The Generation of Multimedia Presentations

Elisabeth André
German Research Center for Artificial Intelligence (DFKI)
Stuhlsatzenhausweg 3, D-66123 Saarbrücken, Germany
Email: andre@dfki.de

Abstract

Multimedia systems—systems which employ several media such as text, graphics, animation and sound for the presentation of information—have become widely available during the last decade. The acceptance and usability of such systems is, however, substantially affected by their limited ability to present information in a flexible manner. As the need for flexibity grows, the manual creation of multimedia presentations becomes less and less feasible. While the automatic production of material for presentation is rarely addressed in the multimedia community, a considerable amount of research effort has been directed towards the automatic generation of natural language. The purpose of this chapter is to introduce techniques for the automatic production of multimedia presentations; these techniques draw upon lessons learned during the development of natural language generators.

Contents

1 Introduction

Rapid progress in technology for the display, storage, processing and creation of multimedia documents has opened up completely new possibilities for providing and accessing information. While the necessary infrastructure is already in place, we still need tools for making information available to users in a profitable way. A number of authoring systems which support the human author in creating and changing multimedia documents are already commercially available. However, at the same time, it is becoming clear that in many applications the manual creation of multimedia documents is no longer feasible. To satisfy the individual needs of a large variety of users, the human author would have to prepare an exponential number of presentations in advance. In the rapidly growing field of online presentation services, the situation is even worse. If live data has to be communicated, there is simply not enough time to manually create and continuously update presentations.

Intelligent multimedia presentation systems (IMMP systems) represent an attempt to automate the authoring process by exploiting techniques originating from Artificial Intelligence, such as natural language processing, knowledge representation, constraint processing and temporal reasoning. The intelligence of these systems lies in the fact that they base their design decisions on explicit representations of application, presentation and contextual knowledge. Such systems go far beyond current multimedia systems since they account for:

- *effectiveness* by coordinating different media in a consistent manner;
- *adaptivity* by generating multimedia presentations on the fly in a context-sensitive way; and
- *reflectivity* by explicitly representing the syntax and semantics of a document.

Because of these benefits, IMMP systems have gained widespread recognition as important building blocks for a large number of key applications, such as technical documentation, traffic management systems, educational software, and information kiosks (see Fig. 1).

From a linguistic point of view, IMMP systems are interesting because communication by language is a specialized form of communication in general. Theories of natural language processing have reached a level of maturity whereby we can now investigate whether these theories can also be applied to other media, such as graphics or pointing gestures. Furthermore, the place of natural language as one of the most important means of communication makes natural language generators indispensible components of a presentation system. Conversely, the integration of additional media may increase the acceptance of natural language components by avoiding communication problems resulting from the deficiencies of using just one medium.

The purpose of this chapter is to survey techniques for building IMMPSs, drawing upon lessons learned during the development of natural language generators. To

Application	Sample Systems
report generation	MAGIC [23], PostGraphe [29], SAGE [43], RoCCo [6]
technical documentation	COMET [30], IDAS [60], PPP [10], Visual Repair [31] and WIP [5]
route directions	MOSES [48]
mission planning and situation monitoring	AIMI [51], CUBRICON [57], FLUIDS [36]
project management	EDWARD [17], IGING [26]
business forms	XTRA [3]
configuration of computer networks	MMI^2 [70]
education and training	PEA [54], MAGPIE [34], Herman the Bug [66], COSMO [45], Steve [62]
information kiosks	ALFRESCO [65], ILEX [44], PEBA-II [24], AiA [10]

Figure 1: Applications for IMMP Systems

faciliate the comparison of these systems, we will first present a generic reference model that reflects an implementation-independent view of the authoring tasks to be performed by an IMMP system. After that, we will present techniques for automating and coordinating these tasks.

2 A Generic Reference Model for Multimedia Presentation Systems

To enable the analysis and comparison of IMMP systems as well as the reuse of components, an international initiative has set up a proposal for a standard reference model for this class of systems (cf. [16]).

Besides a layered architecture, the proposal comprises a glossary of basic terms related to IMMPs. As the authors of the reference model point out, the terms *medium* and *modality* have been the source of confusion since they are used differently in different disciplines. Aiming at a pragmatic merger of a wide variety of approaches, the reference model uses the term *medium* to refer to different kinds of perceptible entities (e.g., visual, auditory, haptic, and olfactory), to different kinds of physical devices (e.g., screens, loudspeakers and printers) and to information types (e.g., graphics, text and video). The term *modality* is then used to refer to a particular means of encoding information (e.g., 2D and 3D graphics, written and spoken language).

Fig. 2 outlines a simplified version of the reference model. It is composed of a knowledge server and a number of layers which stand for abstract locations for tasks, processes or system components:

- *Control Layer*
  The Control Layer embodies components which handle incoming presentation goals, presentation commands, and possibly further commands, such as stop or interrupt, which allow the user or external components to control the presentation process.

- *Content Layer*
  The Content Layer is responsible for high-level authoring tasks, such as selecting appropriate contents, content structuring and media allocation. As a result, the Content Layer delivers a list of media/modality-specific design tasks and a structural description that specifies how these tasks are related to each other.

- *Design Layer*
  The design layer embodies a number of media/modality-specific design components. These design components can be seen as micro-planners which transform the design tasks delivered by the content layer into a plan for the creation of media objects of a certain type. Furthermore, there is a layout design component which is responsible for setting up constraints for the spatial and temporal layout.

- *Realization Layer*
  The task of the realization layer is the media/modality-specific encoding of information according to the design specifications which have been worked out in the superordinate Design Layer. For graphics, there may be a number of rendering components; for text, there may be components for grammatical encoding, linearization and inflection. In the case of layout, realization includes the spatial arrangement of output and the design of a presentation schedule that takes account of the constraints delivered by the Design Layer.

- *Presentation Display Layer*
  The Presentation Display Layer describes the runtime environment for a presentation. It is responsible for dispatching media objects to suitable output devices such as a loudspeaker, a printer or a computer screen (though the reference model abstracts from concrete devices). Furthermore, all coordination tasks between display devices are performed by this layer.

The *Knowledge Server* is a collection of knowledge sources which are shared among the layers. It consists of four expert modules, each of which represents knowledge of a particular aspect of the presentation process: application, user, context and design.

In conventional multimedia systems, selection/organization, media selection and media encoding are usually performed by a human author. However, to classify a system as an IMMP system does not necessarily imply that all tasks sketched above are automated. In most cases, the system developers have concentrated on a specific

subtask while others are treated in a rather simplified way or neglected altogether. For example, XTRA and AIMI generate natural language and pointing gestures automatically, but rely on prestored tax forms and geographical maps respectively. Other systems retrieve existing material from a database and adapt it to the user's current needs. For instance, Visual Repair [31] annotates pre-authored video, while SAGE [21] modifies stored graphics by adding or deleting graphical objects.

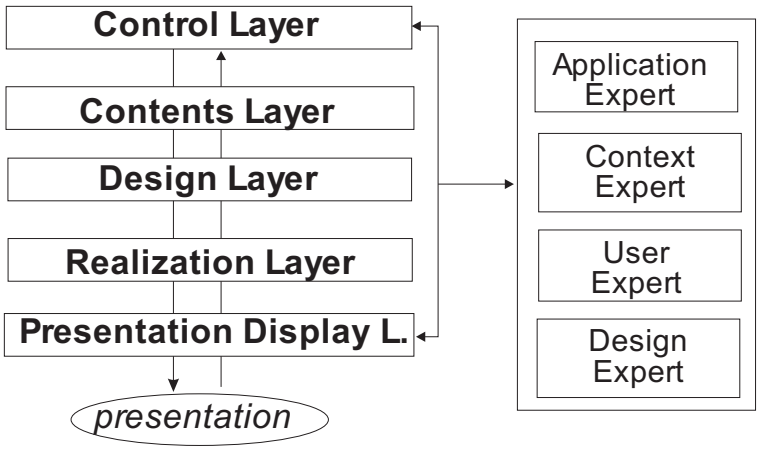

Figure 2: A General Reference Model for IMMP Systems

3 Textlinguistic Approaches as a Methodological Basis

Encouraged by progress achieved in Natural Language Processing, a number of researchers have tried to generalize the underlying concepts and methods in such a way that they can be used in the broader context of multimedia generation. Although new questions arise, e.g., how to tailor text and graphics to complement each other, a number of tasks in multimedia generation, such as content selection and organization, bear a considerable ressemblance to problems faced in Natural Language Generation.

