Script Language for Embodied Agents as Personal Conversational Media in Online Communities

Toru TAKAHASHI [†]　Hideaki TAKEDA [‡]　Yasuhiro KATAGIRI [†]

[†] ATR Media Information Science Laboratories,
2-2-2 Seika-cho, Soraku-gun, Kyoto 619-0288, JAPAN
+81 774 95 1486

[‡] National Institute of Informatics,
2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, JAPAN
+81 3 4212 2543

toru@atr.co.jp　takeda@nii.ac.jp　katagiri@atr.co.jp

ABSTRACT

In this paper, we propose a script language for embodied conversational agents that can function as personal conversational media in asynchronous community systems. Despite the fact that the community interaction is a social event, current online community systems are designed to focus mainly on information exchange through texts, and provide very little support for establishing and maintaining social relationships among participants.

In order to enhance and exploit human prowess in social interaction, we have developed an asynchronous community system that employs embodied conversational agents (ECAs) as conversational media. An ECA, an animated character on a screen, can display various social expressions on behalf of the user. Therefore, we can construct and represent social interactive environment by a group of ECAs acting on a screen. This environment created by ECAs, in turn, induces social and psychological relationships between each ECA and the users.

In this paper, we call such ECAs Personified Media (PM). We propose a script language for PM, PMScript, which enables users to specify and describe the behaviors, both expressive and interactive, of PM together with the handlings of other media contents. Participants in asynchronous community systems have sufficient time to compose a script description for their PM. In addition to the features of ECAs, the social presence of a PM can enhance users' community awareness in terms of the human social environment. Therefore, conversations using PM should be smooth, expressive, informative, and social. By accumulating submitted scripts, PMScript can also serve as materials for further analysis and processing of actual participant behavior data in community interactions.

1. INTRODUCTION

When communicating with others, people act to represent their intentions by integrating their full-body modalities such as voice, gesture, pose, gaze, and motion. Additionally, people use not only their own bodies but also contexts around them such as physical and spatial environments, the presence of other people, and history of the variance they have observed. The idea of context is important for not only interpersonal interactions but also interaction with computer systems. From this viewpoint, a major issue in human-computer interface (HCI) is how to create a sense of cognitive and social context among users and systems. Embodied conversational agents (ECAs) [3] are one of the most promising solutions. By employing one or more ECAs in an information system, the user will recognize a kind of spatial and social context between him/her and the ECA(s) and be able to interact with his/her ECA(s) by using communicative functions of the system under that context.

There has been much research on ECAs in terms of the user interface for interactive systems. Some reports have argued for using the agents not only for information efficiency but also for social interaction. For example, agents in a social context induce a user's social behavior according to the context [8]. Here, it is important that not only users but also agents can make good use of the spatial and social contexts to enhance their communication. For example, by means of full-body emotive expression and deictic representations in the social and spatial context, an embodied tutor agent COSMO [9] could enhance the users' learning effect.

We are interested in contexts represented by ECAs. However, in our research we do not use ECAs as autonomous interface agents. Because we want to see users' natural and long-term interaction with ECAs, we use them as personal conversational media in asynchronous online community systems (not in synchronous community systems like *avatars* in virtual worlds). In this paper, we use the term *Personified Media***,** or PM, for such ECAs that function as personal conversational media in community systems. By means of multiple PM, community systems can form a social and cognitive conversational environment, and participants can make both social and expressive representations in the spatial and social context via their own PM.

Here, we focus on a script language for PM. To make good use of PM, we must know the attributes of interactions via PM. After describing the demand for a script language for PM, we discuss the possibilities of PM in this paper.

In the following sections, we first examine several attempts, both past and ongoing, to design agent behavior specification languages from the view point of how they represent and exploit various types of interaction contexts. We introduce our initial attempt and experience in designing ECA specification script language for community systems. We, then, identify major elements of an ECA specification language which are necessary to provide participants with adequate interaction environments through rich contextual information. We propose our own script language, PMScript, which we design to meet these requirements and discuss its possibilities.

