A Multimodal Presentation Markup Language MPML-VR for a 3D Virtual Space

Naoaki Okazaki **Sohei Aya** **Santi Saeyor** **Mitsuru Ishizuka**

Graduate School of Information Science and Technology
University of Tokyo
Hongo 7-3-1, Bunkyo-ku, Tokyo, 113-8656 Japan, +81-3-5841-8570 (FAX)
{okazaki, s-aya, santi, ishizuka}@miv.t.u-tokyo.ac.jp

Abstract

Toward effective information distribution through WWW and a new style of multimodal media, we are conducting researches on multimodal presentation using lifelike character agents. As part of our research, we have developed MPML (Multimodal Presentation Markup Language), which allows the users to write multimodal contents with ease. By extending MPML, we have designed MPML-VR (Multimodal Presentation Markup Language for Virtual Reality), which facilitates multimodal presentation in a 3D virtual space. For an agent system in VRML environment, we also created our original character, "Aya".

1 Introduction

The main goal of our research is to make a presentation at anytime and anywhere with lifelike character agents in place of human presenters. Although it is possible to produce such an attractive multimodal content, like computer games, with a hard work of professional creators, multimodal presentations have not become widely used yet. Considering that one of the triggers for the rapid growth of WWW contents is the easiness of writing HTML, we should provide a way in which anyone can create multimodal contents without difficulty.

In the prevailing style of presentation aiming to illuminate a concept, a human explains verbally with some materials. This style is very common (i.e. at a conference or lecture), but in some cases, for sale of real estate, explanation of new camera, direction, and educational material of structure, we find the difficulty of transforming a spatial sense from two-dimensional materials. In such situations, we should employ another style of presentation that offers three-dimensional space as it is. Changing the viewpoint of the three-dimensional presentation, a viewer percepts the depth of the space or objects. Yet, the effectiveness of 3D multimodal presentation is not known well.

With this motivation we have designed MPML-VR (Multimodal Presentation Markup Language for Virtual Reality), which facilitates multimodal presentation in 3D virtual space. One can create a dynamic presentation using multimodal lifelike agents with MPML-VR.

2 Related Work

We can see two types of research related to multimodal presentation. One aims at a task-dependent presentation system. For example, Jacob (Evers and Nijholt, 2000) instructs special tasks such as Tower of Hanoi in a VRML space by a character agent. Because this type of systems is not designed for general purpose of presentation, it often needs a certain amount of programming to create other contents.

Another aims at a task-independent presentation system and defines, in most cases, their description languages. There exist some markup languages designed for the multimodal presentation. Virtual Human Presenter (Noma et al., 2000) defines a tex-like description language, and archives multimodal presentation by a character. However, it does not support user interaction. APML - Affective Presentation Markup Language (Carolis et al., 2001) focuses on presenting personality and emotions in agents. The presentation is mainly performed by speech synthesis and facial expression, not coupled with the objects being presented.

VHML – Virtual Human Markup Language (Marriott et. al., 2001) allows interactive Talking Heads to be directed by text marked up in XML. The language is designed to accommodate various aspects of Human-Computer Interaction. As is the case with APML, it does not have a feature to integrate a background space (or targets of explanation) and an agent space into one presentation space.

MPML - Multimodal Presentation Markup Language (Tsutsui et al., 2000), which is the base of this research, has been developed to distribute multimodal contents through WWW. MPML is an XML-compliant description language, by which the users write each step of presentation in the middle level of abstraction. This means, for example, the user does not have to care about how to start a gesture or synchronizes the gesture in the low level (i.e., programming interface of a library). Although some versions of MPML work with 3D agent characters, the background is placed in 2D space or the same flat space as the agents.

```
<?xml version="1.0" encoding="Shift_JIS"?>
<?xml-stylesheet type="text/xsl" href="mpml-vr.xsl"?>

<mpml version="vr">
  <head>
    <title> Demo of MPML-VR </title>
    <agent id="Aya" profile-url="aya.xml" where="posStart" />
  </head>
  <body>
    <page id="p1" world-url="house.wrl">
      <scene id="s1">
        <events>
          <heard id="evExit1" agent="Aya" voice="exit" jump-to="s2" />
          <world-touch id="evExit2" type="Click" location="door" jump-to="s2" />
        </events>
        <seq>
          <set-viewpoint location="camCenter" transition="ON" />
          <listen agent="Aya" />
          <par>
            <play agent="Aya" act="nod" begin="2000" />
            <speak agent="Aya">This is a demo of MPML-VR.</speak>
          </par>
          <speak agent="Aya">Plase say exit or click the door to exit.</speak>
          <move agent="Aya" how="walk" location="posCenter" />
        </seq>
      </scene>
      <scene id="s2">
        <seq>
          <speak agent="Aya"> Bye-bye. </speak>
          <exit />
        </seq>
      </scene>
    </page>
  </body>
</mpml>
```

