Adding Life-Like Synthetic Characters to the Web

Elisabeth André and Thomas Rist

DFKI GmbH, Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany
{andre,rist}@dfki.de

Abstract. With the advent of web browsers that are able to execute programs embedded in web pages, the use of animated characters for the presentation of information over the web has become possible. A strong argument in favour of using such characters in a web interface is the fact that they make human-computer interaction more enjoyable and allow for the emulation of communication styles common in human-human dialogue. In this paper we discuss three ongoing DFKI projects on life-like synthetic characters in the internet. While all agents rely on the same approach for automated script generation, we use different player technologies which will be discussed in the light of different applications.

Introduction

Rapid growth of competition in the electronic market place will boost the demand for new innovative communication styles to attract web users. With the advent of web browsers that are able to execute programs embedded in web pages, the use of animated characters for the presentation of information over the web has become possible. Instead of surfing the web on their own, users can join a tour, ask the lifelike character for assistance or even delegate a complex search task to it.

Despite of the raging debate on the sociological effects that life-like characters may have, yet can't have, and will perhaps never have, it is safe to say that they enrich the repertoire of available options which can be used to communicate with the user. First of all, they add expressive power to a system's presentation skills. For example, cross-references between different media (possibly occurring in different windows) can be effectively established through a two-handed pointing gesture. If one strives for emulating the multimodal interaction that occurs between humans, the presentation agents' repertoire of behaviors may even comprise facial expressions and body gestures to express emotions. Furthermore, a presentation agent can also serve as a guide through a presentation to release the user from orientation and navigation problems known from multi-window/multi-screen settings. Last but not least, there is the entertaining and affective function of such an animated character. Lester and colleagues [11] showed that an animated pedagogical may have a strong positive effect on the students' perception of the learning experience. Mulken and colleagues [16] conducted a study to compare presentations with and without a Persona. The subjects perceived the Persona as being helpful and entertaining.

Furthermore, they experienced learning tasks presented by the Persona as being less difficult than those without a life-like character.

In this contribution, we provide an overview of three DFKI projects which are committed to the development of web-based presentation agents for a broad range of applications including personalized information delivery from the WWW. While all agents rely on the same approach for script generation, we use different technologies for their animation depending on the envisioned application.

Related Work

During the last decade, life-like characters have been discovered as a fascinating and challenging area for research with a large potential for commercial applications. In the following, we restrict ourselves to projects that focus on characters for the web.

Adele (Agent for Distributed Learning Environments) is an animated pedagogical agent that has been developed for Web-based medical instruction [13]. The students are presented with a virtual patient in a simulated clinical environment and may perform a variety of actions on the patient. The role of the pedagogical agent is to monitor the students' actions and provide feedback. The animated agent has been realized as a Java applet that runs within a Web browser environment. An empirical evaluation of the Agent revealed that the system was easy to use and that the students found Adele helpful. Unfortunately, the believability of the agent suffers from the poor quality of speech synthesis.

The Agneta and Frida system incorporates narratives into a Web environment by placing two characters on the user's desktop [8]. These characters watch the user during the browsing process and make comments on the visited Web pages. Unlike the DFKI characters, the system relies on pre-authored scripts and no generative mechanism is employed. Consequently, the system operates on predefined Web pages only. The graphical realization of Agneta and Frida is based on the Microsoft^{TM} package. An empirical evaluation of Agneta and Frida showed that animated characters may encourage users to stay longer on a web page. However, users did not learn more about a web page and some of them –especially web-experienced users - even got disturbed by the characters. As a reason, the authors of the Agneta and Frida system indicate that such users have already developed their own strategies for navigating the web which might interfere with the narrative structure provided by the characters.

Virtual meeting spaces and graphical chat corners are a further application field for life-like characters. For example, Isbister and colleagues [10] developed an animated agent that appears in the role of a party host in a web-based chat corner and tries to find a common topic for guests whose conversation has lagged. An experimental evaluation of the agent's ability to assist in cross-cultural conversations has been performed and revealed that the presence of a lifelike synthetic agent may have a positive influence on the perception of each other and of each others cultural group. Smith and colleagues [14] investigated the influence of 3D features on social interaction in chat spaces. For instance, they showed that even experienced users actively make use of proximity and orientation features to enhance their interactions.

