

Chapter 7

Beyond Text Generation

So far in this book we have considered the generation of natural language as if this was concerned with the production of text abstracted away from embodiment in any particular medium. This does not reflect reality, of course: when we are confronted with language, it is always embodied, whether in a speech stream, on a computer screen, on the page of a book, or on the back of a breakfast cereal packet. In this chapter, we look beyond text generation and examine some of the issues that arise when we consider the generation of text in context.

7.1 Introduction

Real documents do not consist simply of text. Real documents use typographical elements, such as font, face and size changes, to represent headings and to indicate the different functions of particular words and phrases. They use space and layout to indicate the role of textual elements *via*, for example, indentation and the use of list structures. Real documents also often include pictures, diagrams, and other graphical elements. The situation is more complex still in on-line documents, where the text may contain embedded hypertext links, forms, attached query engines and dynamically changing graphical elements such as animated gifs.

In many cases it may be appropriate for a natural language generation system to consider these aspects of real documents when it creates its output. The PEBA system, for example, produces Web pages for display in a browser; this means it has to, if only by default, address some of the above considerations. Figure 7.1 shows a description of the alligator produced by PEBA, demonstrating several of the phenomena we have just alluded to.

Typography: The text lists the different kinds of alligator that are present in the system's knowledge base; these are presented typographically by means of a bulleted list. This information could have been presented without

such a typographic device, using a sentence like the following in the running text:

(7.1) The Alligator has two subtypes: the American Alligator and the Chinese Alligator.

However, as the number of items to be listed increases, expressing the information in this way becomes unwieldy.

Graphics: A picture of an alligator is included in the document: an image like this often conveys much information that would be difficult or impossible to convey in words. In the general case, knowing what information is conveyed by a graphical element can impact on the system's decisions about what information should be communicated *via* text.

Hypertext: All underlined words and phrases in the document are hypertext links that correspond to new discourse goals that the user can provide to the system. If the user clicks, for example, on American Alligator, this is considered by PEBA as a request to generate a description of that entity.

So far we have mentioned aspects of language in use that are a consequence of the need for the printed form to appear in some containing medium. There are also issues that arise in the provision of information *via* speech. The linear nature of the spoken form means that speech does not have counterparts to some of the more complex phenomena possible in visible language; but it is still the case that speech is not simply spoken text. Words and phrases are emphasised in various ways to indicate their informational status or importance, and appropriate pauses help guide the hearer's processing of the information.

A natural language generation system cannot ignore these presentational phenomena for two reasons:

- Although there may be cases where an NLG system can create text without regard for its means of presentation, this generally requires that the system effectively be used in the 'computer as authoring aid' mode, as discussed in Section 1.2.1. A human will then have to intervene to place the generated text in its context. If we want to construct a generation system which generates texts that can be delivered to readers without human intervention, then the system must also embed that text in an appropriate presentational medium. This capability can also be useful, of course, even when the system is required to produce texts that *are* subsequently augmented or edited by a human.
- The containing medium can make a difference to the text that is generated. For example, there may be a physical space restriction that has to be met by the generated text; or there may need to be cross-references in the text between the textual and graphical content. In a spoken language system, the need to produce output that can be spoken may, for example, have an impact on the lengths of sentences that are generated.

The English language does not offer an appropriate term which conveniently covers instances of output in both the spoken and written forms. We will refer to the output of systems that attend to the concerns under discussion here as DOCUMENTS, whether that output is spoken or written. The production of documents rather than just the text they contain has the potential to vastly increase the utility of an NLG system.

In this chapter we look at the issues involved in producing documents. We begin by sketching some relevant background in Section 7.2. We then go on to consider different aspects of embodying text within a medium: we look first at the use of typography in Section 7.3; we extend the discussion to consider the integration of text and graphics in Section 7.4, and the generation of hyper-textual documents in Section 7.5; and we end by looking at the generation of speech in Section 7.6. In each case we provide some discussion of the use of these resources, we look at how they have been used in NLG systems to date, and we show how their use impacts on our NLG tasks of document planning, microplanning, and surface realisation.

7.2 Embodied Text

As natural language generation systems become more sophisticated, it is important that they begin to take account of the medium in which the text they create is situated, whether this be on paper, on a computer screen, or in a speech stream. They must generate EMBODIED TEXT. All text that we experience is embodied, although research in linguistics often carries out analysis of language abstracted away from its embodiment.

Terminology in this area is problematic, because what might seem to be the most appropriate terms for many of the elements we have to distinguish have already been appropriated for other things. Because of this, we can only stipulate our own definitions for already overloaded terms. Figure 7.2 expresses diagrammatically the relationships we will take to hold between TEXT and DOCUMENT, with VISIBLE LANGUAGE and SPOKEN LANGUAGE being the two possible embodiments of text. It is important to note that, in our meaning of the terms, text is something we never see: it is linguistic material which, by the time we perceive it, has been embodied in some medium or CHANNEL.

The study of spoken language is an accepted academic discipline; within linguistics, the fields of phonetics and phonology are well-respected areas of study. Most of what we have to say in this chapter will be concerned, however, with *visible* language. As we have already suggested, the fact that written language always appears in some medium imposes constraints on the ways in which the language can be used; as we will see below, it also offers opportunities that are only apparent in the context of the physical manifestation of the text. In recent years, a not insubstantial amount of work in computational linguistics and natural language processing has concerned itself with the integration of linguistic and graphical expressions of information; however, a great deal of theoretical work remains to be done here. We have only scratched the

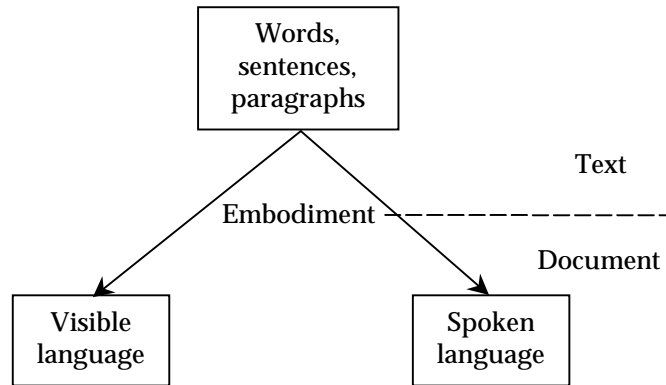


Figure 7.2: The embodiment of text

surface in terms of understanding how the chosen medium plays an important role in communicating a message; this role has long been recognised by professional document designers, but unlike the spoken form, visible aspects of language tend to be ignored in work in theoretical linguistics. Ultimately, a coherent theory of linguistic and graphical communication will not be complete without sufficient attention being paid to the means of embodiment of the elements.

Another way to approach these issues is by considering the notion of MULTIMODALITY. Generally speaking, multimodality is concerned with communication through different MODALITIES such as voice, gesture, animation and written text. Each modality is a communication channel used to convey or acquire information. Within work in natural language processing, the modalities usually considered are speech, text and graphics; as we will see in Section 7.4, there is now a significant body of work concerned with the computer generation of documents that include both text and graphics. Strictly speaking, a system which generates only spoken output is not multimodal; however, a system which makes use of typographic resources is arguably a multimodal system, since it is using both verbal and visual properties to communicate its meaning. All documents, in the sense we are using that term, are therefore MODAL; and some documents are MULTIMODAL.

7.3 Typography

7.3.1 The Uses of Typography

As a discipline, typography is concerned with the appearance and effectiveness of documents. This concern covers a wide variety of phenomena at different levels, including the shapes and sizes of individual letterforms; the spaces between letters, words and lines; the lengths of lines and sizes of margins; and the spatial configuration of larger elements of text. Poor typographic choices, or a neglect of the typographic properties of a document, can have quite a serious impact on the effectiveness of that document.