Figure 1 An example script of MPML-VR

3 MPML-VR

3.1 Fundamentals

In order to cover three-dimensional presentations with keeping the advantage of ease of description, we have designed and developed MPML-VR, which is an extension of MPML into 3D virtual space. An author is supposed to prepare at least two kinds of resources in making MPML-VR contents, i.e., background VRML data and a presentation script written in MPML-VR. A background (this is a relative term to an agent) is a VRML file that depicts objects to be explained or a presentation environment space. In addition, MPML-VR has a feature of importing external objects to the presentation space. Through making the materials that are common to the presentation as templates, we can reuse these templates to save the labor of creating background VRML. In an MPML-VR script, the author writes what he or she wants agents to present in the background, i.e., an acting script for an agent, a reference to the background VRML file, or placements of external object.

By interpreting an MPML-VR script, the viewing system of MPML-VR will automatically integrate a background space and an agent space to generate one.

3.2 Outline of the Specification

Figure 2 is an example of the MPML-VR script. Since the detailed specification is available at MPML-VR homepage, we note here the outline of MPML-VR specification.

Based on the specification of SMIL[1], the children of <seq> element are executed sequentially, and the children of <par> element are executed in parallel. <seq> and <par> can be nested. In the following example, <play> element makes an agent act a gesture, and <speak> element instructs an agent to speak.

```
<seq>
  <par>
    <play agent="aya" act="bow" />
    <speak agent="aya">Hello.</speak>
  </par>
  <speak agent="aya">How are you?</speak>
</seq>
```

<move> element makes an agent move to a specified place (VRML node) by a specified mode of transportation (e.g., walk, run, etc.).

```
<seq>
  <move agent="aya"
    how="run" location="posCenter" />
</seq>
```

<set-viewpoint> element is used for changing camera angle in the virtual space. A viewer can also change the viewpoint during the execution of presentation.

```
<seq>
  <set-viewpoint location="camCenter"/>
</seq>
```

We adopt a state transition as the progress of presentation, and define *scene* as one state and the transition of the scene represents flow of presentation.

An event element describes which scene to transit from and to. The following example describes two transitions; *1)jump to a scene, "scnPanel" when an audience clicks a place, "panel", and 2)jump to a scene, "scnVideo" when an audience says "video".*

```
<scene id="scnStart">
  <events>
    <world-touch id="evPanel"
      type="Click" location="panel"
      jump-to="scnPanel" />
    <heard id="evVideo"
      agent="aya" voice="video"
      jump-to="scnVideo" />
  </events>
  <seq> <!--Do presentation.--> </seq>
</scene>
<scene id="scnPanel">
  <seq> <!-- Presentation when the panel
            is pressed -->
  </seq>
</scene>
<scene id="scnVideo">
  <seq> <!-- Presentation when agent hears
            "video" -->
  </seq>
</scene>
```

We can also control whether to accept an event or not. In the following example, the viewer's touch with the panel can be processed only after he or she hears the instruction to do so.

```
<seq>
  <speak agent="aya">
    Please touch the panel.
  </speak>
  <activate-event event="evPanel" />
</seq>
```

As previously noted, MPML-VR can import some external objects into the current presentation space. In addition to that, MPML-VR can also control the imported object by calling a method as illustrated below.

```
<scene id="scnMovie">
  <object
    id="TV" url="television.xml"
    location="posTable" />
  <seq>
    <call object="TV"
          method="url" value="sample.mpg"/>
    <call object="TV"
          method="start" value="1" />
```

[1] http://www.w3.org/AudioVideo/

```
    </seq>
  </scene>
```

This example shows how to import a television, load a MPEG file, and play it. The url of VRML and the definition of methods are described in "television.xml". We call this kind of XML file *an object profile*, which describes platform-dependent parameters.

4 Character Agents for MPML-VR

4.1 MPML-VR Agent

We have developed a character agent system for MPML-VR (MPML-VR Agent) together with the specification. The main features of MPML-VR Agent are listed below.