There is also an increasing number of commercial web sites that employ animated characters in the interface. Well-known examples include the characters developed by Extempo[1], Haptek[2], Virtual Personalities[3] or Artificial Life[4] [7]. However, due to the constraints imposed by the restricted bandwidth of the internet, these characters have to rely on strongly limited communication channels. For instance, most of them just offer textual output instead of synthesized speech.

Basic Technology

Though a number of similarities may exist, our presentation agents are not just animated icons in the interface. Rather, their behavior follows the equation:

Persona behavior := directives + self-behavior

By *directives* we understand a set of tasks which can be forwarded to a character for execution. To accomplish these tasks, the character relies on gestures that: express emotions (e.g., approval or disapproval), convey the communicative function of a presentation act (e.g., warn, recommend or dissuade), support referential acts (e.g., look at an object and point at it), regulate the interaction between the character and the user (e.g., establishing eye contact with the user during communication) and indicate that the character is speaking. Of course, these gestures may also superimpose each other. For example, to warn the user, a character may lift its index finger, look towards the user and utter the warning. Directives are defined externally, either by a human presentation author or by another system which employs the character as part of its user interface. In what follows, we use the term *presentation script* to refer to a temporally ordered set of directives.

While a script is an external behavior determinant that is specified outside the character, our characters also have an internal behavior determinant resulting in what we call a *self behavior*. A character's self behavior comprises not only gestures that are necessary to execute the script, but also navigation acts, idle time gestures, and immediate reactions to events occurring in the user interface. Note that the borderline between scripted behavior and self behavior is a matter of the degree of abstraction. The more detailed a script prescribes what a character should do, the less there is a need to equip a character with a rich repertoire of reasonable self behaviors.

Since the manual scripting of agent behaviors is tedious, error-prone and for time-critical applications often unfeasible, we aimed at the automation of the authoring approach. Based on our previous work on multimedia presentation design [1], we utilize a hierarchical planner for the automated decomposition of high-level presentation tasks into scripts which will be executed by the presentation agent [2]. To flexibly tailor presentations to the specific needs of an individual user, we allow for the specification of generation parameters (e.g., "verbal utterances should be in

[1] http://www.extempo.com
[2] http://www.haptek.com
[3] http://www.vperson.com
[4] http://www.artificial-life.com

English", or "the presentation must not exceed five minutes"). Consequently a number of presentation variants can be generated for one and the same piece of information, but different settings of presentation parameters. Furthermore, we allow the user to flexibly choose between different navigation paths through a presentation. That is, the course of a presentation changes at runtime depending on user interactions. Knowledge concerning the generation of scripts is represented by means of plan operators.

For the visualization of our Personas, we use three different player technologies depending on the envisioned application: DFKI's Java-based PET toolkit (Persona-Enabling Toolkit), the Microsoft Agent^{TM} toolkit and SMIL (Synchronized Multimedia Integration Language). In the following, we briefly introduce three applications and describe how they are technically realized by using one of these approaches.

The AiA Personas: Java-Based Animated Presentation Agents for the World-Wide Web

In the AiA project (Adaptive Communication Assistant for Effective Infobahn Access), we developed a number of personalized information assistants that facilitate user access to the Web [2] by providing orientation assistance in a dynamically expanding navigation space. These assistants are characterized by their ability to retrieve relevant information, reorganize it, encode it in different media (such as text, graphics, and animation), and present it to the user as a multimedia presentation. The screen shots in Fig. 1 shows one of our applications, which is a personalized travel agent.

Suppose the user wants to travel to Hamburg and is starting a query for typical travelling information. To comply with the user's request, the AiA system retrieves information about Hamburg from various web servers, e.g. a weather, a restaurant and a hotel server, selects relevant units, restructures them and uses an animated character to present them to the user. The novelty of AiA is that the presentation scripts for the characters and the hyperlinks between the single presentation parts are not stored in advance but generated automatically from pre-authored documents fragments and items stored in a knowledge base. For a restricted domain, AiA is even able to combine information units retrieved from different sources and combine them into a single presentation item. For example, the address entry of a hotel is used as input for another web search in order to generate a map display on which the hotel can be located.

