Multimodal Presentation Markup Language on Mobile Phones

Santi Saeyor
Dept. of Information and
Communication Engineering
University of Tokyo
santi@miv.t.u-tokyo.ac.jp

Koki Uchiyama
Hottolink Inc.
1-36-1 Odaka Bldg. 4F
Yoyogi, Shibuya-ku, Tokyo
uchi@hottolink.co.jp

Mitsuru Ishizuka
Dept. of Information and
Communication Engineering
University of Tokyo
ishizuka@miv.t.u-tokyo.ac.jp

Abstract

Multimodal presentation based on embodied conversational agent has been introduced and implemented on various platforms. However, most of them tend to require high computational and graphical capabilities. While mobile phones, which become indispensable device, have extremely limited computational and graphical resources. Implementing multimodal presentation on mobile phones raises a lot of issues because we have to make the most out of limited resources. At the same time, the approaches to creating affective behavior of character are different from other platforms. This paper describes and introduces our approach that successfully brings multimodal presentation to mobile phones.

1 Introduction

Mobile phones have become indispensable device in daily life. At the same time, their functions are increasing both in voice and non-voice services. Thanks to the increasing capabilities, embodied character agent is now available on mobile phones via dedicated 3D graphical engine in Japanese mobile phones.

We have studied the possibility of bringing multimodal presentation to this platform as a successive work to MPML (Multimodal Presentation Markup Language) introduced by Ishizuka et al. MPML has been developed and released on PC based computing environment. Some breeds of MPML are implemented using Microsoft Agents with emotional computing in the backend [Prendinger 2001]. There was a version of MPML that uses embodied character agents in VRML (Virtual Reality Modeling Language) world. [Okazaki 2002] Moreover, MPML has been used for scripting presentation that made by life-like agents on the web. [Prendinger 2002][Saeyor 2002: MPML2.2a]

However, we have found that the original specifications are impossible to be implemented on such a platform due to its limited computational and graphical resources.

This paper describes the issues as well as their solutions when dealing with mobile phones. It also demonstrates how to imp lement a lighter specification of MPML in order to make the most out of using MPML on mobile phones.

2 Using MPML on Mobile Phones

Conceptually, MPML defines the flow of certain entities called ‘scenes’. Each scene comprises of some character agent behavior, display of some web page(s) and usage of certain forms of media like sound. It also has ways of describing how to synchronize them (sequential, parallel etc.). The flow from one scene to another can be natural (i.e. automatic) or triggered by user’s interaction with the character agent.

However, MPML doesn’t define how to control character agent behavior from the user’s actions on various objects/components inside a web page. For example, if we require the action of a character agent to change when user clicks a link / button, we cannot achieve that using just MPML.

Moreover, in tiny handsets like mobile phones, where data entry is a difficult task, storage and reuse of data, once entered or selected by the user, is deemed as a great “convenience” for the user. Though MPML has facilities to define client-side variables, it doesn’t define ways to categorize them as local (transient), session (across current session) and/or global (persistent). It also doesn’t define how to “get”, “set” or “unset” the values of these variables from inside the web pages.

3 Approaches

There are two aspects of MPML that MPML mobile edition inherits, albeit by a different design:

3.1 Sequential Flow of scene: Natural and Dynamic

By invoking <goto> tags inside the web page, we can refer to different MPML scene sequentially. By default the scenes are executed in the order in which they appear inside the web page markup. This is the *Natural* or default order of scene execution.

If the agent has to interact with the user inside a scene to select the next scene, a *Dynamic flow* happens. <ask> tags with any number of <choice> child tags can be used to achieve that. By putting appropriate values for the attributes 'url', 'scene' and 'how', it is possible to move to:

a) A new web page at the background with the agent not executing the remaining scenes of the old page and starting off with the first scene of the new page.
b) A new web page at the background with the agent continuing with its remaining scenes in the old page, discarding the scenes of the new page.
c) Staying at the same web page with the agent changing to a new scene in the current page.
d) Staying at the same web page with the agent changing to a subscene of the current scene.

3.2 Variables

User can be asked a question and the answer typed by the user can be stored for subsequent usage. Three levels of storage can be defined:

- Global
- Session
- Local

The "Global" scope variables are stored persistently and, once defined, all the scenes in all the pages across all future sessions of this application can access them.

The "Session" scope variables are stored persistently across the current session. This means, the "Session" scope variables, once defined, can be used by all the subsequent pages in the current session.

The "Local" scope variables can be used only in the scenes of the current page.