Typographic choices are motivated by a variety of considerations:

Typographic conventions: Sometimes typographic properties are dictated by conventions that apply to the kind of text under consideration. For example, a widespread convention adopted in many publications is that if a document contains words from languages other than the principal language of the document, then these foreign words should be presented in italic face. In the publishing world, conventions such as these are often collected together and referred to as a *HOUSE STYLE*, this being a catalog of rules for the particular ways in which typographic variations should be used. So, as another example, a common house style rule is that references to literary works should be in italic face, as in *Gone with the Wind*. It is important to note that these prescriptions are only conventions, and so there is no one right way of doing things; a different house style, for example, might require that the titles of literary works be presented wrapped in quotes, as in 'Gone with the Wind'. What matters is consistency within a document or collection of documents; this makes it easier for the reader to develop expectations as to what particular typographic variations signify. At the beginning of this book, for example, we have provided a table that indicates the typographic conventions used throughout the book; this is intended as a guide to help the reader determine whether something we are talking about is, for example, a technical term rather or a knowledge base element.

Indicating emphasis: Typography can be used to mark a particular piece of text as being especially important. Figure 7.3 shows a fragment from a conference announcement, where bold face has been used to draw the reader's attention to important dates associated with the conference. Of course, this use of bold face is itself something of a convention, but the visual salience of this device makes it distinct from many other conventions adopted in written documents, where the function of the convention is primarily to mark distinctions by indicating logical difference via typography. Other typographic devices can be used to draw the reader's attention; for example, text can be underlined, different colours can be used, boxes can be drawn around important elements, and so on.¹

¹It is interesting to note that the notion of underlining as emphasis has achieved the status of a

Important Dates

Paper and electronic submissions must be received by **Friday 29th November 1996**. Notification of receipt will be mailed to the first author (or designated author) soon after receipt. Authors will be notified of acceptance by **Monday 16th December 1996**. Camera-ready copies of final eight-page papers must be received by **Monday 27th January 1997**.

Figure 7.3: Using typeface variation for emphasis

Meeting size constraints: Typography also offers opportunities to deal with problems of limited space. Quite different typographic devices can present the same basic information but with different costs in terms of space; we will see an example of this below. Typographic manipulation can also be used to meet space constraints without more radical changes in the devices used. For example, even within the bounds of constraints laid down for the camera-ready copy for a conference proceedings, it is not unusual for an author to succeed in squeezing additional text into the number of pages available by reducing the spaces before and after headings, or even in extreme cases by adjusting font sizes and page margins. It is perhaps a sign of a heightening in end-user awareness of typographic issues that camera-ready instructions, even for relatively informal publications, are becoming increasingly prescriptive; this is at least in part an attempt to prevent the inconsistencies that result from authors adopting such tactics.

Indicating structure: A key use of typography, especially in technical domains but also more generally, is to indicate the logical structure of a document. The use of headings is one part of this: these provide visual cues to the intended segmentation of the textual material, and the use of different sizes or styles in headings is often used to provide an indication of the hierarchical relationships in the document. Particular typographic devices, such as itemized or enumerated lists, can also be used to indicate the structure of information. For example, it is not uncommon for a sequence of instructions to be presented in list form, so that the scope of each distinct step is made clear visually; various levels of indentation can also be used to indicate the hierarchical relationships between elements of the list.

Enabling information access: Just as typographic devices can be used to indicate that which is important, a related use is to help users quickly identify specific pieces of information. For example, presenting information in any regularised form such as a table makes it easier for the reader to access specific elements without having to read the whole text. Figure 7.4 shows the information in our conference announcement expressed in a

metaphor, as in *Let me just underline what I'm saying here*.

Important Dates

Deadline for receipt of paper and electronic submissions: All submissions must be received by Friday 29th November 1996. Notification of receipt will be mailed to the first author (or designated author) soon after receipt.

Notification of reviewing results: Notifications of acceptance or rejection will be provided to authors by Monday 16th December 1996.

Camera-ready due date: Camera-ready copies of final eight-page papers must be received by us by Monday 27th January 1997.

Figure 7.4: Using a labelled list structure to aid information access

different way: here, the significant events that make up the conference reviewing process are used as labels, and the particular spatial configuration used makes it easy for the reader to jump straight to the relevant information. Back-of-book indexes are another example of the same device. As the degree of regularity in the information to be presented increases, it becomes more appropriate to provide this information in an explicit table with labelled rows and columns.

As a larger example of how typography can be used to achieve different effects, consider the following text that might be produced by a space-age travel agency:²

- (7.2) When time is limited, travel by Rocket, unless cost is also limited, in which case go by Space Ship. When only cost is limited an Astrobuss should be used for journeys of less than 10 orbs, and a Satellite for longer journeys. Cosmocars are recommended when there are no constraints on time or cost, unless the distance to be travelled exceeds 10 orbs. For journeys longer than 10 orbs, when time and cost are not important, journeys should be made by Super Star.

The information expressed in this text can be presented typographically in various ways, as demonstrated in Figures 7.5, 7.6 and 7.7. Each presentation is appropriate in different circumstances. So, for example, the logical tree structure in Figure 7.5 might be helpful on a reader's first exposure to the information, since it makes it easy to distinguish those factors the reader considers important. The form in Figure 7.6 is perhaps better suited to experienced users; and the form in Figure 7.7 offers the most compact solution. Note that in each case, the appropriate use of typography has resulted in a presentation of the information that is easier to navigate than the purely textual version shown above.

²This example is from Crystal [1997] and is based on Wright [1977].

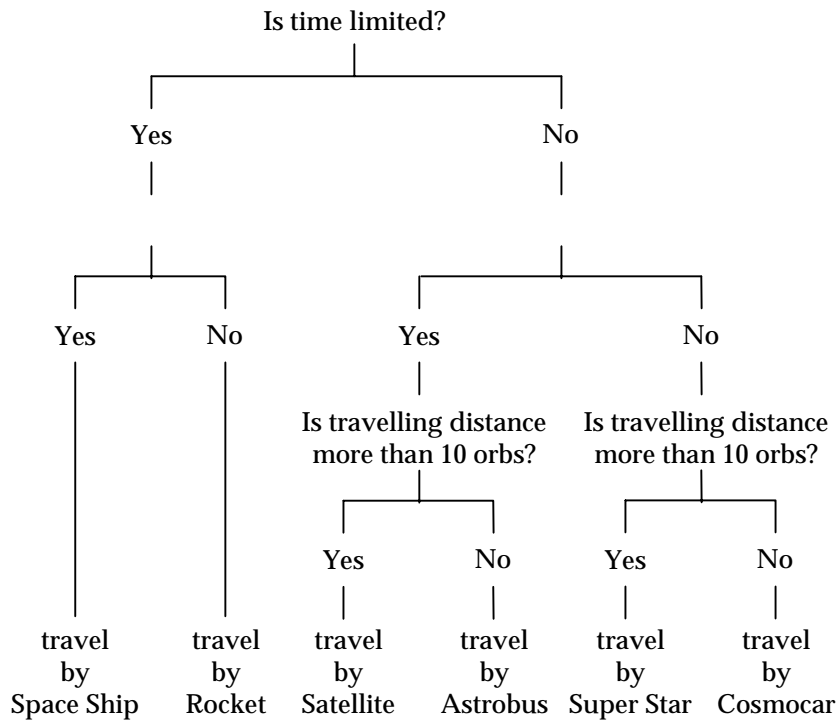


Figure 7.5: Information expressed typographically as a decision tree

	If journey less than 10 orbs	If journey more than 10 orbs
Where only time is limited	travel by Rocket	travel by Rocket
Where only cost is limited	travel by Astrobus	travel by Satellite
Where time and cost are not limited	travel by Cosmocar	travel by Super Star
Where both time and cost are limited	travel by Space Ship	travel by Space Ship

Figure 7.6: Information expressed typographically via a table

Where only time is limited
 travel by Rocket

Where only cost is limited
 travel by Satellite if journey more than 10 orbs
 travel by Astrobus if journey less than 10 orbs