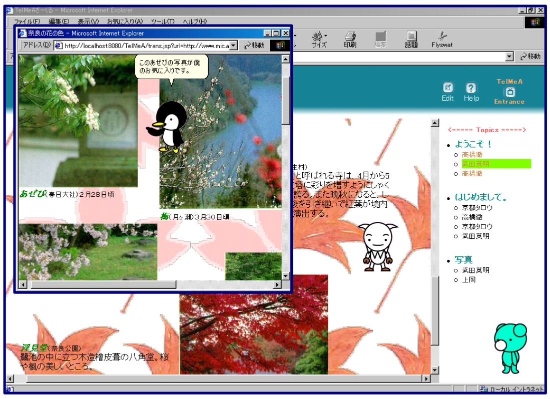
2. RELATED WORKS

From the viewpoint of multimodal presentation with ECA, several presentation systems like Presenter Jack [11] and PPP persona [1] as well as multipurpose presentation languages like MPML [17] have been developed. Moreover, more detailed markup language for ECA like VHML [7] is also proposed. However, these systems and languages are designed not for communication media but for presentation agents. These systems may implicitly assume through interaction some social and spatial contexts between ECAs and their users, but their languages are not designed to store and analyze such social interaction.

On the other hand, there are few languages for PM in online community systems because most systems employ PM for real-time communication by adopting avatars [6]. Users of avatars encounter each other and communicate in the avatars' virtual worlds, but their central communication channel is a simple text chat, or chat with quite simple behaviors like greeting gestures and facial emotions. Because users of avatars have little time to compose their representation, they cannot afford to use various physical modalities in real-time communication.

From the viewpoint of rich physical expression via PM, the machinery-choreographing technology in BodyChat [5] and BEAT [4] are interesting. These technologies add natural and social behaviors to PM automatically when a user inputs text sentences. BEAT has an action form described in XML, and it is possible to change its behavior patterns by modifying the action form. By means of such systems, users of avatars can engage in real-time chatting with multimodal behaviors. This action form helps the unconscious social behaviors of PM. However, many human behaviors are performed consciously in face-to-face conversation. Our script language is designed not only for representation but also for analysis. We aim at a conscious, expressive, and informative environment on the net, where we can analyze social context created by interpersonal interaction through users' PM.

In addition to avatar systems, there are some mailer systems that employ PM (e.g., [16]). Users of such mailer systems compose writing that includes marks for bodily behaviors and exchange utterances using their PM instead of text mail. Because mailer systems are asynchronous communication systems, users have time to consider the behaviors of PM. In such mailer systems, a receiver of an utterance may recognize a social context between him or her and the PM, but PM do not create a shared context between the sender and the receiver at a same time. There are instead two different contexts, i.e., the context between the sender's PM and the receiver and the context between the sender and the receiver's PM.


TelMeAさーくる - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
奈良の花の色 - Microsoft Internet Explorer
アドレス(D)
http://localhost:8080/TelMeA/trans.jsp?url=http://www.mic.a
移動
サイズ
印刷
編集
話題
Flyswat
Edit
Help
TelMeA
Entrance
<===== Topics =====>
ようこそ！
高橋徹
武田英明
高橋徹
はじめまして。
京都タロウ
高橋徹
京都タロウ
武田英明
写真
武田英明
上岡
ローカル イントラネット

このあぜびの写真が僕のお気に入りです。

あぜび(春日大社)２月28日頃

梅(月ヶ瀬)３月30日頃

主村)
と呼ばれる寺は、4月から5
塔に彩りを増すようにしゃく
誇る。また晩秋になると、し
を引き継いで紅葉が境内
演出する。

浮見堂(奈良公園)
鷺池の中に立つ木造檜皮葺の八角堂。桜や楓の美しいところ。

Figure 1. Three PM and multimedia environment in a TelMeA Community

An online community with PM forms a shared context between multiple PM and each user, and each context of each user overlaps with the PM. Therefore, users can share one social and spatial context indirectly on the online community system. As a result, users can communicate while making good use of the shared context on the online community system.

The purpose of this research is to design a script language that describes utterances via PM. For this script language, we need to make the best use of this shared spatial and social context.

3. EXAMPLE OF AN EMBODIED COMMUNITY

In this Section, we construct an image of an online community system that creates social and spatial context via multiple PM by explaining our implemented system named TelMeA [14].

3.1 Description of TelMeA

We designed TelMeA as a web application system. The server-side system of TelMeA is built in Java and the client-side system embedded in the web pages of TelMeA is built in JavaScript.

Anyone can visit the web site of TelMeA via his/her common web browser and participate in several TelMeA communities by registering his/her ID and personal PM from the web page.

