

---

# **NCL Composer Documentation**

*Release 0.4.0*

**Roberto Azevedo**

Oct 26, 2017



<b>1</b>	<b>Prefácio</b>	<b>1</b>
1.1	Público alvo . . . . .	1
1.2	Requisitos . . . . .	1
<b>2</b>	<b>Introdução</b>	<b>3</b>
2.1	O que é NCL ? . . . . .	3
2.2	O que é o NCL Composer? . . . . .	3
2.3	Principais recursos do NCL Composer . . . . .	3
<b>3</b>	<b>Instalando o NCL Composer</b>	<b>5</b>
3.1	Windows . . . . .	5
3.2	Linux (Ubuntu/Debian) . . . . .	6
3.3	MacOS . . . . .	7
<b>4</b>	<b>Executando o NCL Composer</b>	<b>9</b>
4.1	Plataformas conhecidas . . . . .	9
4.2	Línguas . . . . .	9
4.3	Argumentos da linha de comando . . . . .	9
<b>5</b>	<b>Conceitos Básicos</b>	<b>11</b>
5.1	NCM . . . . .	11
5.2	O software NCL Composer . . . . .	12
<b>6</b>	<b>Exemplo 01: Reproduzindo um objeto de mídia</b>	<b>13</b>
6.1	Objetivo . . . . .	13
6.2	Passo 0: Criando um novo projeto . . . . .	13
6.3	Passo 1: Adicionando um objeto de mídia ( <b>O que?</b> ) . . . . .	15
6.4	Passo 2: Informando o objeto de mídia que serão inicialmente exibidos ( <b>Quando?</b> ) . . . . .	16
6.5	Passo 3: Definindo a região onde o objeto de mídia será apresentado ( <b>Onde?</b> ) . . . . .	16
6.6	Passo 4: Executando o exemplo . . . . .	17
<b>7</b>	<b>Exemplo 02: Reproduzindo um vídeo e uma imagem</b>	<b>19</b>
7.1	Objetivo . . . . .	19
7.2	Passo 1: Adicionando os objetos de mídia ( <b>O que?</b> ) . . . . .	19
7.3	Passo 2: Onde os objetos de mídia serão apresentados . . . . .	19
7.4	Passo 3: Quando os objetos de mídia serão apresentados . . . . .	20
7.5	Passo 4: Executando o exemplo . . . . .	20

<b>8</b>	<b>Exemplo 03: Iniciando e terminando dois objetos de mídia simultaneamente</b>	<b>23</b>
8.1	Objetivo . . . . .	23
8.2	Passo 1: Adicionando os objetos de mídia ( <b>O que?</b> ) . . . . .	23
8.3	Passo 2: Onde os objetos de mídia serão apresentados ( <b>Onde?</b> ) . . . . .	24
8.4	Passo 4: Quando os objetos de mídia serão apresentados: <i>Links</i> ( <b>Quando?</b> ) . . . . .	24
8.5	Passo 5: Executando o exemplo . . . . .	25
8.6	Passo 0: Salvar como... . . . .	25
<b>9</b>	<b>Telas e Menus do NCL Composer</b>	<b>27</b>
9.1	Janela Principal . . . . .	27
9.2	Visão Estrutural . . . . .	30
9.3	Visão de Leiaute . . . . .	40
9.4	Visão de Outline . . . . .	42
9.5	Visão de Propriedades . . . . .	42
9.6	Visão Textual . . . . .	44
9.7	Visão de Regras . . . . .	44
9.8	Visão de Validação . . . . .	45
9.9	Visão de Execução . . . . .	45
<b>10</b>	<b>Como mudar o tema do NCL Composer</b>	<b>47</b>
<b>11</b>	<b>Indices and tables</b>	<b>49</b>



Este manual tem o objetivo de agrupar todas as informações sobre o {nclcomposer} de forma detalhada.

## Público alvo

Este manual é destinado a qualquer pessoa que deseja aprender a desenvolver aplicações para o middleware de TV Digital [Ginga-NCL](<http://www.ginga.org.br>), independente de sua experiência em [NCL (Nested Context Language)](<http://www.ncl.org.br>).

## Requisitos

O único requisito que se espera do leitor são conhecimentos básicos de informática.

Para tirar total proveito deste manual você precisará também de um computador com:

- Sistema operacional **Linux**, **Windows** ou **MAC OS X**.

- Versão mais atual do **{nclcomposer}** (disponível em <http://composer.telemidia.puc-rio.br>) instalada e os plugins padrões habilitados.

### O que é NCL ?

Criada no [Laboratório TeleMídia](<http://www.telemidia.puc-rio.br>) da [PUC-Rio](<http://www.puc-rio.br>), a linguagem [NCL (Nested Context Language)](<http://www.ncl.org.br>) é uma linguagem declarativa para autoria de documentos hipermídia baseados no modelo conceitual NCM - Nested Context Model. NCL define como objetos de mídia são estruturados e relacionados, no tempo e espaço.

### O que é o NCL Composer?

O [NCL Composer](<http://composer.telemidia.puc-rio.br>) é uma ferramenta de autoria que visa facilitar o desenvolvimento de aplicações multimídia interativas usando [NCL (Nested Context Language)](<http://www.ncl.org.br>). Para isso, ele utiliza-se de várias abstrações gráficas (**visões**) que permitem ao autor focar em apenas um determinado aspecto, durante cada fase distinta da autoria.

Com isso, o NCL Composer permite que usuários não-programadores possam criar aplicações multimídia interativas. Além disso, o NCL Composer também pode ajudar no ensino/aprendizado dos principais conceitos da linguagem NCL e do modelo NCM.

### Principais recursos do NCL Composer

- Edição gráfica de documentos NCL
- Edição textual de documentos NCL
- Integração com ferramentas de execução
- Multiplataforma (Linux, Windows e MacOS)

Além disso, o NCL Composer foi desenvolvido com o objetivo de ser facilmente extensível. Novas funcionalidades podem ser facilmente adicionadas por meio de *plugins*.



---

## Instalando o NCL Composer

---

### Windows

Para instalar o NCL Composer em ambiente Windows:

- Baixe a versão mais atual do NCL Composer para Windows em: <<http://composer.telemidia.puc-rio.br/en/download>>
- O arquivo deve ter o nome na forma *nclcomposer-installer-X.X.X.exe*, onde X.X.X deve conter a versão do NCL Composer. Execute esse arquivo.
- Caso você tenha alguma versão anterior instalada, o instalador irá perguntar se deseja desinstalá-la antes de prosseguir.
- Caso contrário, a seguinte tela será exibida:

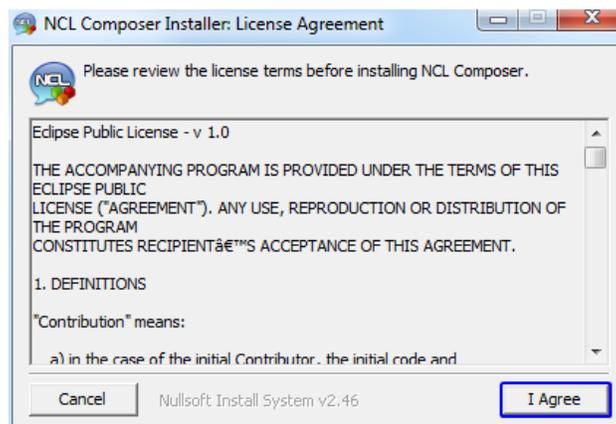


Fig. 3.1: Instalando (Passo 1)

- Clique no botão *I Agree* para aceitar a licença do NCL Composer.

- Após aceitar a licença do software, será possível escolher quais *plugins* serão instalados na sua máquina, por meio da tela (em caso de dúvida não altere):

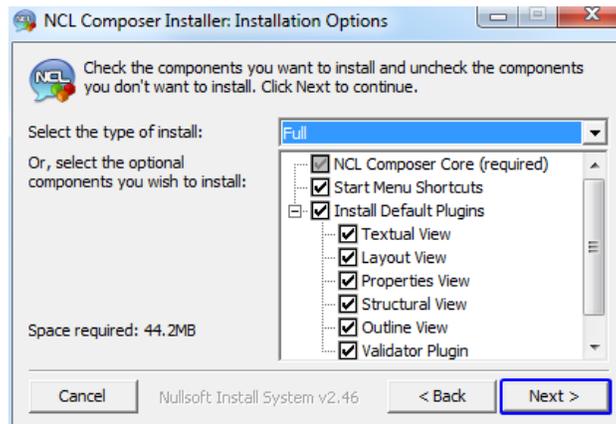


Fig. 3.2: Instalando (Passo 2)

- Clique no botão *Next*, e será possível escolher o local onde o NCL Composer será instalado:

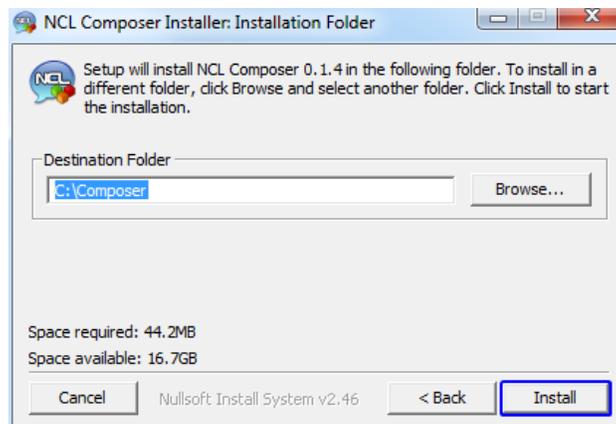


Fig. 3.3: Instalando (Passo 3)

- Por fim, o NCL Composer será instalado, e a seguinte tela exibida:
- Caso não tenha ocorrido nenhum problema, feche o instalador. Será possível executar o NCL Composer através do menu *Iniciar->TeleMídia->NCL Composer*.

## Linux (Ubuntu/Debian)

Para instalar o NCL Composer em ambiente Linux (Ubuntu/Debian):

- Baixe a versão mais atual do NCL Composer para Linux (Ubuntu/Debian) em: <http://composer.telemidia.puc-rio.br/en/download>
- O arquivo deve ter o nome na forma *nclcomposer\_X.X.X\_i386.deb*, onde X.X.X deve conter a versão do NCL Composer. Para instalar o arquivo que foi baixado, execute o seguinte comando na linha de comando:

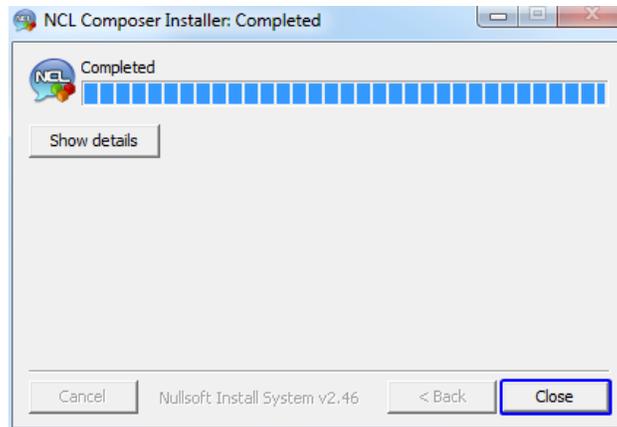


Fig. 3.4: Instalando (Passo 4)

```
$ dpkg -i nclcomposer-X.X.X.deb
```

- Caso a instalação ocorra sem problemas, é possível acessar o NCL Composer através do menu de aplicações, ou por meio do comando:

```
$ nclcomposer
```

## MacOS

Para instalar o NCL Composer em ambiente MacOS:

- Baixe a versão mais atual do NCL Composer para MacOS em: <http://composer.telemidia.puc-rio.br/en/download>
- O arquivo deve ter o nome na forma *nclcomposer-X.X.X-YYYYMMDD.dmg*, onde X.X.X deve conter a versão do NCL Composer e YYYYMMDD a data do lançamento.

---

**Note:** O *release* do NCL Composer para MacOS é realizado através de um arquivo [DMG](<https://pt.wikipedia.org/wiki/dmg>). O arquivo DMG é uma imagem de disco, que se assemelha a um arquivo compactado (ZIP ou RAR). Dessa forma, para instalar a aplicação é necessário extraí-la de dentro do arquivo DMG.

