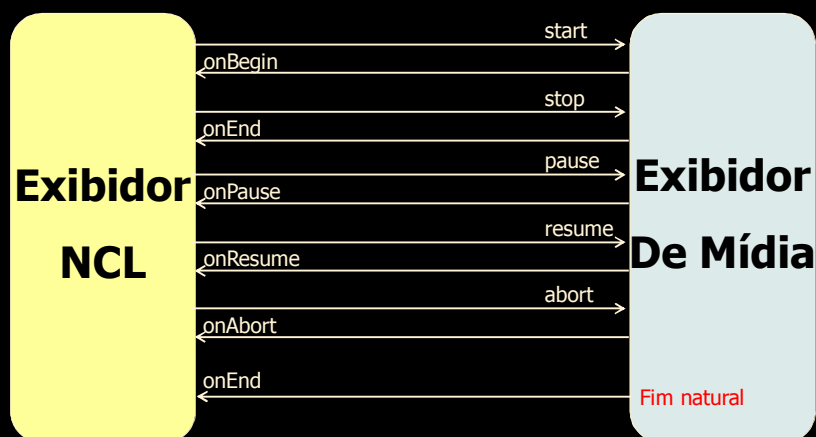
Máquina de Estado de Evento



Copyright © 2012 TeleMídia



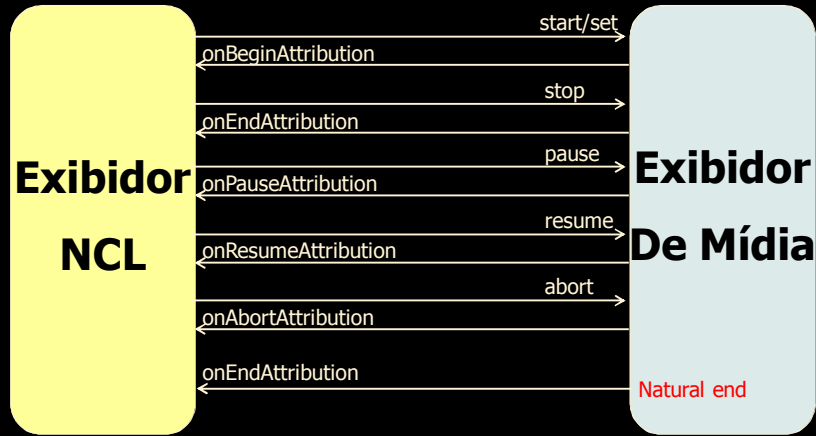
Eventos de Apresentação



Copyright © 2012 TeleMídia



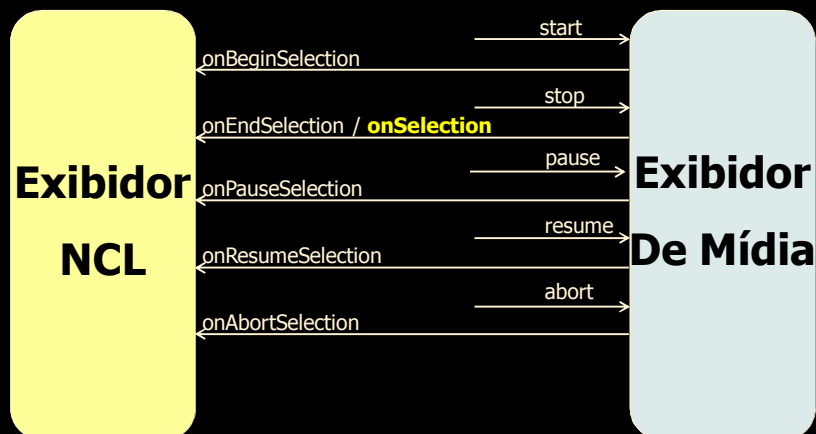
Eventos de Atribuição



Copyright © 2012 TeleMídia



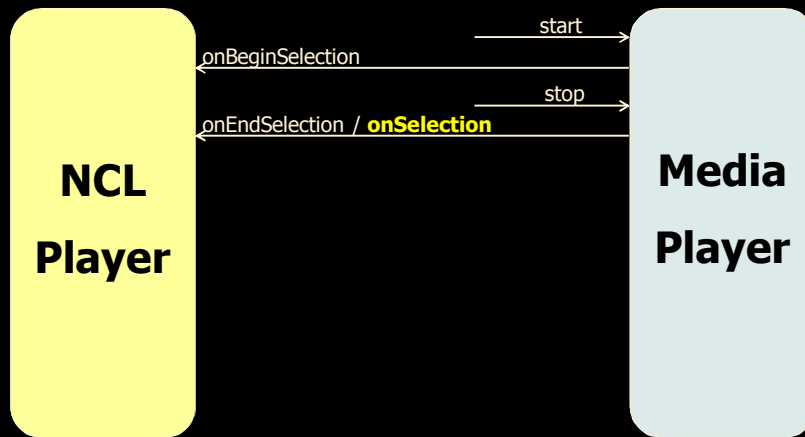
Eventos de Seleção



Copyright © 2012 TeleMídia



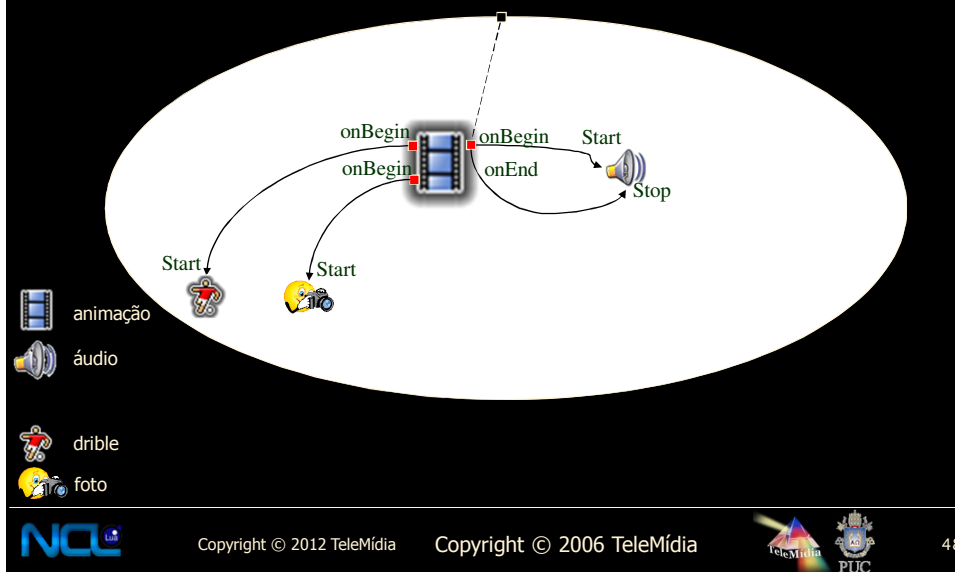
Eventos de Seleção



Copyright © 2012 TeleMídia



Exemplo 1 – Visão Estrutural



Copyright © 2012 TeleMídia

Copyright © 2006 TeleMídia



48

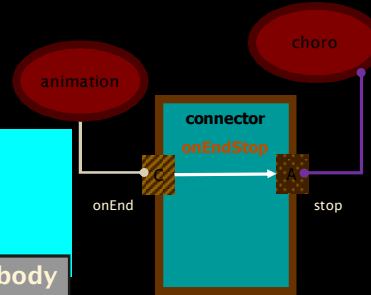
Connector *onEndStop*

```
<causalConnector id="onEndStop">
  <simpleCondition role="onEnd"/>
  <simpleAction role="stop" max="unbounded" qualifier="par" />
</causalConnector>
```

Elo que utiliza o conector *onBeginStart*:
Terminando o áudio *choro*

```
<link xconnector="onEndStop">
  bind role="onEnd" component="animation"/>
  <bind role="stop" component="choro" />
</link>
```