Where both time and cost are limited
 travel by Space Ship

Where time and cost are not limited
 travel by Super Star if journey more than 10 orbs
 travel by Cosmocar if journey less than 10 orbs

Figure 7.7: Information expressed compactly

7.3.2 Typography in Natural Language Generation Systems

For an NLG system to make use of typographic resources, it must in some sense be **TYPOGRAPHICALLY-AWARE**: it must be able to indicate typographic distinctions in its output, whether directly by specifying font and face changes and spacing instructions to some typesetting device, or by marking up its output text in such a way that an external rendering device can produce the appropriate visible results. As discussed in Section 6.2, the latter option is to be preferred, since this absolves the NLG system from having to concern itself with the idiosyncracies of physical typesetting. So, to make use of typographic resources, we might expect our NLG system to output \LaTeX source or HTML markup. Note that each of these systems provides both **LOGICAL** and **PHYSICAL** annotations: Figure 7.8 shows how a number of typographic requirements can be expressed in terms of their logical status, so that the rendering device hides from the source of the specifications the detail of how the particular typographic element will be rendered; Figure 7.9 shows how these annotation schemes also permit more direct physical markup.

Until recently, work in NLG tended to ignore typographic issues; most systems typically rely on the default renderings provided by standard teletype devices, so that the output produced is presented in a uniform face, font and size. This is not so much a reflection of deficiencies within the systems; rather, it is a consequence of the fact that, by and large, researchers in the field tend to see their goal as the generation of texts rather than of documents. Increasingly, however, NLG systems are becoming typographically-aware, and we are likely to see this tendency accelerate as the Web gains in popularity as a delivery medium for NLG system output.

As we indicated above, some uses of typography are determined by convention, whereas others may be the result of conscious decisions made during the design of a document in order to achieve particular effects. For those aspects of typography based on conventions, the issues that arise when building a natural language generation system are quite simple, provided of course that

Element	L ^A T _E X	HTML
Major heading	<code>\section{...}</code>	<code><h1>...</h1></code>
Minor heading	<code>\subsection{...}</code>	<code><h2>...</h2></code>
Itemised list	<code>\begin{itemize}\item ...</code>	<code>...</code>
Numbered list	<code>\begin{enumerate}\item ...</code>	<code>...</code>

Figure 7.8: Declarative markup annotations

Element	L ^A T _E X	HTML
Emboldening	<code>{\bf ...}</code>	<code>...</code>
Italicisation	<code>{\it ...}</code>	<code><i>...</i></code>
Teletype font	<code>{\tt ...}</code>	<code><tt>...</tt></code>

Figure 7.9: Physical markup annotations

the system is typographically-aware: if the system is capable of indicating typographic variation in its output then all that is required is that the appropriate conventions be hard-wired into the system's behaviour. So, for example, whenever the NLG system refers to a literary work or uses a technical term for the first time, it is relatively straightforward for these to be rendered using the predetermined typographic cues.

Typographic issues are more complicated where the meaning-to-form mapping is one to many: that is, when there are different typographical devices available for expressing a given meaning, and the system must somehow choose between these at run-time. One such case would be the choice between a long and complex sentence and a more typographically structured presentation of the same information, as we have seen in a number of the examples above. Formulation of the reasoning required in order to make such choices is a relatively unexplored area of NLG research. The appropriate choice of typographic devices from the wide range available is clearly a knowledge-rich decision process, and to our knowledge there are no systems which attempt to emulate the complex reasoning processes pursued by a typographer or information design specialist.

7.3.3 Implementing Typographic Awareness

Making an NLG system typographically aware impacts on a number of aspects of the generation process. We can review what is involved here in terms of the architecture used in this book:

The meeting was attended by the following:

Company	Representatives
IBM	Tom Jones
Digital	Cecile Smith, Nancy Robinson
Microsoft	Mike Williams
Sun	Richard Adams, Joe Blackford, Sally Su

Figure 7.10: A tabular presentation of aggregated information

Document Planning

Recall that our view of document planning is concerned primarily with decisions of document content and overall structure, where that structure is derived from the information source itself. Typographic *choices* are unlikely to impact on these decision-making processes; however, typographic *constraints* can play a role here. So, for example, if the genre of the text to be generated requires that certain kinds of information be presented in tabular form, this may provide a constraint on the content elements that can be selected for expression. Similarly, if there is a real size limitation on the text to be generated, this imposes a constraint on the content selection task in terms of the quantity of material it chooses for expression.

Microplanning

It is at the level of microplanning that any real typographic decisions to be made by the system will lie. Thus, for example, decisions about aggregation of information may cause selection of an itemized list as the best means of expression. Consider the following two variants:

(7.3) The meeting was attended by Tom Jones from IBM; Cecile Smith and Nancy Robinson from Digital; Mike Williams from Microsoft; and Richard Adams, Joe Blackford, and Sally Su from Sun.

(7.4) The meeting was attended by:

- Tom Jones from IBM;
- Cecile Smith and Nancy Robinson from Digital;
- Mike Williams from Microsoft; and
- Richard Adams, Joe Blackford, and Sally Su from Sun.

The second of these arguably makes the grouping of persons described clearer. As an alternative to these, the microplanner might choose to use the tabular form shown in Figure 7.10.

A number of researchers have noted that typographic resources can be used instead of, or as a supplement to, cue words when expressing some discourse relations. For example, a SEQUENCE might be expressed by including the sequence elements in a numbered list, instead of (or in addition to) inserting cue words such as *first*, *then*, and *finally*. Thus, where a lexicalisation process might have chosen from amongst different cue words that realise some relation, we now have to countenance the possibility that a non-lexical realisation might be appropriate. Again, this kind of decision is best placed within the microplanner.

Realisation

For each typographic convention, the NLG system needs to be able to make the appropriate underlying semantic and logical distinctions; implementing a convention then corresponds to defining the appropriate mapping from semantic type to typographic realisation. This mapping fits very naturally within the surface realisation component.

The most appropriate conventions to use can be derived from a number of sources, often quite specific to the domain or genre of the texts to be generated. For example, the *Chicago Manual of Style* [?] is a good source for general conventions widely used in English; the AECMA guide to writing aircraft maintenance manuals [?] provides specific stylistic advice for documents in that domain, Microsoft's *Help Authoring Guide* provides suggestions for the use of typography in on-line help messages, and there are a wide selection of business writing guides such as Gartside's *Modern Business Correspondence* [?] that provide useful suggestions in the context of producing business letters.

In summary: as a rough rule of thumb, mechanisms for adherence to typographic constraints are best placed within the document planner; decisions about which typographical resource is the best one to use for realising a part of the document's content are best placed within the microplanner; and the application of typographic conventions can be left to the surface realisation component.

7.4 Integrating Text and Graphics

In the previous section we saw how typographic resources can be utilised in an NLG system. Incorporating the use of typography represents a first step towards the generation of documents rather than texts; with this hurdle overcome, a whole new range of possibilities arise for NLG systems.³ In this section, we build on the preceding material to consider what is involved in having a system generate documents that include both text and graphics.

³Indeed, the different means of presentation of information shown in Figures 7.5, 7.6 and 7.7 demonstrate that there is no hard line between relatively straightforward typographic solutions and the use of graphical elements.

7.4.1 The Benefits of Using Graphical Elements

'A picture is worth a thousand words', states the oft-cited adage. This is often true; in other cases, however, words can communicate information more effectively than graphics. We do not know all the answers as to when graphics are better than words or *vice versa*, but it is clear that the following questions are important in determining which form of presentation is most appropriate:

- What kind of information is being communicated?
- How much expertise does the user have?
- Does the delivery medium or user population impose any constraints?

