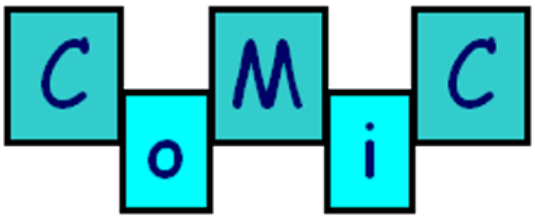
Author: Mary Ellen Foster (University of Edinburgh) Date: 27 September 2002

Deliverable 6.1
State of the art review: Multimodal fission



Document history

Date	Editor	Explanation	Status
27 September	MEF	Removed project-internal discussion; fixed typos; added subsection on research groups; reformatted slightly	Final
15 August	MEF	Content structuring section; various fixes and additions; first release to full COMIC group	Draft
1 August	MEF	More on markup and speech synthesis; comments from Edinburgh COMIC people	Draft
22 July	MEF	Initial version—imitating the formatting of Els's Word document	Draft

COMIC

Information sheet issued with Deliverable 6.1

Title:	State of the art review: Multimodal fission
Abstract:	We describe the tasks that may be performed by the fission module in a system that generates multimodal output. Broadly speaking, these tasks fall into three categories: *content selection and structuring*, *modality selection* and *output coordination*. We summarise the requirements of each of these tasks, and the approaches taken to them by a number of existing multimodal presentation systems. We also give an overview of two of the specific output techniques that will be used in COMIC: speech synthesis and embodied conversational agents. We describe several languages that have been used used to specify the content of multimodal presentations or that have been proposed as standards for this purpose. Finally, we analyse the existing work in the context of COMIC and make some recommendations.
Author(s):	Mary Ellen Foster (University of Edinburgh)
Reviewers:	
Project:	COMIC
Project number:	IST-2001-32311
Date:	27 September 2002
	For public use
Key words:	Multimodal fission, content selection and structuring, modality selection, output coordination, text-to-speech synthesis, unit selection, embodied conversational agents, multimodal markup languages
Distribution list	
COMIC partners:	Edinburgh, DFKI, KUN, MPI-N, MPI-T, Sheffield, ViSoft
External COMIC:	PO, reviewers, and general public

Contents

1 Introduction

In multimodal interactive systems, *fission* is the process of realising an abstract message through output on some combination of the available channels. Broadly speaking, the tasks of a fission module fall into three categories:

- The content to be included in the presentation must be selected and arranged into an overall structure (*content selection and structuring*).
- The particular output that is to be realised in each of the available modalities must be specified (*modality selection*[1]).
- The output on each of the channels should be coordinated so that the resulting output forms a coherent presentation (*output coordination*).

Section 2 outlines the requirements of each of these tasks, and summarises the approaches taken by a number of existing multimodal-presentation systems. Content selection and structuring is not as obviously a part of the fission process as the other two tasks. However, in many systems it is done by the same module and at the same time as the other tasks, so it is also discussed in this document.

The rest of this document deals with other areas of research that are relevant to the task of the fission module.

In Section 3, we give a brief outline of two specific output techniques that will be used in COMIC: Section 3.1 describes current techniques in speech synthesis, concentrating on unit selection, while Section 3.2 describes current work in embodied conversational agents.

In Section 4, we then describe several general-purpose and application-specific representation languages that have been used to specify the content of multimodal presentations or that are proposed as standards for this task.

Finally, in Section 5, we summarise the existing work in the context of the needs of the COMIC project, and make some recommendations for techniques and representations to use.

1.1 A note on terminology

The terms *medium* and *modality*—and the related terms *multimedia* and *multimodal*—tend to be used in different ways by different authors, some of which are contradictory. Oviatt (1999), Larson (2002), and Dahl (2002), for example, use *modality* to refer to forms of input only, while *medium* is used only for forms of output. For Wilson *et al.* (1992), on the other hand, the crucial distinction is that a *multimedia* system is one which simply uses different presentation media, while a *multimodal* system is a special type of multimedia system that is committed to a single internal representation language for the information to be presented.

Bordegoni *et al.* (1997) and Bernsen (1997), who concentrate only on the output side, use the following definitions. A *medium* is the physical realisation of a particular piece of information, or the physical device on which that information is realised. In this framework, there are three broad categories of media: graphics, acoustics, and (less frequently) haptics. A *modality* is a "mechanism of encoding information for presentation to humans or machines in a physically realised form" (Bordegoni *et al.*, 1997).

[1] This task is often called *media allocation*

In this document, we will use the word *modality* in this sense of *encoding mechanism*, and we will not use the term *medium* at all.

1.2 Systems considered

Systems that generate multimodal output have a variety of applications, and employ a variety of output modalities. Table 1 summarises the systems that were considered in the production of this document. For each system, the following information is included: its broad application domain, the modalities used in the presentations it creates, and a reference to a more detailed description of the system.

These systems were selected because there are good descriptions available of the processes they use to produce their output. For this reason, most of the systems in the table are research prototypes rather than commercial systems; the available descriptions of the latter are simply not as complete in general.

The systems that generate statistical graphics alone—APT, BOZ, and SAGE—are not truly multimodal in the sense of the preceding section. However, the choices these systems must make in building graphics from individual graphical elements are similar to those made by fully multimodal systems, and can also provide useful information.

SmartKom, MIAMM, and MagiCster are still being developed as part of ongoing projects, and there is therefore not as much information about such systems as there is about those that have been completed.

1.3 Leading research groups

Many of the systems described in Table 1 are the product of or involve some groups that have done and continue to do a great deal of research in the area of multimodal presentation systems. In particular, the following groups are involved in a number of the systems described in the table.

- The **Intelligent User Interface Lab**[2] at the German Research Center for Artificial Intelligence (DFKI) has been and continues to be involved in a number of multimodal presentation projects. (FLUIDS, MIAMM, PPP, REAL, RoCCo, SmartKom, WIP)
- The **SAGE Visualization Group**[3] at Carnegie Mellon University works in areas including information visualisation, multimedia explanations, and user interfaces. (AutoBrief, SAGE)
- The **Gesture and Narrative Language Group**[4] at the MIT Media Lab works on embodied conversational agents. (BEAT, Rea)

[2] http://www.dfki.de/iui2/
[3] http://www-2.cs.cmu.edu/Groups/sage/
[4] http://gn.www.media.mit.edu/groups/gn/

Table 1: Systems considered

System	Domain	Modalities	Reference
Adele	Education	2D graphics, avatar	(Johnson *et al.*, 2000)
AIMI	Route directions	Speech, 2D graphics	(Maybury, 1993b)
ALFresco	Information kiosk	Text, 2D graphics	(Stock, 1991)
APT	Statistical reports	Graphics	(Mackinlay, 1986)
AutoBrief	Statistical reports	Text, 2D graphics	(Kerpedjiev *et al.*, 1998)
BEAT	Animation	Avatar, speech	(Cassell *et al.*, 2001)
BOZ	Statistical reports	Graphics	(Casner, 1991)
COMET	Instruction manual	Text, 3D graphics	(Feiner and McKeown, 1991)
Cosmo	Education	3D graphics, avatar	(Johnson *et al.*, 2000)
Cuypers	Information kiosk	Speech, text, graphics	(van Ossenbruggen *et al.*, 2001)
DArt$_{bio}$	Information kiosk	Text, graphics	(Bateman *et al.*, 2001)
FLUIDS	Traffic management	Text, speech, graphics, animation	(Herzog *et al.*, 1998)
Herman the Bug	Education	3D graphics, avatar	(Johnson *et al.*, 2000)
MAGIC	Medical reports	Speech, graphics, text	(Dalal *et al.*, 1996)
MagiCster	Embodied conversational agent	Avatar, speech	(de Carolis *et al.*, 2002)
MAGPIE	Education	Graphics, text	(Han and Zukerman, 1997)
MATCH	Mobile information access	Graphics, speech	(Walker *et al.*, 2002)
MIAMM	Database access	Graphics, haptics	(Reithinger *et al.*, 2002)
PostGraphe	Statistical reports	Text, 2D graphics	(Fasciano and Lapalme, 2000)
PPP	Instruction manual	Graphics, avatar	(André *et al.*, 1999)
Rea	Embodied agent	Speech, gestures	(Cassell, 2000)
REAL	Mobile navigation	Graphics, text	(Baus *et al.*, 2002)
RoCCo	Football match reports	Speech, text, 2D graphics	(André *et al.*, 1998)
SAGE	Statistical reports	Graphics	(Roth *et al.*, 1991)
SmartKom	Information kiosk	2D graphics, avatar	(Wahlster *et al.*, 2001)
Steve	Education	3D immersive graphics, avatar	(Rickel *et al.*, 2001)
WhizLow	Education	3D Graphics, avatar	(Johnson *et al.*, 2000)
WIP	Instruction manual	Text, 3D graphics	(André *et al.*, 1993)

2 Tasks in fission

This section describes the requirements of each of the main tasks in fission, and summarises existing and possible approaches to each. Section 2.1 describes content selection and structuring, Section 2.2 deals with modality selection, while Section 2.3 covers with output coordination.

In many existing systems, more than one of these tasks is performed by the same component and at the same time; in particular, modality selection is often done in parallel with content selection and structuring. In such cases, we will discuss the particular aspects of the process that deal with each individual task in the section devoted to that task.

2.1 Content selection and structuring[5]

Content selection and structuring together constitute the task of designing the overall structure of a presentation. Since multimodal presentations generally follow structuring principles similar to those used in text, most multimodal generation systems use techniques derived from text planning.

Researchers in computational linguistics have long argued that coherent discourse has structure, and that recognising the structure is a crucial component of comprehending the discourse. Moreover, early research in language generation showed that producing natural-sounding multisentential texts required the ability to select and organise content according to rules governing discourse structure and coherence.

Although there is still considerable debate about the exact nature of discourse structure and how it is recognized, there is a growing consensus among researchers in computational linguistics that at least three types of structure are needed in computational models of discourse:

Intentional structure describes the roles that utterances play in the speaker's communicative plan to achieve desired effects on the hearer's mental state or the conversational record.

Informational structure consists of the semantic relationships between the information conveyed by successive utterances.

Attentional structure contains information about the objects, properties, relations, and discourse intentions that are most salient at any given point in the discourse.

In addition to these three types of discourse structure, two other types of structure have been discussed in the literature on discourse in computational linguistics. One of them, rhetorical structure, has had considerable impact on computational work in natural language generation.

Information structure consists of two dimensions: (1) the contrast a speaker makes between the part of an utterance that connects it to the rest of the discourse (the *theme*), and the part of an utterance that contributes new information on that theme (the *rheme*); and (2) what the speaker takes to be in contrast with things a hearer is or can be attending to.

Rhetorical structure is used by many researchers in computational linguistics to explain a wide range of discourse phenomena. There have been several proposals defining the set of *rhetorical* (or *discourse* or *coherence*) relations that can hold between adjacent discourse elements, and that have attempted to explain the inferences that arise when a particular relation holds between two discourse entities, even if that relation is not explicitly signalled in the text.

[5] Parts of this section are taken from Moore and Wiemer-Hastings (In press).

Representations that have been used for discourse structure include Discourse Representation Theory (Kamp and Reyle, 1993) and the theory of Grosz and Sidner (1986). However, the theory that has seen the most use in the context of natural language generation is Rhetorical Structure Theory (RST) (Mann and Thompson, 1988). RST characterises coherence in terms of a set of relations between spans of contiguous text in a discourse. A rhetorical relation consists of a *nucleus* and a *satellite*, which are related by a relation such as *motivation*. The nucleus of the relation is the item in the pair that is most essential to the writer's purpose; in general, the nucleus could stand on its own, but the satellite would be considered a non-sequitur without its corresponding nucleus.

In the remainder of this section, we describe approaches to content selection and structuring that have been used in multimodal presentation systems. In some cases, the content selection and structuring is done by another process or by the user; for example, the content that is to be presented in APT or MAGPIE is determined before the fission process begins. However, in other cases, selecting and structuring the content does form part of the fission process, as described below.

2.1.1 Schema-based approaches

The notion of a *schema* was first proposed by McKeown (1985) in the context of text generation. A schema encodes a standard pattern of discourse by means of rhetorical predicates that reflect the function each utterance plays in the text. Examples of such predicates are *analogy* (comparison with a familiar object), *constituency* (description of sub-parts or sub-types), and *attributive* (associating properties with an event or type). By associating each rhetorical predicate with an access function for an underlying knowledge base, these schemas can be used to guide both the selection of content and its organization into a coherent text to achieve a given communicative goal. This schema-based approach has proven successful for many text generation applications.