---

- Para instalar (extrair) o NCL Composer, clique duas vezes no arquivo DMG baixado. Assim, uma janela será exibida contendo a aplicação do NCL Composer. Clique e arraste a aplicação do NCL Composer para dentro da pasta de aplicações do sistema (normalmente */Applications/*).
- Pronto!

---

**Note:** Este processo é semelhante para qualquer arquivo DMG. Caso queira assistir um vídeo explicativo, acesse o [Doutor Apple](<https://www.youtube.com/watch?v=gNqDTE7Vvwc>).

---



---

## Executando o NCL Composer

---

As duas formas mais comuns de iniciar o NCL Composer são clicando no ícone do NCL Composer (por exemplo, na Área de Trabalho ou no Menu Iniciar) ou através de uma linha de comando.

### Plataformas conhecidas

As plataformas nas quais é sabido que o NCL Composer funciona são: “GNU/Linux”, “Apple Mac OS”, e “Microsoft Windows”.

### Línguas

Atualmente o NCL Composer está disponível em três línguas: **Inglês**, **Português** e **Espanhol**. É possível alterar a língua corrente em *Edit->Preferences->General*.

### Argumentos da linha de comando

Ao executar o NCL Composer por meio de uma linha de comando, os seguintes argumentos são reconhecidos.

**--help, -h**

Mostra a lista de opções.

**--version, -v**

Mostra a versão.



---

## Conceitos Básicos

---

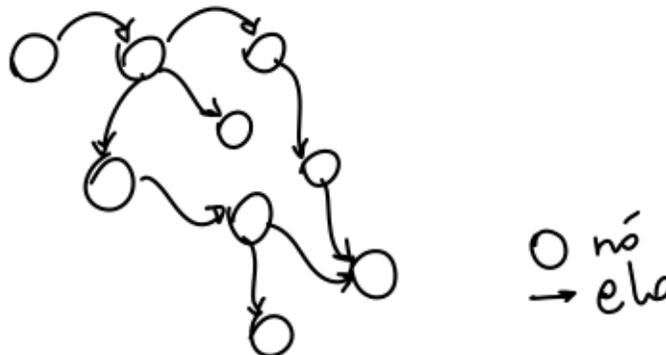
A linguagem NCL—e, consequentemente, o NCL Composer—baseiam-se no modelo NCM (Nested Context Model) para modelar aplicações multimídia interativas. Esta seção discute os principais conceitos do NCM e da NCL, os quais são úteis para utilizar o NCL Composer de forma eficiente.

### NCM

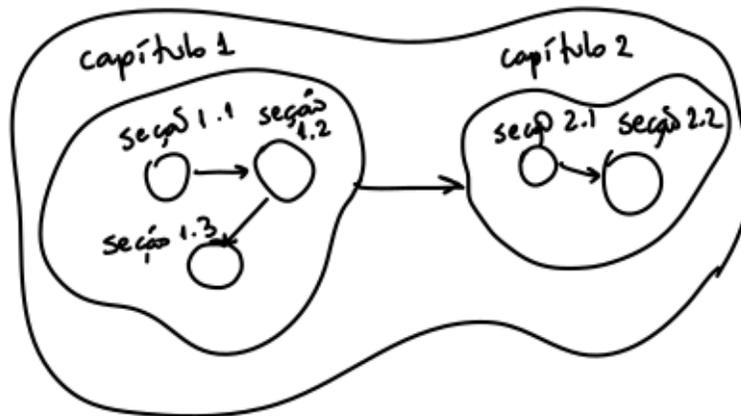
#### Nós e elos

No NCM, aplicações multimídia são modeladas por meio de nós (**nodes**) e elos (**links**) (Figura 1).

Os **nós de mídia** representam os objetos de mídias da aplicação (vídeos, imagens, textos etc.), enquanto os **elos** representam relacionamentos entre esses nós. Exemplos de relacionamentos entre os nós são os **relacionamentos causais** (por exemplo, no NCM é possível criar um elo entre dois nós, informando que quando a apresentação de um nó começar, a apresentação de outro nó também deve ser iniciada).



Além dos nós de mídias, o NCM também permite agrupar nós e links em **nós de composição** (ou **contextos**), que permitem agrupar logicamente os nós e elos de uma aplicação. Contextos podem também agrupar outros contextos, recursivamente (veja figura abaixo).



Modelar uma aplicação multimídia no NCM então é definir **quais** os objetos de mídia fazem parte da aplicação (através dos **nós de mídia**), **como** eles estão estruturados (definindo-se **nós de contextos**) e **quando** esses objetos de mídia serão apresentados (o que é realizado por meio da definição de **elos** entre os nós de mídia).

## Propriedades, regiões e descritores

Objetos de mídia podem conter um conjunto de **propriedades**. Propriedades definem várias características dos objetos de mídia, como, por exemplo, o seu posicionamento (propriedades *top* e *left*), a sua dimensão (propriedades *width* e *height*), a sua transparência (propriedade *transparency*) etc. Para uma lista completa das propriedades suportadas pela NCL é possível consultar o [NCL Handbook](<http://handbook.ncl.org.br/doku.php?id=property#predefined-properties>). Propriedades também podem ser modificadas dinamicamente durante a exibição do objeto de mídia.

As propriedades de um objeto de mídia são definidas no próprio objeto de mídia. Porém, a NCL também define dois outros elementos que possibilitam reusar um conjunto de propriedades em mais de um objeto de mídia, são eles: a: **descritores** e **regiões**.

Um **descriptor** é usado para agrupar um conjunto qualquer de propriedades e, assim, é possível compartilhá-las por mais de uma mídia. Por exemplo, se duas mídias tocam na mesma região da tela e com a mesma transparência, podemos criar um descriptor *meudescritor* que agrupa essas propriedades. Em vez de defini-las novamente para cada um dos objetos de mídia, cada objeto de mídia pode apenas referenciar este descriptor.

Uma **região** permite agrupar o conjunto de propriedades espaciais de um objeto de mídia. Além disso, por meio de regiões também é mais fácil definir

## Portas e âncoras

## Conectores

## O software NCL Composer

O NCL Composer é uma ferramenta de autoria para aplicações multimídia baseada no NCM. Por meio do NCL Composer é possível criar uma aplicação multimídia através de diversas **visões** sobre um documento NCL. Cada visão traz alguma representação diferente do documento NCL que permite visualizar/editar algumas características específicas.

Atualmente, as principais visões do NCL Composer são: Visão Estrutural, Visão de Leiante, Visão de Outline, Visão Propriedades, Visão de Regras, Visão de Validação e Visão de Execução.

---

## Exemplo 01: Reproduzindo um objeto de mídia

---

### Objetivo

Neste primeiro exemplo, o objetivo é apenas exibir um objeto de mídia, em tela cheia.

Para isso, iremos:

1. Criar um novo projeto;
2. Adicionar o objeto de mídia (no exemplo, usaremos um vídeo) na aplicação;
3. Informar que esse objeto será iniciado quando a aplicação começar, e
4. Determinar a região da tela onde esse objeto de mídia será exibido.

### Passo 0: Criando um novo projeto

Para criarmos o nosso exemplo, o primeiro passo é criarmos um novo projeto.

Ao abrir o NCL Composer, a seguinte tela de boas-vindas é apresentada:

Nessa tela, é possível criar um novo projeto por meio do menu *File->New Project*, por meio do botão *New Project* no canto esquerdo inferior da tela de boas-vindas, ou simplesmente pressionando *Ctrl+N*.

Feito isso, a seguinte janela será exibida:

Atribua um nome para o projeto que você está criando (aqui atribuímos o nome *exemplo-01*), selecione o diretório onde deseja salvá-lo e deixe as opções *Copy and Import the default Connector Base* (Copie e importe a base de conectores padrão) e *Add a default region to the project* (adicione uma região padrão ao projeto) selecionadas.

---

**Note:** Nas próximas seções serão discutidos em detalhes o que são os conectores e bases de conectores. Por enquanto, apenas deixe esta opção marcada, pois ela será útil para os próximos exemplos.

---

Por fim, pressione o botão *Finish*, e o seu novo projeto será criado e aberto.

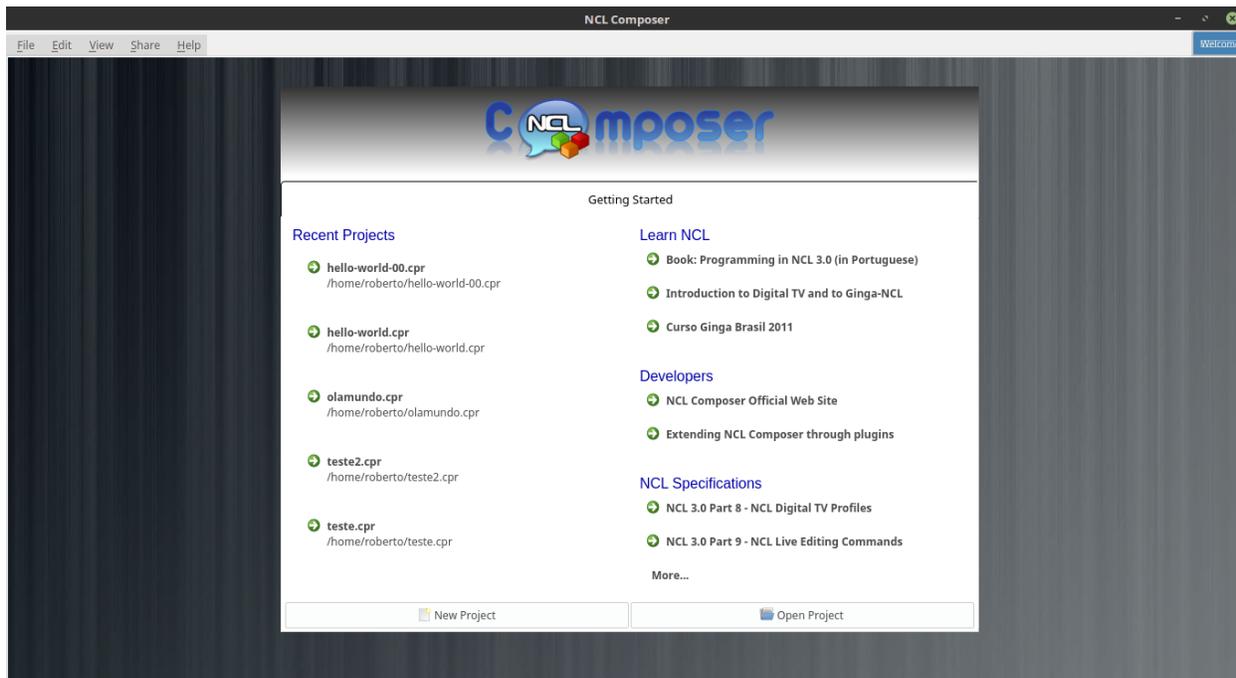


Fig. 6.1: Tela de boas-vindas do NCL Composer.

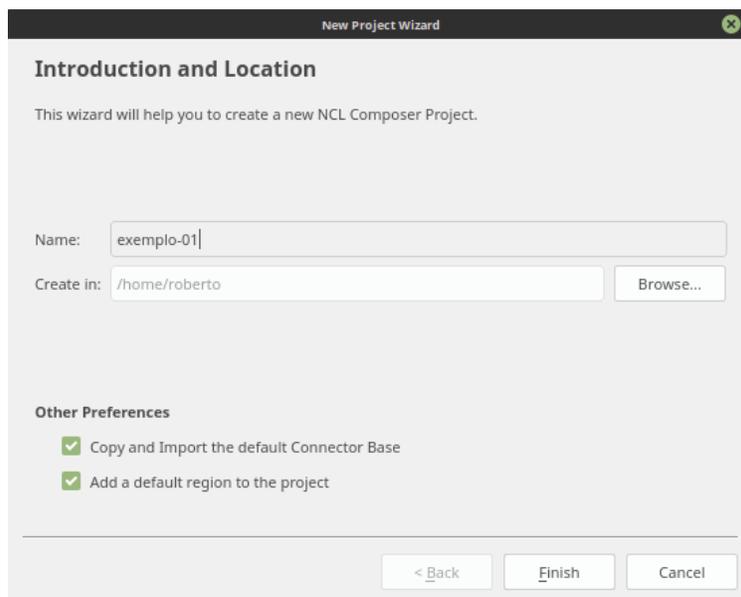


Fig. 6.2: Criação de um novo projeto.