The AiA personas have been realized with DFKI's Java-based player technology. To view a Persona presentation, the user does not need to install any software on his or her local machine. Instead the presentation engine is downloaded as a Java-applet. To support the integration of animated agents into web interfaces, our group has developed a toolkit called PET (Persona-Enabling Toolkit). PET provides an XML-based language for the specification of Persona commands within conventional HTML-pages. These extended HTML-pages are then automatically transformed into a down-loadable Java-based runtime environment which drives the presentation on standard web browsers.

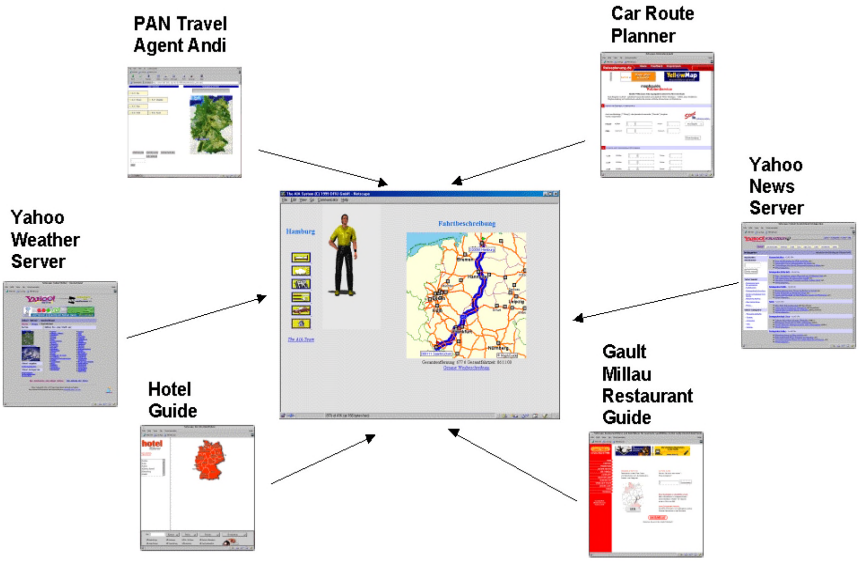


Fig. 1. The AiA travel agent.

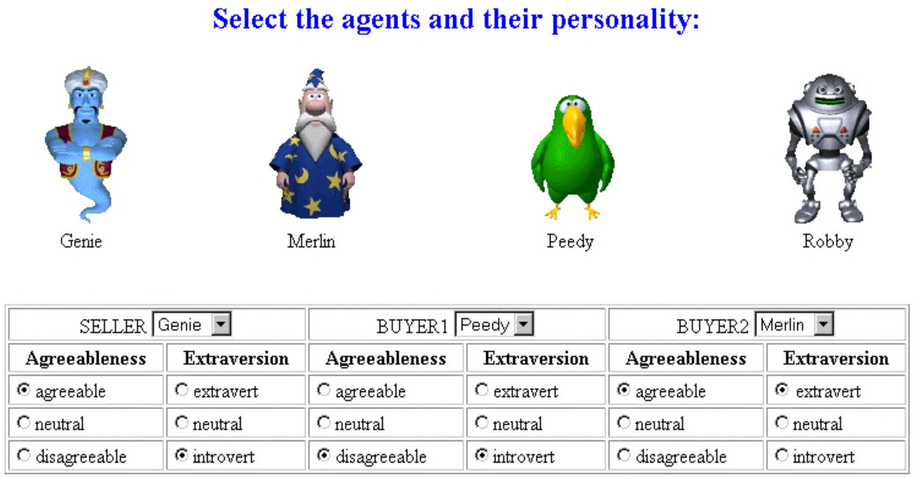
PET may be used in two different ways. First of all, it can be used by a human author for the production of multimedia presentations which include a lifelike character. Second, we have the option to automate the complete authoring process by making use of our presentation planning component to generate web pages that include the necessary PET-commands. Knowledge concerning the script generation process is encoded by means of plan operators. An example of a simple plan operator is listed below. It may be used to instruct a character to describe a hotel by showing a photo and providing a textual explanation.

```
HEADER: (A0 (Describe ?hotel))
CONSTRAINTS:
(*and* (Hotel ?hotel)(Illustrates ?photo ?hotel)
       (Describes ?text ?hotel))
INFERIORS:
(A1(DisplayImage ?hotel))
(A2(Speak ?text))
```

Fig. 2. A simple plan operator for the AiA travel agent.