Figure 1 shows a screenshot of a TelMeA community. This image includes three PM on two web pages. The bigger page is the main web page of TelMeA. This page consists of three frames: a control frame, a conversation list frame, and a central frame. Users can control actions by their PM on the control frame and select the conversations they want to see on the conversation list frame. The central frame shows a web page of the community. Conversations via PM are mainly made on this shared web page. Additionally, participants can open new browser windows and refer to web contents on web pages through their utterances. Therefore, the TelMeA's shared context of a conversation is basically made by PM, the main web page, and other web pages referred to during the conversation.

3.2 Editing Utterances in ALAScript

When a participant clicks a title of conversation from the conversation list frame, all PM participating in the conversation appear. Then, the first of the PM starts to utter. On TelMeA, all utterances are written in a unique script language named ALAScript.

```
< #actor> penguin
< #approach>  bear
< #play>  Smile
< #speak>  I found such a web page.
< #open>  http://www.mis.atr.co.jp
< #refer>  img3@http://www.mis.atr.co.jp
< #speak>  This picture is nice, isn't it?
```

Figure 2. Instance of ALAScript description

On TelMeA, we include Microsoft Agent [2][10] as PM. Behaviors of Microsoft Agent are described via script code in JavaScript, but participants of TelMeA communities never compose utterances in Microsoft Agent's specific way. Instead, TelMeA prepares ALAScript and an editor of this script language. The client-side system of TelMeA includes an interpreter of ALAScript; this component interprets an ALAScript description into Microsoft Agent's specific grammar and a described character acts in accordance with the description.

Figure 2 shows an instance of ALAScript. With tagged annotation in the head, each line describes a uni-modal sentence of an utterance. In this instance, a "penguin" PM approaches a "bear" PM and says "I found such a web page" with smiling animation. The "penguin" PM opens a web page located at "http://www.mis.atr.co.jp," refers to a picture on the page, and says, "This picture is nice, isn't it?" Thus, an utterance is composed by putting together various representations. PM of TelMeA can represent verbal representations with synthesized voice and text balloons and physical representations with associated animations. PM can open a web page as well as move on a display to the location of other PM or an image on the opened web page.

After a participant writes up an ALAScript description and confirms it, he/she submits this description to the server-side system. The system registers the submitted description as part of the dialogue on a database, and the description becomes available to other participants. In this way, participants communicate asynchronously on TelMeA, but share the context of the conversation, represented by the participants' PM.

4. ATTRIBUTES OF EMBODIED CONVERSATIONAL MEDIA

By exchanging descriptions for their PM, participants of online community systems like TelMeA can represent their intentions and communicate with each other. A sequence of the descriptions for PM makes a conversation and develops their shared context in the community. In this Section, we give the attributes of PM in online communities and clarify the guidelines of the script language specification for PM.

4.1 Bodily Multimodal Representation

People in the real world represent themselves by their bodily modalities such as voice, gesture, facial expression, pose, gaze, and finger pointing. With recent technology, PM can also have several verbal and nonverbal bodily modalities as functions. For example, PM can display text sentences with balloons and speak the sentence by synchronously using a text-to-speech technology for the balloon sentences. By means of CG animation or 3D modeling technology, PM can generate natural gestures, facial expressions, poses, gazes, and finger pointing.

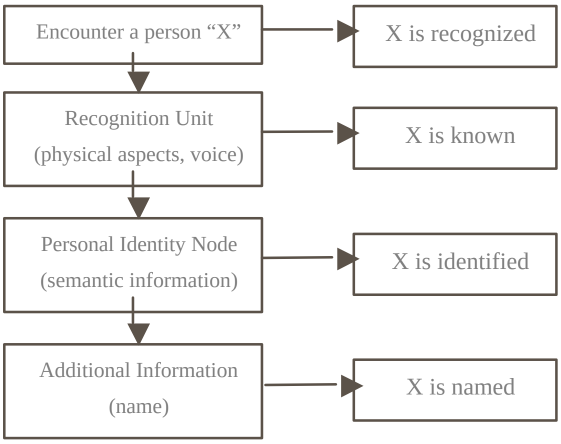


Figure 3. Model of the process for recognizing and naming a person

4.2 Representation in Multimedia Environment

The community environment can play/display multimedia contents such as images, figures, movies, audio, applications, and their combinations like web pages. PM can introduce multimedia content into the multimedia environment and can move and refer to them by finger pointing or by gaze. PM can also move relative to other PM's positions. These interactions among PM and multimedia contents make the spatial and social context of conversation the same as the direction of each PM, and distances between PM have meaning as in face-to-face conversations.