A animação a seguir exemplifica a criação de um projeto.

Fig. 6.3: Criando um novo projeto.

Ao criar o novo projeto, o NCL Composer irá automaticamente abri-lo, onde teremos acesso às visões: Visão Estrutural, Visão de Leiaute, Visão de Outline, Visão Propriedades, Visão de Regras, Visão de Validação e Visão de Execução.

## Passo 1: Adicionando um objeto de mídia (O que?)

Com o nosso projeto criado e aberto, podemos começar a desenvolver a nossa aplicação. Como primeiro passo, temos que adicionar o objeto de mídia da nossa aplicação (neste primeiro exemplo, teremos apenas um objeto de mídia, um vídeo).

Para inserir os objetos de mídia (áudio, vídeo, imagens etc.) em nossa aplicação, a forma mais simples é apenas arrastá-los do seu gerenciador de arquivos para dentro da Visão Estrutural do NCL Composer.

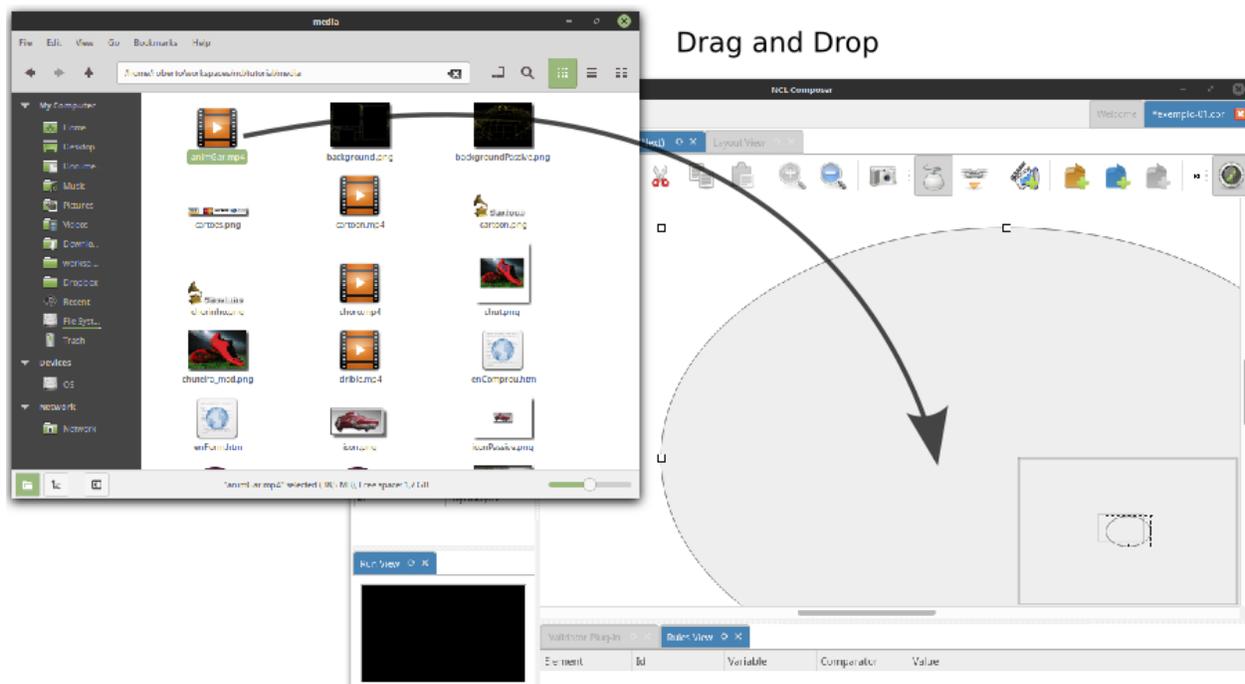


Fig. 6.4: Arraste o objeto de mídia do seu gerenciador de arquivos para a Visão Estrutural do NCL Composer para inserí-lo como objetos de mídia da aplicação

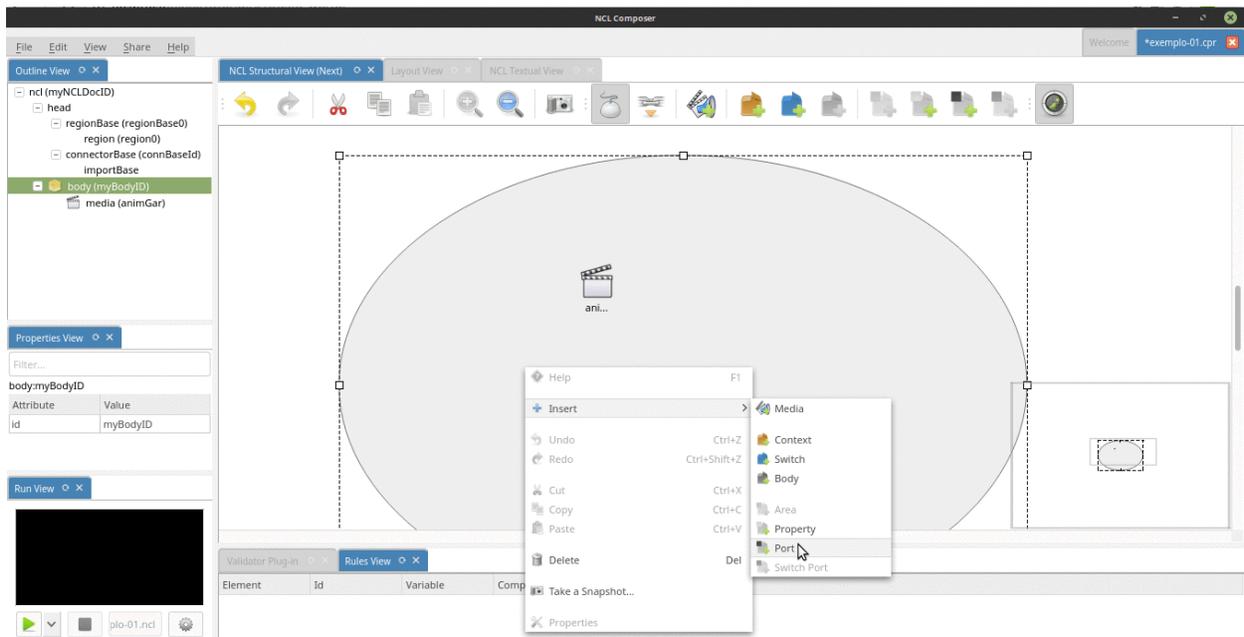
**Note:** Outra forma de inserir um objeto de mídia, na Visão Estrutural é clicar com o botão direito sobre o corpo do documento e selecionar o menu *Insert -> Media*. Neste caso, é importante não esquecer de definir o caminho para o objeto de mídia, o que poder ser realizado selecionando o objeto que você acabou de editar e alterando o valor do atributo *src* (ou na *Visão de Propriedades* ou na *Visão Textual*).

A animação a seguir, exemplica como podemos adicionar um novo objeto de mídia, arrastando a partir do gerenciador de arquivos.

## Passo 2: Informando o objeto de mídia que serão inicialmente exibidos (Quando?)

Para informar que o objeto de mídia que acabamos de inserir deve iniciar quando a nossa aplicação iniciar, é suficiente criarmos uma **porta** no corpo do documento, mapeando-a para o objeto de mídia. As portas de um contexto (inclusive do corpo do documento, que nada mais é do que o contexto inicial do documento) definem quais os objetos daquele contexto serão iniciados quando aquele contexto for iniciado.

Para inserirmos uma porta no corpo do documento pela *Visão Estrutural* é suficiente clicarmos com o botão direito sobre o corpo e selecionarmos o menu *Insert->Port*.



Para mapearmos a porta que acabamos de criar para o objeto de mídia, podemos selecionar a ferramenta *Link* na barra de tarefas e clicar sobre a porta e, mantendo o botão do mouse pressionado, arrastar o ponteiro até o objeto de mídia.

**Note:** Outra forma de ativar a ferramenta *Link* é segurando a tecla SHIFT enquanto faz-se o mesmo procedimento acima.

Com isso, já definimos que nossa aplicação é constituída apenas de um objeto de mídia e que quando a aplicação começar, esse objeto de mídia será executado.

A animação a seguir exemplifica como criar e mapear uma porta no corpo do documento.

## Passo 3: Definindo a região onde o objeto de mídia será apresentado (Onde?)

Para finalizar este primeiro exemplo só falta agora definirmos **onde** (em que região da tela) o objeto de mídia da nossa aplicação será exibido.

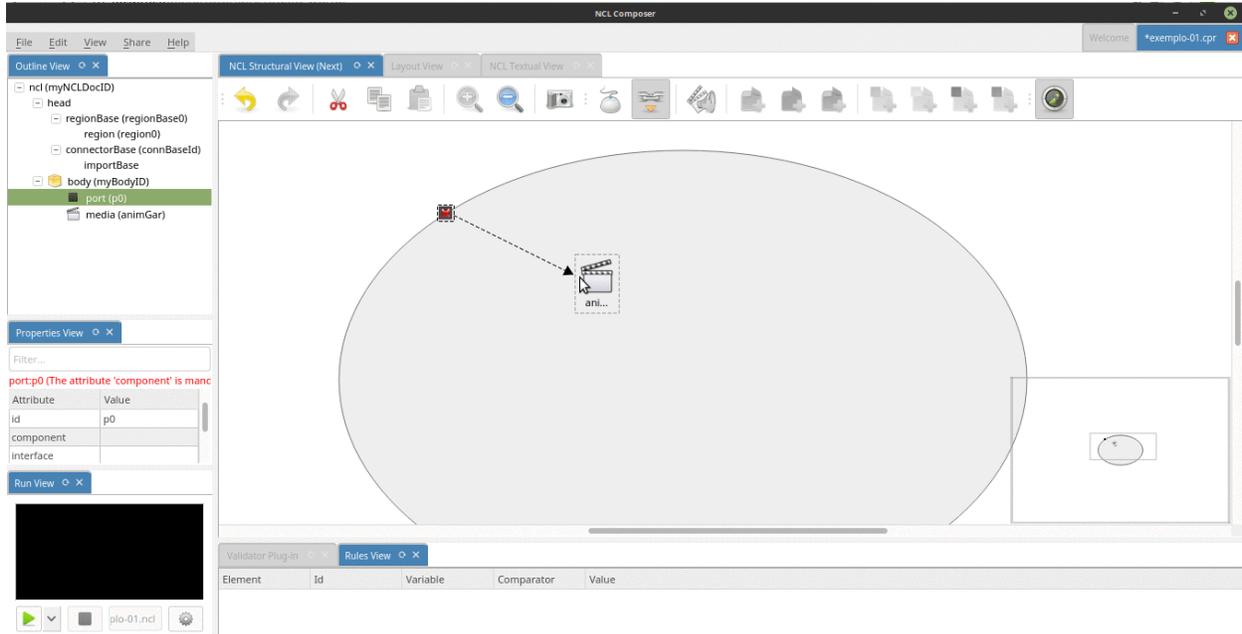


Fig. 6.5: Mapeando porta para objeto de mídia.

Fig. 6.6: Mapeando porta para objeto de mídia.

Como ao criarmos o nosso projeto, deixamos a opção “Add a default region to the project” selecionada, ao criarmos o projeto já criamos também uma região que ocupa toda a tela.

Para associarmos a mídia que adicionamos no nosso projeto à essa região, basta arrastamos essa mídia (por exemplo, a partir da **visão de outline**) para a região na visão de leiaute. Veja a animação a seguir:

Fig. 6.7: Associando uma media à uma região.