3.1 The Generation of Multimedia Presentations as a Goal-Directed Activity

Following a speech-act theoretical point of view, several researchers have considered the presentation of multimedia material as a goal-directed activity. Under this view, a presenter executes communicative acts, such as pointing to an object, commenting upon an illustration, or playing back an animation sequence, in order to achieve certain goals. Communicative acts can be performed by creating and presenting multimedia

material or by reusing existing document parts in another context (see also the distinction between the presentation display layer and the other four layers in the reference model). The following cases may be distinguished:

- *The generation and the use of multimedia material are considered as a unit*
  This case occurs, e.g., when graphics are created and commented upon while they are being viewed by the user.

- *Multimedia material is created and used later by the same person.*
  This case occurs, e.g., when someone prepares in advance the material to be used for a presentation.

- *Multimedia material is created and used by different authors.*
  This case occurs, e.g., when someone uses material retrieved from some other information source, such as the World Wide Web.

In the last two cases, the goals underlying the production of multimedia material may be quite different from the goals which are to be achieved by displaying it. For example, a graphic which has been generated to show the assembly of a technical device may be used on another occasion to show someone where he or she may find a certain component of this device. These examples illustrate that the relationship between a multimedia document and its use is not trivial and cannot be described by a simple one-to-one mapping; instead, we have to clearly distinguish between the creation of material and its use.

Most multimedia systems are only concerned with the production of multimedia material. Recently, however, personalized user interfaces, in which life-like characters play the role of presenters explaining and commenting on multimedia documents, have become increasingly popular (see Section 6). In such applications, the need for a clear distinction between the design of material and its presentation becomes obvious and is also reflected by the systems' architecture.

3.2 An Extended Notion of Coherence

A number of text linguists have characterized coherence in terms of the coherence relations that hold between parts of a text (e.g. see [33, 38]). Perhaps the most elaborated set is presented in Rhetorical Structure Theory (RST, cf. [50]), a theory of text coherence. Examples of RST relations are *Motivation*, *Elaboration*, *Enablement*, *Interpretation* and *Summary*. Each RST relation consists of two parts: a *nucleus* which supports the kernel of a message and a *satellite* which serves to support the nucleus. RST relations may be combined into schemata which describe how a document is decomposed. Usually a schema contains one nucleus and one or more satellites related to the nucleus by an RST relation. For example, the *Request* schema consists of a nuclear request, and any number of satellites motivating or enabling the fulfillment of the request.

To generalize theories of text coherence to the broader context of multimedia, we have to analyze the relations between the component parts of a multimedia presentation. Earlier studies only investigated relations between pictures as a whole and text, i.e., they did not address the question of how a picture is organized (cf. [46, 15]). In [4], an extensive study of illustrated instructions has been carried out in order to find out which relations may occur between textual and pictorial document parts and how these relations are conveyed. It turns out that the structure of most instructions can be described by a slightly extended set of RST relations. New relations that have to be added include *illustration* and *label*.

Fig. 3 shows the rhetorical structure of a document fragment[1]. We use the graphical conventions introduced by [50], representing relations between document parts by curved lines pointing from the nucleus to the satellite. The document is composed of a request, a motivating part, and a part that enables the user to carry out the action. Rhetorical relations can also be associated with individual picture parts. For example, the depiction of the espresso machine serves as a background for the rest of the picture.

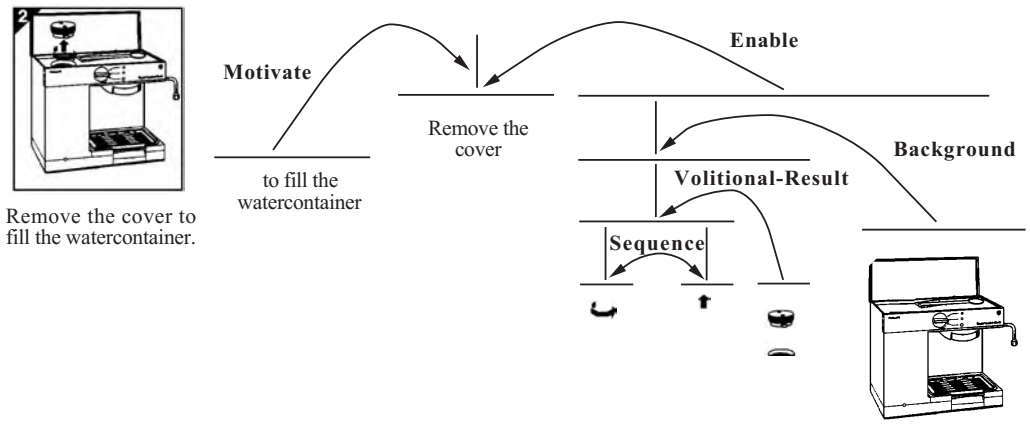

Figure 3: Rhetorical Structure of a Sample Document

3.3 Approaches to Content Selection and Content Organization

Since multimedia presentations follow similar structuring principles to those used in pure text, it seems reaonable to use text planning methods for the organization of the overall presentation, as well as for structuring its textual parts. An essential advantage of a uniform structuring approach is that not only relationships within a single medium, but also relationships between parts in different media can be explicitly represented.

A number of IMMP systems make use of a notion of schema based on that originally proposed by McKeown [52] for text generation. Schema describe standard pat-

[1]The example is a slightly modified and translated version of instructions for the Philips espresso machine HD 5649.

terns of discourse by means of rhetorical predicates which reflect the relationships between the parts of a presentation. Starting from a presentation goal (e.g. "the user should know how to operate a technical device"), a schema is selected. When traversing the schema, information from a given set of propositions is selected. The result of this selection process is forwarded to a media coordinator which determines which generator should encode the selected information. Examples of systems using a schema-based approach are COMET [30] and an earlier prototype of SAGE [64]. While SAGE only relies on schemata to select the text contents, COMET employs schemata to determine the contents and the structure of the overall presentation.

As was shown in [58], information concerning the effects of the individual parts of a schema are compiled out. If it turns out that a particular schema fails, the system may use a different schema, but it is impossible to extend or modify only one part of the schema. For text generation, this has been considered as a major drawback and has led to the development of operator-based approaches (cf. [53]) which enable more local revisions by explicitly representing the effects of each section of the presentation.

In the last few years, extensions of operator-based approaches have become increasingly popular for the generation of multimedia presentations, too. Examples include AIMI [51], FLUIDS [36], MAGIC [23], MAGPIE [34], PPP [9], WIP [5] and a recent extension of SAGE [43]. The main idea behind these systems is to generalize communicative acts to multimedia acts and to formalize them as operators of a planning system. The effect of a planning operator refers to a complex communicative goal while the expressions in the body specify which communicative acts have to be executed in order to achieve this goal. Communicative acts include linguistic (e.g., inform), graphical (e.g., display), physical acts (e.g. gestures) and media-independent rhetorical acts (e.g., describe or identify). A detailed taxonomy of such acts has been proposed by [51]. Starting from a presentation goal, the planner looks for operators whose effect subsumes the goal. If such an operator is found, all expressions in the body of the operator will be set up as new subgoals. The planning process terminates if all subgoals have been expanded to elementary generation tasks which are forwarded to the medium-specific generators. During the decomposition, relevant knowledge units for achieving the goals are allocated and retrieved from the domain knowledge base, and decisions are taken concerning the medium or media combination required in order to convey the selected content.