4.3 Social Presence

The term social presence provides a logical explanation of the image of a social and spatial context between PM and their users. Reeves & Nass [12] demonstrated that the interaction between humans and computers is social forms in a way similar to that between humans. Takeuchi & Katagiri [15] suggested that it is possible for system designers to affect users' behaviors by inducing human interpersonal reactions. Moreover, we suggested that it is also possible for system designers to make good use of social relationships between users and their personal agents of a system and protect the system from users' unsocial/unexpected behaviors [13]. In each case, the personification of interface agents is an essential factor in enhancing interpersonal reactions.

Therefore, PM are not media that simply express their users. PM also enhance social relationships between PM and participants in online community systems. The above research suggests the possibility of designing an online community system where each participant has a strong relationship with his/her own PM and forms social relationships with other participants via their PM.

Regardless of whether a participant of the system is aware of the existence of other participants behind other PM, the social presences of PM have an effect on the context of their conversation; PM reduce the chance of offhand unsocial reactions turning into unintended and wasteful "flaming" arguments, and each participant may behave socially as a natural consequence.

4.4 Basis of Community Awareness

Because of the social presences and unique aspects of PM, participants of the community system can easily identify by each other. Figure 3 is a model of the process for recognizing and naming a person, which was studied by Young et al. [18]. The important point of the model is that the process is strictly sequential: users cannot access information on other people unless familiar aspects can be found, and they cannot access additional information on them, such as names, without accessing semantic information such as biographies, interests, personalities, and episodes.

This model says that the physical aspect is important for identifying people. By applying this model to text-based asynchronous conversations like Newsgroups or BBS (Bulletin Board System), users cannot get enough information to identify people because the participants' names and texts are provided without their physical aspects. Suppose a newcomer to the community cannot get to know other participants well. The newcomer might be able to understand each contribution's message, but would probably encounter difficulty in identifying a participant as the contributor of previous messages. This difficulty prevents participants, especially newcomers, from recognizing other participant's personalities and relationships in the group, the meta-structures in conversations, and, as a result, hinders the participation of newcomers in ongoing conversations. (Detailed experimental results are written in [14].)

Conversely, if messages are provided with physical aspects like PM, visitors can identify and recognize other participants more naturally. Therefore, participants can understand utterances of others when provided with a background to each human environment of online communities. As a result, participants, especially newcomers, feel free to join a conversation and the conversation develops more smoothly.

4.5 Summary of Attributes of PM

Users of PM can represent bodily multimodal representations, including verbal and nonverbal physical representation. Moreover, the environment, including multimedia and multiple PM in online communities, provides social and spatial context among participants. The PM of participants can form representation by using such context. As a result, these representations nurture the contents and contexts of the online communities.

Personified aspects of PM enhance the social nature of interrelationships between humans and PM. In turn, the social relationships between participants and their PM enhance the social behavior of the participants and reduce unsocial reactions in the community. Moreover, social behaviors of personified PM enhance the social presence. This social presence leads to the community's awareness of human environments and enhances smooth communication in online communities.

5. PROPOSAL FOR A NEW SCRIPT LANGUAGE FOR PM

5.1 Requirements for Script Languages for PM

Most script languages used to describe the behaviors of interface agents are designed to control the functions of a whole agent comprehensively. That is, they are languages for developers of agent systems. On the other hand, script languages for PM are languages for users. From the viewpoint of users, these languages for PM should be designed so that users can describe as many various social and informative representations as possible. Additionally, it is desirable that script languages for PM be designed to describe the entire context of the representations in order to analyze each utterance and conversation. Consequently, specifications of script languages for PM must not only allow users to represent social and spatial contexts mentioned in the previous section but also permit the analysis of these contexts in sequences of such representations from the viewpoint mentioned in the previous section.

5.2 Evaluation of ALAScript

From the above viewpoint, we evaluate ALAScript, a script language for TelMeA.

5.2.1 Identifier and Social Presence

ALAScript has several kinds of tag annotations. The <#actor> tag is the identifier for PM. Because each PM of TelMeA has a unique personified aspect, PM enhance community awareness of the human environment.