Ao fazermos isso, o NCL Composer irá nos perguntar se queremos “copiar os dados da região” para a mídia ou se queremos criar um descritor. Um descritor serve para podermos reusar as mesmas propriedades em mais de uma mídia. Exemplos destas propriedade são o posicionamento e a largura e a altura do objeto de mídia. Na grande maioria das vezes (inclusive aqui) iremos querer criar um descritor.

## Passo 4: Executando o exemplo

Feito isso, o nosso primeiro exemplo já está completo.

Para executá-lo, basta executar o projeto no botão “Run NCL Project”  na *Run view*.

Fig. 6.8: Executando a aplicação.

---

## Exemplo 02: Reproduzindo um vídeo e uma imagem

---

### Objetivo

O objetivo deste exemplo é apresentar um vídeo e uma imagem na tela.

Para isso, iremos:

- Adicionar os dois objetos de mídia (um vídeo e uma imagem) na nossa aplicação;
- Criar duas regiões (uma para cada objeto) e associá-las aos objetos de mídia; e
- Informar que os dois objetos devem ser apresentados quando a aplicação começar.

### Passo 1: Adicionando os objetos de mídia (O que?)

Neste exemplo, queremos uma aplicação com dois objetos de mídia (um vídeo e uma imagem). Para inserí-los no seu projeto, basta arrastar as duas mídias do seu gerenciador de arquivos para a Visão Estrutural do NCL Composer.

---

**Note:** Caso precise de mais informações sobre como criar um novo projeto veja o [Passo 0 do Exemplo 01](01.hello\_world).

---

### Passo 2: Onde os objetos de mídia serão apresentados

Agora iremos definir **onde** os objetos de mídia serão apresentados. Para isso iremos criar regiões na tela, por meio da Visão de Leiaute e associar os objetos de mídia à essas regiões.

## Criando as regiões

Como podemos perceber ao acessar a Visão de Leiaute, por padrão, o NCL Composer já cria uma região *region0* que ocupa toda a tela. Iremos reusar essa região para o vídeo (queremos que ele ocupe toda a tela) e iremos criar uma nova região para a imagem. Para isso, podemos clicar com o botão direito do mouse na região existente e acessar a opção *Add region* do menu contextual. Uma nova região será criada, e podemos agora redimensioná-la e posicioná-la da forma que nos convier.

## Associando os objetos de mídia às regiões

Depois de criarmos as regiões, agora devemos associar cada um dos objetos de mídia (o vídeo e a imagem) a cada uma das suas regiões. Para isso, basta arrastar o objeto de mídia a partir da Visão de Outline para a sua respectiva região na Visão de Leiaute.

## Passo 3: Quando os objetos de mídia serão apresentados

Como os dois objetos de mídia iniciarão no começo da aplicação, uma forma simples de definir esse comportamento é criando-se uma porta para cada um desses objetos.

Diferente de objetos de mídia de vídeo e áudio, objetos de mídia estáticos—como, por exemplo, imagens—não possuem duração implícita. Neste caso, caso iniciemos sua execução ela sendo apresentada indefinidamente. Para definirmos a duração de uma mídia (por exemplo, uma imagem) de forma explícita podemos o atributo *explicitDur* no descriptor associado à esta mídia (veja a animação abaixo, onde definimos que a imagem irá durar *5s*).

## Passo 4: Executando o exemplo

Feito isso, o nosso exemplo já está completo e podemos executá-lo na Visão de Execução.





---

### Exemplo 03: Iniciando e terminando dois objetos de mídia simultaneamente

---

#### Objetivo

O objetivo deste exemplo é iniciar e terminar dois objetos de mídia simultaneamente.

Para isso, iremos:

1. Adicionar os dois objetos de mídia que fazem parte da aplicação (um vídeo e uma imagem);
2. Criar duas regiões, uma para o vídeo e uma para a imagem, e associar os respectivos objetos de mídia à suas regiões;
3. Informar que o vídeo começará a ser apresentado no início da aplicação; e
4. Criar dois elos:
  - Um informando que a imagem iniciará sua apresentação quando o vídeo começar a ser executado; e
  - Outro informando que a imagem irá terminar a sua execução quando o vídeo terminar a sua execução.

#### Passo 1: Adicionando os objetos de mídia (O que?)

Da mesma forma que no exemplo anterior iremos arrastar os dois objetos de mídia que ainda fazem parte da nossa aplicação (um vídeo e uma imagem) para a Visão Estrutural.

Fig. 8.1: Arrastando um vídeo e uma imagem para a Visão Estrutural.

## Passo 2: Onde os objetos de mídia serão apresentados (Onde?)

Agora iremos definir **onde** os objetos de mídia serão apresentados. Para isso, iremos criar regiões na tela por meio da Visão de Leiaute e associar os objetos de mídia à essas regiões.

### Criando as regiões

Como podemos perceber ao acessar a Visão de Leiaute, por padrão, o NCL Composer já cria uma região *region0* que ocupa toda a tela. Iremos reusar essa região para o vídeo (queremos que ele ocupe toda a tela) e iremos criar uma nova região para a imagem. Para isso, podemos clicar com o botão direito do mouse na região existente e acessar a opção *Add region* do menu contextual. Uma nova região será criada, e podemos agora redimensioná-la e posicioná-la da forma que nos convier.

Fig. 8.2: Arraste um vídeo e uma imagem para a Visão Estrutural.

### Associando os objetos de mídia às regiões

Depois de criarmos as regiões, agora devemos associar cada um dos objetos de mídia (o vídeo e a imagem) a cada uma das suas regiões. Para isso, basta arrastar o objeto de mídia a partir da Visão de Outline para a sua respectiva região na Visão de Leiaute.

## Passo 4: Quando os objetos de mídia serão apresentados: *Links* (Quando?)

Para informar **quando** os objetos de mídia serão executados iremos:

- Criar uma **porta** para o vídeo, informando que ele deve ser apresentado quando a aplicação iniciar sua execução; e
- Criar dois elos sincronizando a imagem com o vídeo acima. O primeiro elo irá informar que **a apresentação da imagem deve começar quando a apresentação do vídeo começar**, e o segundo elo irá informar que **a apresentação da imagem deve terminar quando a apresentação do vídeo terminar**.

### Informando o objeto que será inicialmente apresentado

Para criarmos uma porta para o vídeo principal podemos, na Visão Estrutural, selecionar o *body* e pressionar o botão *Insert port* na barra de ferramentas. Feito isso, teremos uma nova porta no *body*.

Para associarmos essa porta ao vídeo de nossa aplicação podemos selecionar a *ferramenta de link* na barra de ferramentas e clicar com o botão esquerdo do mouse e (ainda segurando o botão) arrastar o ponteiro para o vídeo.

Com isso, já definimos que **quando a aplicação iniciar o vídeo iniciará a sua apresentação**.

## Sincronizando o início e o fim do vídeo com o início e fim da imagem

Para sincronizarmos a apresentação do vídeo com a apresentação da imagem, iremos usar elos. Para criarmos um elo entre dois objetos de mídia na Visão Estrutural podemos:

1. selecionar a *ferramenta de link* na barra de ferramentas; e, depois
2. clicar com o botão direito do mouse sobre o primeiro objeto de mídia (no caso do nosso exemplo, o vídeo), mantendo o botão do mouse pressionado, arrastar o ponteiro até o segundo objeto (no nosso exemplo, a imagem).

Com isso, iremos criar um elo no qual o primeiro objeto clicado participará como parte da condição, e a ação será executada sobre o segundo objeto (para o qual arrastamos o elo).

Logo após criarmos um elo com o procedimento acima, uma nova janela irá abrir, na qual poderemos detalhar as informações desse elo. Em especial, devemos escolher qual o **tipo de elo** que estamos criando (ou seja, qual o **conector** daquele elo) e quais as **ações** e **condições** dos objetos que selecionamos. Geralmente, só modificamos as ações ou condições quando o elo que estamos criando possuir mais de uma condição ou mais de uma ação.

---

**Note:** Alguns tipos de elo também podem receber parâmetros (e.g. um parâmetro *delay* que informa com quanto tempo depois de a condição satisfeita a ação deve ser executada). E também podemos definir os valores desses parâmetros na janela de criação de elos.

---

No caso do nosso exemplo, iremos escolher o tipo do primeiro elo *onBeginStart* (ou **ao começar** “isto”, **inicie** “aquilo”), informando que quando o primeiro objeto selecionado (o vídeo) começar a sua apresentação, queremos que o segundo objeto (a imagem) também comece a sua apresentação.

Para o segundo elo, iremos escolher o tipo *onEndStop* (ou **ao terminar** “isto”, **termine** aquilo). Dessa forma, iremos informar que o final do vídeo deve também disparar a finalização da imagem.

## Passo 5: Executando o exemplo

Pronto! O nosso Exemplo 03 está pronto. Para testá-lo basta executá-lo na Visão de Execução. O resultado deve ser o início do vídeo e da imagem no início da aplicação e os dois devem continuar apresentando até o final do vídeo, quando os dois devem ser finalizados, e a aplicação termina a sua execução.

---

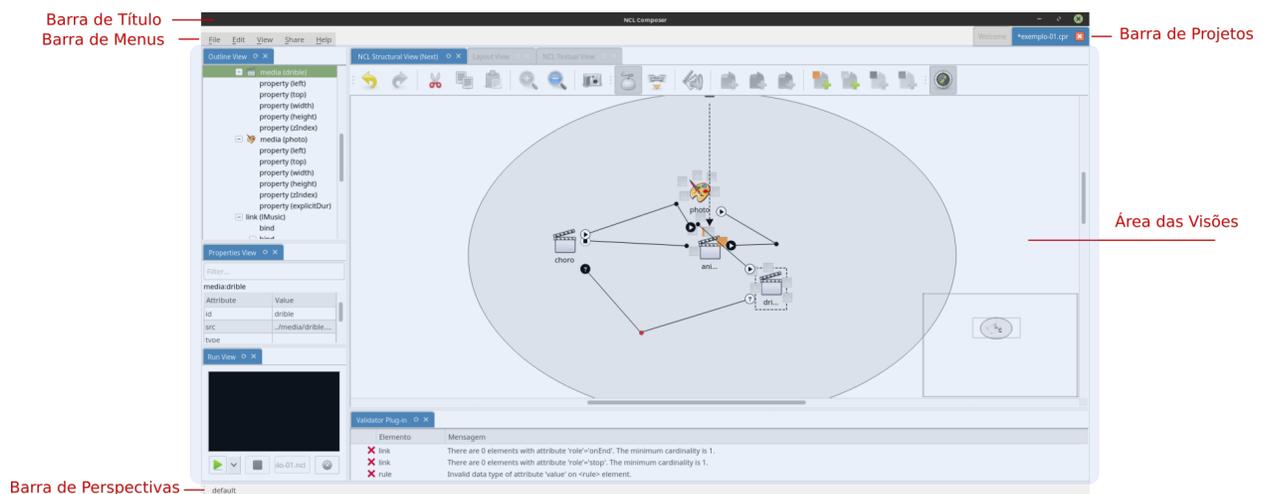
## Passo 0: Salvar como...

Para reusarmos o que já foi feito no exemplo anterior, vamos abrí-lo e criar um novo projeto baseado nele, por meio do menu *File->Save as...*



## Telas e Menus do NCL Composer

## Janela Principal



## Barra de título

Evidencia o nome do NCL Composer e do projeto corrente.

## Barra de projetos

Lista a tela de boas-vindas e a lista de projetos abertos, permitindo navegar entre eles.