An advantage of an operator-based approach is that additional information concerning media selection or the scheduling of a presentation can be easily incorporated and propagated during the content selection process. This method facilitates the handling of dependencies, as medium selection can take place during content selection and not only afterwards, as is the case in COMET (cf. [8]).

While WIP, COMET and AIMI only concentrate on the creation of multimedia material, PPP, FLUIDS and MAGIC also plan display acts and their temporal coordination (cf. Section 4.3). For instance, MAGIC synchronizes spoken references to visual material with graphical highlighting when communicating information about a patient's post-operational status; PPP and FLUIDS synchronize speech with the dis-

play of graphical elements and the positioning of annotation labels to explain technical devices.

4 Media Coordination

Multimedia presentation design involves more than just merging output in different media; it also requires a fine-grained coordination of different media. This includes distributing information onto different generators, tailoring the generation results to each other, and integrating them into a multimedia output.

4.1 Media Allocation

The media allocation problem can be characterized as follows: *Given a set of data and a set of media, find a media combination which conveys all data effectively in a given situation.* Essentially, media selection is influenced by the following factors:

- Characteristics of the information to be conveyed
- Characteristics of the media
- The presenter's goals
- User characteristics
- The task to be performed by the user
- Resource limitations

Earlier approaches rely on a classification of the input data, and map information types and communicative functions onto media classes by applying media allocation rules. For instance, WIP starts from 10 communicative functions (*attract-attention, contrast, elaborate, enable, elucidate, label, motivate, evidence, background, summarize*) and 7 information types (*concrete, abstract, spatial, covariant, temporal, quantification* and *negation*) with up to 10 subtypes. Examples of media allocation rules are as follows:

1. *Prefer graphics for concrete information (such as shape, color and texture).*
2. *Prefer graphics over text for spatial information (e.g., location, orientation, composition) unless accuracy is preferred over speed, in which case text is preferred.*
3. *Use text for quantitative information (such as most, some, any, exactly, and so on)*
4. *Present objects that are contrasted with each other in the same medium.*

This approach can be generalized by mapping features of input data to features of media (e.g. static–dynamic, arbitrary–non-arbitrary). An example of such a mapping rule is:

> *Data tuples, such as locations, are presented on planar media, such as graphs, tables, and maps (cf. [13]).*

However, since media allocation depends not only on data and media features, media allocation rules have to incorporate context information as well. Arens and his colleagues [13] proposed representing all knowledge relevant to the media allocation process in And-Or-Networks like those used by Systemic Functional linguists to represent grammars of various language in a uniform formalism. Presentations are designed by traversing the networks and collecting at each node features which instruct the generation modules how to build sentences, construct diagrams, and so on.

While only simple heuristics have been proposed for media selection, approaches relying on deeper inferences have been developed for the selection of graphical 2D-techniques. For instance, APT [49] checks by means of formal criteria which information may be conveyed via a particular graphical technique (the *criterion of expressivity*) and how effectively such a technique may present the information to be communicated (the *criterion of effectiveness*). Casner [19] describes an approach in which media selection is influenced by perceptual factors. His system first analyses the user's task (e.g. the recognition of differences in temperature) and formulates a corresponding perceptual task (e.g. the recognition of differences in length) by replacing the logical operators in the task description with perceptual operators. Then, an illustration is designed which structures all data in such a way that all perceptual operators are supported and visual search is minimized.

4.2 Generation of Referring Expressions in Multimedia Environments

To ensure the consistency of a multimedia document, the media-specific generators have to tailor their results to each other. Referring expressions are an effective means for establishing coferential links between different media. In a multimedia discourse, the following types occur:

Multimedia referring expressions refer to world objects via a combination of at least two media. Each medium conveys some discriminating attributes which taken together allow for a proper identification of the intended object. Examples are natural language expressions that are accompanied by pointing gestures, and text–picture combinations where the picture provides information about the appearance of an object while the text restricts the visual search space, as in "the switch on the frontside".

Cross-media referring expressions do not refer to world objects, but to document parts in other presentation media (cf. [67]). Examples of cross-media referring expressions are "the upper left corner of the picture" or "Fig. x". In most cases, cross-media

referring expressions are part of a complex multimedia referring expresssion where they serve to direct the reader's attention to parts of a document that have also to be examined in order to find the intended referent.

Anaphoric referring expressions refer to world objects in an abbreviated form (cf. [37]), presuming that they have already been introduced into the discourse, either explicitly or implicitly. The presentation part to which an anaphoric expression refers back is called the *antecedent* of the referring expression. In a multimedia discourse, we have not only to handle linguistic anaphora with linguistic antecedents, but also linguistic anaphora with pictorial antecedents, and pictorial anaphora with linguistic or pictorial antecedents (cf. Fig. 4). Examples, such as "the hatched switch," show that the boundary between multimedia referring expressions and anaphora is indistinct. Here, we have to consider whether the user is intended to employ all parts of a presentation for object disambiguation or whether one wants him or her to infer anaphoric relations between them.

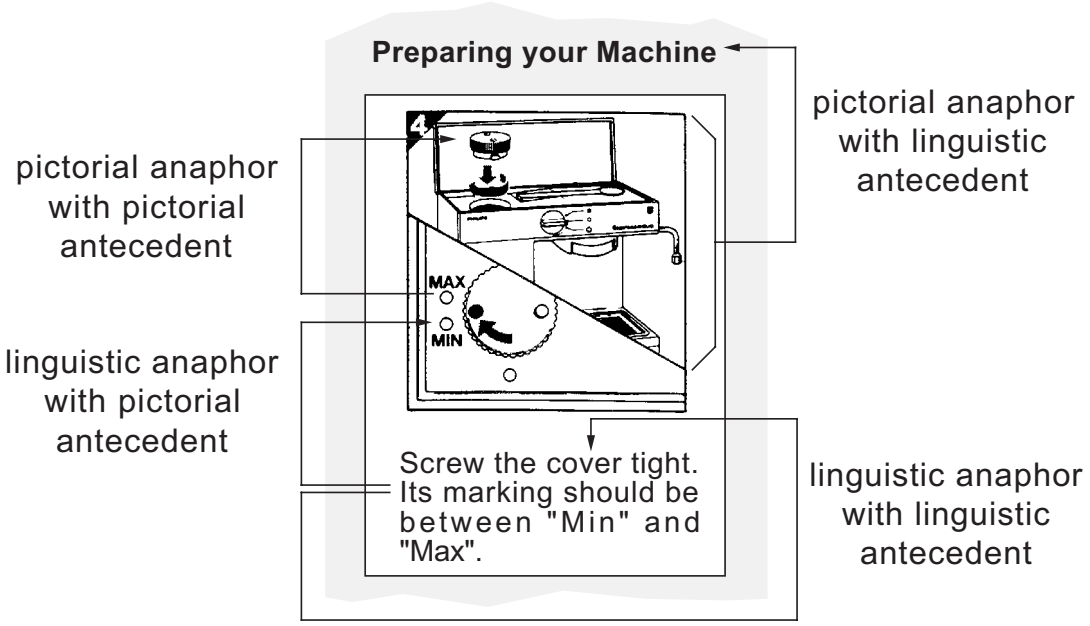

Figure 4: Different types of anaphora occurring in a sample document

In most cases, illustrations facilitate the generation of referring expressions by restricting the focus and providing additional means of discriminating objects from alternatives. A system can refer not only to features of an object in a scene, but also to features of the graphical model, their interpretation, and to the position of picture objects within the picture. Difficulties may arise, however, if it is not clear whether a referring expression refers to an object's feature in the illustration or in the real world.