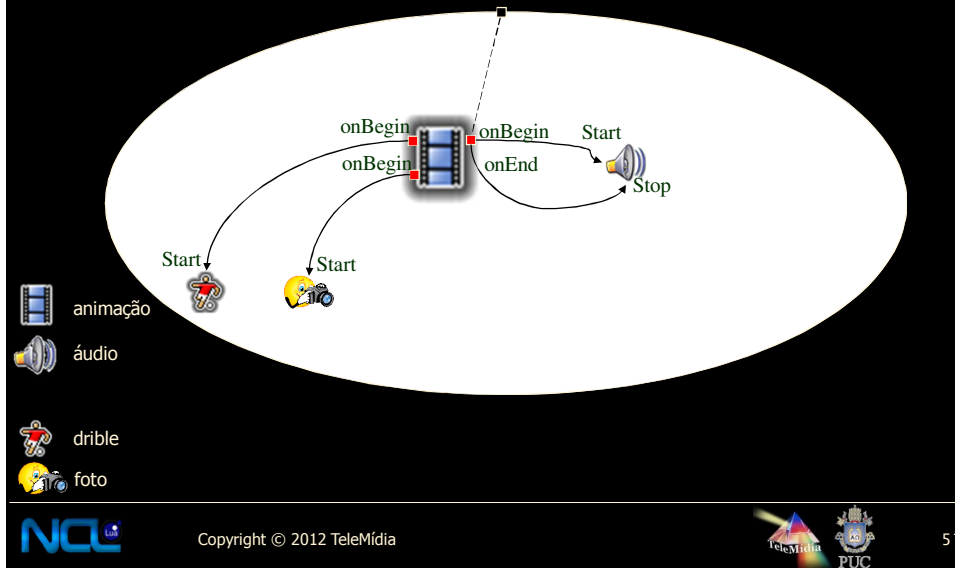
na seção *body*



Exemplo 1

```
{
  <link id="IMusic" xconnector="onBeginStart_delay">
    <bind role="onBegin" component="animation" />
    <bind role="start" component="choro">
      <bindParam name="xyz" value="5s" />
    </bind>
  </link>
}
{
  <link id="IDrible" xconnector="onBeginStart">
    <bind role="onBegin" component="animation" interface="segDrible"/>
    <bind role="start" component="drible"/>
  </link>
}
{
  <link id="IPhoto" xconnector="onBeginStart">
    <bind role="onBegin" component="animation" interface="segPhoto"/>
    <bind role="start" component="photo"/>
  </link>
}
{
  <link id="IEnd" xconnector="onEndStop">
    <bind role="onEnd" component="animation"/>
    <bind role="stop" component="choro"/>
  </link>
}
</body>
</ncl>
```

Exemplo 2 – Visão Estrutural



Entidades Básica

o quê? objetos de mídia

como? propriedades
ou descritores

onde? propriedades
ou regiões

quando? relacionamentos e relações

cabeçalho 1

```

<head>
  <regionBase>
    ...
  </regionBase>
  <descriptorBase>
    ...
  </descriptorBase>
  <connectorBase>
    ...
  </connectorBase>
</head>

```

regions - *onde*
 descriptors - *como*
 connectors - *quando*

corpo 2

```

<body>
  <port id="plnicio" component="video1" />
  <!-- context and media objects -->
  <!-- elos -->
</body>

```

context e media - *o quê*
 links - *quando*

NCL Copyright © 2012 TeleMídia TeleMídia PUC 53

Regions

```

<regionBase>
  <region id="screenReg" height="100%" width="100%" zIndex="2">
    <region id="frameReg" left="5%" top="6.7%" height="18.5%" width="160"
      zIndex="3"/>
  </region>
</regionBase>

```

(100% x 100%)

NCL Copyright © 2012 TeleMídia TeleMídia PUC 54

Descriptors

(how?)

- em que região um objeto será apresentado

```
<descriptor id="photoDesc" region="frameReg" explicitDur="5s"/>
```

- por quanto tempo um objeto será apresentado

```
<descriptor id="photoDesc" region="frameReg" explicitDur="5s"/>
```

- **Como** um objeto é inicialmente apresentado

- p.ex. volume do áudio (soundLevel),
borda de uma mídia textual (border)

```
<descriptor id="audioDesc" >  
  <descriptorParam name="soundLevel" value="70%" />  
</descriptor>
```