Information type

In general, it seems that physical information (such as location) is often best communicated graphically, while abstract information (such as causality) is best communicated via text. Text may also be better at communicating very diverse information (such as the type of information found in a secondary-school yearbook), while graphics may be better at communicating a large number of related and structurally similar facts (such as the interdependencies between subtasks). A general point made by Oberlander and Stenning [?] is that graphical representations do not make clear what information is essential and what is not. For example, if we use a set of pictures or an animation to show a user how to fix a flat tire, he or she might not realise that it is not necessary to stand exactly where the person in the picture is standing.

7.4.2 Degree of user expertise

Another factor in determining whether text or graphics should be used is the ability of the user to decipher the conventions of the form of expression being used. Speakers of a natural language possess a tremendous amount of knowledge of their language, including a vocabulary of tens of thousands of words, plus a very rich set of syntactic, semantic, and pragmatic rules for producing and interpreting sentences made up of these words. This knowledge represents an immense base of shared conventions that make communication possible. The language of graphics, on the other hand, is not so conventionalised. There are, of course, some graphical conventions that are widely shared: the layperson may know the meaning of a few hundred traffic signs, and be aware of a few rules of interpretation that are broadly applicable in reading graphical representations (for example, if two objects are linked with an edge, they are related in some fashion). However, graphical conventions are often specific to the needs of particular user groups, and this may make graphical representations more appropriate for experts. In specialised or restricted domains, graphical conventions can offer a more concise or efficient representation of information than would be possible with words, but the knowledge required

to interpret these conventions is often acquired by virtue of being an expert in the domain in question. This means of expression may not be appropriate if the intended audience is not expert in the domain. Many graphical genres have conventions which novices may not be aware of even if they have learned the formal structure of the graphical system [?]. These conventions allow experts to rapidly interpret chunks of a graphical representation, without having to resort to first-principles reasoning. For example, in electronic circuit diagrams, bistable flip-flops are usually drawn as two vertically aligned NAND gates. Any time an expert sees two vertically aligned NAND gates, he or she is likely to assume that they form a flip-flop, without checking the wiring to verify this. However, a novice who is not aware of this convention may find it much more difficult to interpret the diagram. The general point here is simply that because the conventions for interpretation of linguistic material are widely shared, they do not need to be spelled out; in the case of graphics, however, the rules of interpretation may have to be specified explicitly.

Petre [ref??], Kim [ref??], Shneiderman [ref??], and others have pointed out that people often believe graphical representations to be more effective than is in fact the case. Many graphical notations that intuitively seem appealing have been shown to not be particularly effective when evaluated in controlled psychological experiments. Petre discusses several advantages claimed for graphical representations (that they are informationally rich, that they make it easier to get the big picture, and that they are more understandable) and shows that at least in the domains she studied, these advantages did not exist; indeed, experimental evaluation often showed that graphical representations were more difficult to understand than text.

The delivery medium

Finally, an important real-world influence on the choice of text or graphics in many applications is the limitations of the delivery mechanism or particular characteristics of the user population. For example, it is not possible to use graphical representations if the document is being delivered aurally or via telex; and it may not be desirable to use graphical elements if the document is being delivered on a slow network link. On the other hand, text may not be practical if the user population is a world-wide group which has no language in common, or consists of small children with a limited knowledge of written language.

From a practical perspective, perhaps one of the best (and simplest) ways of determining what information to communicate textually and what to communicate graphically is to study documents produced by human experts. If a certain piece of information is traditionally presented using one mode rather than the other, then this may constitute an argument for having a computer document generation system do the same. Of course, the traditional wisdom has to be balanced against the benefits of the opportunities afforded by new technology: in some cases, human-authored documents may not use graphical elements simply because of the expense of creating them by hand.

7.4.3 The Automatic Generation of Graphical Objects

Considerable research has been carried out in the computer graphics community on systems which automatically create information-conveying graphics: given a chunk of information that needs to be communicated to the user, or a communicative goal that needs to be satisfied, such systems automatically create appropriate pictures, diagrams, maps, and so on to convey this information. Some results here are as follows:

Data graphics: One of the most common uses of graphics is to display a set of data values. Commonly used techniques for doing this include X-Y plots, bar charts, and scatter plots. Designing an appropriate graphic—i.e., choosing the general type of graphic, then determining the most appropriate scale, assignment of axes, and so on—for a particular data set can be a difficult task. Ultimately the choices here should be driven by properties of the human visual system, since at the end of the day this is what will determine what works best. In the interim, a substitute in those areas where there is a lack of strong psychological findings are design guidelines proposed by human experts, such as Tufte [?] and Bertin [?]. APT and SAGE [?] are examples of computer systems which automatically create appropriate data graphics.

Node-link diagrams: Another common type of graphic is a series of shapes (most often boxes or circles) connected by arrows; examples include entity-relationship diagrams and PERT charts. Creating a good node-link diagram can again be a difficult task, and the research community does not completely understand what make such diagrams effective and useful. The best studied aspect of this task is ensuring that diagrams satisfy general aesthetic criteria, such as having as few edge crossings as possible; but it is also important to ensure that diagrams conform to the conventions of the target genre (see [?]).

Perhaps the single most widely used algorithm for creating node-link diagrams is the Sugiyama algorithm [?]. A good source of research papers on this topic is the Proceedings of the Annual Graph Drawing Conference [?].

Maps: One of the hardest problems in automatically creating maps is label placement. If many labels need to appear in a relatively small area, it is hard to find a placement which minimises overlaps and also does not obscure underlying features of the map. Marks [?] discusses possible solutions to this problem for computer systems.

Pictures: Producing an image of what an object looks like from a digitally-represented model of the object is a well-studied problem; see any standard graphics textbook, such as [?]. Some research systems have investigated ways of automatically enhancing images with arrows, highlighting, cut-out views, and all the other techniques that human graphics designers routinely use [].

⟨The WIP diagram with “Use cold tapwater” label: to be obtained⟩

Figure 7.11: Multimodal output from the WIP system

There is, then, a body of work which can help us automatically create a graphical object to express a given set of data. Within the context of natural language generation, the real questions focus on how we choose between graphical and linguistic representations of information, and how the two forms can best be integrated.

7.4.4 Text and Graphics Integration in NLG Systems

The inclusion in a document of graphical elements such as diagrams, graphs, charts, and pictures opens up considerably more scope for having an NLG system reason about the available resources; at the time of writing, this constitutes one of the most exciting areas of research in NLG. Research here focuses on issues such as the following:

- Which types of information are best communicated textually, and which are best communicated graphically?
- What are the underlying similarities and differences between text and graphics as communication media?
- How can we combine the two in a way which produces truly integrated documents?

In NLG systems, choosing how to express a given collection of information is known as the *MEDIA ALLOCATION PROBLEM*: given a set of data to be expressed, how should the elements and properties of that data be represented using the available media?

Media Allocation

The selection of media is influenced by a variety of factors:

- the characteristics of the information to be conveyed;
- the properties of the available media;
- the goals of the system in presenting the information;
- known or assumed characteristics of the user; and
- the nature of the task that the user is faced with.

An example of how different information can be presented using different media is demonstrated in Figure 7.11, which shows a part of a multimodal document produced by the WIP system [?]. This is part of a sequence of instructions for using a coffee machine; the example shown instructs the user to pour water into the device. In this example, the WIP system has used graphics to indicate where the water should be poured, but it has used text to indicate the properties of the water to be used. This is because, at least in this context, physical location is best communicated graphically, while information about the nature of objects is best communicated via text.

Systems such as WIP and COMET make these decisions on the basis of MEDIA ALLOCATION RULES; some examples of such rules are the following:

- Prefer graphics for concrete information (such as shape, colour and texture).
- Prefer graphics over text for spatial information (for example, location, orientation, and composition) unless accuracy is more important than speed, in which case text is preferred.
- Use text for quantitative information (such as that expressed by the quantifiers *most*, *some*, *any* and *exactly*).
- Present objects that are contrasted with each other in the same medium.