If a system refers to the interpretation of a graphical feature, it has to ensure that the user is able to interpret these encodings. For instance, the referring expression

"the left resistor in the figure" is only understandable if the user is able to recognize certain images as resistor depictions. Furthermore, it must be clear whether a referring expression refers to an object's feature in the illustration or in the real world since these features may conflict with each other.

Often, spatial relations between images are used to discriminate objects from alternatives. Since a system cannot anticipate where images will be positioned within a picture or which layout will be chosen for text blocks and pictures, it does not make sense to compute such relations in advance. Some systems, such as COMET and WIP, rely on localization components to determine the position of images relative to an illustration (e.g., *the circle in the right section of the illustration*) or relative to other objects in the illustration (e.g. *the circle next to the triangle*). Furthermore, objects may be localized relative to parts of an illustration (e.g. the corners) or object groups (cf. [69]).

An important prerequisite for the generation of referring expressions in a multimedia discourse is the explicit representation of the linguistic and pictorial context.

In the CUBRICON system [57], the linguistic context is represented by a focus list of entities and propositions to which the user of the system may refer via natural language or pointing gestures. The pictorial context represents which entities are visible, in which window they are located, and which windows are visible on the screen. The XTRA system represents not only the linguistic context, but also maintains a data structure for graphics to which the user and the system may refer during the dialogue (cf. [61]). In the tax domain, the graphical context corresponds to a form hierarchy which contains the positions and the size of the individual fields as well as their geometrical and logical relationships. Furthermore, connections between parts of the form, e.g. *region437*, and the corresponding concepts in the knowledge base, e.g. *employer1*, are explicitly represented. While XTRA and CUBRICON rely on different context models for the linguistic and pictorial context, EDWARD [22] uses a uniform model which considers context factors, such as the position in a sentence or visibility on the screen. Unlike these systems, WIP not only represents the semantics of individual images (e.g. *image-1* depicts *world-object-1*), but also the semantics of image attributes (e.g. the property of being coloured red in a picture encodes the real-world property of being defective). To specify the semantic relationship between information carriers and the information they convey, a relation tuple of the form *(Encodes carrier info context-space)* is used. A set of such encoding relations then forms the semantic description of an image based upon which inferences processes may be performed (cf. [7]).

Since image properties may change during the generation process, a presentation system has to ensure that the pictorial context is updated continuously. For instance, if the graphics generator chooses another viewing angle, some spatial relations will have to be updated as well.

4.3 Spatial and Temporal Coordination of the Output

Another coordination task is the integration of the individual generator results into a multimedia output. This includes the spatial arrangement of text blocks and graphics by means of a layout component. A purely geometrical treatment of the layout task would, however, lead to unsatisfactory results. Rather, layout has to be considered as an important carrier of meaning. In particular, it may help indicate the intentions of the presenter, convey the rhetorical structure of a document or draw the user's attention to relevant document parts. For example, two equally sized graphics can be contrasted by putting them beside one another, or one under the other.

While there has been significant work on multimedia production, only a few approaches make use of the structural properties of the underlying multimedia information. The bulk of previous work on automatic layout has concentrated on single media types—e.g., in the context of graphics generation—and does not consider dependencies between layout design and properties of the raw content. Syntactic aspects of layout have been addressed, but not its communicative function. To handle these syntactic aspects, a large variety of techniques have been used, such as dynamic programming, graph drawing algorithms, relational grammars, rule-based systems, genetic algorithms, constraint processing techniques, and efficient search and optimization mechanisms (see [40] for an overview).

The easiest way to enhance a natural language generator by formatting devices is to embed LaTeX or HTML markup annotations directly in the operators of a conventional planner (see also [39]). These annotations are then used by a rendering component, e.g., an HTML-browser, to produce the formatted document. The disadvantage of this method is that the system does not maintain any knowledge about the formatting commands and is thus not able to reason about their implications. Furthermore, it provides only limited control over the visual appearance of the final document.

To avoid these problems, WIP strictly distinguishes between the structural properties of the raw material and its visual realization (cf. [32]). Coherence relations between presentation parts (such as sequence or contrast) are mapped onto geometrical and topological constraints (e.g., horizontal and vertical layout, alignment, and symmetry) and a finite domain constraint solver is used to determine an arrangement that is consistent with the structure of the underlying information.

If information is presented over time, layout design also includes the temporal coordination of output units. The synchronization of media objects usually involves the following three phases:

1. *high-level specification of the temporal behavior of a presentation*
   During this phase, the temporal behavior of a presentation is specified by means of qualitative and metric constraints. Research in the development of multimedia authoring tools usually assumes that this task is carried out by a human author.

2. *computation of a partial schedule which specifies as much temporal information as possible*

From the temporal constraints specified in step 1, a partial schedule is computed which positions media objects along a time axis. Since the behavior of many events is not predictable, the schedule may still permit time to be stretched or shrunk between media events.

3. *adaptation of the schedule at runtime*
   During this phase, the preliminary schedule is refined by incorporating information about the temporal behavior of unpredictable events.

Most commercial multimedia systems are only able to handle events with a predictable behavior, such as audio or video, and require the authors to completely specify the temporal behavior of all events by positioning them on a timeline. This means that the author has to carry out the first and third steps above manually, and the second step can be left out because events with an unpredictable behaviour are not considered.

More sophisticated authoring systems, such as FIREFLY [18] or CMIFED [35], allow the author to specify the temporal behavior of a presentation at a higher level of abstraction. However, the author still has to input the desired temporal constraints from which a consistent schedule is computed. In this case, the second and third steps are automatically performed by the system while a human author is responsible for the first step.

Research in automatic presentation planning finally addresses the automatization of all three steps. In PPP, a complete temporal schedule is generated automatically starting from a complex presentation goal. Basically, PPP relies on the WIP approach for presentation planning. However, in order to enable both the creation of multimedia objects and the generation of scripts for presenting the material to the user, the following extensions have become necessary (cf. [9]):

- *the specification of qualitative and quantitative temporal constraints in the presentation strategies*
  Qualitative constraints are represented in an Allen-style fashion (cf. [2]), which allows for the specification of thirteen temporal relationships between two named intervals, e.g. *(Speak1 (During) Point2)*. Quantitative constraints appear as metric (in)equalities, e.g. *(5 $\leq$ Duration Point2)*.

- *the development of a mechanism for building up presentation schedules*
  To temporally coordinate presentation acts, WIP's presentation planner has been combined with a temporal reasoner which is based on MATS (Metric/Allen Time System, cf. [42]). During the presentation planning process, PPP determines the transitive closure over all qualitative constraints and computes numeric ranges over interval endpoints and their difference. Then, a schedule is built up by resolving all disjunctions and computing a total temporal order.

A similar mechanism is used in MAGIC. However, MAGIC builds up separate constraint networks for textual and graphical output and temporally coordinates these

media through a multi-stage negotiation process, while PPP handles all temporal constraints within a single constraint network irrespective of which generator they come from.

5 Integration of Natural Language and Hypertext

If documents are presented online, it is quite straightforward to offer the user a hypermedia-style interface which allows him or her to jump from one document to another at the click of a mouse. WWW browsers support the realization of such interfaces and make them available to large variety of users. In the ideal case, WWW documents should be customized to the individual user. Doing this manually, however, is not feasible because it would require anticipating the needs of all potential users and preparing documents for them. Even for a small number of users this may be a cumbersome task; it is simply not feasible for the WWW community of currently more than 80 million potential users.

The integration of natural language generation methods offers the possibility of creating hypermedia documents on demand, taking into account the user profile and his or her previous navigation behavior. For instance, if the user browses through the electronics pages of a online shop, this may be taken as evidence that he or she may also be interested in computer magazines. Therefore, a link may be added to the currently visited page which suggests also having a look at the computer magazine pages.