Multimodal presentation systems that use schemas to plan their content include COMET and PostGraphe. In COMET, the content planner uses schemas to determine the information from the underlying knowledge sources to include in its output—step-by-step explanations of technical procedures. The full content of the explanation is represented as a hierarchy of logical forms, and is then passed on to the media coordinator for realisation (Section 2.2).

PostGraphe selects graphical and textual schemas individually, based on the communicative intentions of the user and the particular characteristics of the data. The taxonomy of intentions is based on the work of Zelazny (1996) and consists of five high-level categories: *presentation*, *comparison*, *evolution*, *correlation*, and *distribution*. Intentions can also be composed to specify that the presentation should show, for example, the evolution of a comparison between datasets. Each intention may also be given a weight and a satisfaction threshold to guide the generation process.

PostGraphe's textual schemas specify both the content that should be included and how it should be structured. For example, the following rules are used to select the textual schema to accompany a presentation of the *evolution* intention:

- Choose *increase/decrease* if each value is greater/smaller than the preceding one.
- Choose *most recent tendency* if there was a reversal of tendency.
- Choose *first and last values* if there were many reversals of tendency.

The rules are tried in order until a condition is satisfied. So, for an increasing dataset, the text would be something like "Profits of company A increased between 1987 and 1990", while for a

more fluctuating dataset the caption might be “The profits of company A were 155 in 1990 and 160 in 1987”. These rules were derived from a study of a small corpus of text-graphics pairs.

The schema-based approach has several advantages. First, because it decouples discourse strategies from low-level details of knowledge representation, knowledge engineers have more flexibility to design knowledge bases to satisfy other desiderata, such as maintainability and runtime efficiency. Second, discourse strategies based on rhetorical knowledge enable systems to generate a range of different texts from the same knowledge representation.

However, a disadvantage of a purely schema-based approach is that it “compiles out” all information concerning the intended effects of the components of the presentation, as well as how those intentions are related to one another or to the informational structure of the utterances produced. This compilation renders the system incapable of responding appropriately if the perceiver does not understand or accept the utterance.

2.1.2 Plan-based approaches

To overcome the limitations inherent in schema-based approaches, researchers have applied techniques from AI planning research to the problem of constructing discourse plans that explicitly link communicative intentions with communicative actions and the information that can be used in their achievement (Moore, 1995). Text planning generally makes use of *plan operators*—discourse action descriptions that encode knowledge about the ways in which information can be combined to achieve communicative intentions. Plan operators may include the following parameters:

Effect(s) The communicative goal(s) the operator is intended to achieve.

Preconditions The conditions that must hold for an act to successfully execute. For example, it may be the case that the hearer must hold certain beliefs or have certain goals for a particular discourse strategy to be effective.

Constraints The specifications of the knowledge resources needed by the discourse strategy.

Subplan Optionally, a sequence of steps that implement the discourse strategy

Extensions of such plan-based approaches have been used in many multimodal presentation systems as well. Examples of such systems include AIMI, FLUIDS, MAGIC, MAGPIE, PPP, WIP, and AutoBrief. Generally, such a system generalises communicative acts to multimedia acts and formalises them as operators in a planning system. To plan a presentation, the system starts with a high-level communicative goal. It then uses hierarchical expansion of plan operators, terminating when all subgoals have been expanded to elementary generation tasks. These elementary tasks are then forwarded to the modality-specific generators for realisation.

Figure 1 shows a sample presentation strategy. This sample comes from PPP, and is used to satisfy the goal of showing the user the location of a hotel. In the example, *Label* represents the subgoal of indicating the location of the hotel—the planner must find a plan to satisfy this subgoal. *S-Display-Map*, on the other hand, represents an elementary presentation act (displaying the map on the screen) that requires no further refinement. The *qualitative* and *metric* fields specify temporal constraints on this operator; see Section 2.3.2 for details on the use of these constraints.

Content selection based on a user model Several recent systems have been designed to generate user-tailored recommendations or evaluations to help select among possible alternatives. For

Figure 1: Sample presentation strategy (André *et al.*, 1999)

```
(defstrategy
  :header (Describe Persona User (Location ?hotel ?location))
  :applicability-conditions (Bel Persona (Includes ?map ?location))
  :inferiors ((A1 (S-Display-Map Persona User ?map))
              (A2 (Label Persona User ?hotel ?location)))
  :qualitative ((A1 (before) A2))
  :metric ((2 <= Duration A2))
  :start A1
  :finish A2)
```

example, Carenini and Moore (2000) represent the reader's values and preferences as an additive multiattribute value function; that is, each attribute of an entity is assigned a user-dependent weight, and the overall value of that entity is the weighted sum of its attribute values. For example, features that might influence a reader's opinion of a house for sale include its location and size, and different readers may attach a different amount of importance to each feature. These preferences are then used to select the content and structure of textual arguments in the real-estate domain.

MATCH's speech planner makes use of a similar user model to choose which restaurant options to present to the user and to select the content expressed about each option. The aim of the speech-plan strategies is to filter the information presented to the user so that only options that are relevant are included. Based on the features that are important to the user—for example, price, food quality, or décor—a set of restaurants are selected and compared.

CoGenTex's Recommender[6] software also uses similar techniques to provide customised recommendations for digital cameras.

2.2 Modality selection

André (2000) sums up the task of modality selection in this way:

> *Given a set of data and a set of media, find a media combination that conveys all data effectively in a given situation.*

Arens *et al.* (1993) describe it as follows:

> *How does the producer of a presentation determine which information to allocate to which medium, and how does a perceiver recognise the function of each part as displayed in the presentation and integrate them into a coherent whole? What knowledge is used, and what processes?*

The remainder of this section examines existing approaches to modality selection. In terms of the quotation above from Arens *et al.*, Section 2.2.1 examines the *knowledge* that can be used to decide among various modalities, while Section 2.2.2 examines the *processes*.

2.2.1 Characterising the required knowledge

To perform modality selection, some or all of the following forms of knowledge may be used.

[6] http://www.cogentex.com/solutions/recommender/index.shtml

1. The characteristics of the available output modalities.
2. The characteristics of the information to be presented.
3. The communicative goals of the presenter.
4. The characteristics of the user.
5. The task to be performed by the user.
6. Any limitations on available resources.

The above list comes from André (2000), although similar lists are presented by many other authors, including Arens *et al.* (1993).

The remainder of this section outlines approaches to representing each of these forms of knowledge. The most explicit and complete characterisations come from the theoretical papers and from the systems that generate information graphics (APT, BOZ, and SAGE, for example). In the other systems, it is often difficult to separate the representation of the different types of knowledge. In many cases, for example, the characteristics of the available modalities and the information types are encoded in a single set of mappings, with neither type of knowledge explicitly represented. The communicative goals of the presenter and the user task are often particularly difficult to distinguish, so these topics are discussed together below.

Modality characteristics In most implemented systems, the available modalities are characterised in terms of either the (application-specific) types of information that they can present, or the perceptual tasks that they permit. However, Arens *et al.* (1993) and Bernsen (1997) each describe a more general-purpose set of dimensions along which modalities may be characterised.

Arens *et al.* propose that modalities can be classified using the following characteristics: the dimension of the carrier itself and of the real-world denotation of the values expressed; the temporal endurance; the granularity; the modality type (aural or visual); the "default detectability" (the intrusiveness of output on the carrier; blinking text would have high detectability); and the amount of "baggage" that must be processed in order to interpret a signal on the carrier.

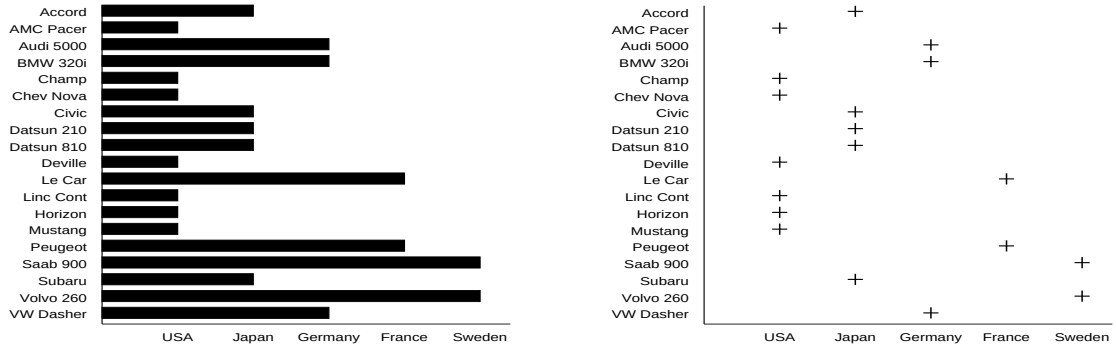
Bernsen defines a basic set of modality properties, on which he then bases a hierarchy of modalities. The properties are: linguistic *vs* non-linguistic; analogue *vs* non-analogue (i.e., how closely the representation corresponds to the represented object); arbitrary *vs* non-arbitrary; static *vs* dynamic; and the type of physical modality (graphics, acoustics, or haptics). These properties together define the top level of the modality hierarchy. Some classes are subdivided; for example, static analogue graphics may be images or graphs. The class of graphs is then further subdivided into bar graphs, line graphs, or pie graphs.

Such theoretical characterisations give a spectrum of possible classification methods. In practice, though, implementations generally use classification schemes that are more specific to the application and the domain.

In many cases, neither the modalities nor the data to be presented are explicitly characterised in isolation; rather, the information is characterised by the modalities that can present it. For example, MAGPIE and COMET tag each part of the input specification with the modality or modalities that can present it.

Many systems that generate information graphics based on statistical data use a classification of graphical presentation techniques first introduced by Mackinlay (1986) in his APT system. For

Figure 2: Examples illustrating APT's second expressiveness criterion (Mackinlay, 1986)



(a) Use of bar chart for *Nation* relation

(b) Use of plot chart for *Nation* relation

each type of data that can be presented, the presentation techniques ("graphical languages", in the terminology of APT) are ranked with respect to *expressiveness* and *effectiveness* criteria.

A set of facts is *expressible* in a graphical language if that language contains a sentence (a collection of graphical objects) that encodes (1) all of the facts in that set, and (2) only those facts. As an example of the first condition, note that a one-to-many relation such as the mapping from car prices to cars cannot be expressed using the *horizontal position* language: that language cannot represent the possibility of two cars having the same price. The second condition essentially means not to use graphical techniques that imply facts that are not part of the actual relation. For example, in addition to presenting the nationality of various car manufacturers, the graph in Figure 2(a) falsely implies that there is some sort of ordering among the car-manufacturing countries by its use of a bar chart. The presentation in Figure 2(b) avoids this implication by using a plot chart instead.

The *effectiveness* of a graphical language depends on both the properties of that language and the capabilities of the perceiver. APT uses the fact that people accomplish the perceptual tasks associated with the interpretation of graphical presentations with different degrees of accuracy. For each type of data (ordinal, quantitative, and nominal), the perceptual tasks are ranked by the accuracy that they permit. For example, *Length* is the second-highest ranked task for quantitative data, while it is much lower on the lists for ordinal and nominal data.

Besides APT, other systems that use this sort of classification include PostGraphe and BOZ. In PostGraphe, the presentation techniques are ranked for their effectiveness at conveying each communicative intention. BOZ clusters its *perceptual operators* (such as *search-object-at-horz-pos* and *determine-area-difference*) into equivalence classes, where every operator in a class can be shown to compute the same function over relational information; this corresponds to Mackinlay's expressiveness criterion. Examples of such equivalence classes are *search* and *subtraction*. The operators in each class are then ranked for effectiveness, using rankings based on those used by APT.

Data characteristics The situation with regard to data characterisation is similar to that regarding modality characterisation. That is, there have been some attempts to provide a principled representation of the information to be presented, but in most practical systems a more ad-hoc approach is generally used.

Again, Arens *et al.* (1993) provide a list of possible aspects of the data that can influence the modality-selection process. These factors include: dimensionality, transience, urgency, density, and "volume" (how much information there is to present). Several of these characteristics interact with other types of knowledge; for example, the urgency of a piece of data is determined largely by the user's task or the presenter's goals, while the amount of data that constitutes "too much" is a user-dependent feature.

It is the systems that generate information graphics that use the most complete set of properties of the data. The primary data characteristic used by APT is the *order*: whether the data is quantitative, ordinal, or nominal. APT also uses other database-style properties, such as the functional dependencies among domain sets—for example, it would make use of the fact that the *Price* relation maps from the set of *Cars* to a dollar range. BOZ also uses the order of the dataset to make its decisions. SAGE uses a similar set of dimensions to Arens *et al.*, as described by Roth and Mattis (1990). These dimensions include dimensionality, order, data domain, and relational properties such as coverage, cardinality, and uniqueness.