Commonalities between Text and Graphics

An interesting theoretical issue is how similar automatic text generation is to automatic graphics generation. Are these fundamentally different processes, or there common underlying principles that apply to all attempts to communicate information to people, regardless of the media used?

Many kinds of similarities have been proposed in the research literature. For example, it has been suggested that text and graphics generation systems could use similar content determination systems [?, ?, ?]. The heart of content determination is determining what information is appropriate and relevant and should therefore be communicated; this is true regardless of whether the information is being communicated by text, speech, a formal language, graphics, animation, or any other means.

Although this hypothesis seems plausible, there is currently insufficient evidence to evaluate it. A few research systems which produce documents that contain both text and graphics have used similar content planning techniques for both the textual and graphical components of the document [?, ?]. If we compare systems which only generate graphics with those which only generate text, however, it can be difficult to find similarities other than in very general terms (for example, good advice in both media is to study documents produced by human experts). However, this may partially be due to the fact that NLG systems and systems which generate graphics are usually used to communicate different kinds of information.

Researchers have also pointed out that some phenomena which have been studied in linguistics seem to have analogues in graphics.

Conversational Implicature: People make inferences from the way in which a piece of information is communicated, as well as the actual content [?]. For example, if someone says *Mrs Jones made a series of sounds approximating the score of Home Sweet Home*, most hearers will assume what Mrs Jones did could not truthfully be described as singing. If this was the case, then the speaker would have simply said *Mrs Jones sang Home Sweet Home*. This phenomena is called CONVERSATIONAL IMPLICATURE, and has been extensively discussed in the linguistics literature. Recently, some researchers [?, ?] have suggested that similar inferences occur with graphics. For example, if someone draws a diagram of a computer network with all servers except one vertically aligned, most viewers will assume that there must be something different about the unaligned server.

Sublanguages: Texts frequently must conform to the rules of a particular genre. As discussed above, conventions are very strong in diagram genres as well [?, ?]. There is nothing in principle wrong with drawing a flip-flop with horizontally aligned gates instead of vertically aligned gates, but a diagram drawn in this way will be more difficult for people (especially experts who have already internalised the interpretation conventions) to understand [?].

Structuring: Texts are hierarchically structured into sentences, paragraphs, sections, chapters, and so on. A textual document consisting of a single paragraph with several hundred sentences would not be easy to comprehend. Similarly, complex diagrams with several hundred elements can be difficult to understand, and many experts recommend producing a hierarchically structured set of small diagrams instead of one large diagram [?].

Discourse Relations: Andre and Rist [?] argue that RST-like discourse relations can also be applied to graphics. For instance, in Figure 7.11, the general depiction of the coffee machine serves as a BACKGROUND for the parts of the picture that specifically describe the pouring action.

Many people in the design community have tried to use GRAPH GRAMMARS or SHAPE GRAMMARS to understand and generate diagrams [?]. These grammars are often loosely modelled on linguistic grammars. Similarly to linguistic grammars, they state how legal diagrams can be assembled from individual icons or symbols, and they often associate semantic interpretation rules with the various assembly operations. To the best of our knowledge, graph grammars have mainly been used in the design community, and have not been much used in the automatic creation of information-conveying graphics.

In summary, researchers have investigated applying a wide variety of ideas developed within linguistics to the domain of graphics. Less work has been done, however, on the application of ideas from graphics generation to NLG.

⟨Document plan for above WIP diagram: to be obtained⟩

Figure 7.12: Document plan for Figure 7.11

7.4.5 Implementing Text and Graphics Integration

Ideally, a computerised document generation system should be able to generate documents which contain both text and graphics, using each medium to its best advantage; this is the media allocation problem described above. There are different ways in which the textual and graphical components of such systems can be combined, and these different architectures offer different opportunities for media allocation:

Minimal Integration: In a system where there is minimal integration of the textual and graphical generation processes, the overall communicative goal is divided by a (human) analyst into textual and graphical components, and generators are built for each. There is no operational integration between the text generator and the graphics generator. For example, an analyst might design a multimodal version of SUMGEN-W by combining the existing text generator (without changing it in any way) with a graphics generator that produces a graph of temperature and rainfall over the month in question; to all intents and purposes we then simply have two different modalities being used quite independently.

Integrated Content Planning: In a system where content planning is integrated, a common content planner will decide what information needs to be communicated, and then call text and graphics generators to convey the relevant parts of this information. The media allocation is typically carried out using rules such as those discussed above. Once the information has been dispatched to either generator, the text and graphics generators do not interact with each other. For example, a multimodal version of IDAS might use graphics to convey all location information (by generating a picture of the machine with appropriate labelled arrows), and text to communicate the actual actions the user was supposed to do.

Integrated Text and Graphics: A system of this kind integrates the textual and graphical representations of information at a very fine-grained level. For example, the text might include referring expressions that refer to properties of the graphical presentation; Figure 7.13 shows an example from the WIP system that demonstrates a range of such phenomena.

Systems which only involve minimal integration of text and graphics are of course the simplest kind to construct. The other types of integration require changes to the underlying algorithms that might otherwise have been used by the separate components. The WIP system, for example, uses an integrated

⟨WIP figure showing multimedia reference: to be obtained⟩

Figure 7.13: Varieties of reference in integrated text and graphics

RST-based discourse planner which produces, in our terms, a document plan whose leaf nodes can be either clauses or graphical primitives. Media allocation in WIP is not fixed, but rather partially depends on a user-specified media preference. Figure 7.12 shows the document plan behind Figure 7.11.

7.5 Hypertext

So far our discussion has focussed on characteristics of document generation that do not explicitly take account of whether the document will be presented on-line or on paper. When we look specifically at on-line document generation, a number of other opportunities open up: the on-line environment makes it possible to increase the kinds of media used in a document, so that, for example, we can incorporate audio, video, and other kinds of animation. Explorations of what is possible here have only barely begun; here, we will focus only on one aspect of on-line documents, this being the scope for creating hypertextual documents rather than simple linear documents.

7.5.1 Hypertext and Its Uses

What Hypertext Is

A hypertext document is an on-line document that can be read in a non-linear fashion by virtue of links within the document which allow the reader to jump from one place to another at the click of a mouse. Although the basic idea of hypertext has been around for some time, it is only in recent years with the explosion of interest in the World Wide Web that hypertext has become a part of everyday life for many people. There are other, more complex, forms of hypertext beyond what is found on the World Wide Web; however, we will focus here on hypertext in the World Wide Web context since this is where we expect to see most visible applications of NLG technology in the next decade.

A typical hypertext contains mouse-sensitive regions—either linguistic units such as words or phrases, or graphical elements—which the user can click on. The result of clicking on one of these ANCHORS is that the user is transported, in a *Star Trek*-like fashion, to somewhere else: either to another document or to somewhere else in the current document. Crucially, the new document may not exist prior to the user's mouse click: the result of clicking can be the invocation of a program whose results are returned for the user to view. It is

this capability that Web-based NLG systems capitalise upon. A good general introduction to the ideas behind hypertext is [ref??].

One of the most straightforward uses of hypertext is to allow ease of access to related documents; we will call these REFERENCE LINKS. On the World Wide Web, most hypertext links are probably of this kind; it is precisely this characteristic that gives the World Wide Web its name. Reference links may appear interwoven into the text on a page or may be collected together at a specific location on the page, perhaps along the top or bottom of the displayed material.

A variant of the reference link which is popular in on-line help systems is the GLOSSARY LINK, which pops-up a definition of a term or a short explanatory note when clicked. The information viewed as a result of clicking on a glossary link typically does not contain further links.

Hypertext links may also be used to assist in navigating the structure of a complex document: for example, if a document contains more than one screenful of information, it may be useful to initially show the user an outline that lists the document's sections, and let the user request which sections to read by clicking on the appropriate section heading. The requested section may exist as a separate document, or it may exist at a specific location within the current document. In some hypertext systems, the outline can be expanded to incorporate the requested section in-line.