A benefit of hypermedia is that it provides an easy way of involving the user in the discourse planning process (see also [25]). Natural language generators often have to start from incomplete knowledge concerning the user's goals and interests. In such cases, it may be hard to determine which information to include in a presentation. The exploitation of hypermedia may alleviate this problem. Instead of overloading the user with information or risking to leave out relevant information, the user may determine whether or not to elaborate on particular presentation parts simply by selecting certain mouse-sensitive items.

To build a natural language generator which creates hypertexts, we have to modify the flow of control within the control/content layers of the architecture. Instead of building up a complete discourse structure in one shot, certain parts of it are only refined on demand. The basic idea is to treat subgoals that should be realized as hyperlinks analogously to the creation of clauses and forward them to design/realization components which generate corresponding mouse-sensitive items. If the user selects such an item when viewing the presentation, the control/content layer modules are started again with the goal corresponding to that item. The question of whether these modules should also be activated if the user clicks on an item a second time is a matter of debate in the hypermedia community. On the one hand, a dynamic hypermedia system should consider whether the user has already seen a page or not. On the other hand, most web users today are only acquainted with static web pages and might get

confused if the contents and form of a web page change each time they visit it.

There are various ways to integrate hypermedia facilities in a natural language generation system.

PEA [54] and IDAS [60] view hypertext as a means of realizing some simple form of dialogue and offer the user a hypertext-style interface that allows them to select mouse-sensitive parts of an explanation. On the basis of such mouse-clicks, the system generates a menu of follow-up questions which may be asked in the current context. In particular, the systems have to decide (1) where to include hyperlinks and (2) which follow-up questions to offer by consulting the user model and the dialogue history. For instance, PEA will not include a question in the menu if it assumes that the user already knows the answer to this question.

ALFRESCO [65] takes a different approach. This system exploits hypermedia as a browsing facility through an existing information space by generating text with entry points to an underlying preexisting hypermedia network. However, since only the initial text is automatically designed, the system has no influence on the links within the hypertext. Once a user enters the hyperspace, the system is no longer aware of his or her activities and has no possibility of adapting it to their needs.

More recent systems like ILEX [44] and PEBA-II [24] automatically compose hypertext from canned text and items from a knowledge base. To smoothly combine canned text with automatically generated text, the canned text may include various types of annotations. For instance, canned text in ILEX contains pointers to domain entities for which the system automatically creates referring expressions. The links between the single pages are automatically created based on the user profile and current situation.

The idea of dynamic hyperlinks can also be found in recent systems for assisted web access, such as WebWatcher [41] and Letizia [47]. However, unlike ILEX and PEBA-II, these systems rely on handcrafted web pages.

6 Personalized Multimedia Presentation Systems

So far, we have only addressed the generation of multimedia material. Although this material may be coherent and even tailored to a user's specific needs, the presentation as a whole may fail because the generated material has not been presented in an appealing and intelligible way. This can often be observed in cases where multimedia output is distributed across several windows, requiring the user to find out herself how to navigate through the presentation. To enhance the effectiveness of user interfaces, a number of research projects have focussed on the development of personalized user interfaces in which communication between user and computer is mediated by life-like agents (see [28] for an overview).

There are several reasons for using animated presentation agents in the interface. First, they allow for the emulation of presentation styles common in human–human communication. For example, they enable more natural referential acts that involve

locomotive, gestural and speech behaviors (cf. [45]). In virtual environments, animated agents may help users learn to perform procedural tasks by demonstrating their execution (cf. [62]). Furthermore, they can also serve as a guide through a presentation to release the user from orientation and navigation problems common in multi-window/multi-screen settings (cf. [11]). Last but not least, there is the entertaining and emotional function of such animated characters. They may help to lower the "getting started barrier" for novice users of computers/applications, and, as Adelson notes, "... interface agents can be valuable educational aids since they can engage students without distracting or distancing them from the learning experience" (cf. [1], pp. 355).

To illustrate this, we use some examples taken from the PPP (**P**ersonalized **P**lan-based **P**resenter) system. The first application scenario deals with instructions for the maintenance and repair of technical devices, such as modems. Suppose the system is requested to explain the internal parts of a modem. One strategy is to generate a picture showing the modem's circuit board and to introduce the names of the depicted objects. Unlike conventional static graphics where the naming is usually done by drawing text labels onto the graphics (often in combination with arrows pointing from the label to the object), the PPP Persona enables the emulation of referential acts that also occur in personal human–human communication. In the example, it points to the transformer and utters "This is the transformer" (using a speech synthesizer). The example also demonstrates how facial displays and head movements help to restrict the visual focus. By having the Persona look into the direction of the target object, the user's attention is directed to this object.

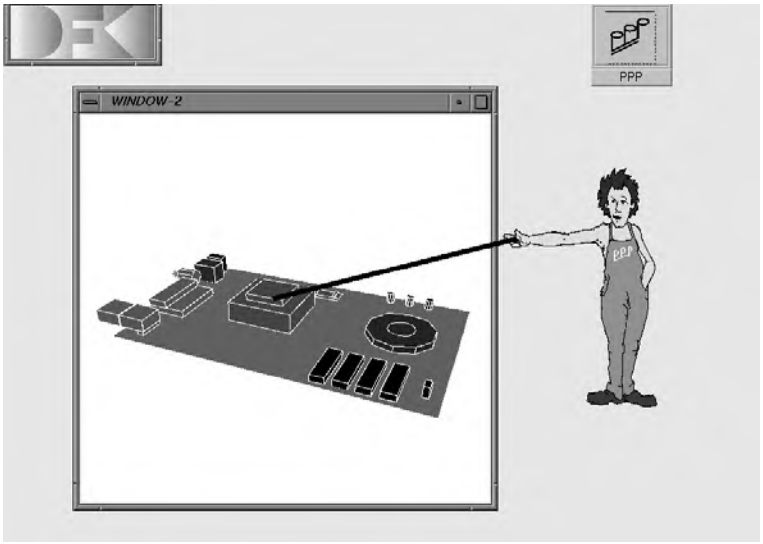

Figure 5: The Persona instructs the user in operating a technical device

In the second example, the Persona advertises accommodation offers found on the WWW. Suppose the user is planning to spend holidays in Finland and is therefore

looking for a lakeside cottage. To comply with the user's request, the system retrieves a matching offer from the web and creates a presentation script for the PPP persona which is then sent to the presentation viewer (e.g. Netscape Navigator^{TM} with an in-built Java^{TM} interpreter). When viewing the presentation, the PPP Persona highlights the fact that the cottage has a nice terrace by means of a *verbal annotation of a picture*; i.e., Persona points to the picture during a verbal utterance (cf. Fig. 6). When the

Figure 6: PPP Persona presents retrieval results from the web using the Netscape Navigator^{TM} and Java^{TM}

graphical elements are generated automatically, as in the modem example, the presentation system can build up a reference table that stores the correspondences between picture parts and domain concepts. Since scanned pictures are used in the travelling agent application, such a reference table has to be set up manually in order to enable pointing gestures to that material. However, in many cases, the author of a web page has already carried out the task of relating image regions to concepts. For example,

many maps available on the web are already mouse-sensitive; in such cases, the system just has to follow the links to find the concepts related to the mouse-sensitive regions.

According to its functional role in a presentation, an animated character must be conversant with a broad variety of presentation gestures and rhetorical body postures. It has to execute 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 articulate what is being said.