In PostGraphe, the input takes the form of spreadsheet-style tables. A set of properties and a unit are assigned to every variable of the input; these properties include order, domain, temporal units, numerical units, measurement type, and specific objects such as countries. In the input, the main type of each variable is specified, along with any auxiliary types. The input also specifies which variables can and cannot be used as relational keys.

As mentioned in the preceding section, many systems in non-information graphics domains characterise the information and the modalities solely in terms of each other, in a set of rules. For example, COMET divides the information that it can present into six broad categories: locations, physical attributes, simple actions, complex actions, abstract actions, and relationships between actions. This characterisation is used to select the output modality. The information itself is represented as a set of nested attribute-value pairs describing the necessary communicative acts.

MAGPIE tags each piece of information with the output modalities that can represent that information; for example, in their domain, an *Assert* can be realised only in text. Arens *et al.*-style information is also attached, but—at least in the system described by Han and Zukerman (1997)—none of this additional information is actually used.

In DArt_{bio}, modality selection is performed using a dependency-lattice representation of the information content associated with the rhetorical structure of the presentation. The dependencies in the lattice come from the domain knowledge; for example, properties of a painter such as nationality or birthdate would be represented as dependencies of the painter node. The complexity of these dependency relationships is used to select modalities as described in Section 2.2.2.

User characteristics While many multimodal presentation systems incorporate a user model, very few make use of it at the point of modality selection. Of the systems examined, only MAGPIE has a user model that guides the selection of modalities. Its model represents the user's age and literacy level; for younger viewers and those with low literacy, non-textual modalities will be preferred if possible over textual presentations.

User task and presenter goals In most multimodal presentation systems, the main goal is to present some information to the user, and the user's task is essentially to understand the information that is presented. In other words, the presentation is designed only to perform information

transfer, rather than to engage in any sort of dialogue. This means that it is often difficult to distinguish the user's task from the presenter's goals in this context.

In most systems that take into account these factors, modality selection and content selection and structuring take place at the same time; in fact, the particular content structure often determines the modalities. In AIMI, for example, the presentation operators include a specification of the modality. See Section 2.1 for more information on techniques of content selection and structuring.

Resource limitations The main form of resource limitation that can affect modality selection is the physical size of the display device. Many systems combine modality selection with physical layout in the presentation-planning process, so if the first-choice combination of modalities cannot be made to fit into the screen space available, the planner can backtrack over other possible modalities until a combination is found that can fit. See Section 2.3.1 for a full discussion of approaches to layout.

Other resource limitations may also be used—for example, if APT is informed that the output device cannot display colour, it will use alternate presentation techniques to produce its graphs. None of the other system descriptions explicitly discuss any considerations of output channels becoming unexpectedly unavailable.

Resource limitations are particularly important in systems that generate presentations designed for use on small mobile devices, such as MATCH and REAL. In the case of REAL, for example, navigational instructions produced for a handheld computer are graphically less sophisticated than those generated for use with a display that clips onto a pair of glasses; however, the handheld presentations may also contain more complex interactive areas, since interaction with the handheld is easier than with the interactive display.

2.2.2 Performing the selection

Existing systems take a variety of approaches to the task of modality selection. In systems that concentrate on other aspects of multimodal presentation, this component can be very simple. ALFresco, for example, has "a higher level, *ad hoc*, pragmatic component [that] decides how to react in the given dialogic situation, considering the type of request, the context, the model of the interest of the user, the things already shown or said to the user, and so on" (Stock, 1991). No more detail is provided about this component. Similarly, the only information about MAGIC's modality allocator is that it uses "a simple algorithm based on semantic properties alone" (Dalal *et al.*, 1996).

Other systems give more details about the approaches they take to modality selection. This section describes several typical approaches. Notice that, in many cases, the task of modality selection is combined with that of content selection and structuring (Section 2.1); the selected content structure essentially determines the presentation modalities.

Composition All of the information-graphics generation systems that use expressiveness and effectiveness criteria—i.e., APT, BOZ, and PostGraphe—also use a similar technique to produce the graphics. In each case, the outline of the procedure is as follows:

1. The components of the output specification are grouped into compatible sets. In APT, the components are individual relations to be presented; in BOZ, they are logical operators in the task specification; while in PostGraphe they are individual intentions.
2. For each grouping, the system selects the graphical presentation techniques that can express that grouping. The techniques are then ranked by their effectiveness.
3. The system tries to combine the selected graphical primitives, using predefined composition operators. If the top-choice candidates cannot be combined using the operators, the remaining candidates are tried in order until one is successful.

There are some differences between the systems that use this technique. BOZ, for example, actually chooses among perceptual operators (such as *determine-height*) rather than graphical primitives; a subsequent step then generates the actual graphs based on the operators. For example, using the *determine-height* operator for the *price* relation could lead to a bar chart in which the height of the bars corresponds to the price of the flight.

In PostGraphe, if none of the combinations of techniques can be combined into a single graph, the system has the option of creating two or more individual graphs, while APT and BOZ are designed to produce only one graph. In addition, PostGraphe produces text to accompany its graphs; however, the modality selection between graphics and text is completely predetermined.

Rules Many systems use rules to allocate the components of the presentation among the modalities. Arens *et al.* (1993), for example, describe a number of possible rules such as the following.

- If the information to be presented is urgent, and it is not yet part of a presentation instance, use a modality whose default detectability is *high*.
- If there is a large amount of information to present (volume is *much*), do not use a transient modality.

COMET's six presentation types are mapped to modality combinations by the following rules: graphics alone for locations and physical attributes, text alone for abstract actions and relations among actions, and graphics and text together for simple and complex actions. This selection is based on a series of informal experiments and a survey of the literature on modality effectiveness.

Cuypers also uses a set of application-specific design rules to determine which parts of the presentation will be produced in which modality. The input to this component is the rhetorical structure of the desired presentation. While van Ossenbruggen *et al.* (2001) do not go into any detail about the nature of these rules, they do list the types of knowledge that they employ, which include: general design knowledge; knowledge of the domain, the task, and the preferences of the end-user; and the capabilities of the device that is used to display the presentation. They also point out that the designers of a particular application will need to write their own rules; that is, the rules are specific to the application.

DArt$_{bio}$ uses the following heuristic to choose between diagrams and text: the more simultaneous dimensions of regularity are present, the more likely it is that a diagram will be chosen over text. In other words, more complex concepts will be presented using a diagram, while simple concepts will be presented in text. For example, when presenting simultaneous information about the school, profession, and time period of a set of artists, the system will choose a diagram over text. DArt$_{bio}$ can also use photographs, but these are "self-selecting"; i.e., if information is only available as a photograph, that will be used.

Plan-based approaches In the systems that use a plan-based approach to content selection and structuring (Section 2.1.2), modality selection takes place as a side effect of selecting among presentation strategies, and the necessary knowledge is encoded in the strategies themselves. For example, FLUIDS can produce four completely different types of presentations of its traffic-control data; based on user requirements, the modalities used in each of these presentation types are specified. This is reflected in the presentation strategies available to satisfy each communicative intention.

RoCCo deals with dynamic presentations; that is, commentating on a (simulated) football match as it takes place, with accompanying visualisations of the actions of the match. It therefore incorporates presentation strategies that select events to describe based on their "topicality", which decreases as the scene progresses.

AIMI also uses plan operators similar to those used by the DFKI systems. Plans with graphical components are chosen only if the information to be presented meets the applicability conditions of those plans; for example, if the information to be presented is a cartographic location. The presentation is built up hierarchically from the individual communicative acts, using a set of preference metrics such as the following to guide the selection of operators. The preference metrics specify, for example, that operators that use text and graphics should be preferred over those that use only graphics, and operators that are more common in naturally-occurring explanations should be selected rather than those that are rarer.

Competing and cooperative agents MAGPIE, which generates presentations of high-school physics concepts, uses a hierarchical system of competing and cooperative agents to plan its presentations, using a blackboard architecture. An individual agent is created to attempt to realise each piece of information in each modality attached to that information (Section 2.2.1). These agents may "hire" sub-agents to produce individual components; for example, a table agent would hire a set of table-cell agents.

When there are alternative modalities that can be used to realise a particular piece of information, all of the agents will work in parallel. The agents then compete to produce output that meets the space constraints of the output device and the other components; the output that is produced first will form part of the final presentation. If time permits, the system may attempt to improve the presentation by negotiating solutions to violations of various preferred constraints—for example, by scaling images so that all of the elements of a table column are the same size. This process is incremental and can be interrupted at any time.

2.3 Output coordination

Output coordination is the task of ensuring that the combined output from the individual generators amounts to a coherent presentation. Coordination may take several forms, depending on the particular modalities that are used and the emphasis of the presentation system. This section surveys approaches to the following three tasks:

Physical layout When more than one visually-presented modality is used—for example, graphics and text—the individual components of the presentation must be laid out.

Temporal coordination If the presentation includes dynamic modalities such as speech or animation, these presentation events must be coordinated in time. This is essentially the dynamic

analogue of the static physical-layout task, although the requirements and approaches differ somewhat.

Referring expressions Some systems further coordinate their output by producing multimodal and cross-modal referring expressions—i.e., making references using multiple modalities or referring to other parts of the presentation.

2.3.1 Physical layout

Some systems use a completely fixed layout for their presentations; examples of such systems are MAGIC and the graphics-text combination component of PostGraphe. However, most systems that must do physical layout use some form of constraint satisfaction. The constraints may come from the top-level presentation planner, or from the individual modality-specific generators.

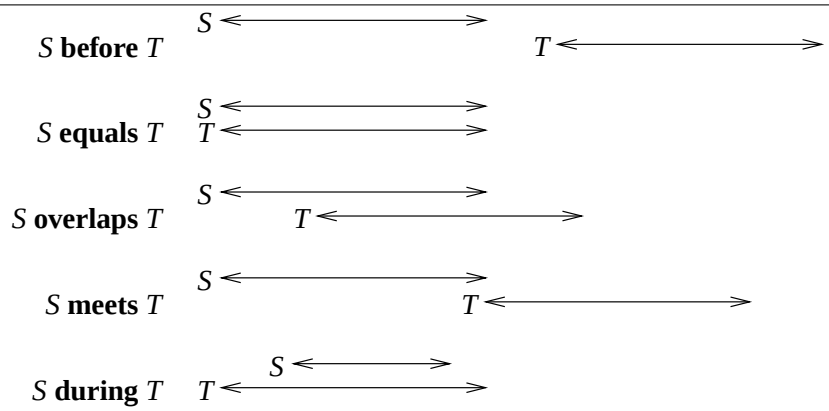
In COMET, for example, an effort is made to ensure that sentence breaks in the text correspond to "picture breaks" in the graphics; this is based on feedback from users. In general, constraints from the graphics and the text can both influence where breaks occur, and both generators could realise their content in multiple different ways. To make sure that the breaks are synchronised, each generator works with its own private instance of the logical form, incorporating information from the other generator whenever an important piece of information is under-specified. For example, if the text generator can produce either a one-sentence version of a description or a two-sentence version, and the graphics generator indicates that it cannot produce a single image to cover the content, the text generator can choose the two-sentence version.

Cuypers uses qualitative constraints to specify the layout of its presentations—for example, "caption *C* is positioned below picture *P*". These constraints are then converted into quantitative constraints (i.e., (x,y) coordinates) for the final output. Cuypers also makes use of meta-constraints to indicate, for example, that images should always appear above their corresponding captions. The qualitative constraints themselves come from a higher-level process, which uses information such as the capabilities of the output device and the cultural background of the user to decide how the presentation should be laid out.

In many systems, layout is performed in parallel with the other parts of presentation planning; this ensures that modalities that will not physically fit into the available space will not be selected. In MAGPIE, for example, the individual agents all attempt to lay out their output in the allocated space, and may request more space from other components if necessary. WIP uses a constraint-satisfaction procedure (Graf, 1997) to lay out its presentations in a grid-based system; it can also backtrack and try other modalities if the constraints cannot be met. More recent systems from DFKI use a similar approach.

DArt$_{bio}$ lays out its automatically-generated text by using the underlying communicative structure to guide the layout decisions. The layout rules were derived from the analysis of a corpus of documents and their layout. Based on the results of this corpus analysis, Bateman *et al.* then implemented a system to do automated page layout based on the rhetorical structure of generated texts. Their system works on the rhetorical structure tree of the text in a recursive, top-down manner; its main decision is whether to break rhetorical links by putting some of the content into different layout units. This process interacts with modality selection by forcing a partition whenever there is a difference in modality—for example, if one segment of a partition is to be realised as a photograph and another as a text block, they cannot be part of the same layout unit.

Table 2: Relations in Allen's interval temporal logic



2.3.2 Temporal coordination

In a presentation incorporating dynamic modalities, such as speech or animation, an important part of producing coherent output is ensuring that the various parts of the presentation occur at the right time with respect to one another.