Under the current development of TelMeA, all PM participating in a conversation are displayed from the beginning of the conversation until the end. Therefore, the presence of each PM simply means participation in a part of the conversation, not participation in the current context. In other words, participants of PM cannot use the presence of PM to represent their emotions or intentions. This implementation should be improved, or such an ambiguous presence of PM may confuse the social context among participants.

5.2.2 Bodily Multimodal Representation

The <#speak> and <#think> tags indicate verbal representation, that is, the text sentence of a speech balloon with/without synthesized voice. The <#play> tag means physical representation with animation such as gesture, facial expression, face movement, and finger pointing.

These tags express bodily multimodal expressions efficiently. However, there may be different meanings in a single animated behavior for several contexts. This ambiguity of the relationsre between users' intentions and their representations makes analysis more difficult.

5.2.3 Spatial Context

The <#open> tag means opening a designated web content designated by a URL. The <#move>, <#approach>, and <#refer> tags mean the movement of PM to a designated point, to other PM, and to web content, respectively. By using these tags,

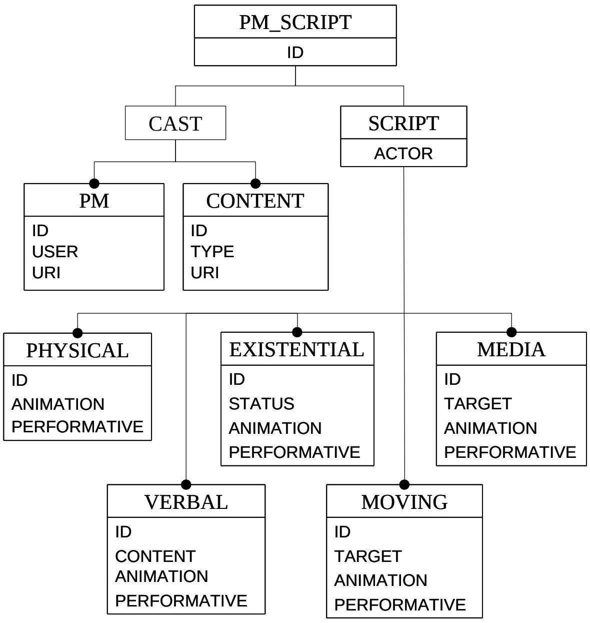


Figure 4. Structure tree of PMScript

participants can utilize and increase the spatial informative contexts in their community.

5.2.4 Summary of ALAScript

ALAScript can represent various social contexts, including presences of PM, multimodal representations through PM, and communicative environments with multimedia contents, by using a small number of tag annotations. Except for representation of social existence, ALAScript has adequate ability to describe the utterances in a PM mediated community system. From the viewpoint of dialogue analysis, however, descriptions in ALAScript includes ambiguous relations between intention for representation and the represented social and spatial contexts.

5.3 Proposal of PMScript

For the purpose of informative and social representation for PM, as well as for PM's analytical capability, we designed a new script language named PMScript. In this script, as with ALAScript, one PMScript description represents one utterance.

Figure 4 shows the structure tree of PMScript. As the figure shows, the content of PMScript consists of two elements, i.e., CAST element and SCRIPT element.

5.3.1 CAST element

The CAST element presents a list of PM and multimedia contents that are involved in the Script element. The PM in the list are presented in the PM element and the multimedia contents in the list are presented in the CONTENT element.

The CAST element has only one attribute, that is, the ID attribute. The ID attribute is an identifier of elements. Most elements in PMScript have this attribute, and it simply shows the identifier, so we will omit the explanation of the ID attribute hereafter.

5.3.2 PM element and CONTENT element

The PM element presents PM of a participant in this script, and the CONTENT element presents multimodal content as well. The PM element and CONTENT element each have three attributes: ID, USER, and URI attributes for the PM element, and ID, TYPE, and URI attributes for the CONTENT element. The USER attribute of the PM element presents the user of the PM. The TYPE element of the CONTENT element presents the MIME content type of this content. The URI attribute of both elements presents the URI (Uniform Resource Identifier) of this PM/content.

5.3.3 SCRIPT Element

The SCRIPT element has only one attribute, the ACTOR attribute. The utterer PM of the script is designated by this ACTOR attribute in the SCRIPT element. The SCRIPT element presents the context of the utterance as a combination of its sub-elements, i.e., VERBAL, PHYSICAL, MOVING, EXISTENTIAL, and MEDIA elements. Each element has its respective ID attribute and system processes elements of PMScript with the ID attributes. We explain the details of each element below.