From a technical point of view, it makes no difference whether we plan presentation scripts for the display of static and dynamic media, or presentation acts to be executed by life-like characters. Basically, we can rely on one of the temporal planners presented in Section 3.3 if we extend the repertoire of presentation strategies by including strategies which control the Persona's presentation behavior. However, the behavior of a character is not only determined by the directives (i.e., presentation tasks) specified in the script. Rather, it follows the equation:

$$\textbf{Persona behavior} := \textbf{directives} + \textbf{self-behavior}$$

Such self-behaviors are indispensible in order to increase the Persona's vividness and believability (cf. [63]). They comprise idle-time actions, such as tapping with a foot, actions for indicating activity, e.g., turning over book pages, navigation acts, such as walking or jumping, and immediate reactions to external events, such as mouse gestures on the presented material.

Though it is possible to incorporate self-behaviors into the plan operators of a script generator, the distinction between task-specific directives on the one hand, and character-specific and situation-specific self-behaviors on the other, bears a number of advantages. From a conceptual point of view, it provides a clear borderline between a *what to present part* which is determined by the application, and a *"how to present" part* which, to a certain extent, depends on the particular presenter. From the practical perspective, this separation considerably facilitates the exchange of characters, and the reuse of characters for other applications.

While planning techniques have been proven useful for the generation of presentation scripts, the coordination of script directives and self-behaviors and their transformation into fine-grained animation sequences has to be done in real-time, and thus requires a method which is computationally less expensive. One solution is to precompile declarative behavior specifications into finite-state machines. Such an approach has been originally proposed for Microsoft's Peedy (cf. [14]) and later been adapted for the PPP Persona. The basic idea is to compute for all possible situations beforehand which animation sequence to play. As a result, the system just has to follow the paths of the state machine when making a decision at runtime, instead of starting a complex planning process. Finite-state automata are also a suitable mechanism for synchronizing character behaviors. For instance, Cassell and colleagues [20] use

so-called parallel transition networks (PaT-Nets) to encode facial and gestural coordination rules as simultaneous executing finite-state automata.

A character's face is one of its most important communication channels. Lip movements articulate what is being said, nose and eyebrow movements are effective means of conveying emotions, and eye movements may indicate interest or lack of interest, just to mention a few examples. To describe possible facial motions performable on a face, most systems rely on the facial action coding system (FACS, [27]) or MPEG-4 facial animation parameters (FAPs). For instance, both Nagao and Takeuchi [56] and Cassell and colleagues [20] map FACS actions, such as Inner Brow Raiser, onto communicative functions, such as Punctuation, Question, Thinking or Agreement. In both systems, speech controls the generation of facial displays: the speech synthesizer provides parameters, such as timing information, which form the input for the animation components.

The believability of a life-like character hinges on the quality of the output speech. Unfortunately, most life-like characters today only use the default intonation of a speech synthesizer. Here, the integration of natural language generation and speech synthesis technology offers great promise, since a natural language generator may provide the knowledge of an utterance's form and structure that a speech synthesizer needs to in order to produce good output. Prevost [59] provides an example of such an approach: this work uses a Categorial Grammer to translate lexicalized logical forms into strings of words with intonational markings. Also noteworthy is the approach to speech synthesis adopted by Walker and colleagues [68]), which also addresses the social background of the speaker and the affective impact of an utterance.

7 Architectures for Multimedia Presentation Systems

While the reference model introduced in Section 2 abstracts away from a concrete implementation, the developer of an IMMP system has to commit to a particular architecture: he or she has to decide how to distribute the tasks corresponding to the single layers onto different components, and how to organize the information flow between these components.

Research on architectures for IMMP systems can be roughly divided into two camps. One group tries to find architectural designs which are able to handle interactions within and between layers. The other group focuses on the technically challenging task of integrating heterogeneous components that were not originally designed to work together.

Architectures proposed by the first group essentially differ in the organization of the processes to be performed by the content layer.

In the early SAGE prototype [64], relevant information is selected first and then organized by the text and graphics generators. Then, the generated structures are transformed into text and graphics. A disadvantage of this method is that text and graphics are built up independently of each other. Therefore, SAGE needs to revise generated

textual and graphical document parts to tailor them to each other.

In COMET [30], a tree-like structure that reflects the organization of the presentation to be generated is built up first. This structure is passed to the media coordinator which annotates it with media information. Then, the tree is extended by the medium-specific generators in a monotonic manner: there is only a one-way exchange of information between the content layer and the design and realization layers. More attention has been devoted to dependencies between text and graphics generation. Examples include the coordination of sentence breaks and picture breaks and the generation of cross references. To facilitate communication between text and graphics generation, all information to be communicated is represented in a uniform logical form representation language. Since both generators rely on the same formalism, they can provide information to each other concerning encoding decisions simply by annotating the logical forms.

Arens and colleagues [12] propose a strict separation of the planning and the medium selection processes. During the planning process, their system fully specifies the discourse structure, which is determined by the communicative goals of the presenter and the content to be communicated. Then, special rules are applied to select an appropriate medium combination. After medium selection, the discourse structure is traversed from bottom to top to transform the discourse structure into a presentation-oriented structure. A problem with this approach is that the presentation structure obviously has no influence on the discourse structure. The selection of a medium is influenced by the discourse structure, but the contents are determined independently of the medium.

MAGPIE [34] relies on an agent-based architecture for presentation design. As in COMET, a blackboard mechanism is used to handle the communication between the single agents. However, in MAGPIE, agents are hierarchically organized on the basis of task decomposition. That is agents may delegate tasks to other subordinated agents with whom they share a blackboard. For example, the group headed by the table generator agent includes the number agent, the icon agent and the text agent.

Whereas the systems mentioned above rely on separate components for content selection/organization and media allocation, WIP uses a uniform planning approach which accomplishes all these tasks concurrently. That is the content layer is realized just by one component, namely the presentation planner. The most important advantage of such an approach is that it facilitates the handling of dependencies among choices. As in COMET, the document plan generated by the presentation planner serves as the main data structure for exchanging information among the planner and the generators.

In the systems above, the realization components forward media objects to the presentation display components, but do not specify how and when they should be presented to the user. Document parts are either shown to the user immediately after their production (incremental mode), or the systems wait until the production process is completed and then present all the material at once (batch mode). Consequently, these systems are not able to influence the order in which a user processes a document,

or the speed of processing. Unlike these systems, MAGIC's and PPP's design and realization components also handle the temporal layout of presentations.

The second group is represented by Moran and colleagues [55] who focus on the combination and reuse of existing software components in new multimedia applications. To support the creation of applications from agents written in multiple languages and running on different platforms, e.g. a text generator running on a Sun and a graphics generator running on a Silicon Graphics, they propose an open agent-based architecture (OAA). The OAA agents communicate with each other in a high-level logical language called the Interagent Communication Language (ICL). A key role in OAA is played by the Facilitator Agent: this decomposes complex requests into elementary requests which are delegated to the individual agents. Thus, the Facilitator Agent may be compared to the content planner in the MAGIC system or the presentation planner in the PPP system. However, while the Facilitator has been conceived as a general-purpose communication agent, the MAGIC and PPP planners have been tailored to the coordination of presentation tasks in a multimedia enviroment. They also include mechanisms for media coordination and synchronization, while this task would be performed by a special agent in OAA.

8 Conclusion

The availability of new media opens up new ways of presenting information and leads to new research issues, such as the selection and coordination of media. On the one hand, the integration of multiple media adds complexity to the generation task because far more dependencies have to be handled. On the other hand, many theoretical concepts already developed in the context of natural language processing, such as speech acts and rhetorical relations, take on an extended meaning in multimedia discourse. A key observation of this chapter is that multimedia presentations follow similar structuring principles to those found in pure text. For this reason, text planning methods can be generalized in such a way that they become useful for the creation of multimedia presentations too. Systems like MAGIC and PPP show that the combination of text planners with a module for temporal reasoning even enables the generation of dynamic multimedia presentations. Indeed, the development of the first generation of IMMP systems has been significantly influenced by research in natural language generation.