The following are the typical phases in the synchronisation process (André, 2000):

1. The temporal behaviour of the presentation is specified at a high level.
2. A partial schedule is computed, specifying as much temporal information as possible.
3. The partial schedule is used at run-time to produce the actual presentation. Depending on the behaviour of unpredictable components, the schedule may be refined or altered during the course of the presentation.

Temporal coordination often amounts to making sure that the other components of the presentation take place at the right time with respect to the speech output; that is, it is the speech that largely determines the temporal behaviour of the presentation.

Approaches to each of the above tasks are outlined in the following sections. The papers about PPP, MAGIC, Cuypers, and BEAT give particularly good descriptions of their approaches to synchronising media objects, so this section will concentrate on those systems.

Specifying temporal constraints The standard representation scheme for temporal constraints is the interval temporal logic described by Allen (1983). This logic can represent the five basic categories of relations between time intervals S and T illustrated in Table 2.

PPP specifies both cross-modal temporal constraints and the duration of media objects using this logic. The primitives in PPP's representations are media objects, as in the following examples:

- $(Speak_1$ **during** $Point_2)$—the speaking act $Speak_1$ occurs during the pointing act $Point_2$.
- $(5 \leq$ **duration** $Point_2)$—the pointing act $Point_2$ takes at least 5 seconds to complete.

Figure 1 on page 7 shows a sample PPP plan operator incorporating such temporal constraints.

MAGIC also uses Allen-style temporal constraints to represent the preferred orderings within each of its modality-specific components. However, the primitives in MAGIC's temporal expressions represent individual facts that can be expressed about a patient, rather than presentation objects as in PPP. In the context of MAGIC, there is a one-to-one mapping between presentation events and facts. The following are examples of temporal constraints in MAGIC:

- $(<$ *name* $(*$ *age gender*$))$—the name should be spoken first, while the age and gender can be spoken in either order.
- $($*age* $($*start* 1.2$)$ $($*stop* 2.7$))$—the act of uttering the patient's age takes place between 1.2 and 2.7 seconds after the presentation starts.

The temporal constraints in Cuypers are represented and used in an identical manner to the physical layout constraints, as described in the preceding section. The temporal constraints also use Allen's temporal logic.

In the case of BEAT, which is used to specify embodied conversational agents (Section 3.2), all visual events (lip movements and gestures) are created and scheduled based on the syntactic and semantic features of the speech. This means that all temporal constraints take the form of mappings between words and actions, which are then instantiated as described in the following section once the word timings are available from the speech synthesiser.

Creating a schedule As is the case in physical layout, most systems use some form of constraint processing to create the schedule for temporally coordinated presentations.

In MAGIC, both the language and the graphics components produce a weighted list of possible partial orders of media actions, using Allen-style constraints as described in the preceding section. A negotiation process then combines these constraints into an acceptable total ordering of actions, backtracking as necessary through successively lower-ranked alternatives until a compatible ordering is produced. The duration of each of the speech actions is then computed in order to come up with the final schedule.

PPP uses a temporal planner to produce its presentation schedules. The constraints are used during the the presentation-planning process, so if a set of temporal constraints cannot be met, the planner can try different strategies. After the presentation is designed, the concrete schedule is produced from the resulting plan, using constraint propagation. The schedule indicates the starting time and duration of each of the presentation acts. For unpredictable acts, the system will indicate an interval within which they may start or end instead of prescribing an exact time point.

In Cuypers, the schedule is created using the same constraint-satisfaction process as the physical layout. The qualitative constraints are mapped to quantitative constraints; for example, the constraint $(S$ **meets** $T)$ would become $X_2^S = X_1^T$ (i.e., the finishing time of S is the same as the starting time of T). These constraints are then solved by an off-the-shelf constraint solver.

To coordinate BEAT's character animation with speech, there are two main approaches, depending on the characteristics of the speech synthesiser. With synthesisers based on recorded audio—such as Festival—the system must obtain an estimate of word and phoneme timings and construct the animation schedule prior to execution. With systems that produce real-time events as the audio is produced, the animation system must compile a set of event-triggered rules to govern the behaviour generation. Cassell *et al.* (2001) describe only the former approach, although their system is in principle capable of using either.

The first step in creating a BEAT animation schedule is to send the text specification to the synthesiser with a request for word and phoneme timings. The TTS engine runs as a separate process, so scheduling continues while the timings are being computed. Next, the non-verbal behaviour suggestions are translated into an intermediate form of animation command and placed in a linear order. Once the timings become available, the schedule is instantiated by mapping the word indices to execution times. At this point, facial animation commands to synchronise the lips with the phonemes are also added to produce the final schedule.

Implementing the schedule at run-time There may be events in the presentation for which the duration cannot be completely determined until run-time, or it may be that some event takes longer than anticipated. In such cases, the schedule must be adapted in order to deal with the actual durations of the events. However, not all systems take such unexpected events into account.

Cuypers marks up its final presentation using SMIL (Section 4.1.1). It is then sent to a SMIL player such as RealOne for realisation. The schedule is thus essentially fixed at plan time, with no adaptation other than what the SMIL player can do.

MAGIC does not take into account unpredictable events in its execution of the schedule. Instead, the complete schedule is sent to each of the generators, which then receive a "start" signal from the media conductor so that the start times are synchronised. Since the schedule is complete, this should mean that the whole presentation remains synchronised, but it cannot deal with operations that take more or less time than predicted.

The temporal behaviour of PPP's presentations is controlled by a *presentation clock*. In the implementation described by André *et al.* (1998), the system always chooses the earliest possible starting point for an action whenever there is a choice. The system may dynamically shorten or lengthen the duration of presentation acts in response to feedback from the system. The system results are added to the temporal constraint network so that, for example, if it takes longer than expected to generate a graphic, the persona will wait for it to be completed before talking about it. In other words, the presentation plan is modified at run-time to account for such unpredictable behaviour.

BEAT's schedule is implemented by compiling the abstract animation schedule into a set of legal commands for the animation subsystem that is used. This process also deals with enabling and disabling animation subsystem features; gesture approach, duration, and relax times (since the abstract schedule specifies only peaks); and any time offsets between the speech-production and animation subsystems.

2.3.3 Referring expressions

There are two basic types of referring expressions that are specific to the multimodal context:

Multimodal referring expressions refer to world objects using a combination of two or more modalities—pointing to, looking at, or highlighting the object that is mentioned in language. For example, MAGIC highlights each part of the patient record (e.g., age) as it speaks that information.

Cross-modal referring expressions refer, not to world objects, but to presentation objects in other modalities. An example of such a referring expression is "The upper left corner of the image".

The representations and requirements for the two types of referring expressions are different. For a multimodal referring expression, there must be some form of communication among the multiple modalities so that the references occur at the same time or in the same location; this interacts with the other coordinating processes described in the preceding sections. For cross-modal referring expressions, the referring modality must have access to some part of the internal representation of the referred-to modality, so that appropriate expressions can be generated.

Many multimodal presentation systems simply do not deal with this type of referring expression. PostGraphe, in particular, explicitly does not do any coordination between its text and graphics generation; instead, both are generated independently from the same underlying representation. Fasciano and Lapalme (2000) argue that, since in their domain, text generally just reinforces what already appears in the graphic, there is no need for explicit coordination at the generation stage.

Animated agents that interact with their environment often make use of cross-modal and multimodal referring expressions. According to Johnson *et al.* (2000), "relatively simple representations of spatial knowledge" have been sufficient to allow agents to move around in and interact with their environment. For example, Cosmo and PPP maintain a representation of the location of objects within their two-dimensional world so that they can point to them and make appropriate referring expressions. Steve, which inhabits a three-dimensional virtual world, relies on the virtual reality software to provide boundaries for objects.

To produce its cross-references, COMET's graphics generator constructs a representation of each illustration it generates; this representation includes information such as the objects depicted and any graphical effects used (such as highlighting). The text generator then queries this representation in order to generate cross-references.

3 Specific generation techniques

This section provides an overview of recent results in two of the specific output modalities that will be used in the COMIC system. Section 3.1 provides an overview of speech synthesis, while Section 3.2 surveys recent work in the area of embodied conversational agents.

3.1 Speech synthesis

This section describes recent results in speech synthesis, particularly in text-to-speech systems that use unit selection. Section 3.1.1 describes the techniques involved in unit selection, while Section 3.1.2 outlines the sort of information other than pure text that can be sent to a text-to-speech system.

3.1.1 Unit selection[7]

The most successful recent speech-synthesis systems—i.e., those that produce the most natural-sounding output—use a technique called *unit selection* to produce their output. Unit selection consists of searching through a large speech database at runtime with the aim of finding the best recorded units to produce the desired speech.

Unit selection falls into the category of speech synthesis known as *concatenative synthesis*—that is, rather than building up the speech signal from first principles (e.g., by using a biomechanical model of the human speech system), it uses actual snippets of recorded speech. The elementary speech segments (units) may be, for example, phones, "diphones" (phone-to-phone transitions), or syllables. Concatenative synthesis selects units from a voice database and strings them together to produce the speech signal. Since concatenative synthesis uses actual recorded speech, it has the highest potential for natural-sounding results.

In the past, most concatenative speech synthesis systems have been based on diphones. For American English, a diphone-based concatenative synthesiser normally has at least 1000 units. These diphones are usually obtained from a specific speaker, reading either a series of nonsense words, or sentences that are rich in diphones; in either case, the recordings are done in a neutral voice. In synthesis, one diphone is then used in all possible contexts. However, the output of such a system tends to sound monotonous and unnatural.

Unit-selection systems, in contrast to such previous systems, choose in real time from a database that may include thousands of examples of a specific diphone, recorded in a more natural style. This increase in coverage has been made possible by advances in computing power and storage, search algorithms, and automatic annotation tools for voice corpora. The optimal choice of units depends on factors including the *join cost* (spectral similarity at unit boundaries) and the *target cost* (how well the unit matches the prosodic targets—see Section 3.1.2). The method of unit-selection synthesis is based on the work of Hunt and Black (1996).

Unit-selection synthesis can produce good-quality output for two main reasons. First, choosing speech segments online allows for long fragments of speech (possibly as long as phrases or sentences) to be used if they are found in the database; this will generally be better than a combination

[7] This section is based on Schroeder (2001).

of shorter fragments. Second, when there are multiple instantiations of a unit in the inventory, there is less need for any prosodic modifications that might decrease naturalness.

To make unit-selection schemes work in real time, various techniques are used to decrease the amount of processing needed, by pre-computing and caching representative sets of target and join costs.

Systems that use unit selection to generate speech include Festival,[8] rVoice,[9] and AT&T Natural-Voices.[10]

3.1.2 Markup for text-to-speech synthesis

In unit-selection systems, information about the prosodic requirements of the text to be generated is used to control the units that are selected. This prosodic information can be inferred by parsing and otherwise analysing the input text, as in BEAT. However, if the text is generated by another system, that system itself can often provide some information to the TTS component to help it in selecting the correct prosody.

Various markup schemes have been proposed to help in this task, several of which are described more fully in Section 4. Speech Synthesis Markup Language (SSML, Section 4.1.2), for example, allows parts of the text to be annotated with semantic tags (e.g., *acronym* or *number*), pronunciation guides, and prosodic features (e.g., emphasis and speech rate). The input to the Festival synthesiser can include markup in an XML-based language called Sable, which is an ancestor of SSML.

In the context of the MagiCster project, which aims to generate a speaking, lifelike avatar, the agent's behaviour as a whole is annotated using APML (Section 4.2.3). The speech to be generated is annotated using a scheme derived from Sable, with some additions to represent pitch accents. The following section gives more details about such embodied agents.

3.2 Embodied conversational agents[11]

Embodied conversational agents, or "virtual humans", are animated characters capable of engaging in conversation and collaborative tasks. Applications of such agents include education and training, therapy, marketing, and entertainment.

The research involved in building a virtual human covers many diverse disciplines: reasoning and planning to allow the agents to act in their environments, natural language processing to allow them to carry on a conversation, computer graphics and animation to provide real-time controllable bodies, and psychology and communication theory to produce realistic behaviour.

Cassell *et al.* (2000b) provide a summary of the work in this area. Two recent workshops also have summaries of recent work: Gratch and Rickel (2002) and Marriott *et al.* (2002). The results of the former are summarised by Gratch *et al.* (2002), on which this section is based.

Sections 3.2.1–3.2.3 discuss three key areas identified by Gratch *et al.* that must be addressed in creating virtual humans: face-to-face conversation, emotions and personality, and human figure

[8] http://www.cstr.ed.ac.uk/projects/festival/
[9] http://www.rhetoricalsystems.com/rvoice.html
[10] http://www.naturalvoices.att.com/
[11] This section is based on Gratch *et al.* (2002).

animation. Section 3.2.4 then outlines the issues involved in integrating work from those three areas.

