

Uma Ferramenta de Autoria para Interação Multimodal em Aplicações NCL 4.0

Pedro Alves Valentim
Laboratório MídiaCom
Universidade Federal Fluminense
Niterói, Rio de Janeiro, Brasil
pedroalvesvalentim@midiaom.uff.br

Débora Christina Muchaluat-Saade
Laboratório MídiaCom
Universidade Federal Fluminense
Niterói, Rio de Janeiro, Brasil
debora@midiaom.uff.br

ABSTRACT

Digital television has become an increasingly important platform for communication and entertainment, especially with the rise of smart devices and smartphones. Digital TVs now serve as an ideal environment for multimodal interaction experiences. However, programming these experiences presents significant challenges, particularly when dealing with specific programming languages like NCL (Nested Context Language). NCL 4.0, as part of the TV 3.0 Project, offers promising support for multi-user identification and interaction in digital TV applications, aiming to create a more personalized and participatory digital TV environment. Despite its potential, developing NCL 4.0 applications for multimodal interaction remains complex due to the need for understanding its syntax and semantics, as well as creating applications capable of recognizing and responding to voice commands and differentiating viewers. This paper presents an authoring tool designed to simplify the creation of multimodal interaction applications in NCL 4.0. The tool, implemented as a VSCode extension, automates the process of converting remote control interactions into multimodal links, thus reducing the authoring effort. By integrating voice, gesture, and facial expression interactions, the tool allows developers to focus more on enhancing user experience rather than technical implementation details. The paper details the functionalities and implementation of the tool, demonstrating how it can streamline the development process and improve the flexibility and usability of NCL 4.0 applications. Future work includes further testing and exploration of additional interaction modalities to enhance the tool's adaptability and inclusiveness.

KEYWORDS

Ferramenta de Autoria, Interação Multimodal, NCL, TV Digital, VSCode

1 INTRODUÇÃO

A televisão digital é uma plataforma cada vez mais importante para a comunicação e o lazer das pessoas. Com o aumento da disponibilidade de aparelhos inteligentes e *smartphones*, as TVs digitais tornaram-se um ambiente ideal para experiências de interação multimodal. No entanto, a programação dessas experiências apresenta desafios significativos, especialmente quando se trata de

linguagens de programação específicas, como NCL (*Nested Context Language*) [1].

Além de outras novidades, o NCL 4.0 [2, 5] é uma tecnologia promissora para o suporte à identificação e interação multiusuário em aplicações de TV digital. Como parte do Projeto TV 3.0, o NCL 4.0 foi selecionado como uma das novas tecnologias a serem incluídas na futura geração do SBTVD (Sistema Brasileiro de Televisão Digital). Ginga [8] é o *middleware* padrão do SBTVD e Recomendação da ITU para IPTV (*Internet Protocol Television*) [10]. Esse projeto tem como objetivo criar um ambiente de TV digital mais personalizado e participativo, permitindo, entre outras coisas, que os usuários interajam de forma multimodal com o conteúdo.

No entanto, a programação de aplicações NCL 4.0 para suporte à interação multimodal representa um desafio significativo. A linguagem requer uma compreensão de sua sintaxe e semântica, o que pode ser complexo para desenvolvedores não familiarizados com a tecnologia. Além disso, a criação de aplicações que sejam capazes de reconhecer e responder às solicitações de voz, assim como diferenciar os telespectadores e adaptar o conteúdo a eles, pode se mostrar um desafio a autores novos.

Neste contexto, o objetivo deste artigo é apresentar uma ferramenta de autoria para interação multimodal em aplicações NCL 4.0.

O restante do texto se estrutura da seguinte maneira: a Seção 2 descreve brevemente as novidades da linguagem NCL, como seus novos papéis de conectores; a Seção 3 descreve as funcionalidades da ferramenta; a Seção 4 descreve como essas funcionalidades foram implementadas na extensão; finalmente, a Seção 5 recapitula as contribuições do trabalho e sugere trabalhos futuros.

2 INTERAÇÃO MULTIMODAL EM NCL 4.0

O modelo NCM (*Nested Context Model*) [4, 9] estrutura a representação de documentos multimídia através de entidades como nós, conectores hipermídia [6] e elos. Cada nó é um conjunto de informações a ser apresentado, enquanto conectores e elos estabelecem as relações entre esses nós. A máquina de estados no NCM, que é fundamental para o modelo, opera com base em uma sincronização de eventos [7, 9]. Esta abordagem permite que documentos hipermídia incluam ações relacionadas a eventos temporais, espaciais e assíncronos, como interações do usuário. À medida que os eventos ocorrem, a máquina de estados altera seus estados conforme as condições associadas aos conectores e elos são atendidas, acionando as ações especificadas. A Figura 1 apresenta a máquina de estados dos eventos NCM.