The Inhabited Market Place: An Animated Presentation Team Based on the Microsoft Agent™ Package

The objective of the Inhabited Market Place is to investigate sketches, given by a team of lifelike characters, as a new form of sales presentation [3]. The basic idea is to communicate information by means of simulated dialogues that are observed by an audience. The purpose of this project is not to implement a more or less complete model of personality for characters, such as a seller and a customer. Rather, the demonstration system has been designed as a testbed for experimenting with various personalities and roles. As suggested by the name, the inhabited market place is a virtual place in which seller agents provide product information to potential buyer agents. For the graphical realisation of the emerging sales dialogues, we use the Microsoft Agent package [12] that includes a programmable interface to four predefined characters: Genie, Robby, Peedy and Merlin. To enable experiments with different character settings, the user has the possibility of choosing three out of the four characters and assigning roles to them (see Fig. 3).



Select the agents and their personality:

SELLER Genie ▾		BUYER1 Peedy ▾		BUYER2 Merlin ▾	
Agreeableness	**Extraversion**	**Agreeableness**	**Extraversion**	**Agreeableness**	**Extraversion**
● agreeable	○ extravert	○ agreeable	○ extravert	● agreeable	● extravert
○ neutral	○ neutral	○ neutral	○ neutral	○ neutral	○ neutral
○ disagreeable	● introvert	● disagreeable	● introvert	○ disagreeable	○ introvert

Fig. 3. Dialogue for character settings.

For instance, he or she may have Merlin appear in the role of a seller or buyer. Furthermore, he or she may assign to each character certain preferences and interests. The system has two operating modes. In the first mode, the system (or a human author) chooses the appropriate character settings for an audience. The second mode allows the audience to test various character settings itself.

Fig. 4 shows a dialogue between Merlin as a car seller and Genie and Robby as buyers. Genie has uttered some concerns about the high running costs which Merlin tries to play down. From the point of view of the system, the presentation goal is to provide the observer – who is assumed to be the real customer - with facts about a certain car. However, the presentation is not just a mere enumeration of the plain facts about the car. Rather, the facts are presented along with an evaluation under consideration of the observer's interest profile.