Exemplo 2 - Descriptors

```
<descriptorBase>  
  <descriptor id="screenDesc" region="screenReg"/>  
  <descriptor id="photoDesc" region="frameReg" explicitDur="5s"/>  
  <descriptor id="audioDesc"/>  
  <descriptor id="dribleDesc" region="frameReg"/>  
</descriptorBase>
```



Exemplo 2

```
<body>
  { <port id="entry" component="animation"/>
  { <media id="animation" src="../../media/animGar.mp4" descriptor="screenDesc">
    <area id="segDrible" begin="12s"/>
    <area id="segPhoto" begin="41s"/>
  </media>
  <media id="choro" src="../../media/choro.mp3" descriptor="audioDesc"/>
  <media id="drible" src="../../media/drible.mp4" descriptor="dribleDesc"/>
  <media id="photo" src="../../media/photo.png" descriptor="photoDesc"/>
```

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Exemplo de base de conectores -->
<ncl id="causalConnBase" xmlns="http://www.ncl.org.br/NCL3.0/causalConnectorProfile">
  <head>
    <connectorBase>
      <causalConnector id="onBeginStartDelay">
        <connectorParam name="delay"/>
        <simpleCondition role="onBegin"/>
        <simpleAction role="start" delay="$delay" max="unbounded" qualifier="par"/>
      </causalConnector>
      <causalConnector id="onBeginStart">
        <simpleCondition role="onBegin"/>
        <simpleAction role="start" max="unbounded" qualifier="par"/>
      </causalConnector>
      <causalConnector id="onEndStop">
        <simpleCondition role="onEnd"/>
        <simpleAction role="stop" max="unbounded" qualifier="par"/>
      </causalConnector>
    </connectorBase>
  </head>
```

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- Exemplo de sincronismo sem a interacao do usuario -->
<ncl id="sync" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <regionBase>
      <region id="screenReg" height="100%" width="100%" zIndex="1">
        <region id="frameReg" left="5%" top="6.7%" height="18.5%" width="18.5%"
          zIndex="3"/>
      </region>
    </regionBase>
    <descriptorBase>
      <descriptor id="screenDesc" region="screenReg"/>
      <descriptor id="photoDesc" region="frameReg" explicitDur="5s"/>
      <descriptor id="audioDesc"/>
      <descriptor id="dribleDesc" region="frameReg"/>
    </descriptorBase>
    <connectorBase>
      <importBase documentURI="causalConnBase.ncl" alias="conEx"/>
    </connectorBase>
  </head>

```

Exemplo 2

```

  <link id="IMusic" xconnector="conEx#onBeginStart_delay">
    <bind role="onBegin" component="animation" />
    <bind role="start" component="choro">
      <bindParam name="xyz" value="5s" />
    </bind>
  </link>

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

  <link id="IPhoto" xconnector="conEx#onBeginStart">
    <bind role="onBegin" component="animation" interface="segPhoto"/>
    <bind role="start" component="photo"/>
  </link>

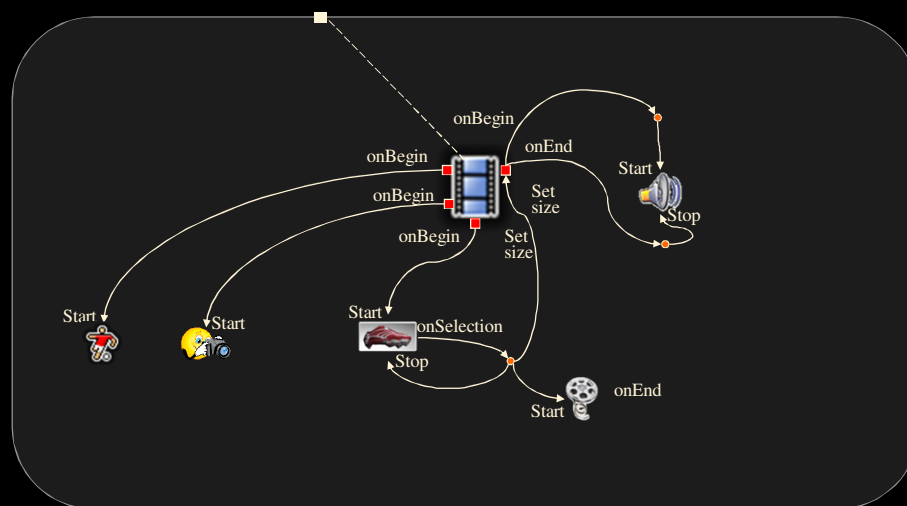
  <link id="IEnd" xconnector="conEx#onEndStop">
    <bind role="onEnd" component="animation"/>
    <bind role="stop" component="choro"/>
  </link>
</body>
</ncl>