Redefined "Local" and "Session" variables are overwritten.

"Global" variables, once defined, cannot be redefined. This is due to a security threat for global variables. For example, if the global variable storing the user name could be re-written, it may be easy to track its name and change its value to something different, like abusive phrases, in another MPML page.

3.3 Controlling character-agent based multimodal presentation

A set of action tags, defined earlier enables control of agent behavior. The execution of the actions is restricted to sequential, due to limitations in CLDC (Connected Limited Device Configuration) devices. The 'act'-s relevant to a particular agent is entirely dependent on the browser implementation. The speed of character agent's movement or change of posture is dependent on the processing power of the respective machines and the browser implementation.

3.4 Controlling character's behavior from user's action on form components / links

In addition, 'MPML mobile edition' has features that deal with 'change of character behavior with user's action on web page components or objects.' An approach similar to the JavaScript approach has been taken for this purpose. 'Extended' attributes for HTML Form components and 'Link' tags have been defined for user events like *OnClick, onSelect, onDeSelect and onChange* on the web page.

The extended attributes are as follows:

- *onClickGoTo*
- *onSelectGoTo*
- *onDeSelectGoTo*
- *onChangeGoTo*

The value of each of these attributes is a valid scene ID or a valid subscene ID in the same page. They make the character jump to the given scene in case of the following respective events:

OnClick, onSelect, onDeSelect, onChange.

3.5 Difference in the concept of MPML's scenes and MPML mobile edition's scenes

Unlike MPML, in MPML mobile edition, a web page is not a part of a scene. In MPML mobile edition, a scene is a set of character agent actions or in programmatic notion, methods that are called from within the web page markup, to execute character's behavior.

4 Tag Specification

MPML mobile edition also conforms to XML but was rearranged to suit mobile phones' capabilities. The tag specifications are shown in the Table 1.

Table 1 MPML mobile edition’s tag specification

Tag	Attr.	Value	Description
Mpml	None	NA	Top element.
Scene	id	*Any string.*	Unique Identifier.
Show	chara	*Character id.*	Act on this character.
Hide	chara	*Character id.*	Act on this character.
Play	chara	*Character id.*	Act on this character.
	act	*Names of all supported states of the character animation.*	Enact this act.
Move_1 (Walk)	chara	*Character id.*	Move this character.
	x	Positive integer.	Move to this x-coordinate.
	y	Positive Integer.	Move to this y-coordinate.
Move_2 (Fly)	chara	*Character id.*	Move this character.
	x	*Positive integer.*	Move to this x-coordinate.
	y	*Positive integer.*	Move to this y-coordinate.
Speak	chara	*Character id.*	Make this character speak.
	[Node value]	*Any string.*	Speak these words.
Wait	chara	*Character id.*	Make this character wait.
	time	*Time in seconds.*	Wait for this time.
Ask	chara	*Character id.*	Make this character ask.
	scene	*Scene Id.*	Pass on to this scene in this page after user inputs answer or select choice.
	var	*Any String.*	Name of the variable, if the answer is to be stored.
	scope	*“global” | “session” | “local”*	The scope of the variable. “local” by default.
GoTo	url	*URL path.*	Go to this page.
	Scene	*Scene Id.*	Go to this scene of the same page if no url is provided, else go to this scene of new page.
	how	*‘jump’ | ‘html’ (default is’ jump’)*	If ‘jump’, do not execute the remaining actions in the current scene and jump onto the first scene of the next page. Else, only change the background HTML display and continue with current list of character behavior.
SubScene	id	*Subscene id.*	Scene within a scene.
Choice	url	*URL path.*	Go to this URL if this choice is selected.
	scene	*Scene Id.*	If this choice is selected, go to this scene of the same page if no url is provided, or go to this scene of the new page if url is provided.
	how	*‘jump’ | ‘html’ (default is’ jump’)*	If ‘jump’, do not execute the remaining actions in the current scene and jump onto the first scene of the next page. Else, only change the background HTML display and continue with current list of character behavior.
	[Node value]	*Any String.*	The choice.