O modelo NCM já inclui diversos tipos de eventos relacionados à interação do usuário, com atributos que descrevem características

In: VII Workshop Futuro da TV Digital Interativa (WTVDI 2024). Anais Estendidos do XXX Simpósio Brasileiro de Sistemas Multimídia e Web (WTVDI'2024). Juiz de Fora/MG, Brazil. Porto Alegre: Brazilian Computer Society, 2024.
© 2024 SBC – Brazilian Computing Society.
ISSN 2596-1683

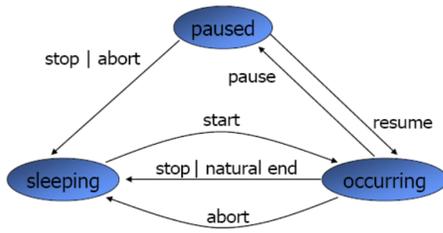


Figura 1: Máquina de estados dos eventos NCM.

específicas desses eventos. Por exemplo, o atributo “key” em um evento de seleção define a tecla pressionada. A linguagem NCL, em sua versão 4.0, apresenta novos tipos de eventos para ampliar os modos de interação. Entre os novos eventos estão “voiceRecognition”, “handPoseRecognition” e “faceRecognition” – para interações via voz, gestos e expressões faciais, respectivamente –, que utilizam o atributo “key” de maneira diferente. Cada evento possui um papel associado – por exemplo, o evento “voiceRecognition” é diretamente relacionado ao papel “onVoiceRecognition”. A listagem 1 apresenta um exemplo de conector e elo para interação por voz. No contexto dos eventos multimodais, o atributo “key” funciona como uma chave que fornece informações adicionais sobre o evento; por exemplo, no evento “handPoseRecognition”, um “key” igual a “OPEN” indica o gesto de mão aberta.

```

1 <connectorBase>
2   <causalConnector id="onVoiceRecognitionStart">
3     <connectorParam name="key"/>
4     <simpleCondition role="onVoiceRecognition" key=
5       "$key"/>
6     <simpleAction role="start" />
7   </causalConnector>
8 </connectorBase>
9 ...
10 <link xconnector="onVoiceRecognitionStart">
11   <bind role="onVoiceRecognition" component="
12     botanicalGardenImage">
13     <bindParam name="key" value="play"/>
14   </bind>
15   <bind role="start" component="botanicalGardenVideo"
16     />
17 </link>

```

Listing 1: Exemplo de conector e elo para interação por voz

É possível configurar múltiplas condições num mesmo conector, permitindo, por exemplo, combinar diferentes modos de interação em um mesmo conector, com condição composta por operador booleano “OR”. A listagem 2 apresenta um exemplo de conector com condição composta.

```

1 <causalConnector id="
2   onKeySelectionOrVoiceRecognitionStart">
3   <connectorParam name="key" />
4   <connectorParam name="key_voice" />
5   <compoundCondition operator="or">

```

```

5   <simpleCondition role="onSelection" key="$key"
6     />
7   <simpleCondition role="onVoiceRecognition"
8     key="$key_voice" />
9 </compoundCondition>
10 <simpleAction role="start" max="unbounded" />
11 </causalConnector>

```

Listing 2: Exemplo de conector que pode ser ativado tanto por controle remoto quanto por voz.

3 PROPOSTA DE FERRAMENTA DE AUTORIA

A ferramenta de autoria para integração multimodal apresentada neste trabalho visa facilitar a criação de aplicativos NCL 4.0 que combina diferentes modos de interação, incluindo voz, gesto e expressão facial. Para isso, desenvolvemos uma extensão VSCode que automatiza o processo de conversão de elos de interação via controle remoto e outros modos de interação em elos multimodais.

A ferramenta utiliza o recurso *Code Actions* do VSCode para indicar ao usuário que é possível adicionar outros modos de interação a elos presentes no código. Este recurso tem a aparência de uma lâmpada no começo da linha do editor. Ao clicar na lâmpada, é aberto um pequeno menu com os modos de interação que podem ser adicionados ao elo. A Figura 2 é um recorte de captura de tela que apresenta tal recurso.

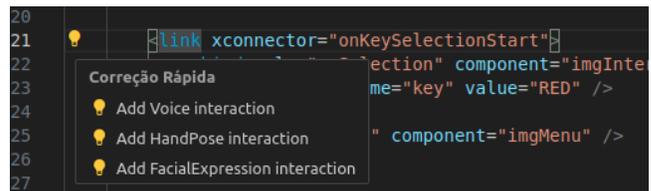


Figura 2: Aparência do elemento de *Code Action* para adição de interação multimodal a um elo de interação via controle remoto.