```

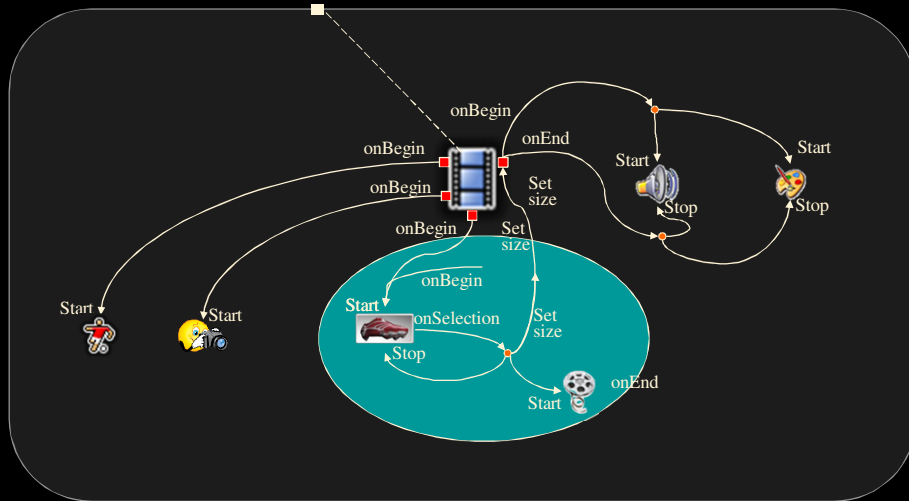
Exemplo - Storyboard



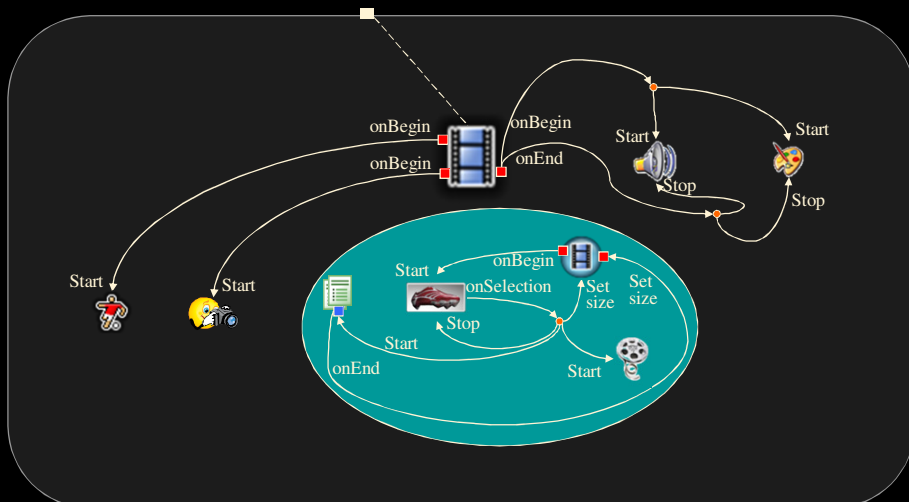
Exemplo



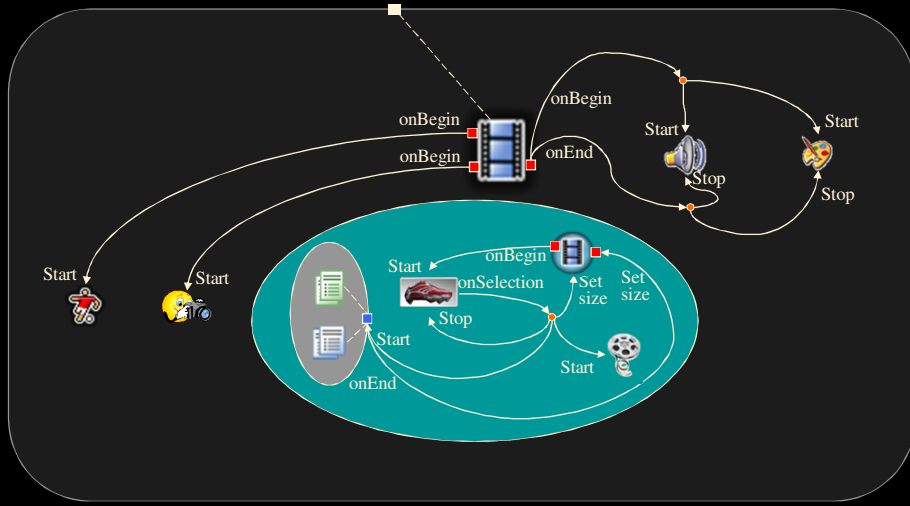
Exemplo



Exemplo



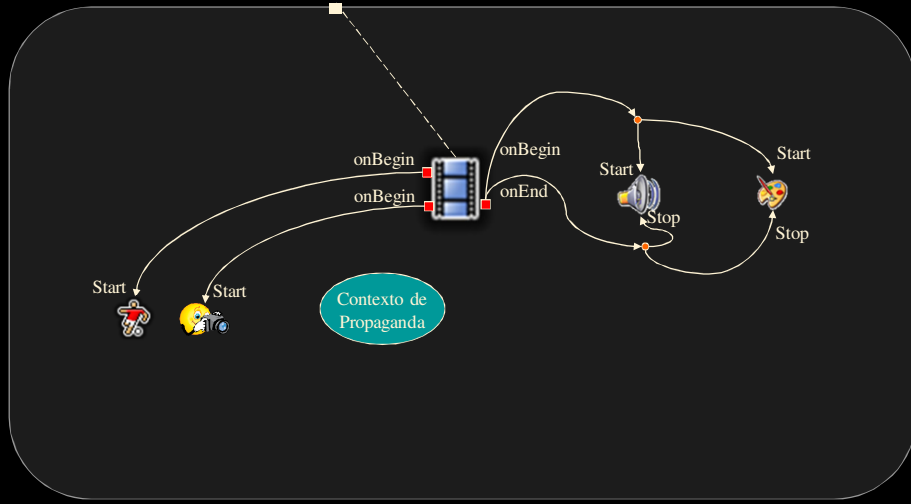
Exemplo



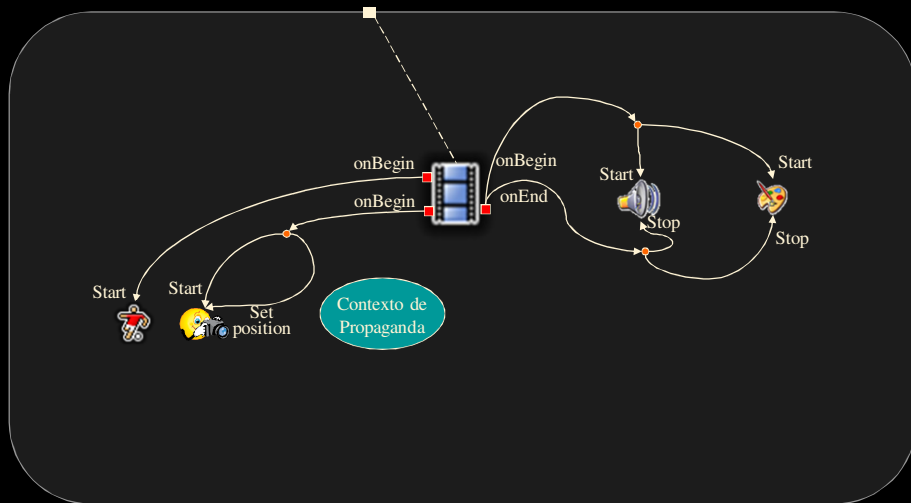




Exemplo



Exemplo



NCL Composer