9 Acknowledgments

This work has been supported by the BMBF under the contracts ITW 9400 7 and 9701 0. I would like to thank Robert Dale and Thomas Rist for their valuable comments.

References

[1] B. Adelson. Evocative Agents and Multi-Media Interface Design. In *Proc. of the UIST'92 (ACM SIGGRAPH Symp. on User Interface Software and Technology)*, pages 351–356, Monterey, CA, U.S.A., 1992.

[2] J. F. Allen. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM*, 26(11):832–843, 1983.

[3] J. Allgayer, K. Harbusch, A. Kobsa, C. Reddig, N. Reithinger, and D. Schmauks. XTRA: A Natural-Language Access System to Expert Systems. *International Journal of Man-Machine Studies*, 31:161–195, 1989.

[4] E. André. *Ein planbasierter Ansatz zur Generierung multimedialer Präsentationen*. DISKI-108, INFIX-Verlag, Sankt Augustin, 1995.

[5] E. André, W. Finkler, W. Graf, T. Rist, A. Schauder, and W. Wahlster. WIP: The Automatic Synthesis of Multimodal Presentations. In M. Maybury, editor, *Intelligent Multimedia Interfaces*, pages 75–93. AAAI Press, 1993.

[6] E. André, G. Herzog, and T. Rist. Generating Multimedia Presentations for RoboCup Soccer Games. In H. Kitano, editor, *RoboCup-97: Robot Soccer World Cup I (Lecture Notes in Computer Science)*, pages 200–215. Springer, 1998.

[7] E. André and T. Rist. Referring to World Objects with Text and Pictures. In *Proc. of the 15 th COLING*, volume 1, pages 530–534, Kyoto, Japan, 1994.

[8] E. André and T. Rist. Generating Coherent Presentations Employing Textual and Visual Material. *Artificial Intelligence Review, Special Volume on the Integration of Natural Language and Vision Processing*, 9(2-3):147–165, 1995.

[9] E. André and T. Rist. Coping with temporal constraints in multimedia presentation planning. In *Proc. of AAAI-96*, volume 1, pages 142–147, Portland, Oregon, 1996.

[10] E. André, T. Rist, and J. Müller. Employing AI Methods to Control the Behavior of Animated Interface Agents. *Applied Artificial Intelligence Journal*, 1998. to appear.

[11] E. André, T. Rist, and J. Müller. WebPersona: A Life-Like Presentation Agent for the World-Wide Web. *Knowledge-Based Systems*, 1998. to appear.

[12] Y. Arens, E. Hovy, and S. van Mulken. Structure and Rules in Automated Multimedia Presentation Planning. In *Proc. of the 13 th IJCAI*, Chambéry, France, 1993.

[13] Y. Arens, E. Hovy, and M. Vossers. Describing the Presentational Knowledge Underlying Multimedia Instruction Manuals. In M. Maybury, editor, *Intelligent Multimedia Interfaces*, pages 280–306. AAAI Press, 1993.

[14] G. Ball, D. Ling, D. Kurlander, J. Miller, D. Pugh, T. Skelly, A. Stankosky, D. Thiel, M. van Dantzich, and T. Wax. Lifelike computer characters: the persona project at microsoft. In J.M. Bradshaw, editor, *Software Agents*, pages 191–222. AAAI/MIT Press, Menlo Park, CA, 1997.

[15] S. Bandyopadhyay. Towards an Understanding of Coherence in Multimodal Discourse. Technical Memo TM-90-01, Deutsches Forschungszentrum für Künstliche Intelligenz (DFKI), Saarbrücken, 1990.

[16] M. Bordegoni, G. Faconti, S. Feiner, M.T. Maybury, T. Rist, S. Ruggieri, P. Trahanias, and M. Wilson. A Standard Reference Model for Intelligent Multimedia Presentation Systems. *Computer Standards and Interfaces: The International Journal on the Development and Application of Standards for Computers, Data Communications and Interfaces*, 18(6-7):477–496, 1997.

[17] E. Bos, C. Huls, and W. Claassen. Edward: Full integration of language and action in a multimodal user interface. *International Journal of Human-Computer Studies*, 40:473–495, 1994.

[18] M.C. Buchanan and P.T. Zellweger. Automatically Generating Consistent Schedules for Multimedia Documents. *Multimedia Systems*, 1:55–67, 1993.

[19] S.M. Casner. A Task-Analytic Approach to the Automated Design of Graphic Presentations. *ACM Transactions on Graphics*, 10(2):111–151, April 1991.

[20] J. Cassell, C. Pelachaud, N.I. Badler, M. Steedman, B. Achorn, T. Becket, B. Douville, S. Prevost, and M. Stone. Animated conversation: Rule-based generation of facial expression,gesture and spoken intonation for multiple conversational agents. In *Proc. of Siggraph'94*, Orlando, 1994.

[21] M.C. Chuah, S.F. Roth, J. Kolojejchick, J. Mattis, and O. Juarez. SageBook: Searching Data-Graphics by Content. In *Proc. of CHI-95*, pages 338–345, Denver, Colorado, 1995.

[22] W. Claassen. Generating Referring Expressions in a Multimodal Environment. In R. Dale, E. Hovy, D. Rösner, and O. Stock, editors, *Aspects of Automated Natural Language Generation: Proceedings of the 6th International Workshop on Natural Language Generation*, pages 247–262. Springer, Berlin, Heidelberg, 1992.

[23] M. Dalal, S. Feiner, K. McKeown, S. Pan, M. Zhou, T. Höllerer, J. Shaw, Y. Feng, and J. Fromer. Negotiation for Automated Generation of Temporal Multimedia Presentations. In *ACM Multimedia 96*, pages 55–64. ACM Press, 1996.

[24] R. Dale and Milosavljevic. Authoring on Demand: Natural Language Generation in Hypermedia Documents. In *Proceedings of the First Australian Document Computing Symposium (ADCS'96)*, pages 20–21, Melbourne, Australia, March 1996.

[25] R. Dale, J. Oberlander, M. Milosavljevic, and A. Knott. Integrating natural language generation and hypertext to produce dynamic documents. *Interacting with Computers*, 1998. to appear.

[26] S. Dilley, J. Bateman, U. Thiel, and A. Tissen. Integrating Natural Language Components into Graphical Discourse. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 72–79, Trento, Italy, 1992.

[27] P. Ekman and W.V. Friesen. *Facial Action Coding*. Consulting Psychologists Press Inc., 1978.

[28] C. Elliott and J. Brzezinski. Autonomous Agents as Synthetic Characters. *AI Magazine*, 19(2):13–30, 1998.

[29] M. Fasciano and G. Lapalme. PostGraphe: a System for the Generation of Statistical Graphics and Text. In *Proceedings of the 8th International Workshop on Natural Language Generation*, pages 51–60, Sussex, 1996.

[30] S.K. Feiner and K.R. McKeown. Automating the Generation of Coordinated Multimedia Explanations. *IEEE Computer*, 24(10):33–41, 1991.

[31] B. A. Goodman. Multimedia Explanations for Intelligent Training Systems. In M. Maybury, editor, *Intelligent Multimedia Interfaces*, pages 148–171. AAAI Press, 1993.

[32] W. Graf. Constraint-Based Graphical Layout of Multimodal Presentations. In M. F. Costabile, T. Catarci, and S. Levialdi, editors, *Advanced Visual Interfaces (Proceedings of AVI '92, Rome, Italy)*, pages 365–385. World Scientific Press, Singapore, 1992.

[33] J.E. Grimes. *The Thread of Discourse*. Mouton, The Hague, Paris, 1975.

[34] Y. Han and I. Zuckermann. Constraint propagation in a cooperative approach for multimodal presentation planning. In *Proc. of the 12 th ECAI*, pages 256–260, Budapest, 1996.

