

# A Proposal for Supporting Sensory Effect Rendering in Ginga-NCL

Renato de Oliveira Rodrigues  
GPMM - CEFET/RJ  
renato.rodrigues@eic.cefet-rj.br

Marina I. P. Josué  
Laboratório MidiaCom - UFF  
marinaivanov@midiacom.uff.br

Raphael S. Abreu  
Laboratório MidiaCom - UFF  
raphael.abreu@midiacom.uff.br

Glauco F. Amorim  
GPMM - CEFET/RJ  
gamorim@eic.cefet-rj.br

Débora C. Muchaluat-Saade  
Laboratório MidiaCom - UFF  
debora@midiacom.uff.br

Joel A. F. dos Santos  
GPMM - CEFET/RJ  
jsantos@eic.cefet-rj.br

## ABSTRACT

This contribution proposes an extension to middleware Ginga-NCL as a way to provide a declarative approach for the creation and execution of interactive mulsemimedia applications. It enables the author to create mulsemimedia applications in NCL, defining sensory effects as nodes. Therefore, one is able to reuse all the language support for content synchronization, user interaction, context adaptation, etc.

## KEYWORDS

Multisensory content, Mulsemimedia, NCL, Ginga-NCL

## 1 Background

A apresentação de conteúdo audiovisual em conjunto com efeitos que geram estímulos a outros sentidos humanos tem o potencial de aprimorar a qualidade de experiência do usuário durante o consumo de conteúdo multimídia. As aplicações são denominadas aplicações *mulsemídia* (do inglês *mulsemimedia* - Multiple Sensorial Media), e possibilitam a apresentação de efeitos sensoriais (tais como vento, aroma, temperatura) sincronizados com objetos de mídia tradicional. A renderização de tais efeitos sensoriais é realizada por dispositivos atuadores (como ventiladores, dispersores de aroma, lâmpadas, etc).

Além da apresentação sincronizada de objetos de mídia com efeitos sensoriais, aplicações *mulsemídia* podem também dar suporte à interatividade. Isto é, além do usuário consumir o conteúdo *mulsemídia*, ele pode interagir com a informação que está sendo exibida, tendo escolha e controle sobre a mesma. Alguns exemplos desse tipo de aplicação interativa são jogos, simuladores e exibições de arte. Consequentemente, a incorporação de efeitos sensoriais em aplicações multimídia

interativas amplia a sensação de imersão do usuário, levando a uma experiência mais rica pois estimula vários sentidos.

Neste contexto, esta proposta propõe a extensão do *middleware* Ginga-NCL, de modo a dar suporte à criação e execução de aplicações NCL com efeitos sensoriais. A extensão aqui proposta consiste em permitir a especificação de efeitos sensoriais como elementos *<effect>*, por meio da criação de um *player* de efeitos sensoriais, bem como a extensão do *parser* do *middleware* para suporte a novos atributos do elemento *<region>* de forma a permitir a especificação de regiões para efeitos utilizando coordenadas esféricas ou por eixos conforme o padrão MPEG-V. Dessa forma, todas as construções do *middleware* Ginga-NCL para tratamento de eventos, comunicação e integração com *players* são aplicadas também aos efeitos sensoriais.

## 2 Proposal

Conforme apresentado na Seção 5.1 da ABNT NBR15606-2, a arquitetura do Ginga-NCL é composta por módulos responsáveis pelo processamento de documentos NCL. Um módulo importante dentro da arquitetura do Ginga-NCL é a máquina de apresentação do conteúdo NCL, também chamada de formatador NCL. O formatador é responsável por controlar as relações de sincronização espaço-temporal entre os objetos de mídia, e pela ativação dos *players* de mídia. Esta proposta propõe que o formatador faça a ativação de efeitos sensorial a partir de um *player*. O *player* de efeitos sensoriais, diferente dos *players* de mídia, deve ser capaz de lidar com dispositivos sensoriais heterogêneos, que podem executar diferentes sistemas operacionais e estarem conectados a uma variedade de protocolos de comunicação.

O *player* de efeitos sensoriais aborda a questão da integração com dispositivos heterogêneos utilizando um modelo de dados compatível com o padrão MPEG-V. A utilização de um padrão livre, possibilita a representação das informações na aplicação de maneira interoperável entre diversos tipos de sistemas. Desse modo, o *player* de efeitos sensoriais proposto processa as informações contidas nas propriedades do nó de efeito sensorial e aquelas relativas ao evento realizado sobre o nó de efeito sensorial,

a fim de obter os metadados dos efeitos sensoriais e gerar uma descrição SEM de acordo com a linguagem SEDL.