5.3.4 PHYSICAL Element

The PHYSICAL element presents a representation of a physical modality by animation, such as gestures, facial expressions, and poses. The PHYSICAL element has two attributes other than ID attribute, i.e., ANIMATION and PERFORMATIVE. The ANIMATION attribute indicates the entity of the animation, which shows a sequence of physical movement of the PM. On the other hand, the PERFORMATIVE attribute presents the user's intention in performing this physical representation. For instance, when a "smile" animation is selected in an utterance, the PERFORMATIVE attribute indicates whether this "smile" animation means the intention of acceptance or sarcasm.

5.3.5 VERBAL Element

The VERBAL element presents linguistic representation such as balloon text and synthesized voice via PM. The VERBAL element has three attributes other than the ID attribute, i.e., CONTENT, ANIMATION, and PERFORMATIVE. The CONTENT attribute presents the linguistic sentences for this representation, and the ANIMATION attribute presents physical representation incidental to this verbal representation, such as lip movement when PM utters. The ANIMATION attribute also indicates the type of verbal representation, in the same way as the distinction between <#speech> and <#thought> in ALAScript. For example, the terms of "speech," "thought," "whisper," "sing," and "write" may be suitable for the ANIMATION attribute. The PERFORMATIVE attribute presents the user's intention in using this verbal representation.

5.3.6 MOVING Element

The MOVING element presents representation through the movement of the PM on the display. The MOVING element has three attributes other than the ID attribute, i.e., TARGET, ANIMATION, and PERFORMATIVE. The TARGET attribute presents the move point of the PM, which is designated as an ID attribute of a PM element or a CONTENT element or value indicating coordinates on the display. The ANIMATION attribute

```
ü ECM_SCRIPTü ä
ü CASTü ä
   ü CONTENT  ID=“em_0” TYPE=“embodied_media”
         USER=“Toru Takahashi”
         URI=“http://telmea.aist-nara.ac.jp/char/goat.acs” /ü ä
   ü CONTENT ID=“em_1” TYPE=“embodied_media”
         USER=“Hideaki Takeda”
         URI=“http://telmea.aist-nara.ac.jp/char/penguin.acs” /ü ä
   ü CONTENT ID=“html_0” TYPE=“text/html”
         TYPE=“text/html”
         URI=“http://ai-www.aist-nara.ac.jp/” /ü ä
ü /CASTü ä
ü SCRIPT  ACTOR=“em_0”ü ä
   ü MOBILE  ID=“exp_0” TARGET=“em_1”
         ANIMATION=“Look”
         PERFORMATIVE=“ACCOST” /ü ä
   ü PHYSICAL  ID=“exp_1” ANIMATION=“Smile”
         PERFORMATIVE=“FAVOR” /ü ä
   ü VERBAL  ID=“exp_2” TYPE=“SPEACH”
         CONTENT=“I found such a web page” /ü ä
   ü MEDIA  ID=“exp_3” TARGET=“html_0”
         ANIMATION=“Look”
         PERFORMATIVE=“EXHIBIT” /ü ä
  ü MOBILE  ID=“exp_4” TARGET=“img_2@html_0”
         ANIMATION=“GESTURE”
         PERFORMATIVE=“REFER” /ü ä
   ü VERBAL  ID=“exp_5” TYPE=“SPEACH”
         CONTENT=“This picture is nice, isn’t it?” /ü ä
   ü VERBAL  ID=“exp_6” TYPE=“SPEACH”
         CONTENT=“Anyway, I’ll get off this conversation.” /ü ä
   ü VERBAL  ID=“exp_7” TYPE=“SPEACH”
         CONTENT=“See you! ” /ü ä
  ü EXISTENTIAL  ID=“exp_8” ANIMATION=“Hide”
         PERFORMATIVE=“Quit” /ü ä
ü /SCRIPTü ä
ü /ECM_SCRIPTü ä
```

Figure 5. Instance of PMScript

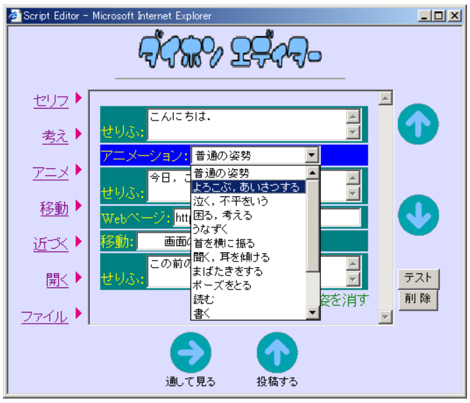
presents physical representation incidental to the verbal representation such as a representation of steps. The ANIMATION attribute also indicates movement speed. The PERFORMATIVE attribute presents the user’s intention in using this movement.