Copyright © 2012 TeleMídia



73

Exemplo - Storyboard



Copyright © 2012 TeleMídia

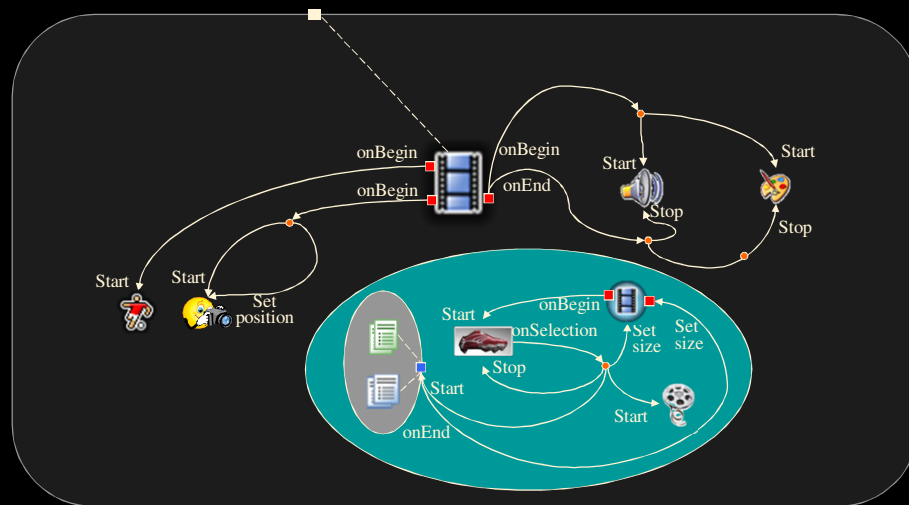


74

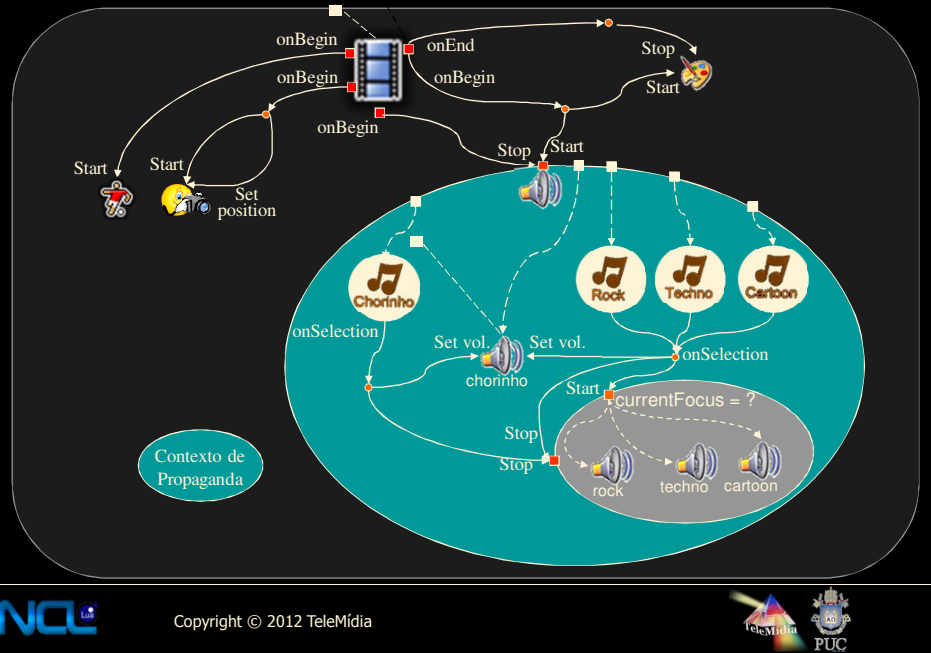
Exemplo - Storyboard



Exemplo



Exemplo 11



Conector *onSelection_orSet_varStopStart*

```
<causalConnector id="onSelection_orSet_varStopStart">
  <connectorParam name="var"/>
  <simpleCondition role="onSelection" qualifier="or" max="unbounded"/>
  <compoundAction operator="seq">
    <simpleAction role="set" value="$var" max="unbounded"
      qualifier="par"/>
    <simpleAction role="stop"/>
    <simpleAction role="start"/>
  </compoundAction>
</causalConnector>
```