- **Control of presentation space**: MPML-VR Agent is an agent system for presentation in the 3D space. The control features of the presentation space include features to change the background space, to import an external object, and to accept the interactions with viewers.
- **VRML based**: We employ VRML2.0 as the platform of 3D virtual space because a lot of contents and authoring tools are available. Some may also argue that VRML should be obsolete because of the shrinking community for it. Several technologies or standards for Web3D are actually being developed to eliminate the defects of VRML, but none of them has reached the stage of diffusion. We are planning to port MPML-VR system to other Web3D platform (e.g. X3D, state-of-the-art successor to VRML) as the next generation of Web3D comes into sight with no or minimum modification of MPML-VR specification.
- **Animation**: MPML-VR Agent can act gestures prepared in the appearance VRML. A creator of an agent may model gestures just as a standard VRML animation. The animations are executed only by putting `<play>` tag while it is quite awkward to describe VRML event synchronizing with other events.
- **Locomotion**: In contrast to the animation that a creator of a character agent has modeled the actions in advance, MPML-VR Agent has a locomotion engine, which enables a character to move freely around the space. It generates dynamically an *Interpolator VRML node* for key framing that presents a movement from the current position to the specified position.
- **Speech and balloon**: MPML-VR agent can communicate with viewers through speech. In some cases, however, the audience might not comprehend synthesized speech. In consideration of the case, we display a balloon synchronized with the speech text.

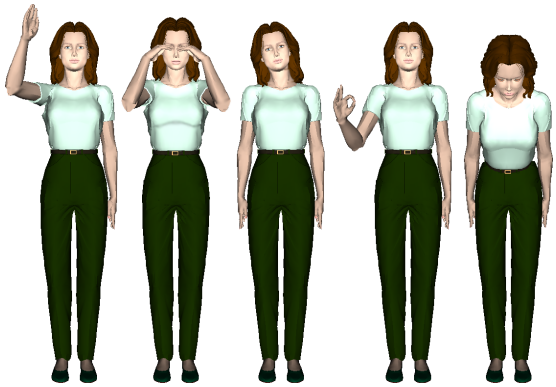

Figure 2. Some examples of gestures that Aya has

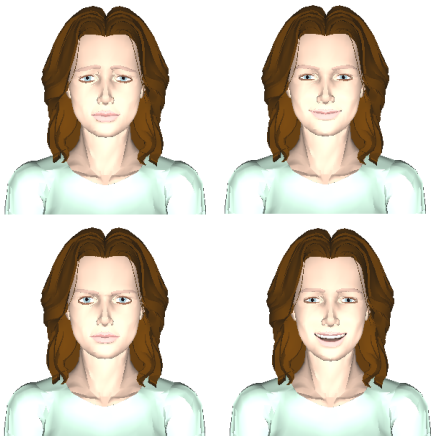

Figure 3. Facial expressions of Aya

- **Emotional expression**: The author can change the emotion of the agent from an MPML-VR script by writing `<emotion>` tag. Five emotions (neutral, happy-for, joy, sad, angry) are defined currently. The change of the emotion turns the facial expression, the pattern of behaviors, the way of emphasizing speech by `<emph>` tag, and the voice pitch/speed.
- **Agent profile**: We introduce an XML database to describe parameters peculiar to an agent, such as URL of appearance VRML file, gender, voice pitch, moving speed, available behaviors, etc. Since such parameters are strongly dependent on TTS engine or rendering format of presentation space, we secede them from the MPML-VR specification to assure the independency of MPML-VR contents. MPML-VR Agent interprets the actual parameters to be sent to the low-layer libraries by consulting an agent profile.

4.2 Our original character agent: "Aya"

"Aya" is a presentation character agent, who has been designed according to the MPML-VR Agent specification, and has detailed appearance and finger joints. She is compliant with Humanoid Animation[2] LOA 2. She acts about a hundred of gestures (see Figure 3) to communicate non-verbally with the

[2] http://www.h-anim.org/

viewers. She can also express her feeling by the facial expressions (Figure 4).

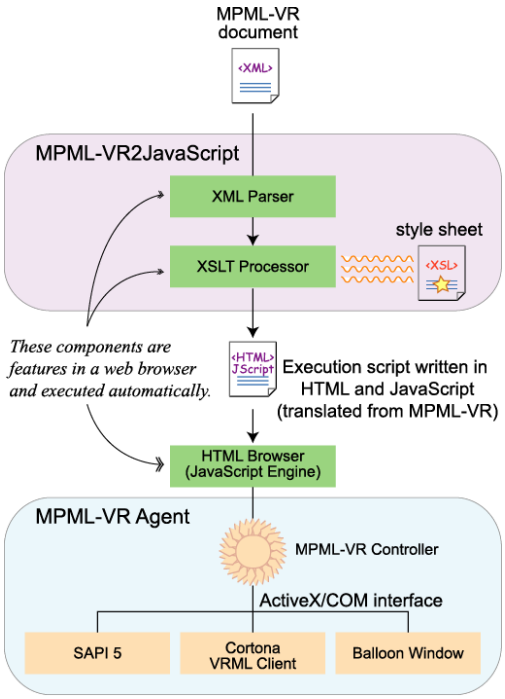

Figure 4. Architecture of viewing system.