Após a seleção, abre-se uma caixa de entrada de texto para que o usuário insira o valor esperado “key” para o novo papel do elo. Feita esta atribuição, a extensão verifica se há na base de conectores um conector correspondente ao novo elo. Caso não haja, ele é adicionado. Finalmente, é reescrito o elo, mantendo os modos originais de interação e o novo, recém-escolhido. Com essa ferramenta, o autor de aplicações NCL não precisa mais se preocupar com a implementação manual da lógica de interação multimodal, reduzindo consideravelmente o esforço de autoria.

Com essa ferramenta, é possível criar aplicativos NCL 4.0 que integrem diferentes modos de interação, tornando-os mais flexíveis e fáceis de usar. Além disso, a redução do esforço de autoria permite que os desenvolvedores se concentrem em melhorar a experiência do usuário, ao invés de se preocupar com aspectos técnicos de sintaxe e dependências.

4 IMPLEMENTAÇÃO NO VSCODE

A extensão é escrita em TypeScript [3] e utiliza a API do VSCode para registrar comandos, fornecer ações de código e manipular o

conteúdo dos documentos abertos no editor. A ativação da extensão ocorre por meio da função “activate”, que registra os comandos e ações a serem disponibilizados aos usuários.

O comando principal da extensão, “editor.writeLink”, é registrado durante a ativação da extensão. Esse comando é responsável por lidar com a escrita de elos multimodais no código NCL. Quando acionado, abre uma caixa de entrada de texto para que o usuário que insira o valor “key” para a nova interação. Após o usuário fornecer o valor, o comando constrói uma nova *tag* de elo que incorpora o novo modo de interação e verifica se o conector correspondente já existe no documento. Se o conector não existir, ele é criado e inserido na base de conectores.

A extensão inclui uma função que verifica se um conector específico já está presente no documento. Isso é feito ao comparar o conteúdo do documento com uma expressão regular que busca pelo identificador do conector. Se o conector necessário não estiver presente, outra função é responsável por compor o conector e inseri-lo na base de conectores. Esta composição inclui a construção de uma *tag* de conector causal que integra a nova ação simples correspondente ao tipo de interação adicionada.

A extensão utiliza o recurso *Code Actions* do VSCode para fornecer sugestões de interação ao usuário, indicando que é possível adicionar modos de interação a elos presentes no código. Um provedor de ações de código verifica se a seleção atual no editor corresponde a um elo de interação. Se corresponder, ele cria ações de código que permitem ao usuário adicionar interações de voz, gestos ou expressão facial ao elo selecionado.

A extensão automatiza o processo de adição de interações multimodais, de modo que o usuário pode simplesmente selecionar e definir novos modos de interação para os elos presentes no código. Ao fazer isso, a extensão verifica e atualiza automaticamente os conectores necessários, assegurando coesão sintática no documento.

A Figura 3 apresenta um diagrama de atividades para o sistema. A versão inicial da extensão está disponível em: ncl-multimodal-qol-0.0.1.vsix

5 CONCLUSÃO

Neste trabalho, desenvolvemos uma ferramenta de autoria voltada para a criação de aplicações NCL 4.0 com interação multimodal, visando facilitar a programação e melhorar a experiência do usuário. Nossa ferramenta permite a integração de diferentes modos de interação, como voz, gestos e expressões faciais, automatizando a conversão de elos de interação e simplificando o processo para os autores.

As principais contribuições deste trabalho incluem a implementação de uma extensão para o VSCode que utiliza o recurso *Code Actions* para sugerir e adicionar novos modos de interação de maneira intuitiva. Esta abordagem reduz significativamente o esforço necessário para programar aplicações multimodais, permitindo que os autores se concentrem em aspectos mais criativos e na melhoria da experiência do usuário.

Além disso, discutimos a estrutura do modelo NCM e como a linguagem NCL 4.0 foi ampliada para suportar novos tipos de eventos multimodais. A ferramenta implementada verifica a presença de conectores necessários e, caso não existam, os cria automaticamente, garantindo a coesão sintática do documento NCL.