Objetos NCLua Embutidos



Copyright © 2012 TeleMídia



79

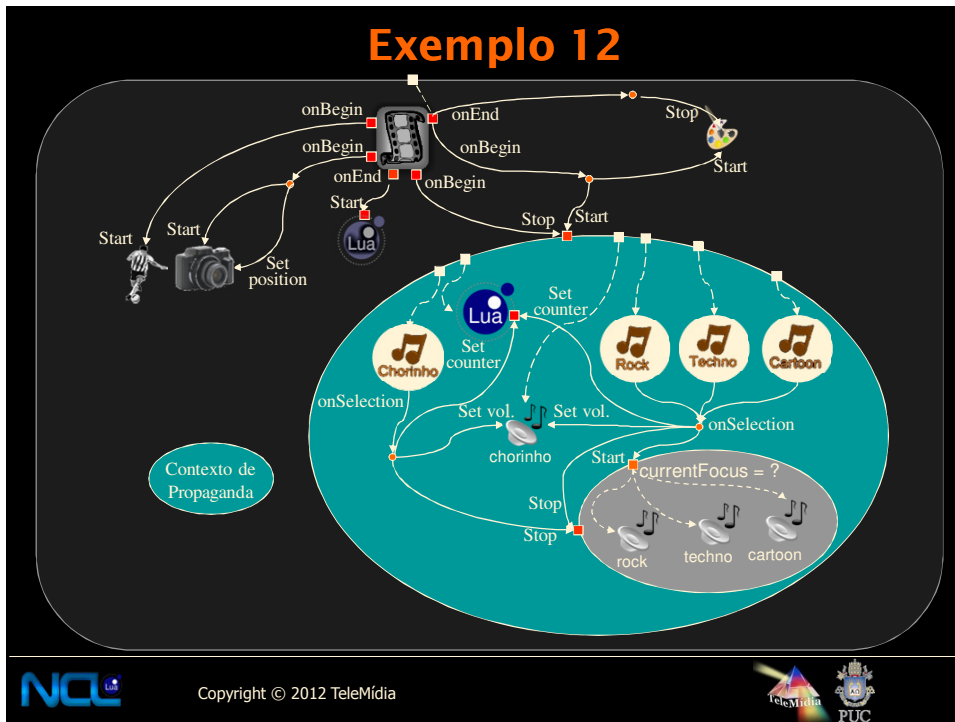
"O Primeiro João"

O número de vezes que você trocou de filme foi: 7



Copyright © 2012 TeleMídia



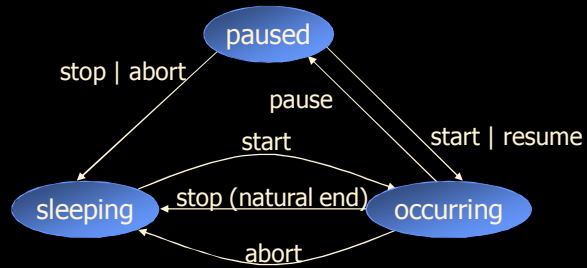


Interfaces

- <area id="a1" label="função interna">
- <property name="função interna" value="parametro">

NCL Copyright © 2012 TeleMídia TeleMídia PUC

Máquina de Estado de Evento



Copyright © 2012 TeleMídia



```
local counter = 0
local dx, dy = canvas:attrSize() -- dimensoes do canvas

function handler1 (evt)
  if evt.class=='ncl' and evt.type=='attribution' and evt.action=='start' and evt.name=='add' then
    counter = counter + evt.value
    event.post {
      class = 'ncl',
      type = 'attribution',
      name = 'add',
      action = 'stop',
      value = counter,
    }
  end
end

function handler2 (evt)
  canvas:attrColor ('black')
  canvas:drawRect('fill',0,0,dx,dy)
  canvas:attrColor ('yellow')
  canvas:attrFont ('vera', 24, 'bold')
  canvas:drawText (10,10, 'O número de vezes que você trocou de ritmo foi: '..counter)
  canvas:flush()
  event.post {
    class = 'ncl',
    type = 'presentation',
    label = 'fim',
    action = 'stop',
  }
end

event.register(handler1)
event.register(handler2,'ncl','presentation','fim','start')
```