## **Área das Visões**

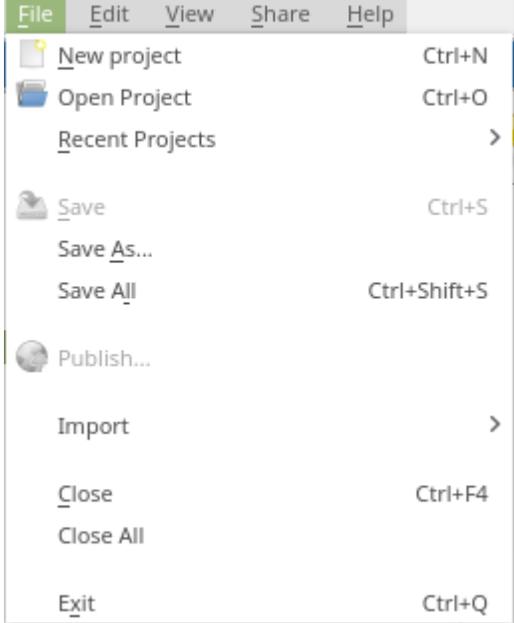
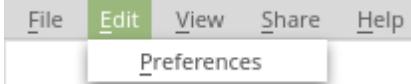
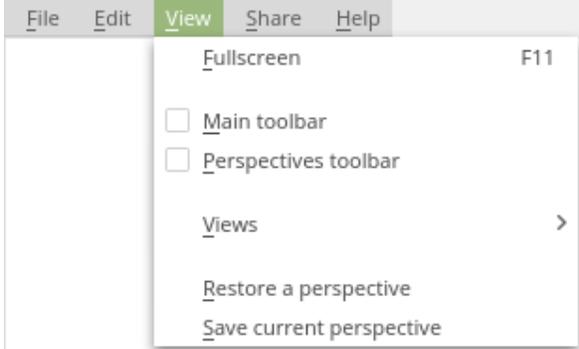
Área de desenho das diversas visões. Cada visão pode ter seus próprios elementos de interface. Também é possível organizar e salvar a disposição das diversas visões por meio das perspectivas.

## **Barra de perspectivas**

Apresenta as perspectivas salvas pelo usuário e permite facilmente alternar entre elas.

## **Barra de menus**

Mostra as entradas de menu disponíveis.

Menu	Screenshot
<p><b>File:</b> Da acesso às seguintes opções:</p> <ul style="list-style-type: none"> <li>• Novo projeto</li> <li>• Abrir</li> <li>• Salvar</li> </ul>	
<p><b>Edit:</b></p>	
<p><b>View:</b></p>	
<p><b>Share:</b></p>	
<p><b>Help:</b></p>	

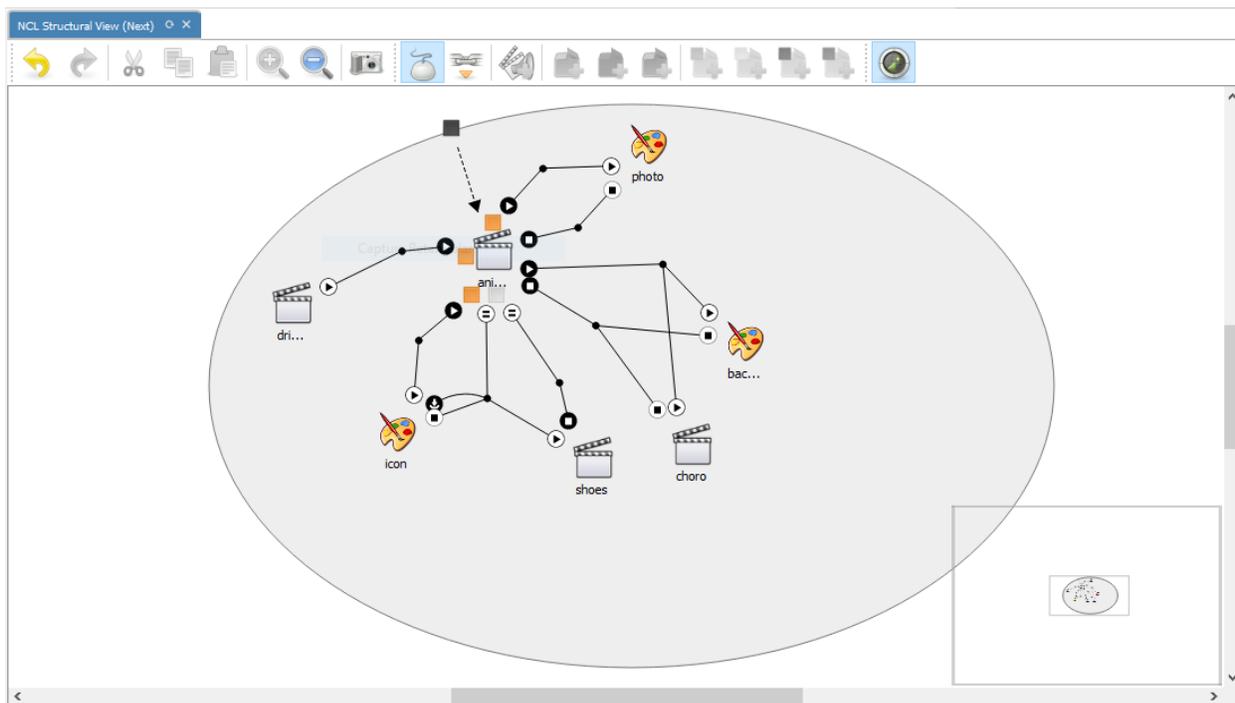
## Atalhos

Os atalhos globais (válidos em qualquer visão) do NCL Composer são:

- “Ctrl”+“n” - Novo projeto.
- “Ctrl”+“o” - Abrir projeto.
- “Ctrl”+“s” - Salvar o projeto corrente.
- “Ctrl”+“Shift”+“s” - Salvar todos os projetos abertos.
- “Ctrl”+“Alt”+“s” - Salvar como...: Salva o projeto corrente com outro nome.
- “Ctrl”+“F4” - Fechar projeto.
- “Alt+F4” ou “Ctrl”+“Q” - Fechar o Inclcomposer!
- “F11” - Habilita/desabilita visualização em tela cheia.
- “F1” - Abre a ajuda do Inclcomposer!
- “F5” - Atualiza o modelo da visão com foco.
- “Ctrl”+“1”, “Ctrl”+“2”, ... - Alterna entre as perspectivas.

## Visão Estrutural

A Visão Estrutural permite que os autores de aplicações interajam visualmente com a estrutura lógica do documento NCL. Em NCL, a estrutura lógica é representada pelos seus objetos (nós de conteúdo e nós de composição) e os elos entre esses objetos (representadas pelos elementos ).



## Ações

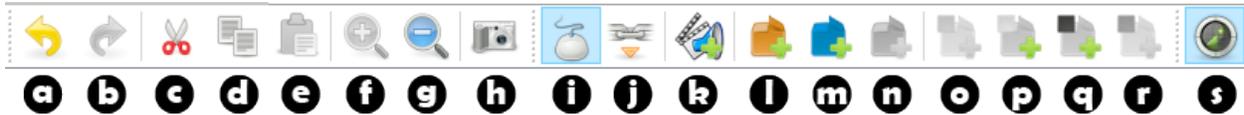
As ações são as diversas atividades que o usuário pode realizar durante o processo de autoria. Uma ação pode ser executada de várias formas: através da barra de ferramenta, do menu contextual, de um atalho de teclado, do *mouse*

e da combinação de várias dessas formas. Abaixo são descritas cada uma das ações disponíveis na Visão Estrutural e como elas podem ser executadas.

Ação	Descrição	
Desfazer	Desfaz a última ação realizada na visão. Esta ação é ignorada caso não exista nenhuma ação para ser desfeita. <i>Inserir, remover, mover e redimensionar</i> são exemplos de ações que podem ser desfeitas. Por outro lado, selecionar não é uma ação que pode ser desfeita. Na versão atual, não existe um limite na quantidade de ações que podem ser desfeitas.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas), <i>menu contextual</i> e <i>atalho</i> .
Refazer	Refaz a última ação realizada na visão. Esta ação é ignorada caso não exista nenhuma ação para ser refeita, ou seja, nenhuma ação foi desfeita anteriormente. Na versão atual, não existe um limite na quantidade de ações que podem ser desfeitas.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas), <i>menu contextual</i> e <i>atalho</i> .
Recortar	Copia a entidade selecionada para a <b>área de transferência</b> e <b>remove</b> ela da visão. Esta ação é ignorada caso nenhuma entidade tenha sido selecionada e se a entidade selecionada é do tipo <i>body</i> .	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas), <i>menu contextual</i> e <i>atalho</i> .
Copiar	Copia a entidade selecionada para a <b>área de transferência</b> . Esta ação é ignorada caso nenhuma entidade tenha sido selecionada e se a entidade selecionada é do tipo <i>body</i> .	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas), <i>menu contextual</i> e <i>atalho</i> .
Colar	Adiciona a entidade copiada para <i>área de transferência</i> como filha da entidade selecionada. Esta ação é ignorada caso nenhuma entidade tenha sido recortada ou copiada e se a entidade selecionada não permita que seja adicionado como um de seus filhos entidades do tipo selecionada. Por exemplo, não é possível colar um contexto em uma mídia.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas), <i>menu contextual</i> e <i>atalho</i> .
Deletar	Remove a entidade selecionada da visão. Esta ação é ignorada caso nenhuma entidade esteja selecionada na visão.	Esta ação pode ser executada através do: <i>menu contextual</i> e <i>atalho</i> .
Mover	Move a entidade selecionada para uma entidade de destino. Para a <i>structural_view</i> , mover significa alterar o parêntesco da entidade selecionada. A entidade de destino é determinada manipulando a entidade selecionada por meio de <i>drag-n-drop</i> para o interior da entidade de destino. Esta ação é ignorada caso a entidade de destino não permita que seja adicionado como um de seus filhos entidades do tipo selecionado.	Esta ação pode ser executada através de <i>atalho</i> .
Aumentar Zoom	Aumenta o <i>zoom</i> na visão. Esta ação é ignorada caso a visão tenha alcançado seu <i>zoom</i> máximo.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas) e <i>atalho</i> .
Diminuir Zoom	Diminua o <i>zoom</i> na visão. Esta ação é ignorada caso a visão tenha alcançado seu <i>zoom</i> mínimo.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas) e <i>atalho</i> .
Tirar uma Foto	Exporta o projeto como imagem (.png). O projeto pode ser exportado mesmo que nenhuma entidade tenha sido adicionada.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas) e <i>menu contextual</i> .
Habilitar Ferramenta de Ponteiro	Altera o modo de edição da <i>structural_view</i> para “ponteiro”. Quando no modo de “ponteiro”, a <i>structural_view</i> permite selecionar, mover e redimensionar as entidades da através do <i>mouse</i> . O modo de “ponteiro” é desabilitado sempre o modo de “link” é habilitado.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas) e <i>atalho</i> .
Habilitar Ferramenta de Link	Altera o modo de edição da <i>structural_view</i> para “link”. Quando no modo de “link”, a <i>structural_view</i> permite criar <i>links</i> (relacionamentos) entre duas entidades. Para criar um novo <i>link</i> , o usuário deve: <b>i</b> ) clicar com o botão esquerdo do <i>mouse</i> na entidade de origem, <b>ii</b> ) arrastar o	Esta ação pode ser executada através da: <i>barra de ferramentas</i> (#barra-de-ferramentas) e <i>atalho</i> .

## Barra de Ferramentas

A partir da versão 0.3.0 a Visão Estrutural disponibiliza uma *barra de ferramentas* com as principais ações da visão.