Para gerenciar a conexão com diversos protocolos de comunicação, o *player* mantém um protocolo, definido pela interface *Protocol*, que especifica alguns métodos básicos de protocolos de comunicação, conforme apresentado no diagrama de classes da Figura 1. Desta forma, é possível estabelecer uma estrutura comum para a criação de classes que encapsulam a interface de bibliotecas que implementam os protocolos de comunicação, e torná-las acessíveis ao *player*, facilitando a integração de novos protocolos.

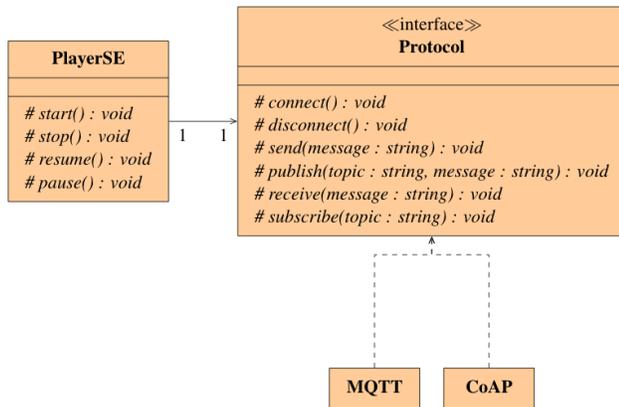


Figure 1: Diagrama de classes do *player* de efeitos sensoriais com os protocolos MQTT e CoAP

Para efeitos sensoriais, a transição no estado de apresentação é realizada a partir do envio de mensagens contendo a descrição dos metadados de efeitos sensoriais (SEM) para os dispositivos onde os efeitos são renderizados. Durante a apresentação de uma aplicação mulsemídia, quando ocorrem as transições *start* ou *resume* no evento de apresentação de um efeito sensorial, o *player* envia uma descrição SEM com a propriedade *activate* igual a *true*. Caso ocorra uma transição *stop* ou *pause* no evento de apresentação de um efeito sensorial, é enviada uma descrição SEM com a propriedade *activate* com o valor *false* para o atuador.

Com o objetivo de tentar reduzir ou até mesmo evitar atrasos durante a apresentação da aplicação mulsemídia, o *player* inclui o tratamento para o evento de preparação de NCL. Durante a preparação, é estabelecida a conexão do formatador com os dispositivos sensores e atuadores.

Com o *player* proposto, o autor de uma aplicação é capaz de especificar um efeito sensorial por meio de um elemento *<effect>*. Este elemento possui um atributo *type*, que define o tipo de efeito sensorial, de acordo com os tipos de efeitos sensoriais especificados no padrão MPEG-V. O *player* dá suporte também ao uso dos atributos *instance*, *refer* e *descriptor* em conjunto com elementos *effect*.

Além do *player*, esta proposta estende o *parser* do *middleware* Ginga-NCL para suporte ao elemento *effect*, bem como a especificação do posicionamento deste efeito no espaço tridimensional do ambiente de execução da aplicação. Assim,

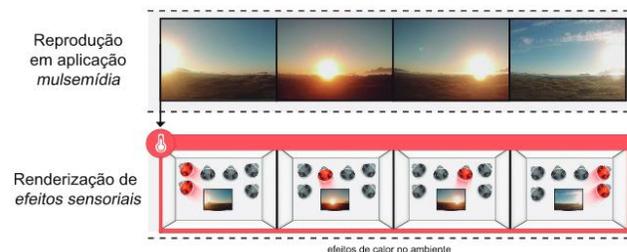
possibilita-se a utilização dos atributos *location*, *polar* e *azimuthal* ao elemento *<region>*. Quando o posicionamento é definido por coordenadas esféricas, os valores dos atributos *width* e *height* devem ser definidos usando o sufixo *deg*, para definir o tamanho da área onde o efeito será ativado e os atributos *polar* e *azimuthal* para definir a posição deste efeito. Nos casos em que o posicionamento é definido por eixos, é utilizado o atributo *location* contendo os valores de cada eixo concatenados separados por dois-pontos, por exemplo, "left:top:front" indica a posição frontal, superior à esquerda (eixos x, y e z, respectivamente).