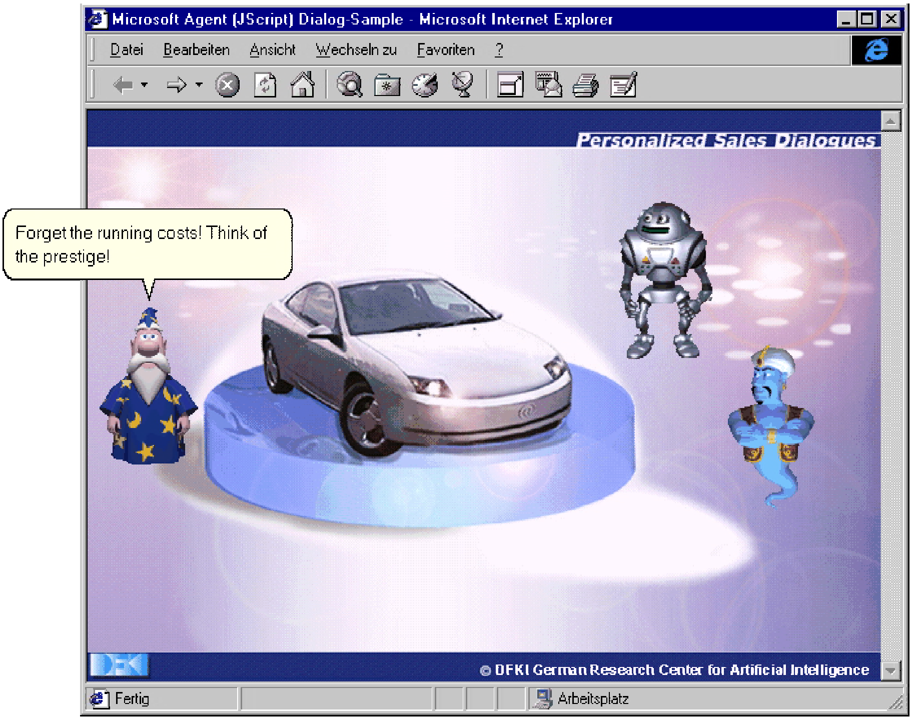

Microsoft Agent (JScript) Dialog-Sample - Microsoft Internet Explorer
Datei Bearbeiten Ansicht Wechseln zu Favoriten ?
Personalized Sales Dialogues
Forget the running costs! Think of the prestige!
© DFKI German Research Center for Artificial Intelligence
Fertig
Arbeitsplatz

Fig. 4. Car sales dialogue example.

While in the travel application sketched above, the information to be presented has been allocated to a single agent, we now have to distribute information to a team of presenters whose activities have to be coordinated. In the sales scenario, the role of the system may be compared with that of a screen writer who produces a script for the actors of a play. The script represents the dialogue acts to be executed by the individual agents as well as their temporal order. From a technical point of view, the approach may be realized by a central planning component that decomposes a complex presentation goal into elementary dialogue and presentation acts that are allocated to the individual agents. To accomplish this task, we use the same plan-based approach as for the AiA personas.[5] The outcome of the planning process is an HTML file that includes control sequences for the Microsoft characters which can be played in the Microsoft Internet Explorer. However, in contrast to the PET approach, the agents' runtime engine is not part of the web page. Rather, to view the presentation with the Microsoft characters, the Microsoft Agent package has to be installed on the user's local machine.

As in the AiA project, knowledge concerning the decomposition of a presentation goal is represented by means of plan operators. However, this time the operators code

[5] An earlier version exploited the Java-based JAM agent architecture [9].

a decomposition of goals into dialogue acts for the agents. For instance, the operator listed in Fig. 5 captures a scenario where two agents discuss a feature of an object. It only applies if the feature has a negative impact on any value dimension[6] and if this relationship can be easily inferred. According to the operator, any disagreeable buyer produces a negative comment referring to this dimension (NegativeResponse). The negative response is followed by a response from the seller (RespNegativeResp).

```
HEADER: (A0 (DiscussValue ?attribute))
CONSTRAINTS:
(*and* (polarity ?attribute ?dimension neg)
       (difficulty ?attribute ?dimension low)
       (Buyer ?buyer)
       (Disagreeable ?buyer)
       (Seller ?seller))
INFERIORS:
(A1 (NegativeResponse ?buyer ?dimension))
(A2 (RespNegativeResp ?seller ?attribute ?dimension))
```

Fig. 5. Plan operator for discussing an attribute value.

An excerpt of a dialogue between Merlin as a seller and Robby and Peedy as buyer is shown in Fig. 6. The last two dialogue turns result from the execution of the operator above.

Robby: How much gas does it consume?
Merlin: It consumes 10 litres per 100 km.
Peedy: That’s bad for the environment!
Merlin: Bad for the environment? It got a catalytic converter, and it is made of recyclable materials.

Fig. 6. Generated dialogue fragment of the sales scenario.

The Virtual News Agency: An Animated Presentation Team Based on SMIL

In the applications described above, the characters’ presentations only included static material, such as written text or images. However, a web presentation may also comprise dynamic media objects, such as video and audio, all of which have to be displayed in a spatial and temporally coordinated manner and need to be synchronized with the agents’ communicative gestures. Fig. 7 shows a typical application scenario with two animated agents - a newscaster and a technician - that watch and comment on a video while it is displayed on a screen.

[6] We adopted the value dimensions from a study of the German car market which suggests that safety, economy, comfort, sportiness, prestige and family and environmental friendliness are the most relevant [15].