5 Implementation

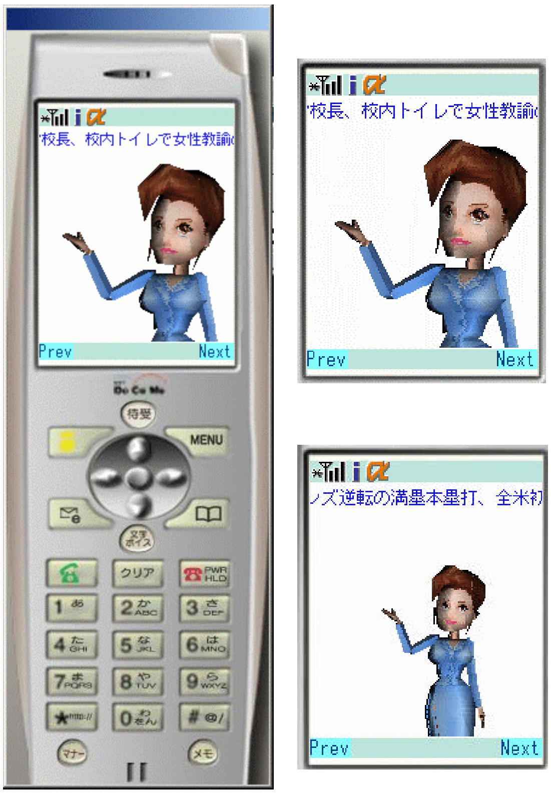
'校長、校内トイレで女性教諭
Prev
Next
MENU
'校長、校内トイレで女性教諭
Prev
Next
ンズ逆転の満塁本塁打、全米初
Prev
Next

Fig. 1 News headline caster using MPML mobile edition

The prototype of MPML player on mobile phones has been created using Java application for Japanese mobile handset based on J2ME: Java 2 Mobile Edition. The application consists of MPML parser, 3D character agent controller, and dialog controller. The handset fetches the MPML data from content server that keeps refreshing news every 30 minutes. The news headlines are reformatted into MPML mobile edition document that contains both content and character's controls. Fig. 1 shows the playback of the MPML mobile document on a mobile phone in NTT Docomo's 504i series. [Saeyor 2002: 3D News Caster]

The MPML parser is an XML parser with a very small foot print of 3 KB in size. We achieve this small profile because we assume that the MPML fed to the parser must be well formed XML so we can neglect a lot of code for error handling in the parsing process.

The character agent controller is a module that manipulate 3D engine inside the mobile phone handset. At the moment, the character agent and its animations must be created before hand. Its animation must be pre-defined and packaged with the model data. We have about 15 animations for the character in this package. This has to be limited to small number of actions because the animation data will occupy a lot of space in Java package (JAR file) and reaches the limit of handset specification (about 30 KB). However, the available actions are adequate for showing affective behaviors and movements in presenting the news headlines to the users.

The dialog controller controls the display of message spoken by the character agent. We use the sliding text, which occupies only one line in the small display of mobile phone in order to make spaces for character agent and background images.

6 Conclusion

This paper shows the approaches we used to bring multimodal presentation to mobile phones based on MPML mobile edition.

The MPML mobile edition also conforms to XML specification but was arranged specially for limited device like mobile phones. Player software that interprets MPML document can be created with small footprint because of simple specification of MPML mobile edition. When combined with 3D character engine in mobile handsets, we successfully realize multimodal presentation system on mobile phone platforms.

The use of MPML mobile edition is not limited to the prototype shown in this paper. It can also be used in mobile educational system. The affective behaviors and attractive movements could be of any help to attract people to view the content that we send in MPML document. However, there are still some issues such as speed, data packet's cost, authoring tools, etc. that are left to be solved. These will be tackled in our successive work.

References

Okazaki, N., S.Aya, S. Saeyor, and M. Ishizuka. 2002. A Multi-modal Markup Language MPML-VR for 3D virtual space. In: Proceedings (CD-ROM) of Workshop on Virtual Conversational Characters: Applications, Methods, and Research Challenges (in conj. With HF2002 and OZCHI2002)

Prendiger, H., S. Descamps, and M. Ishizuka. 2002. Scripting affective communication with life-like characters in web-based interaction systems. Applied Artificial Intelligence, 16:519-553.

Prendiger, H. and M. Ishizuka. 2001 Social role awareness in animated agents. In: Proceeds 5th International Conference on Autonomous Agents (Agents-01), 270-277. ACM Press.

Saeyor. S. 2002. Multi-modal Presentation Markup Language Ver. 2.2a (MPML2.2a) URL: http://www.miv.t.u-tokyo.ac.jp/~santi/research/mpml2a

Saeyor. S. 2002. 3D News Caster for iAppli on NTT Docomo's 504i series and J-Phone's Java Appli http://www.miv.t.u-tokyo.ac.jp/i/3Dcaster