3.2.1 Face-to-face conversation

Face-to-face conversation in humans includes both verbal and nonverbal behaviour, and each of these behaviours can influence the interpretation of the other. Coordinating the generation of verbal and nonverbal behaviour requires an understanding of how the various behaviours create meanings together, how they co-occur in conversation, and what kinds of goals are achieved by the different channels.

The interaction between verbal and nonverbal behaviour is similar to the interactions between the modalities in any multimodal presentation system, and the choices that must be made are similar to the tasks involved in fission described in Section 2. While speech and nonverbal behaviours do not always manifest the same information, what they convey is usually compatible. In some cases, different modalities reinforce one another through redundancy, while in other cases the attributes of the message are distributed across the modalities.

There is a tight synchrony among conversational modalities in human conversations; for example, people tend to accentuate important words by speaking more forcefully, illustrating their point with a gesture, and turning their eyes towards the listener at the end of a thought. Listeners, on the other hand, tend to nod within a few hundred milliseconds of the speaker's gaze shifting. When this synchrony is destroyed, satisfaction with a conversation diminishes.

The different conversational modalities tend to be used to achieve different types of goals. Speakers in natural conversations tend to use gestures to accomplish *propositional* goals (advancing the content of the conversation); for example, a speaker may make a "walking" gesture with the fingers when talking about a walk. Conversely, speakers tend to use eye movements to accomplish *interactional* goals (advancing the conversation process); for example, giving up the conversational turn by looking away. In order to take this into account, computational architectures must represent both the propositional and interactional goals.

The following are the required features proposed by Cassell *et al.* (2000a) for the control architecture of a virtual human:

- The system should be able to send and receive information in any of the conversational modalities.
- It should be able to respond to feedback and turn requests at any time, and should allow different processes to concentrate on activities on different time scales.
- It should deal with both propositional and interactional information. For propositional interaction, it should include a static domain knowledge base, a model of the user's needs and knowledge, and a module for planning the content of presentations. For interactional information, it should maintain a model of the current state of the conversation.
- It should represent conversational functions such as *initiating a conversation* or *giving up the floor*, rather than behaviours such as looking at another person or putting the hands in the lap. This provides modularity and a principled way of combining different modalities.

Synchronisation between modalities is vital in the output. Even if the temporal associations are produced as part of the behaviour-generation process, it is still vital to maintain this synchroni-

sation when the presentation is actually produced. There are two possible mechanisms for this: event-based and time-based. In an event-based system, the speech synthesiser sends events on phoneme and word boundaries; this is usually used for lip synchronisation, but can also be used for other purposes. However, this provides no time for behaviour preparation. In a time-based system, the synthesiser provides exact start times for each word prior to playback; in this way, the behaviours can be scheduled beforehand and played back along with the voice. There is a description of how the Behavior Expression Animation Toolkit (BEAT) (Cassell *et al.*, 2001) goes about this task in Section 2.3.2.

3.2.2 Emotions and personality

Modelling the sorts of emotion and personality that people use in their verbal and nonverbal communication is crucial to building believable virtual humans. The computational approaches that have been used for this task divide into two categories: communication-driven approaches and simulation-based approaches. Communication-driven approaches aim to convey emotions based on the desired impact on the user, while simulation-based approaches aim to simulate "true" emotion.

An example of communication-driven emotion is MagiCster (de Carolis *et al.*, 2002). The agent uses facial expressions to convey affect in combination with other communicative functions. The agent deliberately decides whether to convey a certain emotion.

Simulation-based approaches generally build on *appraisal* theories of emotion, which view emotions as arising from a reaction to events and objects in the light of agent goals, standards, and attitudes. For example, the response of an agent to the winning move in a game should depend on which team the agent prefers.

In addition to generating affective states, the agent must express them in a way that the user can understand—using body gestures, acoustic features, or facial expressions, for example. Bayesian networks are often used to model the relationship between emotion and behaviour, and to resolve conflicts that arise when different communicative functions must be shown on different channels of the face.

More progress has been made on the visual aspects of conveying emotion than on the auditory aspects. Unit-selection schemes, as described in Section 3.1.1, can be used to provide the required prosody, but that prosody must still be derived somehow from the emotions. This is one of the goals of the MagiCster project.

3.2.3 Human figure animation

Most production computer-animation work relies on human animators to design or script movements or to direct the performers in motion capture. However, in the context of a interactive or conversational agent, the system must create novel, contextually appropriate behaviours in real time and cannot rely on an animator. Historically, research in human figure animation has divided into two areas: animation of complete body movements, and animation of the face.

Body animation There are two basic ways that body animation can be used in a real-time setting. The system can use motion capture and other techniques to modify movements to immediate

needs, or it can exert direct control over important movement parameters. Animating a human body requires more than just controlling skeletal motions; truly human qualities arise from intelligent movement strategies, soft deformable surfaces, and clothing.

Implementing an animated human body is complicated because there are not many generally available tools. Body models tend to be either proprietary, optimised for real time (and thus limited in body structure and features), or constructed for particular animations using standard animator tools. Current efforts on the H-Anim project[12] promise to enable model sharing and testing.

Facial animation Facial animation methods fall into three main categories. The earliest method uses manually generated keyframes—images illustrating the extreme of an action or a gesture—and then automatically interpolates frames between the keyframes. This approach provides complete artistic control and is often used in professional animation, but it can be time-consuming.

Another method is to synthesise facial movements based on text or speech. A text-to-speech algorithm or speech recogniser provides a sequence of phonemes, which are then mapped to visemes (visual phonemes). These visemes then drive an articulation model that animates the face. The state of the art using this method provides understandable, but not convincing, facial animation.

The third method, and the most recent one, is to measure human facial movements directly and apply the motion data to a face model. The modelling techniques used with this model range from two-dimensional line drawings to physics-based three-dimensional models with muscles, skin, and bone.

One standard for human figure animation is the MPEG-4 Face and Body Animation standard, which provides anatomically specific locations and animation parameters. It defines Face Definition Parameter feature points and locates them on the face; these points are then displaced by Facial Animation Parameters. Some FAPs are descriptors for visemes and emotional expressions, and new FAPs can be specified by blending two predefined ones.

3.2.4 Integration

When integrating all of the techniques described previously into a complete virtual human, two issues in particular are important: consistency and timing. These issues mirror the considerations for any multimodal presentation as discussed in Section 2.

Consistency When combining a variety of behavioural components, it is crucial that the consistency is maintained between the agent's internal state and its outward behaviour. Just as the intentions should drive all components of a multimodal presentation, as described in Section 5.1.2, it is equally important that the various agent behaviours are consistent with one another. When the output on different channels conflicts, the agent may appear clumsy, insincere, or simply fake. For example, arm gestures without facial expressions look odd, while arm gestures without torso involvement look insincere.

Timing Most existing work has focused on a specific aspect of behaviour, leading to architectures that are tuned to a subset of timing constraints. For example, BEAT's body movements must

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

conform to the timings from the speech synthesiser; other systems that concentrate on emotion often manipulate the length of gestures for emotional effect; while in systems that concentrate on reacting to the environment, behaviour is subordinate to the environmental dynamics.

Approaches to reconciling such different approaches include sharing information among different processes in the “pipeline”, and specifying all of the constraints explicitly and devising an animation system flexible enough to handle them all.

4 Representations for multimodal output

Several representation languages have been proposed for specifying multimodal presentations. This section describes a number of such languages. First, Section 4.1 describes a number of general-purpose, domain-independent language proposals. Then, in Section 4.2, we describe application-specific languages that have been used in several projects.

4.1 General-purpose languages

The W3C Multimodal Interaction group,[13] which was formed in February 2002, has as its goal the development of markup specifications for synchronisation across multiple modalities and devices with a wide range of capabilities. The group's work will build on existing W3C specifications such as XHTML, SMIL, XForms, markup for speech synthesis and recognition, and VoiceXML. Currently (summer 2002), the group is in the process of documenting the requirements for multimodal markup languages, with a goal of publishing the document by September. Another goal is to publish a first Working Draft of the architecture and events that will need to be supported, by the end of 2002. Dahl (2002) provides an overview of the W3C's activities in this area.

Another proposed standard for multimodal markup is the Speech Application Language Tags specification, which comes from the SALT Forum,[14] an industrial coalition that aims to develop a standard for multimodal interaction with computer applications.

A SIGSEM working group on the Representation of Multimodal Semantic Information[15] has recently been formed and will hold its first meeting in January 2003. This group aims to examine the representation of multimodal input and output from a semantic perspective. Bunt and Romary (2002) discuss the issues that this group plans to address.

Some standards from the Motion Picture Experts Group[16] (MPEG) are also relevant to the task of multimodal generation; namely, MPEG-4 (which describes the content of audiovisual presentations), MPEG-7 (which provides a mechanism to add descriptive elements to multimodal content), and MPEG-21 (a proposed standard framework for multimodal interaction).

The remainder of this section describes each of these languages in more detail, with examples where possible.

4.1.1 SMIL

SMIL—Synchronised Multimedia Integration Language (W3C, 2001)—is a meta-language for multimodal presentations. The goal of SMIL is to be an integration format for presentable single-modality formats; that is, to allow authors to specify what should be presented when.

SMIL is a dialect of XML, and is defined with an XML DTD and Schema. All of the media elements of the presentation are referenced from within the SMIL file, in a manner similar to that in which images and applets are included within HTML documents. SMIL allows the integration of images, audio and video clips, animations, and formatted text.

[13] http://www.w3.org/2002/mmi/
[14] http://www.saltforum.org/
[15] http://www.sigsem.org/mmsemrep-web.txt
[16] http://mpeg.telecomitalialab.com/

Figure 3: Sample of SMIL (le Hégaret, 1999)

```
<smil>
    <head>
        <layout>
            <root-layout width="640" height="480" background-color="blue"/>
            <region id="video1" top="50" left="50"/>
            <region id="video2" top="50" left="210"/>
        </layout>
    </head>
    <body>
        <par>
            <img src="background"/>
            <video region="video2" src="video-joe"/>
            <seq>
                <img region="video1" src="web-growth" dur="1min"/>
                <video region="video1" src="tim-video"/>
            </seq>
            <seq>
                <audio src="audio-joe"/>
                <audio src="audio-tim"/>
            </seq>
        </par>
    </body>
</smil>
```

SMIL supports spatial and temporal layout of the presentations it specifies. Spatial layout is described by putting each element of the presentation into a defined "region" on the screen. For temporal layout, various properties can be specified, including the following:

Duration All objects have an intrinsic duration; for discrete media (such as text or images), the intrinsic duration is zero. Explicit durations may also be specified.

Repeat Objects can be specified to repeat a specified number of times, or indefinitely (until the parent stops).

Synchronisation The `<par>` and `<seq>` tags group elements. Elements grouped by `<par>` execute in parallel; elements grouped by `<seq>` execute sequentially. These tags may be nested, and elements may be set to delay execution until a time relative to another element.

SMIL also allows for conditional selection of elements based on, for example, the speed of the user's computer or the size of the screen. The `<switch>` tag allows choosing among different possibilities; the first acceptable element in the list is played.

Several commercial players are available that understand SMIL; for example, the GRiNS player,[17] Internet Explorer 6.0,[18] and RealOne[19] can all understand SMIL 2.0 markup. There are also a number of players for SMIL 1.0, as well as a variety of authoring tools. SMIL is also used to represent the presentations generated by at least one multimodal presentation system: namely, Cuypers (van Ossenbruggen *et al.*, 2001).

Figure 3 shows an example of SMIL markup. This sample represents a video presentation with a

[17] http://www.oratrix.com/GRiNS/SMIL-2.0/
[18] http://www.microsoft.com/windows/ie/preview/default.asp
[19] http://www.realnetworks.com/solutions/ecosystem/realone.html?src=rnhmfs

Figure 4: Sample of SSML (W3C, 2002a)

```
<?xml version="1.0"?>
<speak xmlns="http://www.w3.org/2001/10/synthesis" xml:lang="en">
    <paragraph>
        <sentence>You have 4 new messages.</sentence>
        <sentence>The first is from
            <say-as type="name">
                Stephanie Williams
            </say-as>
            and arrived at <break/>
            <say-as type="time">3:45pm</say-as>.
        </sentence>
        <sentence>
            The subject is <prosody rate="-20%">ski trip</prosody>
        </sentence>
    </paragraph>
</speak>
```

background image and several video and audio streams that play in parallel and in sequence with one another.

4.1.2 SSML

SSML—Speech Synthesis Markup Language (W3C, 2002a)—is designed to provide a rich, XML-based markup language for assisting the generation of synthetic speech. It allows the following types of markup on the input to a text-to-speech system (see Section 3.1 for more information on text-to-speech synthesis):

Structure analysis The `<paragraph>` and `<sentence>` elements indicate the structure of the document.