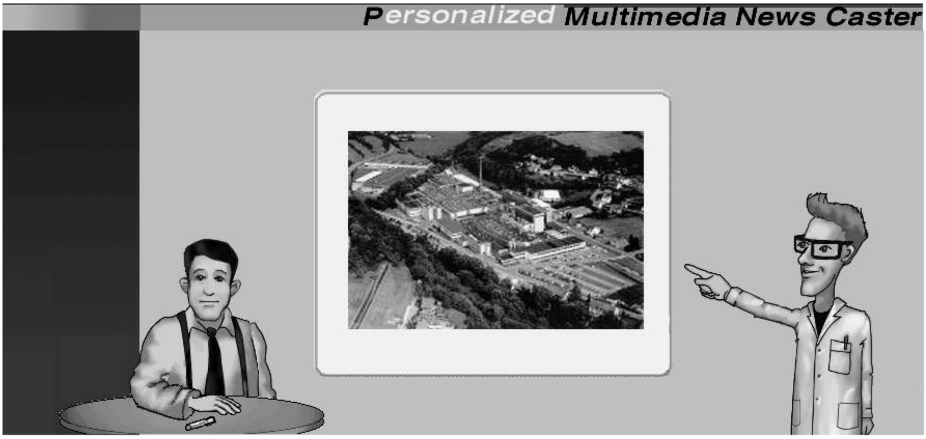


Fig. 7. The Virtual News Agency.

Since the other two player technologies do not include any facilities for controlling the timing of other dynamic media, we decided to rely on the Synchronized Multimedia Integration Language[7] (SMIL) in this application scenario. SMIL is a standard recommended by the W3 consortium with the aim of adding synchronization to the Web. As in the other two applications, we use our presentation planning technology to generate expressions in the SMIL language that are played within a web browser using a SMIL player that supports streaming over the web, such as the Real Player 7 Basic[8]. However, since SMIL requires the specification of a spatial and temporal layout, the following extensions have became necessary:

(1) *the formulation of temporal and spatial constraints in the plan operators*
We distinguish between metric and qualitative constraints. Quantitative temporal and spatial constraints appear as metric (in-)equalities, e.g. (5 ≤ Duration PointInterval) or (5 ≤ bottom CharacterRegion – top CharacterRegion). Qualitative temporal constraints are represented in an "Allen-style" fashion which allows for the specification of thirteen temporal relationships between two names intervals, e.g. (SpeakInterval (During) PointInterval). Qualitative spatial constraints are represented by a set of topological relations, such as LeftOf, CenterHor or TopAlign.

(2) *the development of a mechanism for designing a spatial and temporal layout*
We collect all spatial and temporal constraints during the presentation planning process and use the incremental constraint solving toolkit Cassowary [5] to determine a consistent spatial and temporal layout which is then represented as a SMIL document.[9]

[7] http://smw.internet.com/smil/smilhome.html

[8] http://service.real.com/

[9] When using the Real Player, some further processing of the animation clips is advisable in order to increase the fluency of the overall presentation. Instead of playing one animation after the other, we incorporate all animations that appear in the same region in a single RealPix file that is considered by the RealPlayer as a unit.

In the following, we present a simple plan operator which may be used to synchronize the actions of two characters (Expert and Newscaster) with the display of a video. While the expert provides a summary of the video, the newscaster listens. Both agents then perform a waiting gesture until the video stops. Note that the plan operators do not require a human author to completely specify the layout of a presentation. For instance, we do not specify how long the summarization of the video takes. The spatial constraints specify the position of the inferior actions with respect to A0, the region corresponding to the plan operator.

```
HEADER: (A0 (PresentVideo Expert ?topic))

CONSTRAINTS:
  (*and* (Illustrates ?video ?topic)
         (Summarizes ?audio ?topic))
INFERIORS:
  (A1 (DialogueTurn Expert ?audio))
  (A2 (Listen Newscaster))
  (A3 (SaddSmilCode (?video))
  (A4 (BothWait))
TEMPORAL: ((A1 (starts) A3) (A2 (equals) A1))
           (A1 (meets) A4) (A4 (finishes) A3))
SPATIAL: ((ALIGNLEFT A1) (ALIGNBOTTOM A1)
          (ALIGNLEFT A2) (ALIGNBOTTOM A2)
          (ALIGNLEFT A3) (ALIGNBOTTOM A3)
          (ALIGNLEFT A4) (ALIGNBOTTOM A4))
START: (A3)
END: (A3)
```

Fig. 8. Plan operator for generating a SMIL document.

Conclusions

In this paper, we have described our efforts to develop life-like presentation agents which can be utilized for a broad range of web applications. We sketched three applications that are based on the same presentation planning technology, but exploit different player technologies for the visualization of the Personas. Our experience has shown that there is not yet a general-purpose player technology which meets the needs of all our applications.