Hypertext as Dialogue

If an NLG system produces output which is read on-line, it is usually fairly straightforward to add some hypertext links to this output. Such links can significantly increase the usefulness of the generated text, making it much easier for the user to quickly access related documents. Indeed, in the late 1990s many users expect that on-line documents will contain hypertext links, and may be disappointed in on-line texts which do not contain links.

As noted above, clicking on a link may not cause display of a pre-existing document; instead, it may cause invocation of a program whose results are then displayed on the screen. It is this capability that is used in Web-based NLG systems. For example, all of the hypertext links shown in Figure 7.1 cause new invocations of the NLG system which then generate the requested documents: clicking on Crocodile will cause the generation and display of a document about crocodiles, and clicking on Chinese Alligator will cause the generation and display of a document about Chinese alligators.

Sometimes the term DYNAMIC HYPertext is used for NLG systems where the output consists of hypertext nodes that are created dynamically, and hence can be customised according to a user model, the discourse history, and other contextual factors. Dynamic hypertext is a fairly recent development, and our understanding of the best way to use this medium is still evolving.

Within work on NLG, hypertext can be used to support a dialogue between the user and the computer: under this view, hypertext links are seen not as links to documents, but rather as questions or requests that the user can make,

and to which the system will respond. Hypertext allows more limited interaction than the alternative of allowing users to issue natural language queries and requests. However, a reliable hypertextual interface is easier to implement than a natural language query interface; and more importantly, it makes clear to the user what questions can be dealt with by the system at any given point.

The notion of hypertext as dialogue is present in both IDAS and PEBA. IDAS views every hypertext click as a user request or question. Of all hypertextual NLG systems constructed to date, Moore's PEA system [?] supports the most complex dialogue-like interactions: for example, the system allows users to ask why a particular piece of information has been communicated, or to request further explanation of a point not understood.

Whether this view of hypertext will gain widespread acceptance remains an open question. Most current forms of hypertext adopt a metaphor of browsing through an information space populated by documents; this has somewhat different connotations from a view of hypertext as a conversation with an information-providing entity. For example, there is the subtle question of what should happen when a user clicks a second time on a link that has been clicked on before. In the conception of hypertext as a device for exploring an information space, clicking on a link should bring up the linked document, and exactly the same document should appear every time the link is clicked. In the conception of hypertext as a conversational medium, clicking on a link twice could be construed as asking a question twice, and there is no reason why the response should be the same on both occasions. Indeed, a helpful system should probably make the second response more detailed, since one reason for asking a question again is because the original response was not understood. The ability to offer a different answer the second time around comes naturally to an NLG system, since typically answers will be generated in a context that includes a dynamically-updated discourse model; however, this may not be what the user expects. This and related issues are discussed in [?].

7.5.2 Deciding Where to Add Links

Generally speaking, hypertext systems probably function best when users have a good mental model of what links are present and why. The presence of links which do not fit this pattern may confuse or distract users, even if the links are, from a more abstract perspective, useful. It is often better to have a simple and predictable set of links than a complex and unpredictable set, even if the former requires the user to perform more mouse clicks; and it is important to be wary of adding too many links just because the technology makes it possible.

There are many guidelines for manually creating effective hypertexts, including [ref...]. One useful set of guidelines which is freely available on the Internet is Microsoft's *Help Authoring Guide*. This is based on Microsoft's experience gained in creating on-line hypertext help systems for its many software products, and includes guidelines derived from extensive user evaluation trials. Some of these guidelines are shown in Figure 7.14. Although they may not

-
- Glossary links should only be added to the first occurrence of a term in a text, unless the text is very long.
 - Reference links should appear at the end of a text, not in its body.
 - Users can only remember ‘seven plus or minus two’ chunks of information; don’t give them too many options to explore.
 - Each individual hypertext node should if possible fit on one screen, and never be longer than two screens.

Figure 7.14: Example rules from Microsoft’s *Help Authoring Guide* (rephrased from the original)

be appropriate in all circumstances, they provide some useful initial guidance which can be taken into account when designing a hypertextual NLG system.

Another issue concerns the content of anchors. For instance, consider the anchor that appears as Crocodylidae Family in the PEBA output shown in Figure 7.1. This is only one possible anchor of a number that could have been chosen: other alternatives are Crocodylidae and the Crocodylidae Family. There is no straightforward answer as to what the best thing to do is here, although different solutions may carry different semantic connotations.

7.5.3 Implementing Hypertext-based NLG Systems

Adding hypertextual properties to the documents created by an NLG system impacts on a number of aspects of our architecture.

Document Planning

Hypertext links are essentially about the inclusion of content into a document, even if that content is a mouse-click away from the generated document itself. Given this, the document planner should be responsible for specifying the inclusion of hypertext links. There may be some exceptions to this, as we will discuss below: for some links, the document planner may not know that a link is required, since its inclusion may depend on linguistic content only determined at a later stage in the generation process.

To build an NLG system which generates hypertexts, we need, then, to extend the basic document planning mechanism. For example, a rule-based content planner could have rules for adding hypertext links as well as specifying document content; this is what is done in the IDAS system. A plan-based system can reason about what hypertext links might be useful to the user, as is done in [?]. A schema-based system can have basic operations for adding links

as well as adding clauses; this is what happens in PEBA. In each case, the process of adding a link is treated analogously to that of adding a clause, and similar design decisions need to be made about whether the reasoning involved should be shallow or deep, whether user models should be taken into account, and so on.

As noted above, links should be added in a way which makes the hypertext network as a whole comprehensible to the user, so that he or she can develop a good mental model of what information is present in the network, and how the network can be navigated. One suggestion made by Reiter *et al* [?] is to regard the network as a QUESTION SPACE which users can traverse: in the IDAS system, each node answers a specific question about a specific entity, and users can move from a node to other questions about the same object, or to questions about related objects (such as subcomponents of the current object). This is a fairly simple navigation model, but it is one that seems quite comprehensible to users.

Another approach to links is to treat them as ways of presenting optional information. If the document planner is unsure about whether or not a piece of information should be included, it can include a link to a node giving this information, and only include directly in the current node that information which is known to be important. This approach may have the disadvantage of making the network as a whole less understandable and predictable, but is well suited to a view of hypertext as dialogue, as discussed above; under that view, links to optional information can be seen as natural elaboration questions which a user might want to ask. This model also lessens the burden of choice making on the NLG system; the user can decide what information is relevant and important, instead of the system having to make these decisions [?].

Microplanning

The presence of some links may depend on specific lexical content in the generated text. Since, in our architecture, the lexical content is not determined until we reach the microplanning stage, these links can only be added at that point. This is most obvious in the case of glossary links: if a technical term is used, then the microplanner may have the responsibility for adding a link to a glossary item that explains that term. The microplanner may also take responsibility for adding links to descriptions of entities referred to in the text if the content of the link depends on the content of the description.

One important design choice in microplanning in a hypertextual environment is the question of how to model discourses for the purpose of generating referring expressions. The generation of referring expressions requires a discourse model which lists those objects that have already been mentioned in the text. In a hypertext system, it is unclear whether the discourse model should include objects mentioned in previously visited nodes. On the one hand, if the user has read the previous nodes, then presumably objects mentioned in these nodes are salient for the user, and hence should appear in the discourse model. However, intuition suggests that users frequently only partially read

or skim information presented on-screen, and so it may not be appropriate to automatically add mentioned entities to the discourse model. Of course, in the absence of eye-tracking devices being built into terminals, there is no easy way to determine which parts of a displayed screen a user has read. Given this, it may be best to adopt a cautious strategy, and assume that the previous nodes have not been read; this then has the consequence, of course, that subsequent texts may be more redundant than they need to be.