A figura acima apresenta a *barra de ferramentas*, cujo detalhes de cada ferramenta são descritos a seguir:

- **(a) - Desfazer:** Desfaz a última ação realizada na visão. *Mais detalhes.*
- **(b) - Refazer:** Refaz a última ação reallizada na visão. *Mais detalhes.*
- **(c) - Recortar:** Copia a entidade selecionada para a **área de transferência** e **remove** ela da visão. *Mais detalhes.*
- **(d) - Copiar:** Copia a entidade selecionada para a **área de transferência**. *Mais detalhes.*
- **(e) - Colar:** Adiciona a entidade copiada para *área de transferência* como filha da entidade selecionada. *Mais detalhes.*
- **(f) - Aumentar o \*Zoom\*:** Aumenta o *Zoom* na visão. *Mais detalhes.*
- **(g) - Diminuir o \*Zoom\*:** Diminue o *Zoom* na visão. *Mais detalhes.*
- **(h) - Tirar uma Foto:** Exporta o projeto como imagem (.png). *Mais detalhes.*
- **(i) - Ferramenta de Ponteiro:** Altera o modo de edição da visão para “ponteiro”. *Mais detalhes.*
- **(j) - Ferramenta de \*Link\*:** Altera o modo de edição da visão para “link”. *Mais detalhes.*
- **(k) - Adicionar Mídia:** Adiciona uma entidade do tipo mídia como filha da entidade selecionada. *Mais detalhes.*
- **(l) - Adicionar Contexto:** Adiciona uma entidade do tipo context como filha da entidade selecionada. *Mais detalhes.*
- **(m) - Adicionar \*Switch\*:** Adiciona uma entidade do tipo *switch* como filha da entidade selecionada. *Mais detalhes.*
- **(n) - Adicionar \*Body\*:** Adiciona um *body* na visão. *Mais detalhes.*
- **(o) - Adicionar Área:** Adiciona uma entidade do tipo área como filha da entidade selecionada. *Mais detalhes.*
- **(p) - Adicionar Propriedade:** Adiciona uma entidade do tipo propriedade como filha da entidade selecionada. *Mais detalhes.*
- **(q) - Adicionar Porta:** Adiciona uma entidade do tipo porta como filha da entidade selecionada. *Mais detalhes.*
- **(r) - Adicionar Porta de \*Switch\*:** Adiciona uma entidade do tipo porta de *switch* como filha da entidade selecionada. *Mais detalhes.*

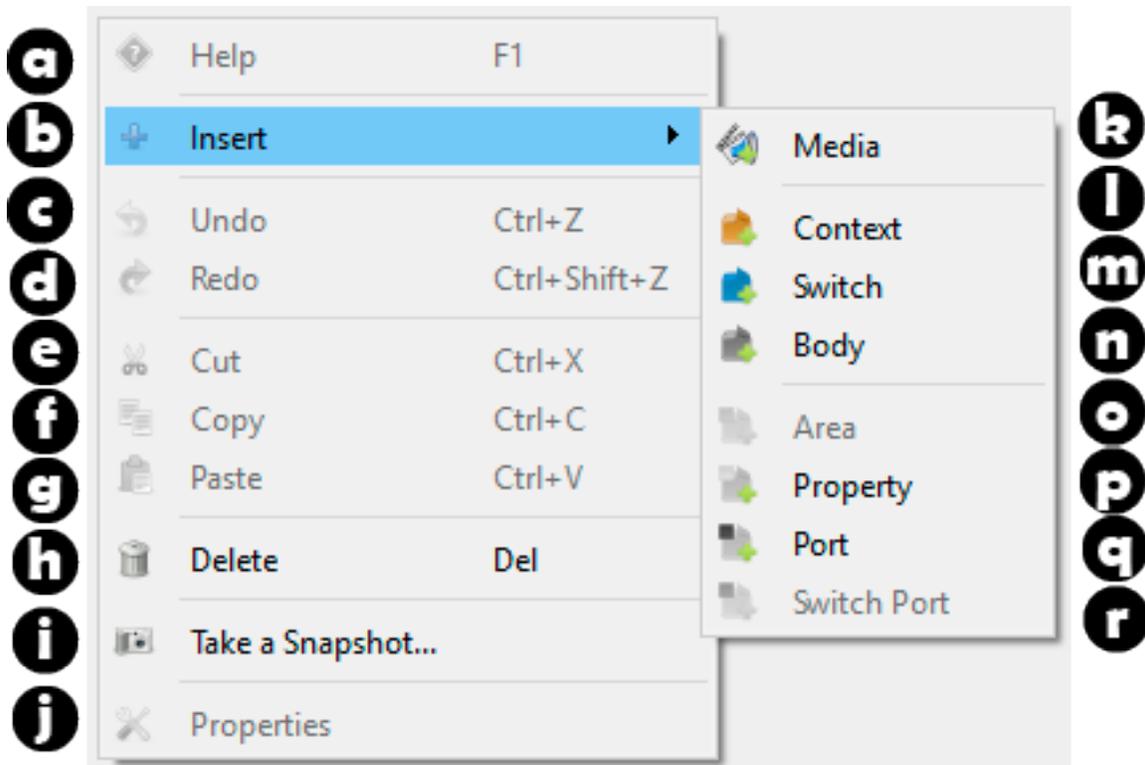
## Menus

### Menu Contextual

O menu contextual é exibido sempre que o usuário clica com o botão direito do *mouse* em qualquer entidade ou nas áreas vazias (em branco) da visão.

A figura acima apresenta o *menu contextual*, cujo detalhes são descritos abaixo:

- **(a) - Exibir Ajuda:** Exibe o diálogo de ajuda. **Desabilitado nesta versão.**



- (b) - **Insert:** Exibe o submenu de adção de entidades.
- (c) - **Desfazer:** Desfaz a última ação realizada na visão. *Mais detalhes.*
- (d) - **Refazer:** Refaz a última ação reallizada na visão. *Mais detalhes.*
- (e) - **Recortar:** Copia a entidade selecionada para a **área de trânsferencia** e **remove** ela da visão. *Mais detalhes.*
- (f) - **Copiar:** Copia a entidade selecionada para a **área de trânsferencia**. *Mais detalhes.*
- (g) - **Colar:** Adiciona a entidade copiada para *área de trânsferencia* como filha da entidade selecionada. *Mais detalhes.*
- (h) - **Deletar:** Remove a entidade selecionada da visão. *Mais detalhes.*
- (i) - **Tirar uma Foto:** Exporta o projeto como imagem (.png). *Mais detalhes.*
- (j) - **Exibir Propriedades:** Exibe o diálogo de propriedades. **Desabilitado nesta versão**
- (k) - **Adicionar Mídia:** Adiciona uma entidade do tipo mídia como filha da entidade selecionada. *Mais detalhes.*
- (l) - **Adicionar Contexto:** Adiciona uma entidade do tipo context como filha da entidade selecionada. *Mais detalhes.*
- (m) - **Adicionar \*Switch\*:** Adiciona uma entidade do tipo *switch* como filha da entidade selecionada. *Mais detalhes.*
- (n) - **Adicionar \*Body\*:** Adiciona um *body* na visão. *Mais detalhes.*
- (o) - **Adicionar Área:** Adiciona uma entidade do tipo área como filha da entidade selecionada. *Mais detalhes.*
- (p) - **Adicionar Propriedade:** Adiciona uma entidade do tipo propriedade como filha da entidade selecionada. *Mais detalhes.*
- (q) - **Adicionar Porta:** Adiciona uma entidade do tipo porta como filha da entidade selecionada. *Mais detalhes.*

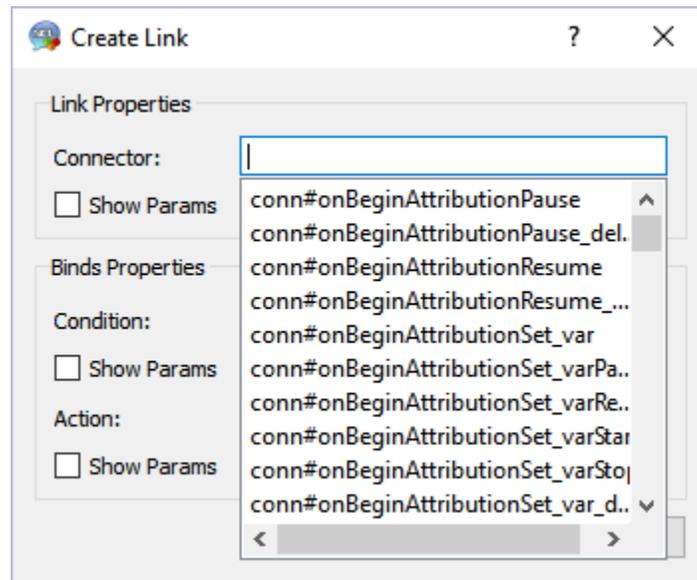
- **(r) - Adicionar Porta de \*Switch\*:** Adiciona uma entidade do tipo porta de *switch* como filha da entidade selecionada. *Mais detalhes.*

## Diálogos

### Diálogo de Link

O diálogo de *link* é utilizado para criação e edição de: *links*, *binds*, parâmetros de *link* e parâmetros de *binds*. O diálogo para criação de *link* é exibido quando um relacionamento entre uma entidade de origem e destino é especificado através da *ferramenta de criação de \*link\**.

Logo que exibido (figura abaixo), o diálogo de *link* apresenta uma lista de conectores para que o usuário possa escolher. A lista de conectores é carregada dinamicamente a partir das especificações da aplicação em desenvolvimento.



Assim que o usuário começar a preencher a caixa de texto “*connector*” (figura abaixo), a lista de conectores é filtrada para facilitar a seleção do conector apropriado.

Uma vez selecionado um dos conectores na lista, o usuário pode escolher qual o tipo de condição e ação na seção de *binds* do diálogo (figura abaixo). Os tipos de condições e ações disponíveis são carregados a partir do conector selecionado.

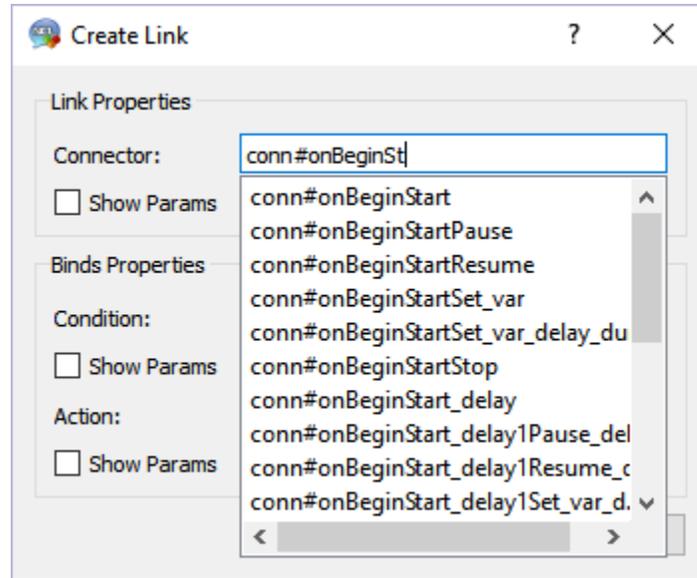
Ainda na criação do *link*, quando o usuário habilita a exibição de parâmetros (“*show params*”), é possível especificar os valores dos parâmetros de *link*, parâmetros do *bind* de condição e parâmetros do *bind* de ação (figura abaixo) que devem ser considerados para criação do relacionamento.

Para editar os parâmetros de um *link*, o usuário deve clicar duas vezes na entidade do tipo *link* na visão. Esta ação resultana exibição do diálogo abaixo.

De modo semelhante, para editar um *bind*, o usuário deve clicar duas vezes na entidade do tipo *bind*, resultando na exibição do diálogo abaixo.

### Referência

É possível especificar uma referência através da ação de *colar*. Assim que a ação de colar é realizada o diálogo abaixo é exibido, solicitando que o autor escolha entre copiar ou criar uma referência da entidade copiada para *área*



de transferência.

Quando uma entidade é uma referência, ela fica levemente transparente (figura abaixo) e herda todas as interfaces da entidade referenciada. Não é possível referenciar uma entidade que já referencia uma outra. Nesta versão ainda não é possível fazer referência para entidades do tipo: contexto, *switch* e *body*.