Text normalisation The `<say-as>` element can be used to indicate that a particular piece of the document should be spoken as a date, number, acronym, currency amount, or other special type of data.

Text-to-phoneme conversion Particular words can be marked up with `<phoneme>` tags to indicate the pronunciation, if it is likely to prove difficult.

Prosody analysis The `<emphasis>`, `<break>`, and `<prosody>` elements can all give hints as to the desired prosodic characteristics of the speech.

Waveform production The `<voice>` tag can select among different voices if the synthesiser supports it, while the `<audio>` tag can insert recorded audio data into the output.

SSML can be integrated with SMIL by inserting links to SSML content into a SMIL document.

Figure 4 shows an example of SSML markup. This example represents part of an email-reading application, and specifies that the system should summarise the user's inbox and then read the headers of the first message.

Figure 5: Sample of VoiceXML (W3C, 2002b)

```
<?xml version="1.0"?>
<vxml version="2.0" xmlns="http://www.w3.org/2001/vxml">
    <form>
        <field name="drink">
            <prompt>Would you like coffee,tea, milk, or nothing?</prompt>
            <grammar src="drink.grxml" type="application/srgs+xml"/>
        </field>
        <block>
            <submit next="http://www.drink.example.com/drink2.asp"/>
        </block>
    </form>
</vxml>
```

4.1.3 VoiceXML

VoiceXML (W3C, 2002b) is designed for creating audio dialogues including synthesised speech, digitised audio, recognition of spoken and DTMF key ("touch-tone") input, recording of spoken input, telephony, and mixed-initiative conversations. It is especially useful for telephone-based applications. VoiceXML provides a form-filling mechanism for handling normal input, as well as a mechanism for handling events (such as help requests) not covered by the form mechanism.

A VoiceXML document or set of documents forms a hierarchical conversational finite-state machine. The user's conversational state is called a *dialog*. There are two types of dialog: *forms*, which collect values for variables, and *menus*, which present the user with a choice of options. Each dialog specifies the next dialog in the sequence; if there is no successor, the execution of the document stops. The system may also specify *subdialogs*, which provide a mechanism for invoking a new interaction and returning to the original form. Subdialogs can be used, for example, to implement a confirmation sequence and or to create a reusable library of dialogs to be shared among documents or applications.

VoiceXML uses speech markup elements from SSML to describe its speech output. It can also specify whether barge-in (interrupting the prompt) is supported within each of its prompts, and what action should be taken if the user does interrupt the prompt.

Figure 5 shows an example of VoiceXML markup. The example asks the user for a choice of drink and then submits the user's response to a server script (*drink2.asp*).

4.1.4 SALT

SALT—Speech Application Language Tags (SALT Forum, 2002)—is an extension of HTML and other markup languages that adds a speech and telephony interface to web applications and services. SALT is a small collection of XML elements that can be used to add a speech interface to a source document. The main elements of SALT are:

`<prompt>` Specifies the content of audio output. The content of a prompt may include inline text, the value of variables, or links link to audio files. Inline text may be marked up with SSML (Section 4.1.2).

`<listen>` Used for speech recognition and audio recording. For speech recognition, a grammar can be specified, using a representation such as the W3C's Speech Recognition Grammar

Figure 6: Sample of SALT (SALT Forum, 2002)

```
<!-- HTML -->
<html xmlns:salt="http://www.saltforum.org/2002/SALT">
    [ ... ]
    <input name="txtBoxCity" type="text" />
    <input name="buttonCityListen" type="button" onClick="listenCity.Start();" />
    [ ... ]

    <!-- SALT -->
    <salt:listen id="listenCity">
        <salt:grammar name="g_city" src="./city.grxml" />
        <salt:bind targetelement="txtBoxCity" value="//city" />
    </salt:listen>
</html>
```

Specification format.

<dtmf> Used in telephony applications to specify DTMF inputs—i.e., telephone touch-tones. Its properties are similar to those of the <listen> element.

<smex> Simple Messaging EXtension—used to communicate with the external component of the SALT platform.

SALT makes use of the DOM (Domain Object Model), the standard interface to the contents of a web page, if it is run in a DOM-capable browser. This means that client-side scripting languages can access and manipulate elements of the SALT markup on a page, in the same way that scripting languages currently access HTML.

SALT elements can be hosted in a SMIL 2.0-compliant environment. In this case, the SMIL timing controls are mapped to the starting and stopping times of the SALT <listen> and <prompt> elements.

Figure 6 gives a sample of SALT. This example shows an HTML page with a button that, when clicked, activates a grammar relevant to an adjacent input field and binds the recognition result into that field.

SALT and VoiceXML The closest W3C counterpart to SALT is VoiceXML. According to Potter and Larson (2002) and Larson (2002), the main difference between the two is that VoiceXML is specifically designed for telephony applications, while SALT targets speech applications across a whole spectrum of devices. Also, SALT concentrates only on the speech interface, while VoiceXML also deals with issues of data and control flow. The basic building blocks in VoiceXML are dialogue turns, while SALT deals with lower-level tags. Both languages use existing W3C standards, such as SSML for output and SRGS (Speech Recognition Grammar Specification) for input.

4.1.5 Relevant MPEG standards

Koenen (2001) provides an overview of the standards from the Moving Picture Experts Group (MPEG). The following are the MPEG standards that are relevant to the task of generating multimodal output.

MPEG-4 (Koenen, 2002) provides a standardised way to describe an audiovisual scene. It represents media objects (still images, video and audio clips) in a hierarchy; the objects can then be composed to produce a scene. MPEG-4 also provides a set of face and body animation tools, which are used by several embodied conversational agents (Section 3.2).

MPEG-7 (Martinez, 2002) complements MPEG-4; where MPEG-4 defines how to represent content, MPEG-7 specifies how to describe it. It provides a mechanism to add descriptive elements to multimodal content; the elements may range from low-level signal features like colour and sound to high-level structural information.

MPEG-21 (Bormans and Hill, 2002) is a proposed framework for multimedia interaction. The goal of MPEG-21 is to describe how different elements build an infrastructure for the delivery and consumption of related multimedia content. It will provide a mechanism for declaring, identifying, and describing "digital items"—structured digital objects incorporating resources (e.g., video or audio tracks), metadata (e.g., MPEG-7 descriptors), and structure (relationships among resources).

Since none of these standards is text-based, samples of the representation cannot be provided.

4.2 Application-specific languages

The preceding section described some emerging standard languages for various types of multimodal presentations. In this section, we summarise some specialised languages that have been used to mark up the content of presentations in the context of specific projects.

4.2.1 M3L

M3L—MultiModal Markup Language (Wahlster *et al.*, 2001)—is the XML-based markup language used to represent all of the information that flows between the processing components of SmartKom. M3L is designed for the representation and exchange of complex multimodal content, of information about segmentation and synchronisation, and of information about the confidence in processing results.

Figure 7 shows a sample of M3L markup, taken from Wahlster *et al.* (2001). The first component of the sample describes a movie theatre called "Europa"; the second describes a display element on the screen corresponding to that theatre. The link between the two is made by means of the *structId* and *contentRef* components.

4.2.2 MMIL

MMIL—Multi-Modal Interface Language (Reithinger *et al.*, 2002)—is the representation language being developed as a part of the MIAMM project, which aims to develop fast and natural access methods for multimedia databases incorporating haptic media. MMIL will act as the unified representation format for all processing in the system. The following are the three basic requirements for the MMIL language:

1. It should be flexible enough to take into account all of the types of information required in the system—high-level and low-level, for input, processing, and output. It should also be extensible to incorporate any further developments.

Figure 7: Sample of M3L (Wahlster *et al.*, 2001)

```
<presentationContent>
    [ ... ]
    <abstractPresentationContent>
        <movieTheater structId="pid3072">
            <entityKey> cinema_17a </entityKey>
            <name> Europa </name>
                <geoCoordinate>
                    <x> 225 </x> <y> 230 </y>
                </geoCoordinate>
        </movieTheater>
    </abstractPresentationContent>
    [ ... ]
    <panelElement>
        <map structId="PM23">
            <boundingShape>
                <leftTop>
                    <x> 0.5542 </x> <y> 0.1950 </y>
                </leftTop>
                <rightBottom>
                    <x> 0.9892 </x> <y> 0.7068 </y>
                </rightBottom>
            </boundingShape>
            <contentRef>pid3072</contentRef>
        </map>
    </panelElement>
    [ ... ]
</presentationContent>
```

Figure 8: Sample of APML (de Carolis *et al.*, 2002)

```
<apml>
    <turn-allocation type="take">
        <performative type="inform" affect="sorry-for" certainty="certain">
            I'm so sorry to tell you that you have been diagnosed as
            suffering from what we call angina pectoris,
        </performative>
        <belief-relation type="ElabObjAttr">
            which
            <performative type="inform" certainty="uncertain">
                appears to be
                <adjectival type="small">
                    mild.
                </adjectival>
            </performative>
        </belief-relation>
    </turn-allocation>
</apml>
```

2. Whenever possible, it should be compatible with existing standardisation initiatives, or designed in such a way that it can be the source of future standardisation.
3. It should be based on XML, but should adopt a schema definition language that is powerful enough to account for the definition of both generic structures and level-specific constraints.

There is no publicly-available example of MMIL markup.

4.2.3 APML

APML—Affective Presentation Markup Language (de Carolis *et al.*, 2002)—is one of the markup languages used within the MagiCster project, whose goal is to create believable conversational agents. APML is used to describe the dialogue moves which the conversational agent is to perform. It specifies the underlying communicative actions, rather than how those actions are to be realised. APML is also capable of specifying the desired prosody for the speech synthesiser.

Figure 8 shows a sample of APML markup. This sample shows one dialogue turn, with associated dialogue-act and affective markup.

4.2.4 MPML

MPML—Multimodal Presentation Markup Language (Tsutsui *et al.*, 2000)—is a language based on SMIL which supports functions for controlling verbal presentation and agent behaviour. It adds to SMIL the following specific tags for controlling agents:

`<agent>` Specifies an agent.

`<move>` Move an agent to a new location at the specified speed.

`<speak>` The specified agent should speak the given text.

`<play>` Play the specified action of the character agent.

Figure 9: Sample of MPML (Tsutsui *et al.*, 2000)

```
<mpml>
    <head>
        <layout>
            <root-layout id="root" width="800" height="600" />
            <region id="spot1" location="500,300" />
            <region id="spot2" location="20%,50%" />
        </layout>
    </head>
    <body>
        <ref region="root" src="http://www.miv.t.u-tokyo.ac.jp/" />
        <agent region="spot1" />
        <par>
            <play act="point" region="spot2" />
            <speak>
                This is MPML Home Page!
            </speak>
        </par>
    </body>
</mpml>
```

The most recent version of MPML also supports the inclusion of Flash animations in the source file.

Figure 9 shows a sample of MPML markup (Tsutsui *et al.*, 2000). This listing specifies that the default agent character will appear from *spot1*, point to *spot2*, and speak the content enclosed by the `<speak>` tags.

5 Conclusions and recommendations

This document has presented various theories, techniques, and representation schemes that are relevant to the task of fission. In this section, we summarise the contents of Sections 2–4 from the perspective of the COMIC project. Section 5.1 summarises the issues involved in performing the actual fission; Section 5.2 discusses the markup languages described in Section 4; and Section 5.3 discusses particular issues from the technical annex.

5.1 Performing the fission

Section 2 presents a number of possible approaches to the task of multimodal fission. In this section, we summarise and comment on the possible approaches in the context of the particular requirements of the COMIC project. Section 5.1.1 discusses each individual task, while Section 5.1.2 discusses issues that affect the whole presentation-planning process.

5.1.1 Specific tasks

The three tasks involved in COMIC are discussed separately in Section 2. However, in practice, these tasks are often done together, with the requirements of one influencing the constraints on another. For example, physical layout and modality selection very frequently constrain each other and take place at the same time.

Nevertheless, this section provides a brief discussion of the particular issues involved in each of these tasks from the perspective of COMIC. The presentation-planning process as a whole is dealt with in Section 5.1.2 below.

Content selection and structuring Content selection and structuring is not always the task of the fission module; in some cases, the content to be included is preselected, either by a previous component in the generation process (BEAT, Rea), or through being directly specified by the user of the system (APT, BOZ). There are advantages to combining this task with the other parts of presentation planning; for example, content can be structured and laid out as it is selected, so the system can avoid choosing more content than can actually be realised by the output system.

In the planned architecture for COMIC, the fission module will perform the task of content selection, guided by the dialogue acts specified by the dialogue manager.

Modality selection The systems that have put a lot of effort into modality selection are generally those that present statistical data using various graphical presentation techniques and natural language. With that sort of well-defined, technical data and those fairly comparable modalities, the question of modality selection is an interesting one, and one that is worth the fairly sophisticated theoretical techniques that have been used for it.