A linguagem NCL (e consequentemente o *middleware* Ginga-NCL) suportam a definição de animações, definido eventos de atribuição sobre as propriedades de posicionamento ou tamanho ao longo de um intervalo de tempo. O *player* proposto também dá suporte à especificação de animações usando os atributos *duration* (para a duração) e *by* (para o incremento) na atribuição de valor a propriedades de um efeito sensorial. A cada passagem do tempo da apresentação, é verificado se uma propriedade deve ser atualizada dado o valor do incremento. Quando o valor de uma propriedade é atualizado, duas mensagens contendo a descrição SEM são enviadas para os atuadores, uma contendo o comando para desativar os efeitos sensoriais na posição antiga, e outra para ativar o efeito na nova posição.

### 3 Use Case(s)

Esta seção apresenta um caso de uso de uma aplicação NCL com efeitos sensoriais, a fim de apresentar a extensão do *middleware* Ginga-NCL proposta. A aplicação NCL utiliza efeitos sensoriais sincronizados com conteúdo audiovisual, o evento de preparação aplicado a efeitos sensoriais e animação do posicionamento de um efeito sensorial. Neste caso de uso, consideramos a preparação de um efeito sensorial como a conexão a priori com os dispositivos atuadores responsáveis pela renderização do efeito.

A Figura 2 apresenta uma visão geral da aplicação. Ela é composta por um vídeo principal que apresenta o movimento do sol no céu ao longo do dia, sincronizado com um efeito de calor. A aplicação é iniciada com um vídeo de espera, indicando o carregamento, que dura o tempo da preparação do efeito sensorial. Após a preparação, o vídeo de carregamento é encerrado e o vídeo principal é iniciado juntamente com um efeito de calor. A posição do efeito de calor é alterada ao longo da aplicação de forma a simular o movimento do sol ao longo do dia.



**Figura 2 -Visão geral da aplicação NCL**

A Listagem 1 apresenta trechos do código NCL dessa aplicação.

```

<head>
...
<region id="rScreen" width="100%" height="100%" />
<region id="rHeat" azimuthal="30deg"
  polar="45deg" width="10deg" height="10deg" />
...
<descriptor id="dHeat" region="rHeat">
  <descriptorParam name="intensity" value="5.0" />
</descriptor>
<descriptor id="dLoading" region="rScreen" />
<descriptor id="dVideo" region="rScreen" />
...
<causalConnector id="onBeginStartPrep">
  <simpleCondition role="onBegin" />
  <simpleAction role="startPreparation" />
</causalConnector>
<causalConnector id="onEndPrepStopStart">
  <simpleCondition role="onEndPreparation" />
  <simpleAction role="stop" />
  <simpleAction role="start" />
</causalConnector>
<causalConnector id="onBeginSet">
  <connectorParam name="var" />
  <connectorParam name="duration" />
  <connectorParam name="by" />
  <simpleCondition role="onBegin" />
  <compoundAction operator="seq">
    <simpleAction role="set" value="$var"
      duration="$duration" by="$by" />
  </compoundAction>
</causalConnector>
...
</head>
<body>
  <port id="pinit" component="loading" />
  <media id="loading" src="loading.ogv"
    descriptor="dLoading" />
  <media id="video" src="sun.mp4"
    descriptor="dVideo" />
  <effect id="heat" type="HeatType"
    descriptor="dHeat" />
  <link xconnector="onBeginStartPrep">
    <bind role="onBegin" component="loading" />
    <bind role="startPreparation"
      component="heat" />
  </link>
  <link xconnector="onEndPrepStopStart">
    <bind role="onEndPreparation"

```

```

      component="heat" />
    <bind role="stop" component="loading" />
    <bind role="start" component="video" />
  </link>
  <link xconnector="onBeginStart">
    <bind role="onBegin" component="video" />
    <bind role="start" component="heat" />
  </link>
  <link xconnector="onBeginSet">
    <bind role="onBegin" component="heat" />
    <bind role="set" component="heat"
      interface="location">
      <bindParam name="var"
        value="30,135,10,10" />
      <bindParam name="duration" value="17s" />
      <bindParam name="by" value="30" />
    </bindParam>
  </link>
</body>

```

**Listagem 1 - Trecho de código NCL com efeitos sensoriais**

## ACKNOWLEDGMENTS

Os autores agradecem à CAPES, CNPq e FAPERJ pelo financiamento parcial do trabalho.