The Microsoft Agent^{TM} toolkit includes a number of useful software packages that support the creation of engaging character applications, such as animation components and components for the recognition and synthesis of natural language. Due to its comfortable application programmer's interface, this toolkit is an excellent choice for rapid prototyping. Being free of charge, it is also a low-cost entry point for new character projects. However, to view a presentation with Microsoft Agent characters, the user has to install the Microsoft package on his or her local machine. Furthermore, HTML-pages including control sequences for Microsoft Agent characters have to be played within the Microsoft Internet Explorer.

The rationale behind the development of the PET toolkit was the aim to display characters on web pages without requiring the user to download a dedicated plug-in or to install other (third-party) software. Being implemented as a Java applet, the PET player runs on any standard web browser. These characteristics are in particular of interest for commercial web sites. By making use of the HTML event model, PET also fulfills the necessary prerequisites for the creation of interactive Personas. On the other hand, there is a tradeoff between the player functionality and the overall download time of the applet. Therefore, the approach is less suitable for applications that require a rich repertoire of character behaviors and complex media combinations.

SMIL provides a declarative language for the specification of the spatial and temporal layout and in combination with players, such as RealPlayer 7, sophisticated streaming and synchronization technology. These features are in particular useful if a character's communicative actions have to be coordinated with other dynamic media, such as video and audio. However, SMIL does not support the incremental design of presentations and thus does not allow for the integration of interactive Personas. Furthermore, defining plan operators for SMIL presentations requires some training since it presumes a basic understanding of spatial and temporal constraints. A major deficiency of RealPlayer 7 lies in the fact that it does not support the concept of a transparent color value. As a consequence, characters always occupy a rectangular region within a SMIL presentation, and some extra design effort may become necessary to achieve a smooth integration into the surrounding screen space.

Some Future Directions

Even though an increasing number of companies populate their web sites with characters, the web still appears as a computerized environment. A study by Mulken and colleagues [17] revealed that animated agents do not necessarily appear more believable and trustworthy than text- or audio-based agents. Reasons might be the strongly limited communicative skills of the agents and their mechanical behavior. To create a social atmosphere on the web, we have to allow for more natural interaction by exploiting a broader range of communicative means, such as intonation, mimics and gestures. In addition, man-machine communication has to be enhanced by psychological and social factors. In the long run, web characters should not just support the user in accomplishing certain tasks, but make the visit of a web site an individual experience. While earlier work in the agent community focused on the design of agents that behave in an intelligent manner, the great challenge for the future will be the creation of believable individuals with their own personality and emotions [4].

Currently, most character developers focus on the Web site owner, e.g. an E-Commerce company, who owns the character. Even if such characters adapt themselves to the specific user, it is still the provider who has the final control over the characters' behavior which is essentially determined by the provider's and not the user's goals. A great challenge for future research is the development of personalized user-owned characters. Whereas provider-owned characters inhabit a specific web site to which they are specialized, user-owned characters may take the user to unknown places, make suggestions, direct her attention to interesting information or simply have a chat with her about the place they are jointly visiting. Most likely, the user will more easily build up a social relationship with a character that is committed to him or her than a character that is owned by a company. Furthermore, users might hesitate to chat with a company-owned character since they are afraid of a potential misuse of private data, such as their personal preferences.

Clearly, the overall quality of an animated information agent's support will depend to a large extent on the information gathered from a web site. There are several approaches to tackling this issue. One direction is to rely on sophisticated methods for information retrieval and extraction. However, we are still far from robust approaches capable of analyzing arbitrary Web pages consisting of heterogeneous media objects, such as text, images, and video. Another approach uses so-called annotated environments [6] which provide the knowledge that agents need to appropriately perform their tasks. These annotations can be compared to markups of a Web page. Our hope is that with the increasing popularity of agents, a standard for such annotations will be developed that will significantly ease the characters work.

Another interesting area is the design of internet agents for portable application domains. For instance, if the user is travelling, she may only have access to a small hand-held computer with a tiny display. Nevertheless, she may still want to communicate with her personal assistant. However, a standalone version of the agent is of little use. Rather, it needs to be connected to an information source which provides the data to be presented. It is a challenging task to make best use of the strongly restricted expressive means in this impoverished environment.