The following are the output modalities that will be used in COMIC:

- Synthesised speech, with (eventual) coordinated movement of the lips of the avatar.
- Non-lip movements of the avatar (head turning, facial expressions, blinking, etc.).

- Manipulation of the ViSoft application's display by, for example, zooming in, changing perspective, or showing a new menu.
- Graphical "gestures"—for example, circling or highlighting some part of the display. This differs from the previous modality in that it will be adding information on top of the ViSoft display, rather than changing the underlying display.

The type of information that will be presented is not completely defined yet, but it is certain to be rather less concrete than statistical data.

In this case, the output modalities are much less interchangeable than graphics and text, and the message types are much less quantitative. Given this, the only type of modality selection that seems likely to be useful is choosing how to realise a reference to an on-screen object; in general such a reference is most likely best made with a multimodal referring expression.

The selection of modalities could also be influenced by the capabilities of the particular device being used to access the COMIC system, and possibly the circumstances of the user. For example, if the user is in a noisy environment, the system could avoid using the speech modality. However, since the underlying ViSoft application is so graphics-intense, it does not seem realistic to consider running it either on a small, mobile device, or in scenarios in which the visual channel is unavailable.

Output coordination Output coordination is likely to play a larger role than modality selection in COMIC's fission process. Physical layout is not really an issue; however, temporal coordination is very important. In the context of the strict fission module in particular, the various components of the presentation must be scheduled so that they occur in the correct relation to one another. Allen's linear temporal constraints will probably be useful in this task, as they have been in the other systems that have used them. See Section 5.3.2 for more information on the difference between the sloppy and strict modules.

The fission module will also need to have access to a model of what is displayed on the ViSoft application screen so that it can highlight the correct components, make appropriate linguistic references, and possibly instruct the avatar to look in the direction of a particular object. If the avatar can be placed in different locations on the screen, the system will also need to know where it is located in relation to the other on-screen objects.

As in most previous systems, the temporal behaviour of the presentation will be driven by the synthesised speech. We can use a technique similar to that employed in BEAT, for example, to ensure that visual events are coordinated with the spoken output when necessary.

5.1.2 Overall considerations

Two important considerations in the context of fission are the importance of communicative intentions throughout the presentation-planning process, and the choice between of domain-dependent and domain-independent techniques and representations.

The importance of communicative intentions The communicative intentions of the presenter can play a very important role in both the content and the form of a presentation. Consider, for example, the (fictional) newspaper-readership data in Table 3 (Moore, 1998). There are many different communicative tasks that this data could support.

Table 3: Western Pennsylvania newspaper readership data (Moore, 1998)

Paper	Readers	Paper	Readers
Buttler Eagle	1,890	Pittsburgh Post-Gazette	371,650
Greensburg Tribute Review	37,020	USA Today	93,210
McKeesport Daily News	73,590	Valley News Dispatch	35,540
New York Times	22,610	Wall Street Journal	55,640
North Hills News Record	44,550	Washington Observer-Reporter	4,500

Suppose, for example, that the presenter wanted to convince the reader to advertise in the Post-Gazette. Figure 10(a) shows a sample presentation that supports this goal, by emphasising the fact that more people read that paper than any of the other newspapers under consideration.

Note that the data alone is not sufficient to determine the text and graphics that should be generated. For example, the same data can also be used to support a presentation with the opposite communicative intent, as in Figure 10(b). The goal of this presentation is to persuade the reader that advertising in the Post-Gazette may *not* be effective.

Finally, it is important that all components of the presentation be generated using the same communicative intentions. Consider Figure 10(c): the presenter's goal is to compare the readership of the Post-Gazette to the total of all other newspapers in the area, but the graphic requires the reader to mentally add together all of the individual bars and compare the total to the length of the Post-Gazette bar.

In the context of COMIC, the communicative intentions will be equally vital to determining the content, structure, and form of the presentations. For these reasons, the particular sorts of communicative goals that the output system will support must be set out soon, so that appropriate presentations can be generated.

Domain-independent *vs* domain-dependent techniques The only systems that use only domain-independent rules to generate their presentations are those that generate information graphics from statistical data—BOZ, APT, PostGraphe, and SAGE. In most other cases, the process of mapping from abstract communicative intention to concrete output uses a great deal of domain-dependent knowledge. This knowledge may take the form of, for example, domain-specific schemas or plan operators, or domain-specific rules for choosing among modalities such as those used by COMET or MAGPIE.

This use of domain-dependent knowledge seems unavoidable when generating any sort of interesting multimodal output, and COMIC will have to employ some such rules when creating its output.

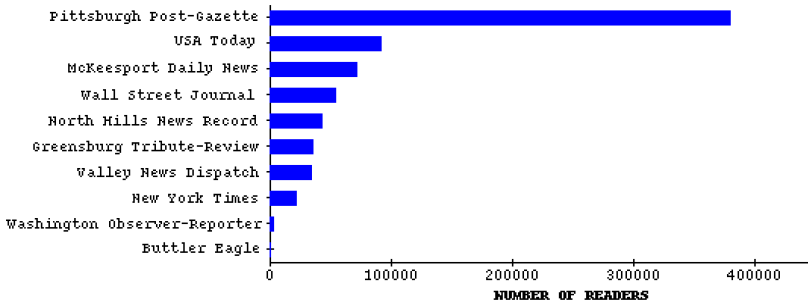
5.2 Representation languages

Of the languages described Section 4, VoiceXML and SALT are definitely not relevant to this particular task; both are aimed more at providing telephony-type services or voice interfaces to web applications.

For the input to the text-to-speech engine, SSML seems acceptable. MagiCster also uses a similar language, based on SABLE (an ancestor of SSML) with the addition of tags for specifying pitch

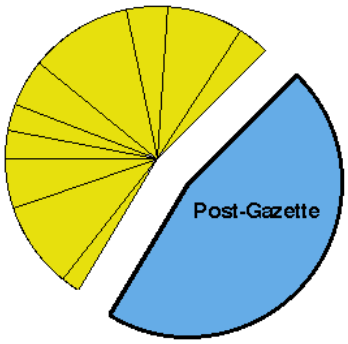
Figure 10: Intentions and graphic design (Moore, 1998)

"Advertise in the Pittsburgh Post-Gazette. More people read it than any other newspaper in Western Pennsylvania."



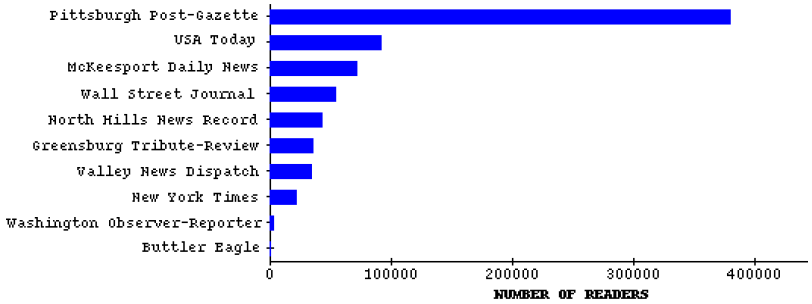
(a) One set of compatible intentions

"Don't advertise in the Post-Gazette. The majority of Western Pennsylvanians read other newspapers."



(b) Another set of compatible intentions

"Don't advertise in the Post-Gazette. The majority of Western Pennsylvanians read other newspapers."



(c) Conflicting intentions

accents. This language may prove useful, particularly as MagiCster is also using Festival for its speech synthesis, is based at Edinburgh, and also requires coordination with its animated talking heads. Essentially, any markup language that is able to specify the required prosodic information can be used.

For specifying the content of the overall presentation, SMIL is one possibility; however, further investigation is required to ensure that it provides enough flexibility to represent the output that COMIC will generate. Of the project-specific languages in Section 4.2, MMIL looks the most general-purpose and promising; however, more information is required about M3L and MMIL in particular before making any decisions.

5.3 Goals from the technical annex

Two particular parts of the COMIC technical annex describe aspects of the fission process that still require further specification: the concept of multimodal "unit selection", and the difference between the "sloppy" and "strict" fission modules. This section discusses each of these concepts briefly.

5.3.1 Multimodal unit selection

One of the stated aims of the COMIC project is to adapt unit-selection techniques from speech synthesis (Section 3.1.1) to the multimodal context. The basic premise of unit selection is that the synthesiser chooses from among a large set of pre-recorded snippets to produce its output. In the context of COMIC, however, it will not be feasible to follow this principle strictly and use a set of multimodal units. In particular, the face recordings that will be used for the avatar will be completely separate from any voice recordings that will be used in speech synthesis.

One option is to cache particular multimodal combinations—for example, saying "put *[object]* here" while making a gesture indicating the relevant spot. These units could be derived from the results of the SLOT-task experiments; they could also come from an independent study with subjects rating various combinations of text and gestures. These multimodal units would only be used in the strict fission module. The particular content of the units depends on the sorts of messages that COMIC aims to produce.

5.3.2 Sloppy *vs* strict fission

As stated in the technical annex, there will be two different versions of the fission module: *sloppy* and *strict*. The difference between these modules will lie in their use of the multimodal units described in the preceding section.

The sloppy fission module will perform whatever allocation is required and send messages to the selected modality-specific generators independently. It may perform some temporal coordination on the content produced by each generator, or it may just send off the individual messages completely independently.

The strict module, on the other hand, will make use of the multimodal units whenever any are available that meet the requirements of the message to be communicated. It may also do more fine-grained control of the temporal coordination across the modalities than the sloppy module.

6 References

André (2000) and Maybury (2001) provide useful surveys of the work in the area of multimodal presentations. Many of the relevant papers are collected in one or both of Maybury (1993a) and Maybury and Wahlster (1998); a number of systems are also described in the context of the "Standard Reference Model for Intelligent Multimedia Presentation Systems" in Rist *et al.* (1997).

ALLEN J F (1983). Maintaining knowledge about temporal intervals. *Communications of the ACM* 26(11):832–843.

ANDRÉ E (2000). The generation of multimedia documents. In: *A Handbook of Natural Language Processing: Techniques and Applications for the Processing of Language as Text*, edited by R Dale, H Moisl, and H Somers, pp. 305–327. Marcel Dekker Inc. http://www.dfki.de/imedia/papers/handbook.ps.

ANDRÉ E, FINKLER W, GRAF W, RIST T, SCHAUDER A, and WAHLSTER W (1993). WIP: The automatic synthesis of multimodal presentations. In: Maybury (1993a), pp. 75–93.

ANDRÉ E, HERZOG G, and RIST T (1998). Generating multimedia presentations for RoboCup soccer games. In: *RoboCup-97: Robot Soccer World Cup I*, edited by H Kitano, Lecture notes in Computer Science, pp. 200–215. Springer. http://www.dfki.uni-sb.de/vitra/papers/robocup97/.

ANDRÉ E, RIST T, and MÜLLER J (1999). Employing AI methods to control the behaviour of animated interface agents. *Applied Artificial Intelligence Journal* 13(4–5):415–448.

ARENS Y, HOVY E, and VOSSERS M (1993). On the knowledge underlying multimedia presentations. In: Maybury (1993a), pp. 280–306. http://www.isi.edu/natural-language/multimedia/knowledge-models.ps. Reprinted in Maybury and Wahlster (1998), pp. 157–172.

BATEMAN J, KAMPS T, KLEINZ J, and REICHENBURGER K (2001). Towards constructive text, diagram, and layout generation for information presentation. *Computational Linguistics* 27(3):409–449.

BAUS J, KRÜGER A, and WAHLSTER W (2002). A resource-adaptive mobile navigation system. In: *Proceedings of IUI2002: International Conference on Intelligent User Interfaces 2002*. http://w5.cs.uni-sb.de/~baus/publications/mobile-navigation-iui.pdf.

BERNSEN N O (1997). Defining a taxonomy of output modalities from an HCI perspective. In: Rist *et al.* (1997), pp. 537–553.

BORDEGONI M, FACONTI G, FEINER S, MAYBURY M T, RIST T, RUGGIERI S, TRAHANIAS P, and WILSON M (1997). A standard reference model for intelligent multimedia presentation systems. In: Rist *et al.* (1997), pp. 477–496.

BORMANS J and HILL K (eds.) (2002). *MPEG-21 Overview*. http://mpeg.telecomitalialab.com/standards/mpeg-21/mpeg-21.htm. Version 4—Fairfax, May 2002.

BUNT H and ROMARY L (2002). Towards multimodal semantic representation. In: *Workshop 'International Standards of Terminology and Language Resources Management', LREC 2002*. Las Palmas (Spain). http://let.kub.nl/people/bunt/docs/lrec.doc.