5.3.7 *EXISTENTIAL Element*

The EXISTENTIAL element presents changes in the state of a PM, such as the size of PM, the appearance of PM, or whether the PM is displayed. The EXISTENTIAL element, for example, may make the PM bigger, reclothe the PM, or make the PM disappear. The EXISTENTIAL element has three attributes other than ID, i.e., STATE, ANIMATION, and PERFORMATIVE. The STATE attribute presents the state of PM after this representation is performed. The ANIMATION attribute presents physical representation incidental to the state changing from the current to the goal. The PERFORMATIVE attribute presents the user’s intention in changing this state.

5.3.8 *MEDIA Element*

The MEDIA element presents representation with multimedia contents. The MEDIA element has three attributes other than the ID attribute, i.e., TARGET, ANIMATION, and PERFORMATIVE. The TARGET attribute presents the target of a multimedia content to operate such as letter string, image, movies, web pages, and so on. This value of the TARGET attribute is designated as an ID attribute of the CONTENT element. The ANIMATION attribute presents physical representation incidental to the verbal representation and operation for the multimedia content. The PERFORMATIVE attribute presents the user’s intention in performing this operation.



せりふ: こんにちは.

アニメーション: 普通の姿勢

せりふ: 今日，こ

Webページ: htt

移動: 画面の

せりふ: この前の

Figure 6. Editor for PMScript

5.4 Design of an Editor for PMScript

The structure of PMScript becomes more complex than the structure of ALAScript. However, a community system with PM should not make the user aware of the structure of the language. By designing a good GUI for an editor, we can enable a user to edit the description of complexly structured language without knowledge about it.

We introduce a PERFORMATIVE attribute into PMScript to clarify the intentions of speakers when they compose the script. For instance, we suppose users select their behavior not from the representation of animations but from the PERFORMATIVE, that is, the intention of selecting this behavior. The same behavior often has different meanings in various contexts. Therefore, even when people can understand the meaning of a behavior in the context of a conversation, a log of this conversation cannot provide distinct information. However, if the PERFORMATIVE of the behavior is designated, the context of the conversation can be analyzed by the relations between representations and their PERFORMATIVE.

Figure 5 shows an instance of PMScript. This instance is nearly the same as that of ALAScript in Figure 2. Compared with that instance of ALAScript, this instance of PMScript is much more informative. However, composing ALAScript is probably easier than composing PMScript thanks to the improved GUI of the editor.

Figure 6 shows a screenshot of our editor for PMScript, which is under construction. When a user wants to make his/her PM speak, for example, the user simply selects the word “speak” on the left-hand side of the window and inputs a sentence into the text area that appears in the center space. When a user wants to make his/her PM smile, the user simply selects the word “animation” on the left-hand side and selects a behavior’s name associated with its PERFORMATIVE from the menu list that appears. Smiling behavior has various meanings depending on context, i.e., friendship, having fun, a quizzical smile, and a derisive smile. Since the user selects a behavior based on its meaning, the system can use the intention of the user in analysis. Therefore, the PERFORMATIVE attribute affects both the user interface and system analysis.

6. CONCLUSIONS

In this paper, we proposed a script language for ECA that can function as the personal conversational media on asynchronous community systems. Personified Media (PM) is an environment that provides social and spatial context for online communities by using multiple embodied conversational agents. We analyzed our existing TelMeA system and identified five functions in creating script languages for PM, i.e., bodily multimodal representation, representation in multimedia environments, spatial context, social presence, and community awareness. We designed a script language for PM called PMScript, which has two basic elements, i.e., CAST and SCRIPT. The former represents agents and other multimedia contents including web pages. The latter realizes utterances by various multimodal methods, i.e., verbal, physical, MOVING, existential, and media. PMScript is currently written in XML syntax, so an utterance by PMScript is a tagged sentence. We also provide an editor for PMScript to make it easy for users to compose utterances.