Acknowledgements

The work described here has been partially funded by the BMBF (Bundesministerium für Bildung und Forschung). We are grateful to Peter Rist and Bernhard Kirsch for the graphical design of the DFKI Personas. We would also like to thank Jochen Müller and Stefan Neurohr for their work on the design and implementation of the PET server, Thomas Kleinbauer for the extension of the presentation planner to SMIL, Martin Klesen for the implementation of the interface for the Inhabited Market Place and Patrick Gebhard for fruitful discussions and technical advice.

References

1. André, E., and Rist, T. (1995). Generating coherent presentations employing textual and visual material. *Artificial Intelligence Review,* Special Issue on the Integration of Natural Language and Vision Processing 9(2–3):147–165.
2. André, E., Rist, T. and Müller, J. (1999). Employing AI Methods to Control the Behavior of Animated Interface Agents. *Applied Artificial Intelligence* 13:415-448.

3. André, E., Rist, T. , van Mulken, S., Klesen, M. and Baldes, S. (2000). The Automated Design of Believable Dialogues for Animated Presentation Teams. In: Cassell et al. (eds.): *Embodied Conversational Agents*, 220-255, Cambridge, MA: MIT Press.
4. Bates, J. (1994). The Role of Emotion in Believable Agents. *Communications of the ACM* 37(7): 122-125.
5. Borning, A., Marriott, K., Stuckey, P. and Xiao, Y. (1997). Linear Arithmetic Constraints for User Interface Applications, Proc. of the 1997 *ACM Symposium on User Interface Software and Technology*, 87-96.
6. Doyle, P. and Hayes-Roth, B. (1998). Agents in Annotated Worlds. Proc. of the Third International Conference on *Autonomous Agents*, 173-180, New York: ACM Press.
7. Gaudiano, P. and Kater, P. (2000). Alife-WebGuide: An Intelligent User Interface for Web Site Navigation. Proc. of IUI 2000, 121-124, New York: ACM Press.
8. Höök, K., M. Sjölinder, A.-L. Ereback, and P. Persson. (1999). Dealing with the lurking Lutheran view on interfaces: Evaluation of the Agneta and Frida System. Proc. of the *i3 Spring Days Workshop* on Behavior Planning for Lifelike Characters and Avatars. 125-136. Sitges, Spain.
9. Huber, M. (1999). JAM: A BDI-theoretic mobile agent architecture. Proc. of the Third Conference on *Autonomous Agents,* 236–243. New York: ACM Press.
10. Isbister, K. (2000). Helper Agent: Designing an Assistant for Human-Human Interaction in a Virtual Meeting Space. Proc. of *CHI'2000*, 57-64. New York: ACM Press.
11. Lester, J. C., Converse, S. A. Kahler, S.E., Barlow, S.T., Stone, B.A. and Bhogal, R.S. (1997). The persona effect: Affective impact of animated pedagogical agents. Proc. of *CHI'97*, 359–366. New York: ACM Press.
12. Microsoft Agent: Software Development Kit (1999). Microsoft Press, Redmond Washington.
13. Shaw, E., Johnson, W. L. (1999). Pedagogical Agents on the Web. Proc. of the Third International Conference on *Autonomous Agent '99s*, New York: ACM Press.
14. Smith, M.A., Farnham, S.D. and Drucker, S.M. (2000). The Social Life of Small Graphical Chat Spaces. Proc. of *CHI'2000*, 462-469. New York: ACM Press.
15. Spiegel-Verlag. (1993). SPIEGEL-Dokumentation: Auto, Verkehr und Umwelt. Hamburg: Augstein.
16. van Mulken, S., André, E. and Müller, J. (1998). The Persona Effect: How Substantial is it? In: Proc. of *HCI'98*, Sheffield, pp. 53-66.
17. van Mulken, S., André, E. and Müller, J. (1999). An empirical study on the trustworthiness of lifelike interface agents. In H.-J. Bullinger and J. Ziegler, eds., *Human-Computer Interaction* (Proc. of HCI-International 1999), 152–156. Mahwah, New Jersey: Lawrence Erlbaum Associates.