CARENINI G and MOORE J D (2000). A strategy for generating evaluative arguments. In: *Proceedings of the 1st International Conference on Natural Language Generation (INLG-00)*. Mitzpe Ramon, Israel. http://www.cs.ubc.ca/~carenini/PAPERS/cr1-inlg00-arg.doc.

DE CAROLIS B, CAROFIGLIO V, BILVI M, and PELACHAUD C (2002). APML, a mark-up language for believable behavior generation. In: Marriott *et al.* (2002). http://www.vhml.org/workshops/AAMAS/papers/decarolis.pdf.

CASNER S M (1991). A task-analytic approach to the automated design of graphic presentations. *ACM Transactions on Graphics* 10(2):111–151. Reprinted in Maybury and Wahlster (1998), pp. 204–225.

CASSELL J (2000). Embodied conversational interface agents. *Communications of the ACM* 43(4):70–78. http://gn.www.media.mit.edu/groups/gn/publications/CACM.pdf.

CASSELL J, BICKMORE T, CAMPBELL L, VILHJÁLMSSON H, and YAN H (2000a). Human conversation as a system framework: Designing embodied conversational agents. In: Cassell *et al.* (2000b), pp. 29–63. http://gn.www.media.mit.edu/groups/gn/publications/ECA_GNL.chapter.to_handout.pdf.

CASSELL J, SULLIVAN J, PREVOST S, and CHURCHILL E (2000b). *Embodied Conversational Agents*. MIT Press.

CASSELL J, VILHJÁLMSSON H H, and BICKMORE T (2001). BEAT: The Behavior Expression Animation Toolkit. In: *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp. 477–486. ACM Press. http://gn.www.media.mit.edu/groups/gn/pubs/siggraph2001.final.PDF.

DAHL D (2002). W3C The World Wide Web Consortium's activities in multimodal interaction. *VoiceXML Review* 2(4). http://www.voicexmlreview.org/Jun2002/features/w3cnatural.html.

DALAL M, FEINER S, MCKEOWN K, PAN S, ZHOU M, HÖLLERER T, SHAW J, FENG Y, and FROMER J (1996). Negotiation for automated generation of temporal multimedia presentations. In: *Proceedings of ACM Multimedia '96*, pp. 55–64. http://www.cs.columbia.edu/~shaw/papers/mm96.pdf.

FASCIANO M and LAPALME G (2000). Intentions in the coordinated generation of graphics and text from tabular data. *Knowledge and Information Systems* 2(3):310–339. http://www.iro.umontreal.ca/~scriptum/IntentionsKAIS.ps.gz.

FEINER S K and MCKEOWN K R (1991). Automating the generation of coordinated multimedia explanations. *IEEE Computer* 24(10):33–41. Reprinted in Maybury and Wahlster (1998), pp. 89–98.

GRAF W H (1997). LayLab from the perspective of the IMMPS standard. In: Rist *et al.* (1997), pp. 651–656.

GRATCH J and RICKEL J (eds.) (2002). *Virtual Humans Workshop*. Marina del Rey, CA. http://www.ict.usc.edu/~vhumans/.

GRATCH J, RICKEL J, ANDRÉ E, CASSELL J, PETAJAN E, and BADLER N (2002). Creating interactive virtual humans: Some assembly required. *IEEE Intelligent Systems* 17(4):54–63. http://www.ict.usc.edu/~vhumans/ieee-is02-workshop.pdf.

GROSZ B L and SIDNER C L (1986). Attention, intention, and the structure of discourse. *Computational Linguistics* 12(3):175–204.

HAN Y and ZUKERMAN I (1997). A mechanism for multimodal presentation planning based on agent cooperation and negotiation. *Human-Computer Interaction* 12(1–2):187–226. ftp://ftp.cs.monash.edu.au/pub/ingrid/HCI98.hanyi.ps.gz. Special Issue on Multimodal Interfaces.

LE HÉGARET P (1999). SMIL: The multimedia for everybody. Talk delivered at the WAP forum, Montreux. http://www.w3.org/Talks/990415SMIL-Montreux/.

HERZOG G, ANDRÉ E, BALDES S, and RIST T (1998). 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*, edited by A Sutcliffe, J Ziegler, and P Johnson, pp. 143–157. Kluwer, Stuttgart. http://www.dfki.de/fluids/docs/deums98/deums98.ps.gz.

HUNT A J and BLACK A W (1996). Unit selection in a concatenative speech synthesis system using a large speech database. In: *ICASSP-96*, volume 1, pp. 373–376. Atlanta, Georgia. http://www-2.cs.cmu.edu/~awb/papers/icassp96.ps.

JOHNSON W L, RICKEL J W, and LESTER J C (2000). Animated pedagogical agents: Face-to-face interaction in interactive learning environments. *International Journal of Artificial Intelligence in Education* 11:47–78. http://www.csc.ncsu.edu/eos/users/l/lester/Public/apa-ijaied-2000.ps.gz.

KAMP H and REYLE U (1993). *From Discourse to Logic: Introduction to Model Theoretic Semantics of Natural Language, Formal Logic and Discourse Representation Theory*. Kluwer Academic Press, Dordrecht.

KERPEDJIEV S, CARENINI G, GREEN N, MOORE J, and ROTH S (1998). Saying it in graphics: from intentions to visualizations. In: *Proceedings of the IEEE Symposium on Information Visualization (InfoVis '98)*, pp. 97–101. Research Triangle Park, NC. http://www-2.cs.cmu.edu/afs/cs.cmu.edu/project/sage/mosaic/Papers/sayit.html.

KOENEN R (2001). Object-based MPEG offers flexibility. *EE Times* http://www.eetimes.com/story/OEG20011112S0042. 12 November 2001.

KOENEN R (ed.) (2002). *MPEG-4 Overview*. http://mpeg.telecomitalialab.com/standards/mpeg-4/mpeg-4.htm. Version 21—Jeju, March 2002.

LARSON J (2002). Speaking standards with Jim Larson. *Speech Technology Magazine* http://www.speechtechmag.com/pub/industry/849-1.html. Industry News, 18 June 2002.

MACKINLAY J (1986). Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics* 5(2):110–141. Reprinted in Maybury and Wahlster (1998), pp. 177–193.

MANN W C and THOMPSON S A (1988). Rhetorical structure theory: Towards a functional theory of text organization. *TEXT* 8(3):243–281.

MARRIOTT A, PELACHAUD C, RIST T, RUTTKAY Z, and VILHJÁLMSSON H (eds.) (2002). *Embodied conversational agents—let's specify and evaluate them!* Bologna, Italy. http://www.vhml.org/workshops/AAMAS/. A workshop in conjunction with the First International Joint Conference on Autonomous Agents and Multi-Agent Systems.

MARTINEZ J M (ed.) (2002). *MPEG-7 Overview*. http://mpeg.telecomitalialab.com/standards/mpeg-7/mpeg-7.htm. Version 7—Jeju, March 2002.

MAYBURY M T (ed.) (1993a). *Intelligent Multimedia Interfaces*. AAAI Press.

MAYBURY M T (1993b). Planning multimedia explanations using communicative acts. In: Maybury (1993a), pp. 59–74. Reprinted in Maybury and Wahlster (1998), pp. 99–108.

MAYBURY M T (2001). Intelligent user interfaces: An introduction. http://www.mitre.org/resources/centers/it/maybury/mayburyiui2001.ppt. Tutorial Notes, International Conference on Intelligent User Interfaces (IUI '01).

MAYBURY M T and WAHLSTER W (eds.) (1998). *Readings in Intelligent User Interfaces*. Morgan Kaufmann.

MCKEOWN K (1985). *Text Generation: Using Discourse Strategies and Focus Constraints to Generate Natural Language Text*. Cambridge University Press.

MOORE J D (1995). *Participating in Explanatory Dialogues: Interpreting and Responding to Questions in Context*. MIT Press, Cambridge, Massachusetts.

MOORE J D (1998). Producing multimedia explanations: A computational model. Invited presentation at the Winter Text Conference, Jackson Hole, WY.

MOORE J D and WIEMER-HASTINGS P (In press). Discourse in computational linguistics and artificial intelligence. In: *Handbook of Discourse Processes*, edited by A C Graesser, M A Gernsbacher, and S Goldman. Lawrence Erlbaum Associates, Hillsdale, NJ.

VAN OSSENBRUGGEN J, GEURTS J, CORNELISSEN F, RUTLEDGE L, and HARDMAN L (2001). Towards second and third generation web-based multimedia. In: *The Tenth International World Wide Web Conference*, pp. 479–488. Hong Kong. http://www.cwi.nl/~media/publications/www10.pdf.

OVIATT S (1999). Ten myths of multimodal interaction. *Communications of the ACM* 42(11):74–81.

POTTER S and LARSON J A (2002). VoiceXML and SALT. *Speech Technology Magazine* 7(3). http://www.speechtechmag.com/issues/7_3/cover/742-1.html.

REITHINGER N, LAUER C, and ROMARY L (2002). MIAMM—multidimensional information access using multiple modalities. In: *International CLASS Workshop on Natural, Intelligent and Effective Interaction in Multimodal Dialogue Systems*. Copenhagen, Denmark. http://www.class-tech.org/events/NMI_workshop2/papers/reithinger-class-final.pdf.

RICKEL J, GRATCH J, HILL R, MARSELLA S, and SWARTOUT W (2001). Steve goes to Bosnia: Towards a new generation of virtual humans for interactive experiences. In: *Proceedings, AAAI Spring Symposium on Artificial Intelligence and Interactive Entertainment*. http://www.ict.usc.edu/pdfs/sss01.pdf.

RIST T, FACONTI G, and WILSON M (eds.) (1997). *Computer Standards & Interfaces*, volume 18(6–7). Elsevier. Special issue on Intelligent Multimedia Presentation Systems.

ROTH S F and MATTIS J (1990). Data characterization for intelligent graphics presentation. In: *Proceedings of the Conference on Human Factors in Computing Systems (CHI '90)*, pp. 193–200. Reprinted in Maybury and Wahlster (1998), pp. 194–203.

ROTH S F, MATTIS J A, and MESNARD X (1991). Graphics and natural language as components of automatic explanation. In: *Proceedings of Intelligent User Interfaces (IUI '91)*.

SALT FORUM (2002). *Speech Application Language Tags (SALT)*. 1.0 Specification, 15 July 2002, SALT Forum. http://www.saltforum.org/.

SCHROEDER J (2001). The fundamentals of text-to-speech synthesis. *VoiceXML Review* 1(3). http://www.voicexmlreview.org/Mar2001/features/tts.html.

STOCK O (1991). Natural language and exploration of an information space: the ALFresco interactive system. In: *Proceedings 12th IJCAI*, pp. 972–978. Sydney, Australia. Reprinted in Maybury and Wahlster (1998), pp. 421–428.

TSUTSUI T, SAEYOR S, and ISHIZUKA M (2000). MPML: A multimodal presentation markup language with character agent control functions. In: *Proceedings of WebNet 2000 World Conference on the WWW and Internet*. San Antonio, TX. http://www.miv.t.u-tokyo.ac.jp/papers/santiWebNet2000.pdf.

W3C (2001). *Synchronized Multimedia Integration Language (SMIL 2.0)*. Recommendation, 07 August 2001, W3C SYMM Working Group. http://www.w3.org/TR/2001/REC-smil20-20010807/.

W3C (2002a). *Speech Synthesis Markup Language Specification*. Working Draft, 5 April 2002, W3C Voice Browser Working Group. http://www.w3.org/TR/2002/WD-speech-synthesis-20020405/. Work in progress.

W3C (2002b). *Voice Extensible Markup Language (VoiceXML) Version 2*. Working Draft, 24 April 2002, W3C Voice Browser Working Group. http://www.w3.org/TR/2002/WD-voicexml20-20020424/. Work in progress.

WAHLSTER W, REITHINGER N, and BLOCHER A (2001). *SmartKom: Towards Multimodal Dialogues with Anthropomorphic Interface Agents*. Technical Report 15, SmartKom project. http://www.smartkom.org/reports/Report-NR-15.pdf.

WALKER M, WHITTAKER S, STENT A, MALOOR P, MOORE J D, JOHNSTON M, and VASIREDDY G (2002). Speech plans: generating evaluative responses in spoken dialogue. In: *Proceedings, Second International Natural Language Generation Conference (INLG'02)*. http://www.cs.rutgers.edu/~mdstone/inlg02/150.pdf.

WILSON M, SEDLOCK D, BINOT J L, and FALZON P (1992). An architecture for multimodal dialogue. In: *Proceedings Second Vencona Workshop for Multimodal Dialogue*. Vencona, Italy.

ZELAZNY G (1996). *Say It with Charts: The Executive's Guide to Visual Communication*. McGraw Hill, 3rd edition.