The benefits of PMScript are that it (1) provides a standard protocol that enables different agent-based community systems, and (2) provides a good structure of utterances in order to analyze and reuse them. Future work on PMScript includes implementation of PM that is capable of representing wide variety of expressions in PMScript. Because PM are media for representation in online communities, the ECAs for PM do not need to completely express human representation. Therefore, current scriptable ECA systems such as MS Agent can be used as PM in TelMeA, but its representation ability is limited; for example, it cannot speak while it moves on-screen. We have to improve both the design of PMScript and ECAs as PM.

We are currently adapting our new community system using PM so that it can understand and process PMScript. Furthermore, we are going to propose to the agent research community that the specification of PMScript be used as the standard in order to stimulate further discussion.

7. ACKNOWLEDGEMENT

This research was supported in part by the Telecommunications Advancement Organization.

8. REFERENCES

[1] André, E. and Rist, T.: Coping with Temporal Constraints in Multimedia Presentation Planning, Proc. AAAI-96, vol. 1, pp. 142-147. AAAI Press/The MIT Press, 1996.

[2] Bell, G. et al., Lifelike computer characters: The persona project at Microsoft, in Jeffrey M. Bradshaw (ed.), Software Agent, pp. 191-222, 1997.

[3] Cassell, J., Sullivan, J., Prevost, S., and Churchill, E. (eds.), Embodied Conversational Agents, Cambridge; MIT Press, 2000.

[4] Cassell, J., Vilhjalmsson, H., and Bickmore, T., BEAT: the Behavior Expression Animation Toolkit, Proc. SIGGRAPH 2001, ACM, 2001.

[5] Cassell, J. and Vilhjalmsson, H., Fully embodied conversational avatars: Making communicative behaviors autonomous, Autonomous Agent and Multi-Agent Systems, Vol. 2, No. 1, pp. 45- 69, 1999.

[6] Damer, B., AVATARS!, Jeremy Judson (ed.), Peachpit Press, 1998.

[7] Gustavsson, G., Strindlund, L. and Wiknertz, E., Verification, Validation and Evaluation of the Virtual Human Markup Language (VHML), M.Sc Thesis No. LiTH-ISY-EX-3188, Image Coding Group, Linköping University, Sweden, 2002.

[8] Hayes-Roth, B., Gent, R., and Huber, D., Acting in Character, in Trappl, R., and Petta, P. (eds.), Creating Personalities for Synthetic Actors: Towards Autonomous Personality, Springer Verlag, 1997.

[9] Lester, C., Towns, S., Callaway, C., Voerman, J., and Fitzgerald, P., Deictic and Emotive Communication in Animated Pedagogical Agents, in [3], pp. 123-154, 2000.

[10] Microsoft Corporation, *Microsoft Agent Documentation*, 1998. http://msdn.microsoft.com/workshop/imedia/agent/alldocs.zip

[11] Noma, T. and Badler, I.: A Virtual Human Presenter, *Proc. IJCAI-97 Workshop on Animated Interface Agents*, pp. 45-51, 1997.

[12] Reeves, B. and Nass, C., The Media Equation. Cambridge University Press, 1996

[13] Takahashi, T., Takeuchi, Y. and Katagiri, Y., Change in Human Behaviors Based on Affiliation Needs – Toward the Design of a Social Guide Agent System --, Proc. KES2000, Vol. 1, pp. 64-487, 2000.

[14] Takahashi, T. and Takeda, H., TelMeA: An Asynchronous Community System with Avatar-like Agents, Proc. INTERACT2001, pp. 480-487, 2001.

[15] Takeuchi, Y. and Katagiri, Y., Social Character Design for Animated Agents, Proc. RO-MAN99, pp. 53-58, 1999.

[16] Tosa, N., NEURO-BABY Internet-Mail, 2000. http://www.mic.atr.co.jp/~tosa/EmotionTransMail/image/panel_En.jpg

[17] Tsutsui, T., Saeyor, S., and Ishizuka, M., PML: A Multimodal Presentation Markup Language with Character Agent Control Functions, Proc. (CD-ROM) WebNet 2000 World Conf. on the WWW and Internet, 2000.

[18] Young, A. W., Hay, D. C., and Ellis, A.W., The faces that launched a thousand slips: Everyday difficulties and errors in recognizing people, British Journal of Psychology, Vol. 76, pp. 495-523, 1985.