To take a concrete example of this issue, a fairly plausible strategy for referring expression generation would refer to a well-known former Prime Minister of the UK by means of the expression *Prime Minister Margaret Thatcher* for the initial reference, and by means of the form *Thatcher* in subsequent references. In a hypertextual system, we have to decide whether this means that the expression *Prime Minister Margaret Thatcher* should be used for the first reference to this person in *every* node which refers to her, or whether it means that *Prime Minister Margaret Thatcher* should only be used in the *first* node visited which refers to her? In principle the 'right' answer to this question may depend on how thoroughly the user has read the first node which mentioned Thatcher, but this is information that the NLG system does not have.

Surface Realisation

The realiser's job is a simple one here. If the phrase specification language being used supports hypertextual annotations, then the realiser's job is to map these into the appropriate markup for the hypertext delivery system being used. For the World Wide Web, of course, this will be HTML.

7.6 Speech Output

So far in this chapter we have considered some of the questions that arise when we look at the generation of what we have called visible language. We should not forget, of course, that in many contexts the most appropriate means of delivery may be speech; and so, in this final section, we turn to the generation of spoken language.

7.6.1 The Benefits of Speech Output

There are a number of situations where the delivery of information via speech is to be preferred over delivery via text. Spoken output is especially useful in cases where users are unable or unwilling to read written text on a computer monitor or other display device. This is the case, for example, in telephone-based information systems, such as the special information numbers that people can call to obtain weather forecasts, sports results or airline scheduling updates. It is also useful in 'eyes-busy' situations where the user cannot look at a display screen because they have to look elsewhere, as might be the case when a doctor is examining a patient or a driver is being provided with route

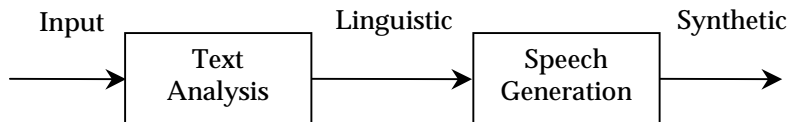


Figure 7.15: The architecture of a text-to-speech system

information via a GPS device. Last but not least, spoken output is essential for users who are unable to read because of some disability.

Some users may also prefer spoken output, even if they are able to look at a monitor. For example, people who are not experienced computer users may in some cases feel more comfortable hearing information presented orally, because that is what they are used to. Also, speech may be a natural mode of communication in some computer games, in part because speech can be inflected to indicate emotional states such as excitement.

Software for generating spoken output from text, called SPEECH SYNTHESIS software, has improved significantly in recent years, and there now are a number of packages available both on the market and as free downloadable software from a number of research institutions. Generally speaking, the quality of synthesized speech is such that it is most intelligible in short segments: a single synthesized sentence is usually much easier to understand than a segment of spoken text several paragraphs in length. As synthesis technology improves, however, the intelligibility of longer speech segments is improving.

Most applications which use speech synthesis to generate spoken output do not currently use NLG techniques. However, as well as improving the appropriateness of what is said, NLG techniques also hold promise for improving the quality of synthesised speech, as we will discuss below; accordingly, there is growing interest in the research community in combining NLG with speech synthesis technology.

7.6.2 Text-to-Speech Systems

To understand the issues here, we first require a basic understanding of how existing text-to-speech (TTS) systems work. Figure 7.15 shows a simple diagram of the component functionality of most TTS systems. The important point to note here is that, before synthesizing speech from an input text stream, a TTS system first typically carries out an analysis of the input text. This analysis is required in order to determine some important aspects of the text that have a bearing on how the text should be spoken. The most obvious of these is that different sentence forms have different INTONATION PATTERNS, as is eas-

ily demonstrated by comparing how the following two sentences sound when spoken naturally:

- (7.5) a. John can open the door.
 b. Can John open the door?

Here, the second sentence is typically spoken with a rising intonation at the end; this is generally true of question forms. Declarative sentence forms, exemplified by the first sentence above, are typically spoken with a falling intonation at the end—although as we will see below, in reality the situation is more complex than this. In order to speak a sentence appropriately, then, a TTS system must first work out what kind of sentence it is. This is relatively straightforward in the above case, since all it requires is identification of the final punctuation mark; however, there are more complex aspects of linguistic analysis that a TTS system needs to perform.

Text normalisation

Speaking a text is not just a matter of sending the words in that text to be spoken, one at a time: some textual elements need to be converted into something speakable. So, for example, any expressions involving numbers or special characters are spoken in ways that are not apparent from the linguistic form itself. Such forms have to be NORMALISED, so that, for example, *28/3/98* may be spoken as *the twenty-eighth of March nineteen ninety eight*, and *80kph* may be spoken as *eighty kilometers an hour*. In a TTS system, these phenomena are generally dealt with by means of special-case pronunciation rules.

Homograph disambiguation

Natural languages contain HOMOGRAPHS: words which are spelled the same but are spoken differently, depending upon their part-of-speech or their meaning. Compare, for example, the spoken form of the word *lives* in the following two sentences:

- (7.6) a. John lives in Edinburgh.
 b. A cat has nine lives.

A text-to-speech system has to work out which form of the word is intended. This is generally done by carrying out some analysis of the input string in order to assign part-of-speech tags, since in many cases homographs are distinguished by part-of-speech: typically the noun and verb instances of a homograph will be pronounced differently. There are cases, however, where the part-of-speech is the same for both instances, as in the case of words like *bow* in the following pair of sentences:

- (7.7) a. At the end of the performance, John took a bow.
 b. John shot the target with a bow and arrow.

Homograph disambiguation is not entirely distinct from text normalisation: for example, the abbreviation *St.* is ambiguous as to whether it should be pronounced *Street* or *Saint*.

Prosody assignment

This is the most complicated aspect of generating synthesised speech. Prosody is the term used to cover the pitch, speed, and volume with which syllables, words, phrases, and sentences are spoken. Our example of the two sentence types above is one case where prosody plays a role, but the assignment of prosody is made more complex by the fact that real utterances tend to break down into a number of INTONATION UNITS, each requiring appropriate prosody to be applied. Consider how the following sentences sound when spoken naturally:

- (7.8)
- a. When I came in from the cold, the spy I saw on the floor was not the person I had spoken to on the phone.
 - b. John, not a good archer at the best of times, only managed to hit the target once—not with *every* shot, which is what he claimed he could do.

Each sentence breaks down into a number of INTONATION UNITS or INTERNATIONAL PHRASES, with the overall pronunciation of the sentence taking on a rhythmic structure. These intonation units can sometimes be identified by looking for punctuation marks such as commas; but in the general case there may be no such surface clues, and the intonation may be related to syntactic structure in subtle ways which are still the subject of ongoing research. This means that, for a TTS system to produce good prosody, it has to carry out quite sophisticated analysis of the text.

The basic point, then, is that TTS systems need some knowledge of the structure and content of the text to be spoken if they are to do a good job of speaking the text. Most TTS systems provide some mechanism for augmenting the bare text stream with additional information that can be used to improve the output quality, either by overriding the results of analysis the TTS system might perform or by giving the system information that it would not be able to work out for itself. Figure 7.16 shows some of the annotations provided by SABLE, a recent attempt to provide a standard language for speech synthesis markup.⁴ Each tag here is interpreted appropriately by the TTS system.

Concept-to-Speech

As we have seen, text-to-speech systems effectively reverse engineer their input in order to determine information they need to produce appropriate out-

⁴All examples here are drawn from the SABLE version 0.1 specification; information about the SABLE specification can be found at http://www.cstr.ed.ac.uk/projects/sable_spec.html.

Phenomenon	Example
Emphasis	The leaders of <EMPH>Denmark</EMPH> and <EMPH>India</EMPH> meet on Friday.
Breaks	Without style, <BREAK LEVEL="large"> Grace and I are in trouble.
Pitch change	Without his penguin, <PITCH BASE="-20%"> which he left at home, </PITCH> he could not enter the restaurant.
Pronunciation	<PRON SUB="tomahto">tomato</PRON>
Special processing	At <SAYAS MODE="time"> 2pm </SAYAS> on <SAYAS MODE="date" MODETYPE="YM"> 98/3 </SAYAS> Mike will send <SAYAS MODE="currency"> \$4000 </SAYAS> to <SAYAS MODE="net" MODETYPE="email"> me acme.com </SAYAS>.