[35] L. Hardman, D.C.A. Bulterman, and G. van Rossum. The Amsterdam Hypermedia Model: Adding Time and Context to the Dexter Model. *Communications of the ACM*, 37(2):50–62, 1994.

[36] G. Herzog, E. André, S. Baldes, and T. Rist. Combining Alternatives in the Multimedia Presentation of Decision Support Information for Real-Time Control. In *IFIP Working Group 13.2 Conference: Designing Effective and Usable Multimedia Systems*, Stuttgart, Germany, 1998. to appear.

[37] G. Hirst. *Anaphora in Natural Language Understanding*. Springer, Berlin, Heidelberg, 1981.

[38] J. Hobbs. Why is a discourse coherent? Technical Report 176, SRI International, Menlo Park, CA, 1978.

[39] E. H. Hovy and Y. Arens. Automatic Generation of Formatted Text. In *Proc. of AAAI-91*, pages 92–97, Anaheim, CA, 1991.

[40] W. Hower and W. H. Graf. A bibliographical survey of constraint-based approaches to cad, graphics, layout, visualization, and related topics. *Knowledge-Based Systems*, 9(7):449–464, December 1996.

[41] T. Joachims, D. Freitag, and T. Mitchell. Webwatcher: A tour guide for the world wide web. In *Proc. of the 15 th IJCAI*, pages 770–775, Nagoya, Japan, 1997.

[42] H. A. Kautz and P. B. Ladkin. Integrating metric and qualitative temporal reasoning. In *Proc. of AAAI-91*, pages 241–246, 1991.

[43] S. Kerpedjiev, G. Carenini, S.F. Roth, and J.D. Moore. Integrating Planning and Task-Based Design for Multimedia Presentation. In *Proceedings of the 1997 International Conference on Intelligent User Interfaces*, pages 145–152, Orlando, Florida, 1997.

[44] A. Knott, C. Mellish, J. Oberlander, and M. O'Donnell. Sources of Flexibility in Dynamic Hypertext Generation. In *Proceedings of the 8th International Workshop on Natural Language Generation*, Sussex, 1996.

[45] J. Lester, J.L. Voerman, S.G. Towns, and C.B. Callaway. Deictic Believability: Coordinated Gesture, Locomotion, and Speech in Lifelike Pedagogical Agents. *Applied Artificial Intelligence Journal*, 1998. to appear.

[46] J.R. Levin, G.J. Anglin, and R. N. Carney. On Empirically Validating Functions of Pictures in Prose. In D.M. Willows and H. A. Houghton, editors, *The Psychology of Illustration, Basic Research*, volume 1, pages 51–85. Springer, New York, Berlin, Heidelberg, 1987.

[47] H. Lieberman. Letizia: An agent that assists web browsing. In *Proceedings of the 14th International Joint Conference on Arti ficial Intelligence*, pages 924–929, Montreal, August 1995.

[48] W. Maaß. From Vision to Multimodal Communication: Incremental Route Descriptions. *Artificial Intelligence Review, Special Volume on the Integration of Natural Language and Vision Processing*, 8(2-3):159–174, 1994.

[49] J. Mackinlay. Automating the Design of Graphical Presentations of Relational Information. *ACM Transactions on Graphics*, 5(2):110–141, April 1986.

[50] W. C. Mann and S. A. Thompson. Rhetorical Structure Theory: A Theory of Text Organization. Report ISI/RS-87-190, Univ. of Southern California, Marina del Rey, CA, 1987.

[51] M. T. Maybury. Planning Multimedia Explanations Using Communicative Acts. In *Proc. of AAAI-91*, pages 61–66, Anaheim, CA, 1991.

[52] K. R. McKeown. *Text Generation*. Cambridge University Press, Cambridge, MA, 1985.

[53] J. D. Moore and C. L. Paris. Planning Text for Advisory Dialogues. In *Proc. of the 27 th ACL*, pages 203–211, Vancouver, 1989.

[54] J. D. Moore and W. R. Swartout. Pointing: A Way Toward Explanation Dialogue. In *Proc. of AAAI-90*, pages 457–464, Boston, MA, 1990.

[55] D.B. Moran, A.J. Cheyer, L.E. Julia, D.L. Martin, and S. Park. Multimodal User Interfaces in the Open Agent Architecture. In *Proceedings of the 1997 International Conference on Intelligent User Interfaces*, pages 61–70, Orlando, Florida, 1997.

[56] K. Nagao and A. Takeuchi. Social Interaction: Multimodal Conversation with Social Agents. In *Proc. of the 32 nd ACL*, volume 1, pages 22–28, 1994.

[57] J. G. Neal and S. C. Shapiro. Intelligent Multi-Media Interface Technology. In J. W. Sullivan and S. W. Tyler, editors, *Intelligent User Interfaces*, pages 11–43. acm press, New York, NY, 1991.

[58] C. L. Paris. Generation and Explanation: Building an Explanation Facility for the Explainable Expert Systems Framework. In C. L. Paris, W. R. Swartout, and W. C. Mann, editors, *Natural Language Generation in Artificial Intelligence and Computational Linguistics*, pages 49–82. Kluwer, Boston, 1991.

[59] S. Prevost. An Information Structural Approach to Spoken Language Generation. In *Proc. of the 34 th ACL*, pages 294–301, Santa Cruz, CA, 1996.

[60] E. Reiter, C. Mellish, and J. Levine. Automatic Generation of On-Line Documentation in the IDAS Project. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 64–71, Trento, Italy, 1992.

[61] N. Reithinger. The Performance of an Incremental Generation Component for Multi-Modal Dialog Contributions. In R. Dale, E. Hovy, D. Rösner, and O. Stock, editors, *Aspects of Automated Natural Language Generation: Proceedings of the 6th International Workshop on Natural Language Generation*, pages 263–276. Springer, Berlin, Heidelberg, 1992.

[62] J. Rickel and W.L. Johnson. Animated Agents for Procedural Training in Virtual Reality: Perception, Cognition, and Motor Control. *Applied Artificial Intelligence Journal*, 1998. to appear.

[63] T. Rist, E. André, and J. Müller. Adding Animated Presentation Agents to the Interface. In J. Moore, E. Edmonds, and A. Puerta, editors, *Proceedings of the 1997 International Conference on Intelligent User Interfaces*, pages 79–86, Orlando, Florida, 1997.

[64] S. F. Roth, J. Mattis, and X. Mesnard. Graphics and Natural Language as Components of Automatic Explanation. In J. W. Sullivan and S. W. Tyler, editors, *Intelligent User Interfaces*, pages 207–239. acm press, New York, NY, 1991.

[65] O. Stock. Natural Language and Exploration of an Information Space: The ALFresco Interactive System. In *Proc. of the 12 th IJCAI*, pages 972–978, Sidney, Australia, 1991.

[66] B.A. Stone and J.C. Lester. Dynamically sequencing an animated pedagogical agent. In *Proc. of AAAI-96*, volume 1, pages 424–431, Portland, Oregon, 1996.

[67] W. Wahlster, E. André, W. Graf, and T. Rist. Designing Illustrated Texts: How Language Production is Influenced by Graphics Generation. In *Proc. of the 5 th EACL*, pages 8–14, Berlin, Germany, 1991.

[68] M. Walker, J. Cahn, and S. J. Whittaker. Improving linguistic style: Social and affective bases for agent personality. In *Proceedings of the First International Conference on Autonomous Agents*, pages 96–105, Marina del Rey, 1997. ACM Press.

[69] P. Wazinski. Generating Spatial Descriptions for Cross-Modal References. In *Proceedings of the Third Conference on Applied Natural Language Processing*, pages 56–63, Trento, Italy, 1992.

[70] M. Wilson, D. Sedlock, J.-L. Binot, and P. Falzon. An Architecture For Multimodal Dialogue. In *Proceedings of the Second Vencona Workshop for Multimodal Dialogue*, Vencona, Italy, 1992.