5 Implementation

Considering distribution of MPML-VR contents through the WWW, we have created a viewing system in cooperation with a web browser (currently for Microsoft's Internet Explorer). Viewers have only to open the URL where an MPML-VR content is located. To archive this, we use XSLT (XSL Transformations) for converting an MPML-VR document to HTML and JavaScript to be executed in the browser, and ActiveX/COM for controlling Cortona VRML Client[3] and Microsoft Speech API[4].

It is necessary that we make up for VRML's lack of speech recognition/synthesis functions when we establish a character agent system in VRML environment. VRML has several means (e.g., EAI: External Authoring Interface) for interacting with other modules. Since Cortona supports ActiveX Automation technology, we have integrated SAPI 5.1 and Cortona and other modules by ActiveX/COM.

Figure 6 is a screenshot of the MPML-VR presentation, in which the audience can get the spatial knowledge of the house as if he or she follows "Aya". The audience can ask her to move to other rooms. In figure 6, "Aya" is presenting a living room (the content of her speech is shown in the balloon in the bottom left corner), pointing to her left side.

[3] ParallelGraphics, Ltd. http://www.parallelgraphics.com/products/cortona/

[4] Microsoft Speech Technologies Homepage: http://www.microsoft.com/speech/

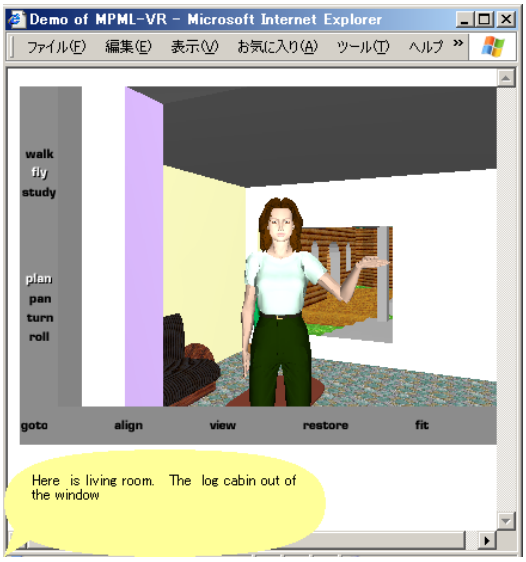

Figure 5. A screenshot of MPML-VR presentation.

6 Conclusion

We have designed and developed MPML-VR, which is an extension of MPML into 3D virtual space and enables us to make multimodal content easily. We have also developed MPML-VR Agent, a character agent system in VRML, and a viewing system that executes on a web browser. We will port the MPML-VR system to other 3D platforms as the next generation of Web3D comes into sight.

References

Andrew Marriott, Simon Beard, John Stallo, and Quoc Huynh. 2001. VHML – Directing a Talking Head. In *Active Media Technology*. Proc. The Sixth International Computer Science Conference Vol. LNCS2252. Springer. Hong Kong.

Berardina De Carolis, Fiorella De Rosis, Valeria Carofiglio, Catherine Pela-chaud, and Isabella Poggi. 2001. Interactive Information Pres-entation by an Embodied Animated Agent. International Workshop on Information Presentation and Natural Multimodal Dialogue.

Mark Evers and Anton Nijholt. 2000. Jacob - An Animated Instruction Agent in Virtual Reality. In *Advances in Multimodal Interfaces - ICMI 2000*. Proc. Third International Conference on Multimodal Interaction. 526-533.

Takayuki Tsutsui, Santi Saeyor and Mitsuru Ishizuka. 2000. MPML: A Multimodal Presentation Markup Language with Character Agent Control Functions. In Proc.(CD-ROM) of WebNet 2000 World Conf. on the WWW and Internet.

Tsukasa Noma, Liwei Zhao and Norman I. Badler. 2000. Design of a Visual Human Presenter, *IEEE Computer Graphics & Applications Magazine*, 20(4):79-85.

MPML-VR Homepage: http://www.miv.t.u-toyko.ac.jp/mpmlvr/