Figure 7.16: SABLE markup for controlling speech synthesis

put. The key problem here is that, given the current state of the art in natural language analysis, this process provides insufficient knowledge about the sources of constraints on intonation, with the result that it is difficult to determine the appropriate assignment of intonation. Ultimately, syntactic analysis—which itself is at the bounds of current NL analysis capabilities where unrestricted text is concerned—is not sufficient. Intonation is more than a reflection of the surface syntactic structure of the text, and in the final analysis depends on communicative goals underlying the utterance.

Of course, this is information that we can reasonably expect an NLG system to already be capable of providing: an NLG system will know which particular word sense is intended when a homograph is used, it will know what the type and the structure of a sentence is, and, most importantly, it will know why it is using a particular utterance. It makes sense, then, to consider linking an NLG system to a TTS system in such a way that the need for analysis of the input stream by the TTS system is removed. By doing this, we provide an input to the speech synthesis process that is richer than that normally assumed by TTS systems; accordingly, this approach is sometimes referred to as CONCEPT-TO-SPEECH.

To see the benefits of having access to this rich information source, consider the utterance *John washed the dog*. This can be uttered as a response to each of the following questions, but the intonation used in uttering this response will vary depending on the question:

- (7.9) a. What did John wash?
b. Who washed the dog?

Although the one sentence serves as a response to each of the two questions, in each case the sentence will carry a different stress pattern in order to mark the information that is *NEW*: in the case of the first question, the noun phrase *the dog* carries this new information whereas in the case of the second question it is the noun phrase *John* that carries the new information. An NLG system, since it has deliberately constructed the sentence in order to satisfy the request for information, will know what is given and what is new, and can assign appropriate intonation in line with this. Similarly, provided we have an appropriate theory that correlates syntactic structure with prosodic structure, an NLG system will know where the appropriate intonation boundaries should fall in a longer utterance.

7.6.3 Implementing Concept-to-Speech

If we want to speech-enable an NLG system in a richer way than simply applying a TTS system to the NLG system's output, there are two general issues we need to address:

- How can the appropriate prosodic and intonational annotations described above be produced?
- How should the content, structure, and style of the text be modified to reflect that fact that it will be spoken instead of read?

We can review how these questions impact on each of the components of our NLG architecture as follows.

Document Planning

First of all, it seems reasonable to suppose that a document generated to be presented via spoken language may vary in terms of its linguistic content and structure from a document generated for visual presentation. Speech and writing are very different forms of language, and indeed the kinds of uses to which each are put vary considerably: there are linguistic acts which would make little sense, or seem very odd, if transplanted from one medium to the other. The document planner may therefore need to adopt rather different strategies for both content determination and text structuring, depending upon the intended output modality. For any given domain of application, what is required here is best determined by a careful empirical analysis of information provision in that domain.

As a simple example, the document planner may need to apply different considerations with regard to the length of the generated output: because speech takes time to utter and cannot be skimmed by the recipient in the way that text can, the document planner may have to limit the size of spoken documents; in contrast, length considerations are probably less of an issue in written text. More speculatively, it may be appropriate to repeat information more often in spoken documents than in written text, because there is no correlate in the spoken form of the scope for rereading provided by the written form.

Overall characteristics of the spoken output, such as speed of talking and the emotional attitude expressed in the speech, are probably best determined within the document planning component, since these are likely to be based on the system's communicative goals and known characteristics of the user.

One possible impact of length limitations and the difficulty of revisiting already uttered text is that a dialogic mode of interaction may be more important in spoken systems than in text-based systems: dialogue provides a natural way for users to 'home in' on the information they need, or to request that a previous utterance be repeated or clarified.

Microplanning

Since the microplanner is responsible for lexicalisation, it is the appropriate place to include information that will allow the later stages of the process to ensure that words are pronounced properly. In the context of an NLG system, there would seem to be little need for a text normalisation stage: instead, the microplanner should build lexicogrammatical constructions that are already in a form suitable for speaking. Since the microplanner also determines the overall sentential structures to be used, it should ensure that the phrase specifications constructed clearly indicate the information structural properties of the utterance, so that these can be mapped into appropriate prosodic contours by the realisation process.

From a stylistic perspective, spoken text is clearly different from written text [?]. In the context of NLG systems, McKeown (1997) suggests that, among other things, spoken output should be short, syntactically simple, and relatively informal. Further research is needed to better understand how to interpret these criteria, especially as they may conflict: for example, as McKeown points out, brevity may conflict with syntactic simplicity.

Realisation

The function of the realiser is to map distinctions at the level of meaning—here including aspects of information structuring—into prosodic distinctions. The relevant information will almost all have been determined by earlier stages of processing, although where the realiser has some latitude with respect to the choice of syntactic constructions, it also then needs to know enough about prosody to be able to annotate these structures appropriately. For our purposes, it is reasonable to assume that the role of the realiser is to take the information specified in the input phrase specification and to build an output stream that includes markup along the lines of that shown in Figure 7.16; some of the information provided here replaces the punctuation symbols that the realiser would insert in the case of written output.

7.7 Further Reading

In this chapter we have covered a very broad range of research issues in a relatively small space, and so we have only been able to scratch the surface of many of the issues involved. We provide here some pointers to the relevant literature for the interested reader who wants to pursue these topics further.

While many papers have been published on the individual topics of typography, hypertext generation, the integration of text and graphics, and speech generation, very little has been published on integrating all these aspects into a general architecture for document generation. Probably the most advanced work on generating rich multimodal documents has been carried out at Saarbrücken and Columbia; see [?] and [?] for examples of multimodal systems built at these laboratories.

Relatively little has been published on typographic issues in NLG, with ?? and [?] being two exceptions. There is, however, a rich literature aimed at human writers and document designers on the best use of fonts, list structures, boxes, and so forth; this literature may suggest rules which can be incorporated into NLG system. See, for example, [?] and [?].

There is a large general literature on hypertext. One introductory textbook on hypertext is [?]; a good up-to-date source of recent research papers on hypertext is the annual ACM Hypertext Conference. There is also a growing literature on dynamic hypertext, and on how NLG systems can produce it; [?] is a good review article with pointers to other work. [?] and [?] describe how hypertext is used in the PEBA and IDAS systems, respectively.

There is a very large general literature on computer graphics; [?] is a standard introductory textbook to this area. Two of the best known systems which generate documents which contained both text and graphics are WIP [?] and COMET [?]. Maybury [?] is a useful collection which contains many papers which discuss the integration of text and graphics. Petre [?] points out some of the problems with graphics, and emphasises that graphics is often less effective than people believe it to be. Roth and Hefley [?] provide some discussion on media choice, including examples of rules used by different systems. Many of the other chapters in Maybury [?] also touch on this issue.

Systems which combine text and graphics have been constructed for a wide range of tasks and text types, including the generation of instructions for operating technical devices ([?], [?], [?], and [?]); mission planning and situation monitoring ([?], [?]); project management ([?]); business forms ([?]); and computer network configuration ([?]).

There is a general literature on speech synthesis; one textbook which covers this topic is [?]. There are many companies which sell commercial speech-synthesis systems. Microsoft Research currently makes available for free its speech synthesis package; see <http://www.research.microsoft.com> for details. Halliday [?] argues convincingly for speech and writing being viewed as quite distinct forms of language use. There is relatively little work on combining NLG with speech, since this is a very new area, but two complete systems which do this are DYD [?] and MAGIC [?]. Prevost's work [?] on integrating

intonation with categorial grammar shows how existing linguistic theories can be extended to allow the incorporate of prosody.