## Atalhos

Abaixo segue a lista de atalhos da Visão Estrutural:

- “Ctrl”+“z” - **Desfazer**: Desfaz a última ação realizada na visão. *Mais detalhes*.
- “Ctrl”+“Shift”+“z” - **Refazer**: Refaz a última ação realizada na visão. *Mais detalhes*.
- “Ctrl”+“x” - **Recortar**: Copia a entidade selecionada para a **área de transferência** e **remove** ela da visão. *Mais detalhes*.
- “Ctrl”+“c” - **Copiar**: Copia a entidade selecionada para a **área de transferência**. *Mais detalhes*.
- “Ctrl”+“v” - **Colar**: Adiciona a entidade da *área de transferência* como filha da entidade selecionada. *Mais detalhes*.
- “Del” - **Deletar**: Remove a entidade selecionada da visão. *Mais detalhes*.
- \*Segurar\* “Ctrl”+\*Drag-n-drop\* - **Mover**: Remove a entidade selecionada da visão. *Mais detalhes*.
- “Ctrl”+“+” ou “Ctrl”+\*Roda do mouse para cima\* - **Aumentar o \*zoom\***: Aumenta o *zoom* na visão. *Mais detalhes*.
- “Ctrl”+“-” ou “Ctrl”+\*Roda do mouse para baixo\* - **Diminuir o \*zoom\***: Diminui o *zoom* na visão. *Mais detalhes*.
- “1” - **Habilitar Ferramenta de Ponteiro**: Altera o modo de edição da visão para “ponteiro”. *Mais detalhes*.
- “2” ou \*Segurar\* “Shift” - **Habilitar Ferramenta de Link**: Altera o modo de edição da visão para “link”. *Mais detalhes*.
- “3” - **Adicionar Mídia**: Adiciona uma entidade do tipo mídia como filha da entidade selecionada. *Mais detalhes*.

**Create Link** ? X

**Link Properties**

Connector:

Show Params

Name	Value
var	

**Binds Properties**

Condition:

Show Params

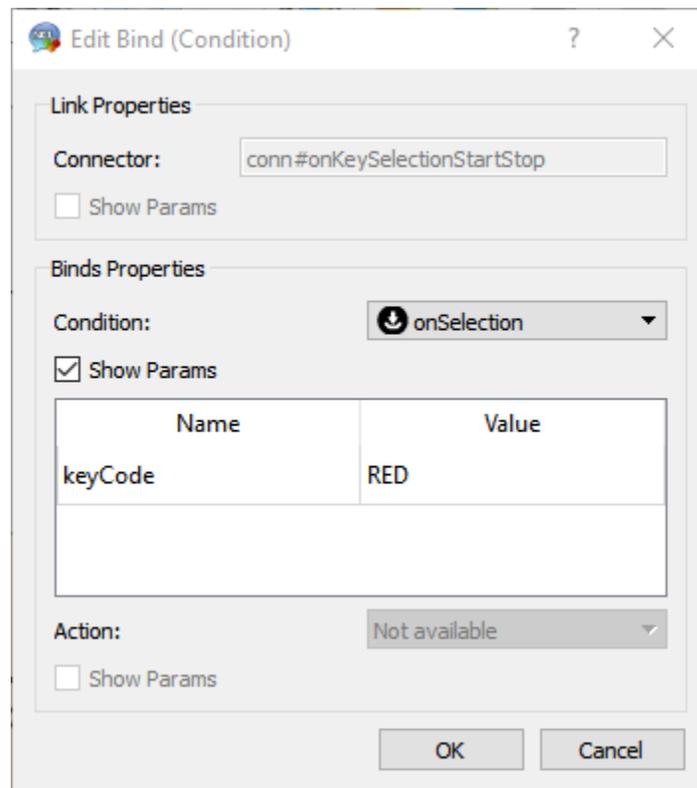
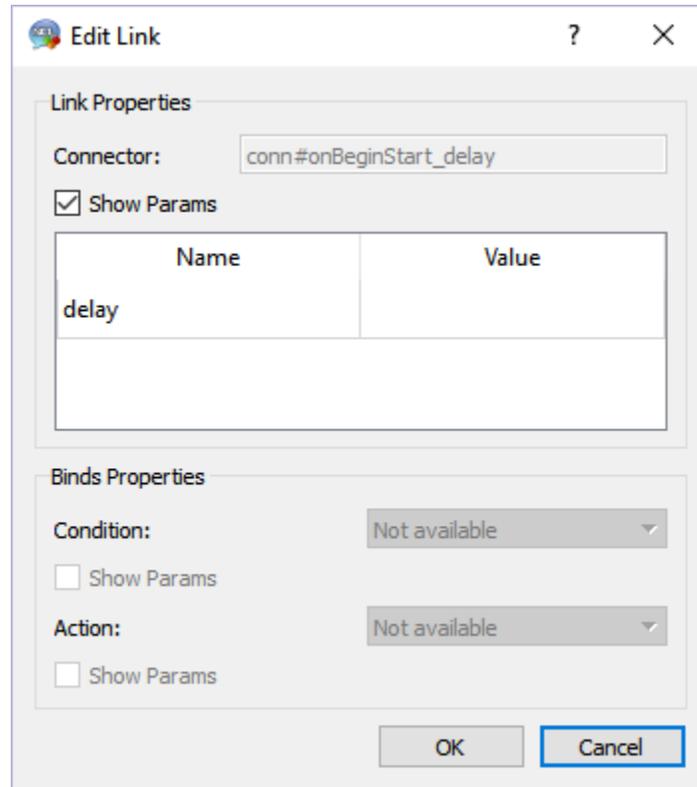
Name	Value
var	

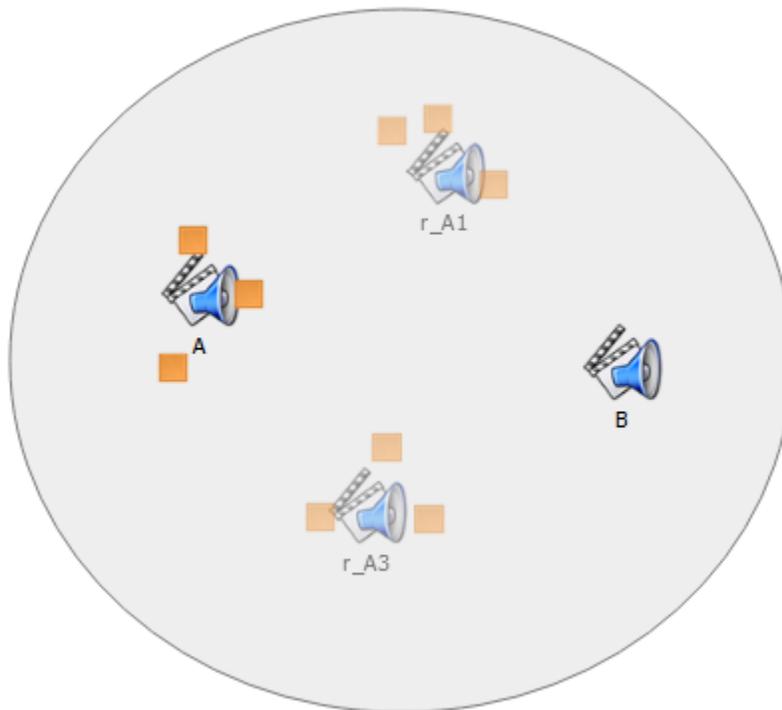
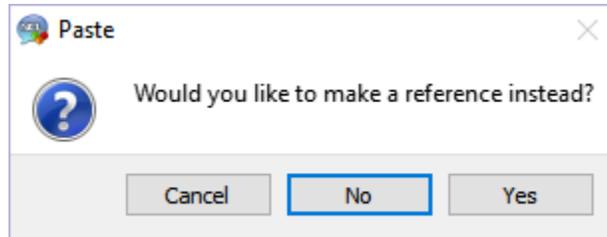
Action:

Show Params

Name	Value
var	

OK Cancel





- “4” - **Adicionar Contexto**: Adiciona uma entidade do tipo `context` como filha da entidade selecionada. *Mais detalhes.*
- “5” - **Adicionar \*Switch\***: Adiciona uma entidade do tipo `switch` como filha da entidade selecionada. *Mais detalhes.*
- “6” - **Adicionar \*Body\***: Adiciona um `body` na visão. *Mais detalhes.*
- “7” - **Adicionar Área**: Adiciona uma entidade do tipo `área` como filha da entidade selecionada. *Mais detalhes.*
- “8” - **Adicionar Propriedade**: Adiciona uma entidade do tipo `propriedade` como filha da entidade selecionada. *Mais detalhes.*
- “9” - **Adicionar Porta**: Adiciona uma entidade do tipo `porta` como filha da entidade selecionada. *Mais detalhes.*
- “0” - **Adicionar Porta de \*Switch\***: Adiciona uma entidade do tipo `porta de switch` como filha da entidade selecionada. *Mais detalhes.*
- “Ctrl”+“m” - **Habilitar/Desabilitar Minimapa**: Habilita/Desabilita o minimapa. *Mais detalhes.*

### Outros detalhes

Todo plugins do NCL Composer permite sincronizar (ou resincronizar) o estado atual da visão com o modelo do núcleo da ferramenta. Esta tarefa é realiza através de um botão na aba de cada visão. No caso da Visão Estrutural, sempre que a visão é resincronizada com o modelo do núcleo da ferramenta, todo o histórico de ações é perdido, de modo que não é possível desfazer ou refazer uma ação.

### Visão de Leiaute

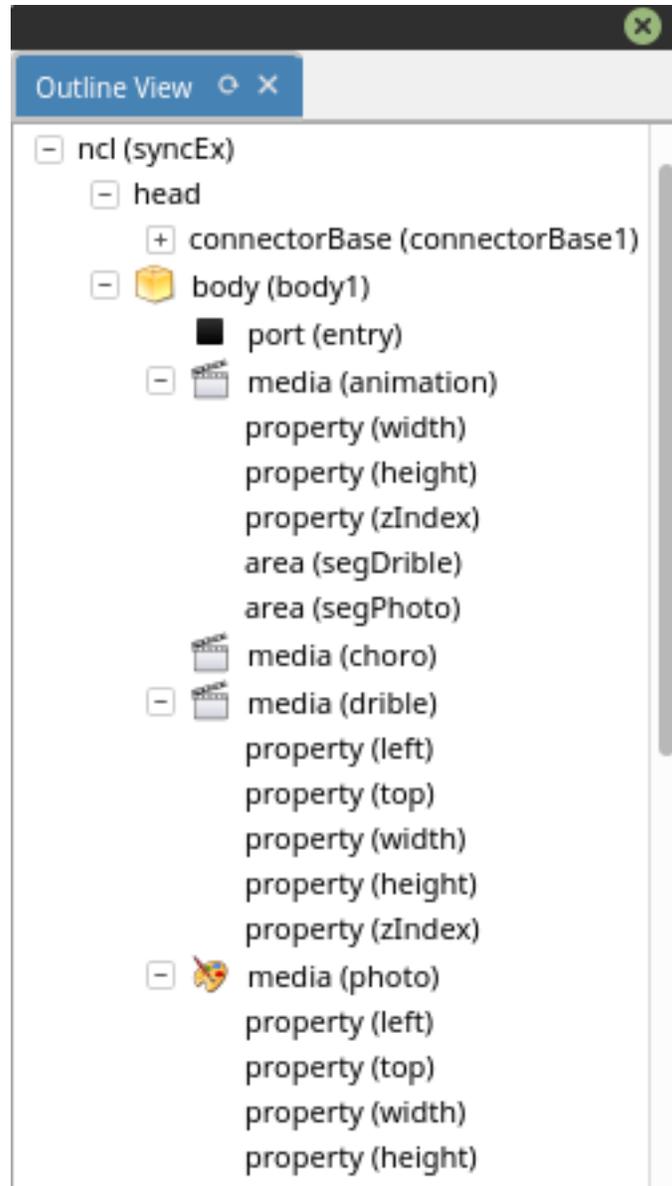
A Visão de Leiaute permite que os autores de aplicações NCL interajam visualmente com as regiões onde os objetos de mídia serão inicialmente apresentados. Nesta visão, os autores podem criar e manipular as regiões e a hierarquia de regiões do documento.