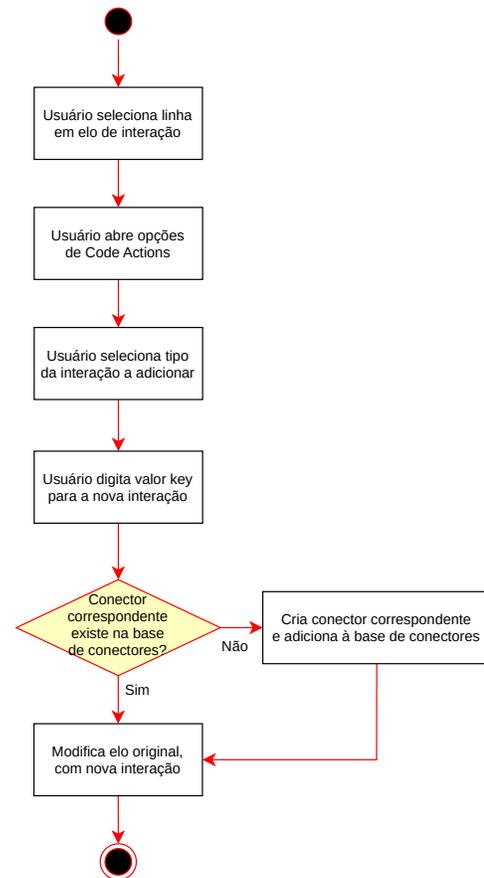


Figura 3: Diagrama de atividades da extensão.

Para trabalhos futuros, sugerimos a realização de testes adicionais da ferramenta em diferentes cenários e com um conjunto maior de dados para validar sua robustez e versatilidade. Outra direção promissora é a exploração de modalidades de interação adicionais, como o uso do evento “eyeGaze” (fixação do olhar), que pode proporcionar uma experiência de usuário ainda mais inclusiva. A implementação de novos recursos que permitam uma maior personalização e adaptabilidade da ferramenta às necessidades específicas dos usuários também é um caminho a ser explorado.

Acreditamos que esta ferramenta pode trazer contribuições significativas para a comunidade de desenvolvimento de aplicações NCL, promovendo uma experiência de TV digital mais interativa e personalizada.

AGRADECIMENTOS

Os autores agradecem o apoio recebido da FAPERJ, CAPES, CAPES PRINT, CNPq, MCOM e RNP.

REFERÊNCIAS

- [1] ABNT. 2021. *Digital terrestrial television - Data coding and transmission specification for digital broadcasting - Part 2: Ginga-NCL for fixed and mobile receivers - XML application language for application coding (In portuguese)*.

- [2] Fábio Barreto, Raphael S de Abreu, Eyre Brasil B Montevecchi, Marina IP Josué, Pedro Alves Valentim, and Débora C Muchaluaat-Saade. 2020. Extending Ginga-NCL to Specify Multimodal Interactions With Multiple Users (in Portuguese). In *Proceedings of the Brazilian Symposium on Multimedia and the Web*. 281–288.
- [3] Gavin Bierman, Martin Abadi, and Mads Torgersen. 2014. Understanding typescript. In *ECOOP 2014—Object-Oriented Programming: 28th European Conference, Uppsala, Sweden, July 28–August 1, 2014. Proceedings 28*. Springer, 257–281.
- [4] Marco A Casanova, Luiz Tucherman, Maria Julia D Lima, José L Rangel Netto, Noemi Rodriguez, and Luiz FG Soares. 1991. The nested context model for hyperdocuments. In *Proceedings of the third annual ACM conference on Hypertext*. 193–201.
- [5] Eyre Brasil B Montevecchi, Marina IP Josué, Fábio Barreto, Raphael S de Abreu, and Débora C Muchaluaat-Saade. 2020. Providing Eye Gaze Interaction for Ginga-NCL Applications (in Portuguese). In *Proceedings of the Brazilian Symposium on Multimedia and the Web*. 297–303.
- [6] D. C. Muchaluaat-Saade and L. F. G. Soares. 2002. XConnector & XTemplate: Improving the Expressiveness and Reuse in Web Authoring Languages. *The New Review of Hypermedia and Multimedia Journal* 8, 1 (2002), 139–169.
- [7] Luiz Fernando Gomes Soares and Rogério Ferreira Rodrigues. 2005. Nested context model 3.0: Part 1—ncm core. *Technical Report of the Informatics Department of PUC-Rio* 12 (2005).
- [8] Luiz Fernando Gomes Soares, Rogério Ferreira Rodrigues, and Márcio Ferreira Moreno. 2007. Ginga-NCL: the declarative environment of the Brazilian digital TV system. *Journal of the Brazilian Computer Society* 12, 4 (2007), 37–46.
- [9] Luiz Fernando G Soares, Rogério F Rodrigues, and Débora C Muchaluaat-Saade. 2000. Modeling, authoring and formatting hypermedia documents in the Hyper-Prop system. *Multimedia systems* 8, 2 (2000), 118–134.
- [10] International Telecommunication Union. 2009. ITU-R H761: Nested Context Language (NCL) and Ginga-NCL for IPTV Services. (April 2009).