Ação	Descrição	
Criar uma região	Cria uma nova região, filha região que está selecionada.	Esta ação pode ser executada através da:
Remover uma região	Remove a região que está atualmente selecionada e seus filhos.	Esta ação pode ser executada através da: <i>barra de ferramentas</i>
Copiar região	Copia a região selecionada para a <b>área de transferência</b> . Esta ação é ela da visão. Esta ação é ignorada caso nenhuma entidade esteja selecionada.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> ( <i>#barra-de-ferramentas</i> ), <i>menu contextual</i> e <i>atalho</i> .
Colar região	Cola a região que está na <b>área de transferência</b> como filha da região selecionada.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> ( <i>#barra-de-ferramentas</i> ), <i>menu contextual</i> e <i>atalho</i> .
Esconder região	Esconde a região atualmente selecionada da visão de leiaute. A região continua existindo no modelo (e pode ser mostrada depois) apenas não é visualizada.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> ( <i>#barra-de-ferramentas</i> ), <i>menu contextual</i> e <i>atalho</i> .
Mostrar/esconder grade	Copiar a entidade selecionada para a <b>área de transferência</b> . Esta ação é ignorada caso nenhuma entidade tenha sido selecionada e se a entidade selecionada é do tipo <i>body</i> .	Esta ação pode ser executada através da: <i>barra de ferramentas</i> ( <i>#barra-de-ferramentas</i> ), <i>menu contextual</i> e <i>atalho</i> .
Tirar um instantâneo	Permite salvar uma imagem com a disposição atual das regiões da visão de leiaute.	Esta ação pode ser executada através da: <i>barra de ferramentas</i> ( <i>#barra-de-ferramentas</i> ), <i>menu contextual</i> e <i>atalho</i> .

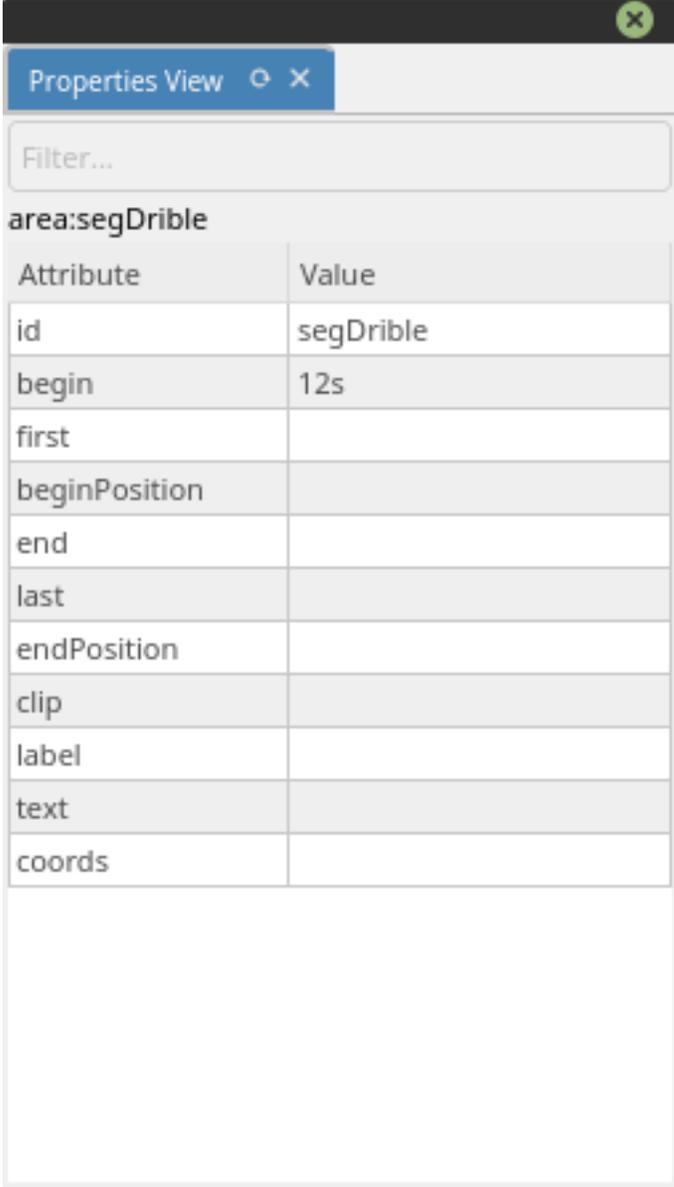
## Visão de Outline

A Visão de Outline apresenta a estrutura do documento NCL como uma árvore, permitindo aos autores navegarem no documento.



## Visão de Propriedades

A Visão de Propriedades apresenta o conjunto de propriedades da entidade selecionada. Essa entidade selecionada por ser, por exemplo, uma região na Visão de Leiaute, um objeto de mídia na Visão Estrutural etc.

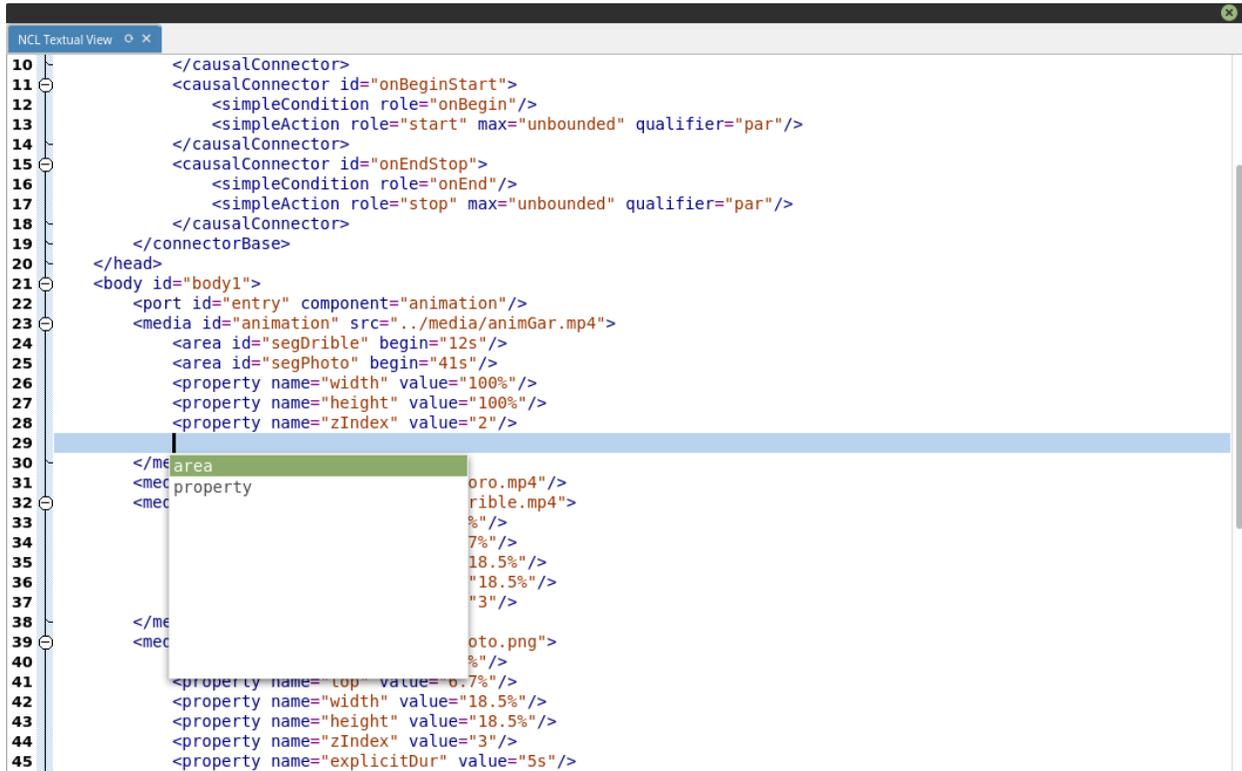


Attribute	Value
id	segDrible
begin	12s
first	
beginPosition	
end	
last	
endPosition	
clip	
label	
text	
coords	

## Visão Textual

A Visão Textual permite aos usuários interagirem diretamente com o código-fonte da aplicação NCL. Ela é um editor de texto avançado focado na linguagem NCL e traz funcionalidades como: coloração sintática, sugestão de código contextual e formatação de código.

A Visão Textual é recomendada especialmente para usuários avançados e que se sentem à vontade com a sintaxe XML.



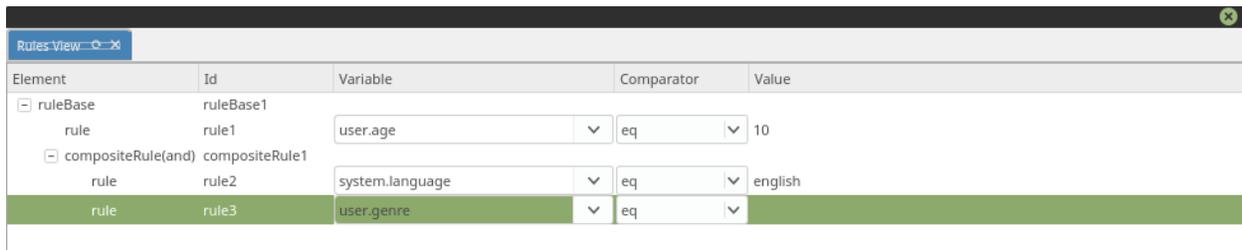
```

10 </causalConnector>
11 <causalConnector id="onBeginStart">
12 <simpleCondition role="onBegin"/>
13 <simpleAction role="start" max="unbounded" qualifier="par"/>
14 </causalConnector>
15 <causalConnector id="onEndStop">
16 <simpleCondition role="onEnd"/>
17 <simpleAction role="stop" max="unbounded" qualifier="par"/>
18 </causalConnector>
19 </connectorBase>
20 </head>
21 <body id="body1">
22 <port id="entry" component="animation"/>
23 <media id="animation" src="../media/animGar.mp4">
24 <area id="segDribble" begin="12s"/>
25 <area id="segPhoto" begin="41s"/>
26 <property name="width" value="100%"/>
27 <property name="height" value="100%"/>
28 <property name="zIndex" value="2"/>
29
30 </media>
31 <media id="segDribble" src="../media/segDribble.mp4">
32 <property name="width" value="100%"/>
33 <property name="height" value="100%"/>
34 <property name="zIndex" value="3"/>
35 <property name="explicitDur" value="5s"/>
36 </media>
37
38 </body>
39 </ncl>
40
41 <property name="top" value="0.7%"/>
42 <property name="width" value="18.5%"/>
43 <property name="height" value="18.5%"/>
44 <property name="zIndex" value="3"/>
45 <property name="explicitDur" value="5s"/>

```

## Visão de Regras

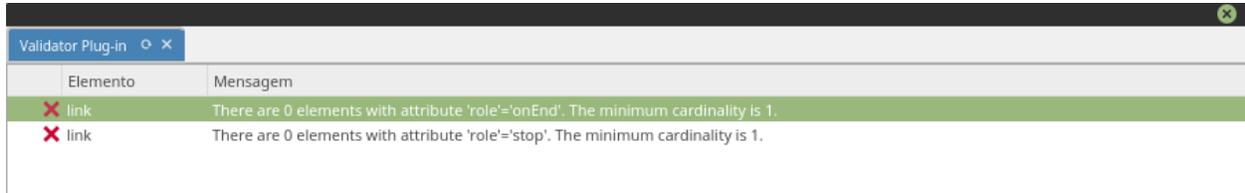
Permite a definição de uma base de regras, as quais podem ser utilizada em *switchs* para adaptação do conteúdo a ser apresentado.



Element	Id	Variable	Comparator	Value
ruleBase	ruleBase1			
rule	rule1	user.age	eq	10
compositeRule(and)	compositeRule1			
rule	rule2	system.language	eq	english
rule	rule3	user.genre	eq	

## Visão de Validação

A Visão de Validação é responsável por notificar erros e alertas no documento NCL. A figura a seguir apresenta um exemplo.



Elemento	Mensagem
link	There are 0 elements with attribute 'role'='onEnd'. The minimum cardinality is 1.
link	There are 0 elements with attribute 'role'='stop'. The minimum cardinality is 1.

## Visão de Execução

Possibilita ver o resultado da execução do seu projeto.



---

### Todo

Executar embedded or not embedded

---



## CHAPTER 10

---

Como mudar o tema do NCL Composer

---



# CHAPTER 11

---

## Indices and tables

---

- `genindex`
- `modindex`
- `search`



## Symbols

- help, -h
  - nclcomposer command line option, 9
- version, -v
  - nclcomposer command line option, 9

## N

- nclcomposer command line option
  - help, -h, 9
  - version, -v, 9